UNIVERSITY OF CALIFORNIA,
IRVINE

Building Gene Regulatory Networks using Self-Organizing Maps

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Biological Sciences

by

Camden Jansen

Dissertation Committee:
Associate Professor Ali Mortazavi, Chair
Professor Ken W. Cho
Associate Professor Maksim Plikus
Assistant Professor Zeba Wunderlich

2019

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

First, I want to offer my deepest thanks to my mentor and chair, Dr. Ali Mortazavi. Learning from him how to be a scientist has been the greatest adventure of my life. His patience for me and my work has been incredible, and he has imparted wisdom to me that will last a lifetime. Without his guidance and persistent help this dissertation would not have been possible. We have done some amazing work together, and I can't wait to see where this path for me continues.

I would like to thank my committee members, Dr. Ken W. Cho, Dr. Maksim Plinkus, and Dr. Zeba Wunderlich, who have been instrumental in guiding my research. I would like to additionally thank Dr. Cho for all of the support he has given me.

I'd like to thank my whole family, mom, dad, and Trenton, my brother, for always being there for me whenever things looked bleak and for their never-ending support throughout my graduate school experience. You are the best family I could ever ask for.

Finally, I want to thank my friends at UCI. You have made the last 6 years the best years of my life. I would like to especially thank Bryan for grabbing coffee with me and being an amazing friend. I would also like to thank all of the Mortazavi lab members for making work as fun and exciting as possible. I'd especially like to thank Gaby and Kate for organizing events for the lab and department and being the best outing buddies a guy could hope for. I will never forget you all.

CURRICULUM VITAE

EDUCATION

Doctor of Philosophy in Developmental and Cell Biology, Advisor: Dr. Ali Mortazavi
University of California, Irvine 2016 - 2019

Master of Science in Biology, Developmental and Cell Biology
University of California, Irvine 2013 - 2016

Bachelor of Science, Computer Science
University of California, Irvine 2011 - 2012
California Institute of Technology 2005 - 2008

RESEARCH EXPERIENCE

University of California, Irvine, Ph.D. Candidate 2013 - 2019

- Developed software package, SOMatic, to perform unsupervised learning through self-organizing maps on big highly-dimensional genomic datasets https://github.com/csjansen/SOMatic
- Developed the "Linked SOMs" algorithm to analyze data from multiple sources and build regulatory maps between them
- Resulted in 3 completed publications and 2 in preparation, 1 of which is first-author[1,2,3,5,6]

California State University, Fullerton, Research Associate 2010

- Developed the "MissMech" R package for analyzing the incidence of missing data in a collection
- Resulted in a completed publication[4]

California Institute of Technology & Jet Propulsion Lab, SURF Research Fellow 2005

- Developed and tested the Mars Rover Navigation Algorithm for the Mars Science Laboratory (Creativity) using the "Fido" test rover
- Created random and realistic testing environments by implementing a mathematical model of the Martian terrain

Massachusetts Institute of Technology, Summer Research Position 2004

- Created a fast and accurate face recognition system by combining eigenfaces and affine transformations

California State University, Fullerton, High School Researcher 2002-2004

- Developed a number of theorems that together classified a majority of matrices as being "Rootless," the property of having no roots
- Resulted in a completed publication[7]

BIBLIOGRAPHY

[1] **Jansen, Camden**, Ricardo Ramirez, Nicole C. El-Ali, David Gomez-Cabrero, Jesper Tegner, Matthias Merkenschlager, Ana Conesa and Ali Mortazavi, *Building gene regulatory networks from scATAC-seq and scRNA-seq using Linked Self-Organizing Maps,* in review**.**

[2] **Jansen, Camden** and Ali Mortazavi, *Progressive clustering and characterization of increasingly higher dimensional datasets with Living Self-Organizing Maps,* accepted in **Advances in Intelligent Systems and Computing – WSOM+ 2019 Proceedings.**

[3] Partridge, Christopher, Surya B. Chhetri, Jeremy W. Prokop, Ryne C. Ramaker, **Camden S. Jansen**, Say-Tar Goh et al., *Genomic characterization of 208 transcription factors and other DNA-associated proteins in a single cell type,* in review.

[4] Longabaugh, William JR, Weihua Zeng, Jingli A. Zhang, Hiroyuki Hosokawa, **Camden S. Jansen**, Long Li, Maile Romero-Wolf et al., *Bcl11b and combinatorial resolution of cell fate in the T-cell gene regulatory network*, **Proceedings of the National Academy of Sciences** (2017): 201610617.

[5] Jamshidian, Mortaza, Siavash Jala Jalal, and **Camden Jansen,** *MissMech: an R package for testing homoscedasticity, multivariate normality, and missing completely at random (MCAR*), **Journal of Statistical Software** (2014): 1-31.

[6] Yue, Feng, Yong Cheng, Alessandra Breschi, Jeff Vierstra, Weisheng Wu, Tyrone Ryba, Richard Sandstrom, **Camden Jansen** et al., *A comparative encyclopedia of DNA elements in the mouse genome*, **Nature, 515,** v. 7527 (2014):355.

[7] Mortazavi, Ali, Shirley Pepke, **Camden Jansen**, Georgi K. Marinov, Jason Ernst, Manolis Kellis, Ross C. Hardison, Richard M. Myers, and Barbara J. Wold, *Integrating and mining the chromatin landscape of cell-type specificity using self-organizing maps,* **Genome Research** (2013).

[8] **Jansen, Camden**, and Scott Annin, *Rootless Matrices: A New Approach to Their Classification*, **NCUR Proceedings**, August 2004.

TEACHING EXPERIENCE

Teaching Assistant - Fall 2017
Introduction to personalized medicine
Instructor: Dr. Ali Mortazavi

Teaching Assistant - Winter/Spring 2015
Lab in developmental and Cell Biology
Instructor: Dr. Debra Mauzy-Melitz

Teaching Assistant - Fall 2015
DNA to Organisms
Instructor: Dr. Justin Shaffer

EMPLOYMENT HISTORY

DTI Software Inc.
388 St-Jacques Street, 1st Floor
Montreal, QC H2Y 1S1, Canada

Certification Programmer, Aug 2009-Oct 2010
I worked as a programmer in their Certification team based in California. My principal responsibility was to examine submitted game software code (C++ language), perform necessary modifications as required by the end users (commercial airlines), and repair any discovered anomalies within the game code. These activities commonly required large-scale changes in the game codes, and required interaction with the software customers and Panasonic Corp. (game hardware providers). This work activity gave me incredible experience in code modification, game development using software development tools, and interacting with clients and other vendors.

Location Based Tech Inc/ PocketFinder
Jenner #100, Irvine, CA 92618

Contract Software Developer, Jan 2009-Apr 2009
At PocketFinder, I worked as part of their software development team. This involved working on the back end of their website to allow users to download their movement history to a pdf, creating a tech support system from scratch, and various bug fixes in both the server side and client side code. I also had to build large sections of their ASP.NET website as well as perform large edits to the existing architecture. I have learned how to use ASP.NET as both a debugging tool and also a development platform. My experiences also include web apps for the Iphone, G1, Blackberry, and Windows Mobile phones. Finally, I have learned how to work on several large web projects with several programmers and how to use SVN to manage code.

LaserFische Inc.
3545 Long Beach Boulevard
Long Beach, CA 90807

Software Developer Intern, May 2006 – Aug 2006
I developed an automated process for the recognition of documents from within photographs, which is especially useful on near-white backgrounds. I developed many specific user tools and the GUI for processing both scanned and photographed documents, including a lens correction tool, an area tool that allows for more than one region to be processed, a polygon tool for selecting regions, and I also created paper prototypes for the workflow web client.

SELECTED PRESENTATIONS

Taught Tutorials

Center for Complex Biological Systems (CCBS) Short Course
Irvine, CA 2018
Analyzing RNA-seq data

Encyclopedia of DNA Elements (ENCODE) 2015: Research Applications and Users Meeting
Washington DC 2015
SOMatic Tutorial
https://www.youtube.com/watch?v=zddlCTNQ7XA

Oral Presentations

ENCODE Consortium Meeting
Washington DC 2019
Progressive clustering and characterization of growing highly-dimensional datasets with Living Self-Organizing Maps

Center for Complex Biological Systems (CCBS) Retreat
Pasadena, CA 2018
Analyzing Cell-Cell Signaling in Breast Cancer Through scRNA-seq and Doublet RNA-seq

Tsukuba Global Science Week
Tsukuba, Japan 2017
Integrative Analysis of scRNA-seq and scATAC-seq data using Self-Organizing Map Fusion

ENCODE Consortium Meeting
Cold Spring Harbor, NY 2015
Spliced self-organizing cubes for integrative analysis of chromatin and gene expression data

Poster Presentations

ENCODE Consortium Meeting
Washington DC 2019
Progressive clustering and characterization of growing highly-dimensional datasets with Living Self-Organizing Maps

International Society for Computational Biology (ISMB)
Chicago 2018
Global Analysis of Transcription Factor ChIP-seq binding using Self Organizing Maps

Encyclopedia of DNA Elements (ENCODE) Consortium Meeting
Stanford 2018
Global Analysis of Transcription Factor ChIP-seq binding using Self Organizing Maps

Tsukuba Global Science Week
Tsukuba, Japan 2017
Integrative Analysis of scRNA-seq and scATAC-seq data using Self-Organizing Map Fusion

Systems Biology of Cancer Journal Club
Irvine, CA 2016
Spliced self-organizing cubes for integrative analysis of chromatin and gene expression data

Cold Spring Harbor Laboratory (CSHL) Systems Biology: Networks
Cold Spring Harbor, NY 2015
Spliced self-organizing cubes for integrative analysis of chromatin and gene expression data

Encyclopedia of DNA Elements (ENCODE) Consortium Meeting
Cold Spring Harbor, NY 2015
Spliced self-organizing cubes for integrative analysis of chromatin and gene expression data

ABSTRACT OF THE DISSERTATION

Building Gene Regulatory Networks using Self-Organizing Maps

By

Camden Jansen

Doctor of Philosophy in Developmental and Cell Biology

University of California, Irvine, 2019

Associate Professor Ali Mortazavi, Chair

Gene expression is a tightly controlled process in all cells and at all stages of life. Expressing the wrong gene at the wrong time in the wrong cell can be deadly to an organism and is one of the hallmarks of disease. The primary point of control of gene expression is transcriptional regulation, which is the process gating the transcription of a gene into mRNA. This process is largely controlled by protein-DNA interactions where specific proteins recognize a specific DNA sequence potentially in combination with other proteins in order to bind to that location and either recruit or repel the general transcriptional machinery. These protein-DNA interactions can be abstracted into connections on a gene regulatory network (GRN) for visualization. GRNs have been drawn for many cellular functions from the bottom-up, in which each interaction is exhaustively studied one-at-a-time, representing months or years of work. In this thesis, I present two works that build these networks from the top-down with self-organizing maps using (1) single cell gene expression and single cell chromatin accessibility drawn from a mouse pre-B cell differentiation system and (2) a large dataset of bulk functional genomics assays of

mesendodermal development in *Xenopus tropicalis*. The resulting networks not only recapitulate known interactions, but they also introduce a large number of new potential regulatory connections for each system. Finally, I present a process for performing this analysis on a growing dataset with iterative releases without requiring a full re-classification. In all, the results of this work provide a novel way to study the regulation of gene expression using integrative analysis of large functional genomics datasets.

THEME OF THESIS

The complex process of transcriptional regulation is abstracted into gene regulatory networks (GRNs). These networks have been built for a number of cellular functions in an exhaustive bottom-up manner, where each connection in these networks represents months or years of work. Recently, the availability of large-scare functional genomics datasets prompted the development of methods for building GRNs from the top-down. The central theme of my thesis is to use self-organizing maps to find regulatory patterns in highly-dimensional functional genomics data and build gene regulatory networks from these patterns.

In Chapter 1, I review the published methods for analyzing highly-dimensional data and provide a sample analysis and experimental pipeline. Recent advances in single-cell experimentation have given even small research teams the ability to generate very large datasets. There is a pressing need for a new generation of analysis tools capable of providing insights from functional genomics "big data". The recent popularity of single-cell RNA-seq has led to the development of over 400 scRNA-seq tools alone, and it can be difficult to know which is the proper tool to use. In this chapter, I describe many of these recent tools and their proper function. First, I show the proper way to prepare genomics data for downstream analysis. Next, I go through an example pipeline for classifying experimental states, such as cell subtypes, through dimensionality reduction and machine learning. I describe the most powerful techniques for identifying biological modules through clustering by observation. Finally, I describe multiple methods for jointly analyzing RNA-seq and epigenomic data.

In Chapter 2, I introduce the Linked Self-Organizing Maps (SOM) method of integrating multiple sets of highly-dimensional data from different sources. SOMs are an unsupervised learning method for clustering highly-dimensional data, which I apply here to the clustering of functional genomics data into similarly regulated sets of genome region-gene pairs. These pairs are then scanned for motifs and used to build gene regulatory networks with significantly more connections than motif scanning would return on its own. We apply this method to a mouse pre-B cell differentiation system in which the level of Ikaros in the nucleus was doubled driving these cells to differentiate 1 step towards becoming full B-cells. We measured chromatin accessibility and gene expression in single cells. The Linked SOM method generated a gene regulatory network immediately downstream of Ikaros with 43 known and 20 new connections.

In Chapter 3, I apply the Linked SOM method to a large bulk functional genomics data set built around mesendoderm development in *Xenopus tropicalis* with a number of perturbation experiments and treatments. Due to the ease of the model system and the importance of this key point in development, there has been a substantial amount experiments performed such that the complexity of the resulting data set is high enough to apply the Linked SOM method. After determining that each data-type was appropriately clustered into modules, I convolved the two separately trained SOMs for gene expression and chromatin state to identify motifs for 12 Xenopus transcription factors in the resulting Linked Metaclusters (LMs). This generated a large network of over 30,000 unique connections with the largest fraction of these belonging to foxh1, an important maternal transcription factor. In all, we recovered all known foxh1 targets and ~150 new potential transcription factor targets including some that are known from other organisms.

In Chapter 4, I introduce a new method for using SOMs on a growing data set. Typical analysis workflows require data sets to be completely static. If new data is released, the entire pipeline including all down-stream analyses needs to be rerun from scratch. The Living SOM method, on the other hand, uses previous analyses to slightly anchor the SOM during a data insertion. This allows the topology to only change enough to include the new data, keeping the down-stream analysis as intact as possible. To test this new method, I compare its reproducibility to normal Kohonen SOMs. In normal circumstances, the Jaccard indices, which is a measure of reproducibility, were not impacted. However, the Living SOM performs significantly better than a standard Kohonen SOM following a simulated data release. To finish this work, I investigate the effect of data insertion order on this method and discuss the ability of Living SOMs to find improperly labeled or extremely error-prone data through a drop in reproducibility.

In Chapter 5, I discuss possible future work for each of the chapters above. I suggest several new analyses using new state-of-the-art techniques. I focus heavily on new neural network techniques particularly in motif discovery and scanning. I also discuss a potential way to determine the effect size of each regulatory connection in a gene regulatory network using deep stacked neural networks.

CHAPTER 1

Introduction

Analyzing and Integrating Highly-Dimensional Gene Expression and Epigenomics Data

Chapter 1

Introduction - Analyzing and Integrating Highly-Dimensional Gene Expression and Epigenomics Data

1.1 Abstract

The recent availability several commercial single-cell platforms has let to the proliferation of data sets with thousands of dimensions in the form of individual cell measurements for even the smallest project with relative ease. However, new computational tools are necessary to analyze this new form of sparse, noisy data. In recent years, there has been an explosion of computational tools to fill that need, but new users can still find it daunting to choose between the tools that will work best for them or the best practices for using them. In this review, we provide a sample analysis and experimental pipeline that can be used to study any genomically-supported model system such as exprloratory experiments, finding sub-populations, or detailed integrative regulatory analysis of those sub-populations.

1.2 Introduction

Recent advances in functional genomics technologies, especially in the realm of single-cell experiments, have given rise to vast data sets in an ever-growing number of organisms and experimental conditions. For example, comprehensive single cell atlases of gene expression have been built for tissues such as the Drosophila brain throughout its lifespan[1] to an entire mouse[2]. Large scale surveys of chromatin accessibility and gene regulation through projects like ENCODE[3] have revealed a wealth of insights. New droplet-based technologies such as 10X[4] or DDseq (Bio-Rad) have enabled even small research teams to have access to a high resolution survey of their model system of interest.

These advances came alongside a host of new bioinformatics tools to analyze these results. Each tool has its strengths and weaknesses and is meant to be used on data of a certain type at a certain stage of analysis. According to scrna-tools[5] at the time of this publication, there are 397 different analysis platforms in the literature for scRNA-seq alone. There are so many, in fact, that it is becoming difficult to determine, given a certain type of data, which is the best tool to use. This is why one of the biggest advantages of using a commercial product such as 10X is access to their proprietary software packages like CellRanger[4] which will do all of the initial steps in your analysis automatically at the cost of flexibility. However, analysis of data from ad-hoc methods typically requires ad-hoc pipelines to shape it into the form that the platform of choice requires. The purpose of this work is to help the reader to determine which programs to use on their data and to help plan their analysis.

1.3 Preparing highly-dimensional genomics data for downstream analysis

The initial goal of all functional genomics analyses is to create a data matrix in the proper format. The creation of this matrix is specific to each type of experiment, but there are plenty of resources to help for highly-dimensional RNA-seq[1, 3, 6-10], ATAC-seq[7, 10, 11], or ChIP-based assays[12, 13]. Each of the following analysis platforms requires a specific input format for the data and experimental metadata that will require reading through their individual documentations. For RNA-seq, some packages require the matrix to contain gene counts, and some require TPMs. For DNA-based technologies, some require peaks to be called using tools such as Homer[14] and reads under those peaks reported, and some such as Dr.seq2[15] will work directly with your fastQ read files to build the matrix automatically.

With this matrix in hand, it is important to do several rounds of quality control (QC) to prune experiments that will bias results in the wrong direction. For most applications, there are tools for each type of data to perform this. For example, Seurat[16] has been designed to QC and visualize scRNA-seq data and Dr.seq2[15] does the same for DNA-based single cell assays such as scATAC-seq and Drop-ChIP. However, for new assays these methods may not exist yet, so we have detailed some possible QC steps here.

The first round of QC should involve some form of simple dimension reduction, such as a principal component analysis (PCA)[17] or t-distributed stochastic neighbor embedding (T-sne) plots[18], to see if any experiment is an outlier that does not cluster with its replicates or if a cell in a single-cell experiment is vastly different compared to the others. This is followed by a second round o QC metrics that depend on the type of data you are using. For RNA-type experiments, a few metrics that are used by the community[19] include number of genes detected, total gene count per experiment, and percentage of mitochondrial mapped reads. For DNA, metrics include percentage of reads under peaks (efficiency or FRiP)[20] and percentage of experiment-wide peaks detected per sample. The final round of QC should involve removing observations (genes or peaks) that are only seen in a few experiments to improve processing time and avoiding clustering on noise and empty vectors.

After QC, it is important for downstream analysis to do data normalization if there are batch effects. A popular choice is quantile normalization[21] even though there is some work in Information Theory that states that this process removes up to 40% of the information stored in a data set[22]. Others use spike-ins in their original experiment, but there are not many tools available to use these to great effect. For scRNA-seq, Seurat[16] will handle many normalization steps for you. One step that Seurat helps the most on is scaling cell cycle

genes to put each of your cells in the cell cycle step *in-silico*, but this must be deactivated if cell cycle genes are important to your specific condition as we do in Chapter 2. Down-stream analysis can begin after a final data matrix is computed.

1.4 Classifying experimental states through dimensionality reduction and machine learning

Given a model system that is not yet well-studied, it is currently best to begin with exploratory experiments to classify experimental populations using droplet-based scRNA-seq platforms such as 10X or DDseq. These experiments typically survey a large number of experimental conditions at low-to-medium coverage to find new populations, to classify known ones further, or to determine combinatorial signals of interest. The analysis pipeline for these experiments can be abstracted to a general pipeline of observing the data's structure through dimensionality reduction and grouping experiments/cells using advanced clustering methods. These clusters can then be classified either manually, by using detected gene/genomic region dynamics, or by integrating multiple types of highly dimensional data.

There are multiple dimensional reduction techniques, but there are two that have become the most popular: T-sne plots[18] and Uniform Manifold Approximation and Projection (umap)[23]. Both are nonlinear dimensionality reduction methods that embed highly dimensional data into a two- or three-dimensional space, but each have their specific uses. T-sne plots are designed to separate data points into clusters and push these clusters apart in the display. The method is so effective at this that it has been shown that even un-clustered data will be organized[24], so it is important to choose the parameters correctly[25].

Additionally, the visual clusters are positioned in the embedding such that the distance between them is skewed, and therefore, has little meaning. Thus, this method should only be used if easily visible cell clusters, but not their relationship, is the goal. On the other hand, umaps were designed with the intent of keeping the distance between clusters relevant. However, this comes with the trade-off that the clusters are less visually distinct, and thus, should only be chosen if cluster relationships are important to the analysis.

After viewing the structure of your data, it is important to choose the clustering algorithm that will work best for your particular data. For most non-differentiation experiments, cells should separate fairly well using the dimensional reduction techniques from the previous paragraph, and thus, one can normally draw clusters by hand. However, it is possible that some clusters may be difficult to visually determine. In this case, a clustering algorithm needs to be chosen. A recent review of scRNA-seq clustering methods[26] for cells made a list of the following types of possible methods (and their tools): hierarchical clustering (ascend[27], CIDR[28], PCAHC[17, 29]), self-organizing maps (FlowSOM[30]), density-based clustering (monocle[31]), k-means clustering (PCAKmeans[17, 32], pcaReduce[33], RtsneKmeans[18, 32, 34], SC3[35]), k-medoid clustering (RaceID2[36]), k-nearest neighbor (Seurat[16]), and model-based clustering (TSCAN[37]). Of these, the two methods specifically made for scRNA-seq data (SC3[35] and Seurat[16]) performed the best overall on scRNA-seq data using default parameters, but FlowSOM[30] performed the best at a higher number of clusters. So, self-organizing maps may be a better option if the expected number of clusters is quite high.

Single cell DNA-based assays have an entirely separate set of tools designed for clustering cells. For example, multiple scATAC-seq tools have been released to address its

inherent sparsity. There are methods (such as chromVAR[38] and BROCKMAN[39]) that make use of well-curated lists of transcription factor (TF) motifs to separate cells by estimated TF activity in open regions. These have been shown to be effective in very-well-studied systems, but require highly detailed motif databases. Next, there are techniques that use identified "landmark" regions such as scABC[40], to cluster cells, but these have only been shown to separate very different cell-lines from each-other and may have trouble on time course data where changes are more granular. Finally, there are methods that correct scATAC-seq's sparcity by imputing cell differentiation orderings, such as Cicero[41] and work by Beunrostro and colleagues[42]. These were capable of separating cells in systems with a strong differentiation lineage. Finally, cisTopic[43] was specifically designed to cluster scATAC-seq data with an unsupervised Bayesian framework and does quite well at separating cell types on a number of different data sets, even on huge (>30,000 cells) data sets. However, it hasn't been formally compared to other clustering methods, so its effectiveness is still relatively unknown.

In single-cell differentiation experiments, the typical clustering workflow involves separating cells by lining them up from the experimental start point to the possible end points with "pseudotime" analysis[8]. The general idea behind this is that some cells in an experiment will receive the differentiation signal at different times due to random diffusion and, as such, will be at different points on the developmental axis even if all of the cells are collected at the same time point. Thus, this process allows the discovery of intermediate cell states between experimental time points and provides a simple classification. There is a great volume of works that apply this method to find previously unknown cell states with both scRNA-seq and scATAC-seq. A number of tools exist for data with different properties

to help with this process. A recent work[44] benchmarked 45 of these tools on real and synthetic datasets and found that no one tool worked best for all expected topologies. Instead, they published a decision tree of 4 questions to decide between their "best" algorithms of PAGA[45], Slingshot[46], STEMNET[47], SCORPIUS[9], or simply computing the angle with respect to the origin in a 2D PCA.

After experiment/cell clustering, cluster classification is the next step for analysis. For systems with well-known sub-populations with known markers, simply graphing the expression of these markers on the discovered clusters is generally enough to decipher their meaning. However, in less-studied systems, new classifications and markers need to be discovered with supervised learning, which can help determine the difference between clusters. Classically, this has been done by decision trees[13], which provide a good balance between interpretability and predictability. However, a recently-released tool, SuperCT[48], trains an expandable supervised-classifier neural network to reveal seemingly highly-accurate cluster identities. In this work, the authors show this promising method has high accuracy, robustness, compatibility, and expandability when classifying scRNA-seq data from mouse and human sources. However, one will have to rely on classical methods for other systems.

1.5 Identifying dynamic biological modules through clustering by observation

Due to the poor coverage inherent in large-scale single cell experiments, it is normal to miss detecting lowly-expressed genes or rarely-active genomic regions. Unfortunately, some of those lowly-expressed genes are important transcription factors that regulate the very cell-states being studied. This sometimes makes it impossible to determine the exact

underlying biological process that generates a population of interest. To further investigate a system of interest, one should either isolate those populations experimentally using detected cell markers and sequence them again at a higher coverage or use one of many tools available to impute the missing data. This higher coverage allows more powerful tools to cluster the data by observation into biological modules. The dynamics of these modules can be tracked over multiple experimental conditions to determine that are likely to be the biological cause of the population of interest.

There have been many released tools to mask the presence of dropout events from low-coverage sequencing through imputation. A recent work compared 8 of the these methods for scRNA-seq on several real and simulated datasets with random drop outs[49]. They concluded that no one method is the best with LLSimpute[50] performing well on homogenous cell populations, the low-rank method[51, 52] doing well on data with known cell labels, and BISCUIT[53] performing best on recovering correct cell clusterings and psuedotime structures. Each of these tools is capable of adding additional synthetic coverage to single-cell experiments, but it is not meant to be a replacement for additional experiments. Thus, it is important to only use them for data exploration and not fully trust results from further down-stream analysis.

After imputation or experiments with sufficient coverage have been performed, this data can be clustered into gene and chromatin modules through unsupervised learning for further regulatory analysis. In Chapter 2, we perform this clustering through self-organizing maps, but that option is only truly viable if you have hundreds of samples/cells. If the dataset is significantly smaller, there are several tools to choose from depending on the experiment. For time-course-based gene expression data, maSigPro[54] is a popular

choice, which uses two-regression steps to separate genes into temporally-correlated modules. For highly dimensional or single cell differentiation systems, SCORPIUS[9] can use the imputed cell trajectories to find gene modules. For other types for gene expression data, WGCNA[55] will build gene modules using correlation networks. For mining chromatin modules, ChromClust[12] can cluster regions based on <20 histone marks at a time, and cisTopic[43] was specifically designed to cluster chromatin regions in scATAC-seq data, although we show in Chapter 2 that this tool may be prone to underclustering. Finally, if no other tool exists, t-nearest neighbor[56] can be used to find modules, but this process is not nearly as effective clustering observations as it is clustering cell-states as can be seen in Chapter 2.

Determining the functional purpose of these modules can be difficult. For gene expression, there are a number of tools available to perform Gene Ontology (GO) term enrichment[57, 58]. Many genes in some model systems have been annotated with functional terms. Given a cluster of genes, it is possible to determine if some of these terms are statistically overrepresented which provides clues into the overall biological function of the cluster. There are a number of web-based options for this from DAVID[59, 60], which will also perform pathway analysis, to PANTHER[61], which provides a clearer GO enrichment tree. For clusters of chromatin regions, nearby genes can be analyzed using the GREAT[62] web-tool for GO enrichment. These clusters can also be searched for an overrepresentation of transcription factor motifs with tools such as FIMO[63]. Differences in motif enrichments can indicate a difference in regulation. Each of these techniques was used in Chapters 2 and 3.

1.6 Linking the regulation and output of biological programs through jointly analyzing RNA-seq and epigenomic data

There are many systems in which just one type of measurement does not provide a full picture. For example, transcription factor gene expression data cannot fully explain future gene expression levels due to possible epigenetic factors. Similarly, chromatin accessibility or histone mark ChIP data alone are not sufficient to determine if the transcription factors required for gene expression are present. To overcome this challenge, many works have shifted to making multiple measurements on the same set of experimental conditions, normally mRNA expression levels and some combination of epigenomics assays. In conjuction with these experimental data sets, software tools have been developed with the goal of integrating these different combinations of assays.

Computationally integrating highly-dimensional data from different sources is a difficult problem. Even aligning the same cell population over multiple scRNA-seq experiments across different conditions presents a number of challenges. The newest version of Seurat[16] and LIGER[64] attempt to combat this issue through canonical correlation analysis to find shared structures across data sets and align cell clusters in a low dimensional space. Both of these techniques were shown to properly integrate multiple scRNA-seq experiments, and in a recent pre-print[65], it was suggested that Seurat's new functionality can be used to calculate cell states through scRNA-seq analysis and anchor scATAC-seq data through converting chromatin accessibility into a "gene activity matrix" using Cicero[66]. This separation of cell identities allowed the authors identify chromatin state changes across multiple cell types in the mouse visual cortex.

Rather than performing this anchoring *in-silico*, there have been a multiple recent works that have shown that multiple assays can be done on the same cell simultaneously[67]. For example, the sciCar[68] assay is capable of measuring gene expression and chromatin accessibility in the same single cells, albeit at an extremely low coverage of ~4000 reads per cell in the RNA-seq and ~1500 reads per cell in the ATAC-seq. Measuring single cell gene expression and methylation in the same cells has been possible for a while now in multiple assays[7, 11, 69, 70]. Each of these works developed publically-available computational tools to cluster cells using multiple sets of data.

There has been several recent works on integrating multiple single cell datasets measured separately or simultaneously to find new cell types or fine sub-populations of cells. However, there has been very little in the recent literature on building un-supervised learning methods to take advantage of these multi-omic techniques from a gene/genome region perspective. In Chapter 2, we will introduce a new method of building pairwise gene-genome region sets using multiple self-organizing maps using scRNA-seq and scATAC-seq data. These region sets have very similar regulatory properties and thus have a high motif density, which vastly improves detection. We used this increased detection to build a gene regulatory network of our model system. This is followed up in Chapter 3, in which we use the same method on a highly-dimensional set of bulk RNA-seq and ChIP-seq data in *Xenopus tropicalis* mesendoderm development to build another gene regulatory network.

1.7 Future Directions and Conclusions

Sparked by the recent release of single-cell technologies and other methods that generate highly-dimensional genomic data sets inexpensively, there has been a high demand for computational methods that can analyze the results of these assays, and there has been a large wave of software tools to meet those demands with 400 developed packages for scRNA-seq alone. Additionally, with the release of more and more multi-omic data sets, integrative tools have been developed to analyze them. However, there are still many obstacles, both computational and experimental, that need to be overcome.

We are reaching a limit in our ability to mine information from low-signal-to-noise experiments such as the current generation of single-cell DNA sequencing experiments and are resorting to imputation to provide the data-set complexity required for analysis of this type. New versions of protocols such as the scATAC-seq and scChIP-seq assays to improve the capture rate would vastly improve the quality of regulatory analysis. Additionally, assays capable of measuring both chromatin accessibility and gene expression in the same cells such as sciCar have a ~10 fold reduction in reads per cell compared to performing the experiments separately, which limits the potential results drastically, so there is plenty of work to be done in this area as well.

From the computational perspective, the amount of available highly-dimensional data is growing by the day. Currently, methods like LIGER and Seurat v3 allow users to integrate data from the same cell type with different conditions. It should be possible to improve current integration methods to apply them on vast growing data sets taken from multiple labs, projects, cell types, and conditions to find patterns in regulation on a global scale. Additionally, when building these methods, it should be kept in mind that genomic

data sets are not meant to be static and thus neither should be the analysis. For example, in Chapter 4, we discuss the Living SOM, which is a system for analyzing a growing data set with a self-organizing map that uses previous analyses to improve future clusterings.

1.8 Figures

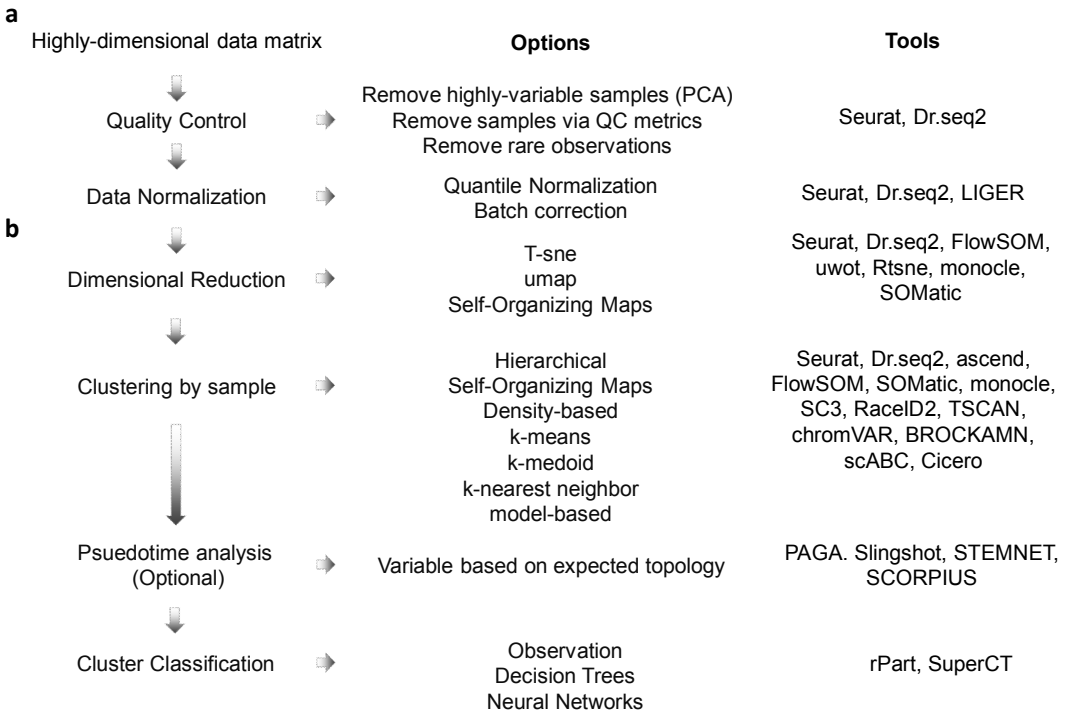


Figure 1.1. Sample analysis pipeline for initial exploratory experiments
(a) To prepare a data matrix for analysis, certain quality control measures must be followed. Following this, data should be normalized between experiments if any batch effects are present. There are several tools to help with this, with the best choices labeled. **(b)** To find new sub-populations, the improved data matrix should have its dimensionality reduced for visualization and then clustered. In differentiation experiments, psuedotime analysis can help find cells between 2 states. Finally, these clusters need to be classified to allow for further experimentation and analysis.

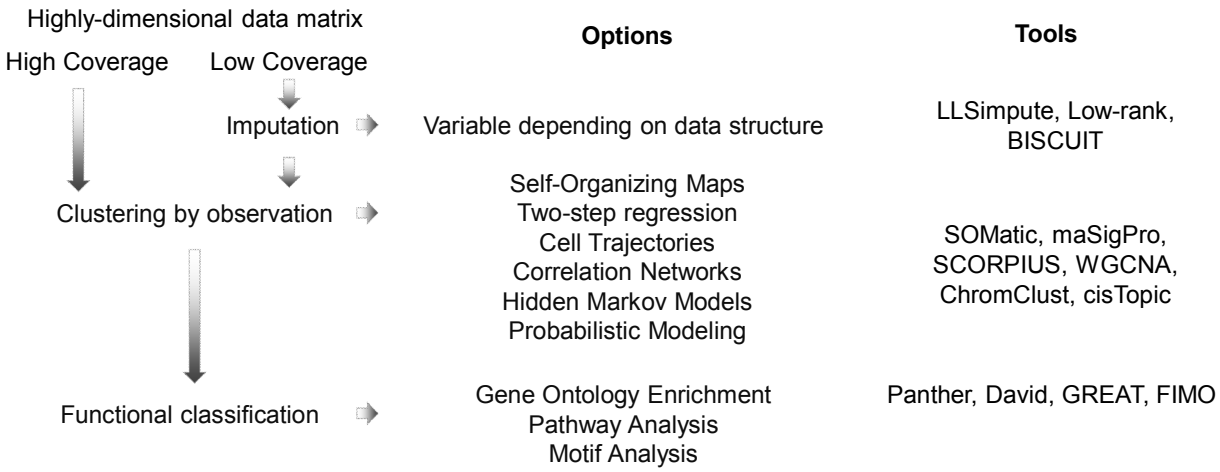


Figure 1.2. Sample analysis pipeline for identifying gene/chromatin modules
When analyzing a low coverage data set, it may be required to perform an imputation step to provide the complexity required for identifying regulatory modules. Clustering by observation requires powerful methods that typically require hyper-parameter exploration. Finally, the discovered modules need to be classified for functional significance.

1.9 References

1. Davie, K., et al., *A Single-Cell Transcriptome Atlas of the Aging Drosophila Brain.* Cell, 2018.
2. Han, X., et al., *Mapping the Mouse Cell Atlas by Microwell-Seq.* Cell, 2018. **172**(5): p. 1091-1107.
3. Consortium, T.E.P., *An integrated encyclopedia of DNA elements in the human genome. .* Nature 2012. **489**(57-74).
4. Zheng, G.X.Y., J.M. Terry, and J.H. Bielas, *Massively parallel digital transcriptional profiling of single cells.* Nature Communications, 2017. **8**.
5. L, Z., P. B, and O. A., *Exploring the single-cell RNA-seq analysis landscape with the scRNA-tools database.* PLOS Computational Biology, 2018.
6. Dijk, D., et al., *Recovering Gene Interactions from Single-Cell Data Using Data Diffusion.* Cell, 2018. **174**(3): p. 716-729.
7. Angermueller, C., et al., *Parallel single-cell sequencing links transcriptional and epigenetic heterogeneity.* Nature Methods, 2016. **13**: p. 229-232.
8. Trapnell, C., *Defining cell types and states with single-cell genomics.* Genome Reseach, 2015. **25**: p. 1491-1498.
9. Cannoodt, R., et al., *SCORPIUS improves trajectory inference and identifies novel modules in dendritic cell development.* bioRxiv, 2016.
10. Ramirez, R.N., et al., *Dynamic Gene Regulatory Networks of Human Myeloid Differentiation.* Cell Systems, 2017. **4**: p. 416-429.
11. Cheow, L.F., et al., *Single-cell multimodal profiling reveals cellular epigenetic heterogeneity.* Nature Methods, 2016. **13**: p. 833-836.
12. Noureen, N., et al., *ChromClust: A semi-supervised chromatin clustering toolkit for mining histone modifications interplay.* Genomics, 2015. **106**(6): p. 355-359.
13. Jiang, S. and A. Mortazavi, *Integrating ChIP-seq with other functional genomics data.* Briefings in Functional Genomics, 2018. **17**(2): p. 104-115.
14. S, H., et al., *Simple Combinations of Lineage-Determining Transcription Factors Prime cis-Regulatory Elements Required for Macrophage and B Cell Identities.* Molecular and Cellular Biology, 2010. **38**(4): p. 576-589.
15. Zhao, C., et al., *Dr.seq2: A quality control and analysis pipeline for parallel single cell transcriptome and epigenome data.* PLOS One, 2017. **12**(7).
16. Butler, A., et al., *Integrating single-cell transcriptomic data across different conditions, technologies, and species.* nature Biotechnology, 2018. **36**: p. 411-420.
17. Pearson, K., *On Lines and Planes of Closest Fit to Systems of Points in Space.* Philosophical Magazine, 1901. **2**(11): p. 559-572.
18. van der Maaten, L.J.P. and G.E. Hinton, *Visualizing Data Using t-SNE.* Journal of Machine Learning Research, 2008. **9**: p. 2579-2605.
19. Ilicic, T., et al., *Classification of low quality cells from single-cell RNA-seq data.* Genome Biol, 2016. **17**(29).
20. Landt, S.G., et al., *ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia.* Genome Reseach, 2012. **22**: p. 1813-1831.

21. Bolstad, B.M., et al., *A comparison of normalization methods for high density oligonucleotide array data based on variance and bias.* Bioinformatics, 2003. **19**(2): p. 185-193.
22. Qiu, X., H. Wu, and R. Hu, *The impact of quantile and rank normalization procedures on the testing power of gene differential expression analysis.* BMC Bioinformatics, 2013. **14**(124).
23. McInnes, L., J. Healy, and J. Melville, *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.* arXiv, 2018.
24. Schubert, E. and M. Gertz, *Intrinsic t-Stochastic Neighbor Embedding for Visualization and Outlier Detection.* In: Beecks C., Borutta F., Kröger P., Seidl T. (eds) Similarity Search and Applications. SISAP 2017. Lecture Notes in Computer Science, vol 10609. Springer, Cham, 2017.
25. Pezzotti, N., et al., *Approximated and User Steerable tSNE for Progressive Visual Analytics.* IEEE Transactions on Visualization and Computer Graphics, 2017. **23**(7): p. 1739-1752.
26. Duò, A., M.D. Robinson, and C. Soneson, *A systematic performance evaluation of clustering methods for single-cell RNA-seq data.* F1000Res, 2018. **7**: p. 1141.
27. Senabouth, A., et al., *ascend: R package for analysis of single cell RNA-seq data.* BioRxiv, 2017.
28. Lin, P., M. Troup, and J.W.K. Ho, *CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data.* Genome Biology, 2017. **18**(59).
29. Ward JH, J., *Hierarchical grouping to optimize an objective function.* J Am Stat Assoc, 1963. **58**(301): p. 236–244.
30. S, V.G., C. B, and V.H. MJ, *Flowsom: Using self-organizing maps for visualization and interpretation of cytometry data.* Cytometry A, 2015. **87**(7): p. 636–645.
31. Trapnell, C., et al., *The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. .* Nature Biotechnology 2014. **32**: p. 381-386.
32. JA, H. and W. MA, *Algorithm as-136: A k-means clustering algorithm.* J R Stat Soc Ser C Appl Stat, 1979. **28**(1): p. 100–108.
33. J, Ž. and Y. C, *pcaReduce: hierarchical clustering of single cell transcriptional profiles.* BMC Bioinformatics, 2016. **17**(1): p. 140.
34. L, V.D.M., *Accelerating t-SNE using tree-based algorithms.* J Mach Learn Res, 2014. **15**(1): p. 21.
35. VY, K., K. K, and S. MT, *SC3: consensus clustering of single-cell RNA-seq data.* Nat Methods, 2017. **14**(5): p. 483–486.
36. D, G., M. MJ, and B. JC, *De Novo Prediction of Stem Cell Identity using Single-Cell Transcriptome Data.* Cell Stem Cell, 2016. **19**(2): p. 266–277.
37. Z, J. and J. H, *TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis.* Nucleic Acids Research, 2016. **44**(13): p. e117.
38. Schep, A.N., et al., *chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data.* Nature Methods, 2017. **14**: p. 975–978.
39. Boer, C.G.d. and A. Regev, *BROCKMAN: deciphering variance in epigenomic regulators by k-mer factorization.* BMC Bioinformatics, 2018. **19**(253).

40. Zamanighomi, M., et al., *Unsupervised clustering and epigenetic classification of single cells.* Nature Communications, 2018. **9**.
41. Pliner, H., et al., *Chromatin accessibility dynamics of myogenesis at single cell resolution.* bioRxiv, 2017.
42. Buenrostro, J.D., et al., *Integrated Single-Cell Analysis Maps the Continuous Regulatory Landscape of Human Hematopoietic Differentiation.* Cell, 2018. **173**(6): p. 1535-1548.
43. González-Blas, C.B., et al., *cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data.* Nature Methods, 2019. **16**: p. 397-400.
44. Saelens, W., et al., *A comparison of single-cell trajectory inference methods.* Nature Biotechnology, 2019. **37**: p. 547-554.
45. Wolf, F.A., et al., *PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells.* Genome Biol, 2019. **20**: p. 59.
46. Street, K., et al., *Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics.* BMC Genomics, 2018. **19**: p. 477.
47. L, V., et al., *Human haematopoietic stem cell lineage commitment is a continuous process.* Nature Cell Biology, 2017. **19**(4): p. 271-281.
48. Xie, P., et al., *SuperCT: a supervised-learning framework for enhanced characterization of single-cell transcriptomic profiles.* Nucleic Acids Research, 2019. **47**(8): p. e48.
49. L, Z. and Z. S, *Comparison of computational methods for imputing single-cell RNA-sequencing data.* IEEE/ACM Trans Comput Biol Bioinform, 2018.
50. H, K., G. GH, and P. H., *Missing value estimation for DNA microarray gene expression data: local least squares imputation.* Bioinformatics, 2005. **22**(11): p. 1410-1411.
51. Chen, C., B. He, and X. Yuan, *Matrix completion via an alternating direction method.* IMA J. Numer. Anal., 2012. **32**: p. 227-245.
52. Zhu, L., et al., *A unified statistical framework for single cell and bulk rna sequencing data.* bioRxiv, 2017.
53. Azizi, E., et al., *BAYESIAN INFERENCE FOR SINGLE-CELL CLUSTERING AND IMPUTING.* Genomics and Computational Biology, 2017. **3**(1).
54. A, C., et al., *maSigPro: a method to identify significantly differential expression profiles in time-course microarray experiments.* Bioinformatics, 2006. **22**(9): p. 1096-1102.
55. Langfelder, P. and S. Horvath, *WGCNA: an R package for weighted correlation network analysis.* BMC Bioinformatics, 2008. **9**(559).
56. Altman, N.S., *An introduction to kernel and nearest-neighbor nonparametric regression.* The American Statistician, 1992. **46**(3): p. 175-185.
57. al., A.e., *Gene ontology: tool for the unification of biology.* Nat Genet, 2000. **25**(1): p. 25-29.
58. Consortium, T.G.O., *The Gene Ontology Resource: 20 years and still GOing strong.* Nucleic Acids Research, 2019. **47**(D1): p. D330-D338.
59. DW, H., S. BT, and L. RA, *Systematic and integrative analysis of large gene lists using DAVID Bioinformatics Resources.* Nature Protoc, 2009. **4**(1): p. 44-57.
60. DW, H., S. BT, and L. RA, *Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists.* Nucleic Acids Research, 2009. **37**(1): p. 1-13.

61. H, M., et al., *Large-scale gene function analysis with the PANTHER classification system.* Nat Protoc, 2013. **8**(8): p. 1551-1566.
62. McLean, C.Y., et al., *GREAT improves functional interpretation of cis-regulatory regions.* Nature Biotechnology, 2010. **28**: p. 495-501.
63. Grant, C.E., T.L. Bailey, and W.S. Noble, *FIMO: scanning for occurrences of a given motif.* Bioinformatics (Oxford, England), 2011. **27**(1017-1018).
64. Welch, J., et al., *Integrative inference of brain cell similarities and differences from single-cell genomics.* bioRxiv, 2018.
65. Stuart, T., et al., *Comprehensive integration of single cell data.* bioRxiv, 2018.
66. Cusanovich, D.A., et al., *A Single-Cell Atlas of In Vivo Mammalian Chromatin Accessibility.* Cell, 2018. **174**(5): p. 1309-1324.
67. Colomé-Tatché, M. and F.J.Theis, *Statistical single cell multi-omics integration.* Current Opinion in Systems Biology, 2018. **7**: p. 54-59.
68. Cao, J., et al., *Joint profiling of chromatin accessibility and gene expression in thousands of single cells.* Science, 2018. **361**(6409): p. 1380-1385.
69. Hou, Y., et al., *Single-cell triple omics sequencing reveals genetic, epigenetic, and transcriptomic heterogeneity in hepatocellular carcinomas.* Cell Research, 2016. **26**: p. 304-319.
70. Hu, Y., et al., *Simultaneous profiling of transcriptome and DNA methylome from a single cell.* Genome Biology, 2016. **17**: p. 88.

CHAPTER 2

Building gene regulatory networks from scATAC-seq and scRNA-seq using Linked Self-Organizing Maps

Note: (1) Dr Ricardo Ramirez and I equally contributed to the material in this chapter. He built the single cell and bulk libraries used in this work and provided suggestions on interpreting the results.

Chapter 2

Building gene regulatory networks from scATAC-seq and scRNA-seq using Linked Self-Organizing Maps

2.1 Abstract

Rapid advances in single-cell assays have outpaced methods for analysis of those data types. Different single-cell assays show extensive variation in sensitivity and signal to noise levels. In particular, scATAC-seq generates extremely sparse and noisy datasets. Existing methods developed to analyze this data require cells amenable to pseudo-time analysis or require datasets with drastically different cell-types. We describe a novel approach using self-organizing maps (SOM) to link scATAC-seq genes with scRNA-seq regions that overcomes these challenges and can generate draft regulatory networks. Our SOMatic package generates chromatin and gene expression SOMs separately and combines them using a linking function. We applied SOMatic on a mouse pre-B cell differentiation time-course using controlled Ikaros over-expression to recover gene ontology enrichments, identify motifs in genomic regions showing similar single-cell profiles, and generate a gene regulatory network that both recovers known interactions and predicts new Ikaros targets during the differentiation process. The ability of linked SOMs to detect emergent properties from multiple types of highly-dimensional genomic data with very different signal properties opens new avenues for integrative analysis of heterogenous data.

2.2 Introduction

The ability to analyze hundreds to thousands of individual cells using new functional sequencing assays has revolutionized the current state of scientific and biomedical research[1]. For example, single-cell gene expression studies have allowed the identification of rare cell populations in a variety of samples ranging from immune cell systems[2] to circulating tumor cells[3]. Comprehensive atlases of gene expression are being built for tissues such as the Drosophila brain throughout its lifespan[4] to an entire mouse[5]. Inspired by the wealth of new insights from single-cell RNA-seq, there has been a plethora of single cell genomic technologies developed in the last few years[6]. For example, single-cell profiling of chromatin accessibility[7-9] has generated a lot of excitement because of the wealth of insights generated within large scale surveys of chromatin accessibility and gene regulation through projects like ENCODE[10].

However, unlike single-cell RNA-seq, chromatin accessibility mapping from individual cells yields sparse information of the open chromatin landscape[11, 12] due to the intrinsic limitation of numbers of chromosomes per nucleus. It has been difficult for previous analysis platforms to handle the sparsity and noise inherent in data of this type.

Recently, a number of tools have been developed to try and combat this issue. chromVAR[13] uses cells with the highest proportion of reads to build a model of the expected number of fragments per total reads for every respective motif site in the genome, and computes deviation scores from this model to cluster single-cells. This method, while effective, requires the generation of a list of transcription factor binding sites through mass motif scanning which, in this work, necessitated the loosening of strict Type I error control and the creation of a custom, well-curated list of transcription factor motifs. Another

application, scABC[13], manages to cluster cells of different cell-types well by using the total cell accessibility signal to provide weights to an unsupervised clustering of the cells using K-medioids and thus identifies landmark regions that are only open in each found population. The cells are then re-clustered using the respective landmarks. However, this technique would likely become confused by time course data from the same cell-type as it may be too similar to generate proper landmarks. BROCKMAN[14] uses gapped 8-mer factorization to calculate variation in DNA sequences in reads across scATAC-seq experiments and can separate cell types across multiple scATAC experiments to determine TF activity through mapping of known TF motifs with gapped k-mers. Unfortunately some TF motifs such as Ikaros (which has only a 5bp motif) are difficult to map properly with a gapped 8-mer. Others, such as Cis-topic[15], were designed to cluster scATAC-seq data alone, but have not been shown to work on multiple data types simultaneously or are only capable of clustering the cells in these experiments, such as latent semantic indexing[16, 17].

Additional recent techniques attempt to correct for the scarcity of scATAC-seq data by leveraging imputed pseudo-time orderings[18]. For example, Cicero[19] uses the ordering of cells to make small aggregate pools before computing correlations. Alternatively in a study of human hematopoietic cell differentiation, Buenrostro and colleagues[20] also assigned pseudotime ordering so that accessibility peaks could be smoothed by a lowess function. Both of these methods make extensive use of pseudotime orderings, and thus, require systems that have a strong differentiation lineage (with preferably known markers). Here we introduce a method for jointly analyzing scRNA-seq and scATAC-seq data that cannot be ordered by pseudotime by taking a "gene/region-centric" approach using self-organizing maps.

Self-organizing maps (SOMs) are a type of artificial neural network, also referred to as a Kohonen network[21, 22](Fig. 5). SOMs are trained using unsupervised learning to generate a low-dimensional representation of data and can be visualized using two-dimensional maps. Individual SOM nodes (or neurons) have a weight vector that is in the same dimension as the input data vectors and neighboring nodes on a SOM reflect similarities across the input data space vector. Additionally, SOMs have been known to provide extremely robust clusterings, with typical Rand indexes in the 99.9% range at the unit level and 95% range for continuity-constrained metaclustering[23]. Thus, trained SOMs provide an intuitive platform for identifying clusters in high-dimensional datasets. For example, SOMs trained on gene expression data or chromatin data[24] from multiple cell types in human and mouse have identified complex relationships across high-dimensional genomic data[10, 25-27]. Additionally, SOMs have been used to structure and interrogate the transcriptome in single-cells during cellular reprogramming[28]. SOMs provide a natural visual and powerful platform for the analysis and integration of high-dimensional data of different types.

As part of our work in the STATegra consortium (STATegra.eu), we performed single-cell RNA-seq and single-cell ATAC-seq using a mouse pre-B cell model system[29] during cellular differentiation. This system provides a high-resolution view into a narrow transition in pre-B cell development, whereby we induce cell differentiation in response to a sudden doubling of Ikaros expression. Our data only contains two time points and represents a fairly drastic change in chromatin accessibility and gene expression over that period, and thus, would be a poor candidate for pseudo-time analysis. In addition, this data is sufficiently sparse and noisy to give even powerful algorithms like UMAP[30] difficulty

from a gene or genome region perspective (Fig. 6, 7) even if the data can be visualized and clustered from a cell perspective (Fig. 8).

We used SOMatic to create two SOMs in order to identify significant groups of expressed genes and chromatin elements that jointly change during the time course. The two SOMs were then linked using a novel algorithm to find metaclusters of genes and associated genomic regions that show similar profiles during pre-B cell differentiation. The regulatory regions in these clusters were mined for enriched motifs that allowed us to infer a predicted regulatory network downstream of Ikaros. Our flexible and comprehensive approach is first of its kind to provide an analysis platform that combines these, scRNA-seq and scATAC-seq, single-cell data types without leveraging cell ordering and effectively identifies regulatory programs.

2.3 Results

2.3.1 Integration of single-cell data types using SOM

In order to study changes in gene expression and chromatin accessibility for single-cells, we utilized an inducible pre-B model system[29] and performed single-cell RNA-seq and single-cell ATAC-seq before and after cellular differentiation (Experimental methods). The goal was to link the data from these methods in a meaningful way to study individual genome region/gene interactions, and this was accomplished by developing the computational pipeline shown in Figure 1. We began by training separate self-organizing maps (SOMs) for each dataset. The result is a set of SOM units that contain genes and genome regions that have a very similar signal profile across each of the single cells at both time points (Summary maps in Fig. 9). To reduce the signal dropout and technical noise

prevalent in single cell data, our SOM analysis tool produces clusters of these units, called metaclusters[24], which maintain the SOM's scaffold topology by only combining adjacent units and contain similar gene expression and chromatin accessibility profiles. Finally we combine the patterns found in each SOM using a pipeline that links metaclusters from both gene expression and chromatin accessibility. These linked metaclusters (LM) contain sets of chromatin regions that have similar open chromatin signal profiles that are in the proximity of genes that also share a similar profile (although not necessarily the same profile in RNA and ATAC) and can be mined using gene ontology, pathway analysis, and motif discovery. Our method easily extends a traditional single data-type analysis to one that focuses on the integration of fundamentally different data like single-cell RNA-seq and ATAC-seq in order to recover evidence of co-regulation.

2.3.2 Identification of dynamic gene expression metaclusters

We trained a 40x60 SOM on the scRNA-seq dataset (62 single-cells for 0-hour; 66 single-cells for 24-hour) using 12,380 genes that had expression greater than 1 FPKM in at least 5% of cells (Experimental methods). As expected, slices of this map (Fig. 2a), which correspond to single cells, show a general reduction of gene expression over time. SOMatic identified 39 RNA metaclusters that reflect the various gene expression profiles present in the data (Fig. 2b). We validated that these metaclusters were properly determined by calculating the UMatrix and density map for this SOM and overlaying the metacluster boundaries on top of these maps (Fig. 10) for visual inspection. The metaclusters followed the breaks in these maps as expected and thus provide a robust representation of the different profiles present in the single-cell data.

One of the strengths of the SOM approach is that we can perform logical operations on the feature maps. We computed a map by averaging maps from the cells in each time point and subtracting them to determine which metaclusters reflect meaningful gene expression differences across time (Fig. 2c). We performed a correlation analysis to determine which metaclusters were consistently enriched across the cells in each time-point as previously described[24]. We found statistically-significant differences across time in 9 RNA metaclusters, 5 of which were enriched in 0-hour and 4 in 24-hour (Fig. 2d, p-value $<10^{-4}$-10^{-10}). Sizes for these metaclusters can be found in Fig 2d. For example, RNA metacluster 15 consists of 11 SOM units and contains 19 genes enriched in 0-hour single-cells such as *Igll1* and *Vpreb1* (Fig. 2e). Similarly, metacluster 16 consists of 42 units and contains 151 genes enriched in 24-hour cells such as *Mier1*, which has been shown to control mature B-cell survival in mice[31] (Fig. 2e). Gene ontology analysis revealed a series of genes enriched for antigen presentation and negative regulation of cell cyle in 24-hour cells, while DNA replication genes were represented in 0-hour cells (Fig. 2f). This is consistent with the transition of gene programs necessary for coordinating pre-B cell differentiation[32].

2.3.3 Mapping the pre-B single-cell chromatin landscape architecture using SOMs

We performed single-cell ATAC-seq[8] with a total of 227 cells passing our quality controls to explore the change in chromatin accessibility over the differentiation time-course. We recovered on average 53,864 unique chromatin fragments per cell. Using peaks taken from a set of pooled ATAC-seq experiments over three biological replicates with 50,000 cells for each time-point, we quantified the ATAC-seq signal in these peaks for each

cell. We built a data matrix from chromatin regions detected in at least 2% of cells (5 cells) for a total 25,466 ATAC-seq peaks due to the sparse nature of single-cell ATAC-seq.

A 40x60 SOM was trained on this scATAC data matrix (Experimental methods). Similar to the RNA SOM, scATAC feature maps (Fig. 3a) revealed a general closing of the chromatin in 24-hour cells, which is normal for cells undergoing differentiation. Clustering the units from this SOM resulted in the identification of 107 chromatin metaclusters (Fig. 3b). Visual inspection of these clusters confirmed that these clusters properly follow the breaks in the UMatrix and density map (Fig. 11).

A SOM difference map and hypothesis analysis for all 107 chromatin metaclusters revealed 48 metaclusters that exhibit open chromatin signal in 0-hour cells and 3 metaclusters in with higher signal in the 24-hour cells (Fig. 3c-d). Gene ontology enrichments for genes in the vicinity of the regions from two of the most significant metaclusters (Fig 3e), 62 (0-hour enriched; 191 peaks) and 70 (24-hour enriched; 160 peaks), reveal that these genes are enriched for cell cycle and cell division programs as predicted (Fig. 3f). Thus, SOMs are capable of revealing patterns of chromatin accessibility from sparse single-cell ATAC-seq data in a dynamic model system.

2.3.4 Comparison of chromatin SOM results to cisTopic clustering

In order to compare the performance of our SOM clustering on the scATAC-seq data, we also analyzed that dataset using cisTopic[15](Experimental Methods), which determined that there were only 15 region clusters ("factors", Fig. 12a). Umaps built using these 15 factors clustered the cells from this experiment into coherent groups (Fig. 12b) with several factors (3, 6, 8, 13) being enriched in one timepoint over the other. However,

GREAT analysis of these factors did not reveal any significant GO terms that were biologically relevant, which may be due to the large sizes of these topics (Fig. 12c). Additionally, visual inspection of the Igll1 locus (Fig. 12d) showed that the promoters of Igll1 and Vpreb1 and one of the nearby enhancers with extremely different scATAC-seq profiles were all assigned to the same topic. Thus, while cisTopic performs well at separating and clustering cells from scATAC-seq experiments, it is unable to get a high-resolution view of the inner-time-point dynamics at the same granularity as the SOM.

2.3.5 Application of multi-omic single-cell data integration using Linked SOMs

Cellular differentiation occurs as a consequence of dynamics in expression of networks of genes controlled by cis-regulatory elements, which must be open in order to function properly. The linker pipeline within SOMatic attempts to convolve the metaclusters from RNA and chromatin accessibility SOMs in order to interrogate the dynamics of the system. In brief, the pipeline subsets chromatin regions within the same chromatin metacluster into linked metaclusters (LM) using the expression of the gene whose regulatory region (using the same algorithm as GREAT[33]) overlaps the element. Thus, if a set of regions are in a LM, these regions share a similar chromatin accessibility profile and are in the vicinity of genes that also share a similar gene expression profile (See Fig. 13 for an overview). This coherence of joint profiles gives a much higher expectation that these regions will be similarly regulated than grouping on accessibility or gene expression alone.

We applied this new pipeline to our scRNA and scATAC SOMs and analyzed a total of 107 x 39 = 4,173 LMs to identify 459 LMs that were significantly dynamic in both

chromatin accessibility and their nearby genes (Experimental methods) (Fig 4a). Based on our assumption that these LMs were similarly regulated, we mined each LM separately for known transcription factor binding site motifs using FIMO with a q-value cutoff of .05. This generated ~9.3 million candidate motifs, which is substantially more than results from motif analysis on bulk data with less than 50k and 500k for peaks and enriched peaks respectively (Fig. 14a) and is greater than the ~4.4 million using the ATAC-seq SOM on its own (Fig. 14b). Random LMs also gave us fewer candidate motifs, with an average of ~1.46 million motif positions in 100 trials (Fig. 15). Additionally, to determine enrichment, LMs with a percentage of regions containing each transcription factor motif that was significantly (p-value < .05) enriched over the baseline were reported, (Fig. 4b), reducing the ~9.3 million candidate motifs to 265,715 high-confidence potential gene regulatory network connections or 5,268 high-confidence active transcription factor/active transcription factor connections.

The differentiation of the B3 cell line is initiated by doubling the amount of Ikaros in the nucleus of each cell and we therefore focused our analysis on Ikaros as the root node of our gene regulatory network. A majority of the differential LMs contain the Ikaros motif (3,672 total instances compared to an average of 1,232 instances in shuffled clusters), including 35 where Ikaros reaches statistical significance (compared to none in the shuffled clusters). In total, we found 307 genes, with 328 nearby potential cis-regulatory regions that contain the motif, that may be regulated directly by Ikaros (Fig. 4c), including genes known to be differentially expressed in this system, such as *Igll1* (Fig. 16) and *Vpreb2*[34] as well as the transcription factor *Nr3c1*[35], which is a factor that has been previously implicated as being downstream of Ikaros. To validate these connections, *Ikzf1* ChIP data[32]

was interrogated at the same 0hr and 24hr time points for each of the 328 potential cis-regulatory regions. Of these, 312 (~95%) of these regions overlap *Ikzf1* ChIP peaks in one or both of the time points, including 84 (~26%) that overlap *Ikzf1* ChIP peaks in only one time point. Loci for the 3 transcription factors predicted to be regulated by Ikaros were further visually inspected and each of the nearby potential cis-regulatory regions had a significant change over the time course (Fig. 17-19).

We built a gene regulatory network of transcription factors that we predicted were connected to Ikaros to identify indirect, secondary changes to gene expression as a direct result of changes in Ikaros concentration at the direct targets TFs. This network is tied directly to the model system in that it only uses genome segments that are open in either time-point. We determined which factors downstream of Ikaros showed a significant change in expression across the time-series (Fig. 4d) and determined the connections between them (Fig. 4e). Each of these genes has been shown to be important in B-cell differentiation. For example, the activation of *Hbp1* has been shown to prevent c-Myc-mediated transcription[36] and, together with a down-regulation of *Myc* expression, stops B-cell proliferation. The temporal enrichment of predicted targets downstream of *Myc* can be found in Fig. 20.

About 16% of connections in this network have been previously described[35, 37-40], which include *Mef2c* to Ikaros[41] and *Pax5* and *Myc*'s negative feedback loop[42, 43], or have been previously computationally predicted[44-46](52%), and we identify 20 new connections like *Rreb1* to *Myc* (Fig. 21). The identification of both direct and indirect regulation from a sudden doubling of Ikaros demonstrates the power of the Linked SOMs for analyzing highly-dimensional multi-omics data.

2.3.6 Scalability of the Linked SOM for larger datasets

In order to demonstrate the scalability of our approach to larger datasets, we applied the Linked SOM to the recently published sciCar dataset that measured chromatin accessibility and gene expression in the same DEX-treated A549 single cells[47]. This dataset features ~6,000 cells for each experiment with an average of 100,000 reads per cell for the scRNA-seq and 55,000 reads per cell for the scATAC-seq (each of which are an order of magnitude lower than the pre-B cell data above). We built a training matrix of 3,234 cells that passed our filters in both datasets and applied the Linked SOM methodology. (Experimental Methods) We found that the individual RNA and ATAC SOMs called a similar number of DE genes and DA regions to those found in the publication (Fig. 22a-c). After SOM linking, we measured the correlations on the average signal across each timepoint (0h, 1h, and 3h) in both experiments in each LM and compared the distribution to correlations in the differential LMs and correlations when the timepoints for the cells are randomly scrambled. (Fig. 22d). We found that the differential LMs have a lower density of combinations with no correlation and are more skewed towards the positive end, and both distributions are significantly (pvalue < .003) different in the scrambled dataset. To explore this further, we computed a heatmap of the correlations for the differential LMs (Fig. 21e). Investigating the contents of the positively correlated differential LMs revealed the promoter-gene connections for genes known to be targeted by GR activation such as Ckb, Per1, Nfkbia, Cdh16, and Scnn1a. Additionally, after motif analysis in those LMs, we recovered the motif of Nr3c1 (the GR receptor) in the promoter of each of the above genes (Fig. 21f). These results show that Linked SOMs are capable of analyzing data from larger

single-cell experiments with fewer reads per cell and can recover biological insights by leveraging separate measurements of RNA expression and chromatin accessibility without leveraging the same cell measurement of sciCar while demonstrating similar results.

2.4 Discussion

In this work, we used a gene- and chromatin-centric analysis using SOMs on a mouse pre-B time-course data of single-cell RNA-seq and ATAC-seq separately and, then, convolved them to find synergistic effects. Combining the metaclusters from multiple SOMs as a pair-wise set generates a data-space that combines the properties from both without any assumptions about how the data relates to each-other. Due to the inheritance of each SOM's properties, the linked metaclusters (LMs) contain genome regions that should be similarly regulated: not only is the chromatin accessibility of those regions similar across the cells, but the nearby genes they regulate share expression patterns. Thus, these LMs can be mined for motif enrichment and return a higher number of significant motif sites than simply dividing the data set randomly or by signal changes in either data set separately.

We used this SOM linking technique to explore the regulatory control of the lymphoid regulator *Ikzf1* during one step of B-cell development. 35 LMs enriched in the *Ikzf1* motif contained regions that had similarly-differential chromatin accessibility between time points and had had differentially expressed genes. Our analysis successfully recovers known biology about *Ikzf1* regulation on target genes *Igll1*, *Vpreb2*, and *Nr3c1* and novel regulatory information through discovery of possible downstream mechanisms for

B-cell activation. Following the interactions around the network provides many exciting, new avenues for research.

It is important to note, however, that these predicted regulatory connections use an extremely stringent statistical cutoff to be as confident as possible, and thus, do not recover some of the linkages predicted based on *Ikzf1* ChIP data[32] such as Ikaros's involvement in the regulation of *Myc* and *Foxo1*. While we do detect these connections at an early portion of the pipeline, the genome sequence in those regulatory regions are too different from the canonical motif to pass our stringent filters. *Foxo1* had an Ikaros motif in an open chromatin region near its transcription start site, but the motif only had a q-value of 0.762, which was above the threshold.

Our approach for combining multi-omic data through linked SOMs is amenable to integrating other single-cell technologies for the purpose of multi-omic data analysis as long as a linking function can be found. For example, the profiling of small RNAs, such as miRNAs[48], in single cells could be linked with a standard scRNA-seq experiment through the use of target prediction algorithms. The hypothetical LMs in that case would include groups of miRNAs with similar expression patterns such that their target RNA also has similar expression patterns. Following identification of these groups, functional analysis could be done on each group target RNAs and these functions could be passed back to the miRNA in the group. This is just one example of an exciting experimental and computational design that linked SOMs enable.

The ability to perform multi-omic experiments from a single-cell is now achievable for several biochemical and genomic platforms[49-52] with more being developed every day.

We foresee the ability to connect the patterns in multi-omic data using algorithms like linked SOMs to be integral in using this new technology to the fullest.

2.5 Figures

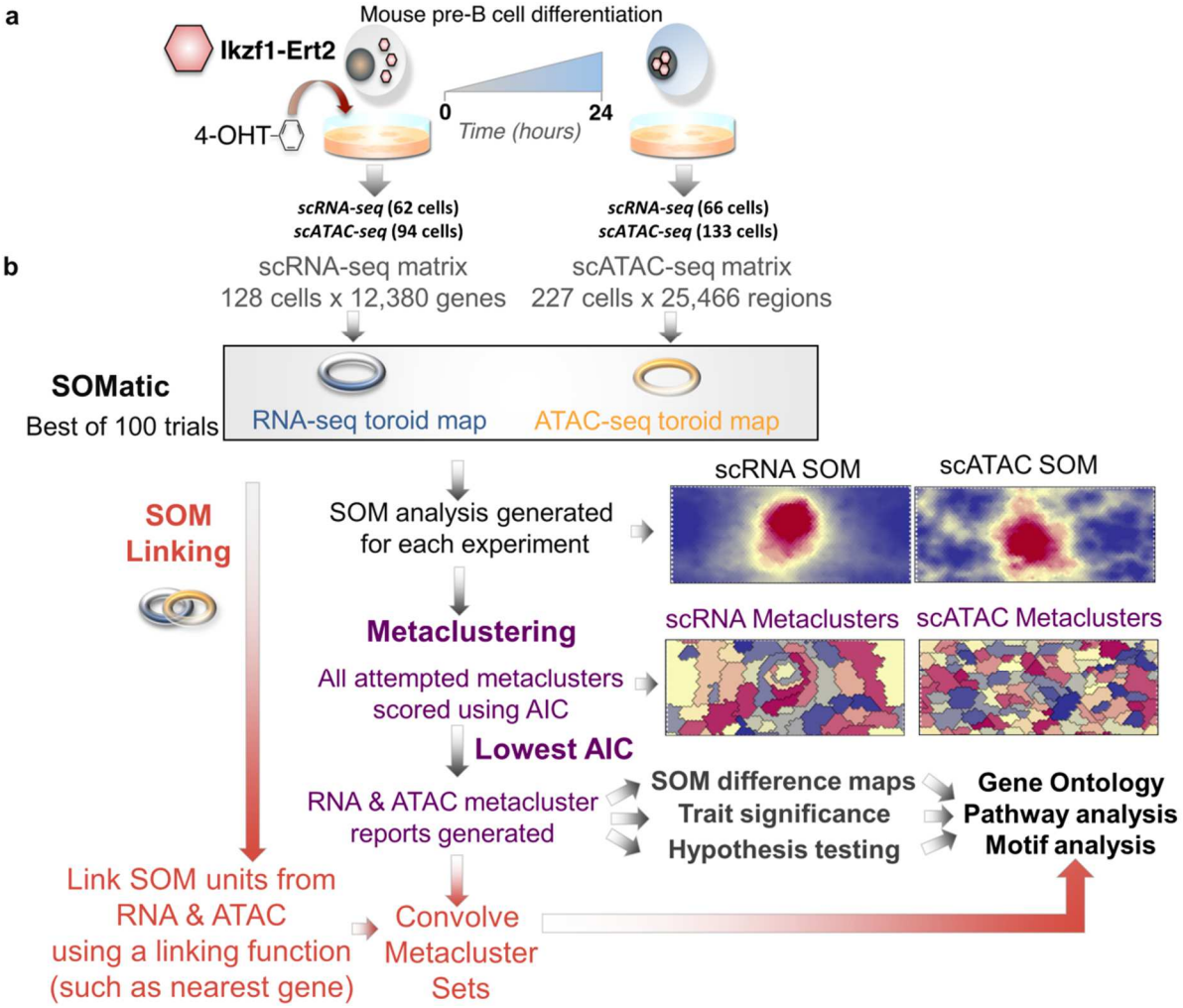


Figure 2.1. Single-cell multi-data integration using SOMs
(a) An inducible Ikzf1 mouse pre-B cell-line was used to track changes in gene expression and chromatin accessibility during differentiation (0 and 24-hours) in single-cells. (b) Single-cell RNA-seq and ATAC-seq data from an inducible mouse pre-B cell-line were independently trained using SOMatic to generate single-cell SOMs and metaclustered using AIC scoring. These clusters were convolved with the new SOM fusion algorithm to generate pair-wise metaclusters of chromatin regions with similar profiles across the single-cell dataset that regulate genes that also share similar profiles. These pair-wise clusters were mined for regulatory connections through motif enrichment analysis.

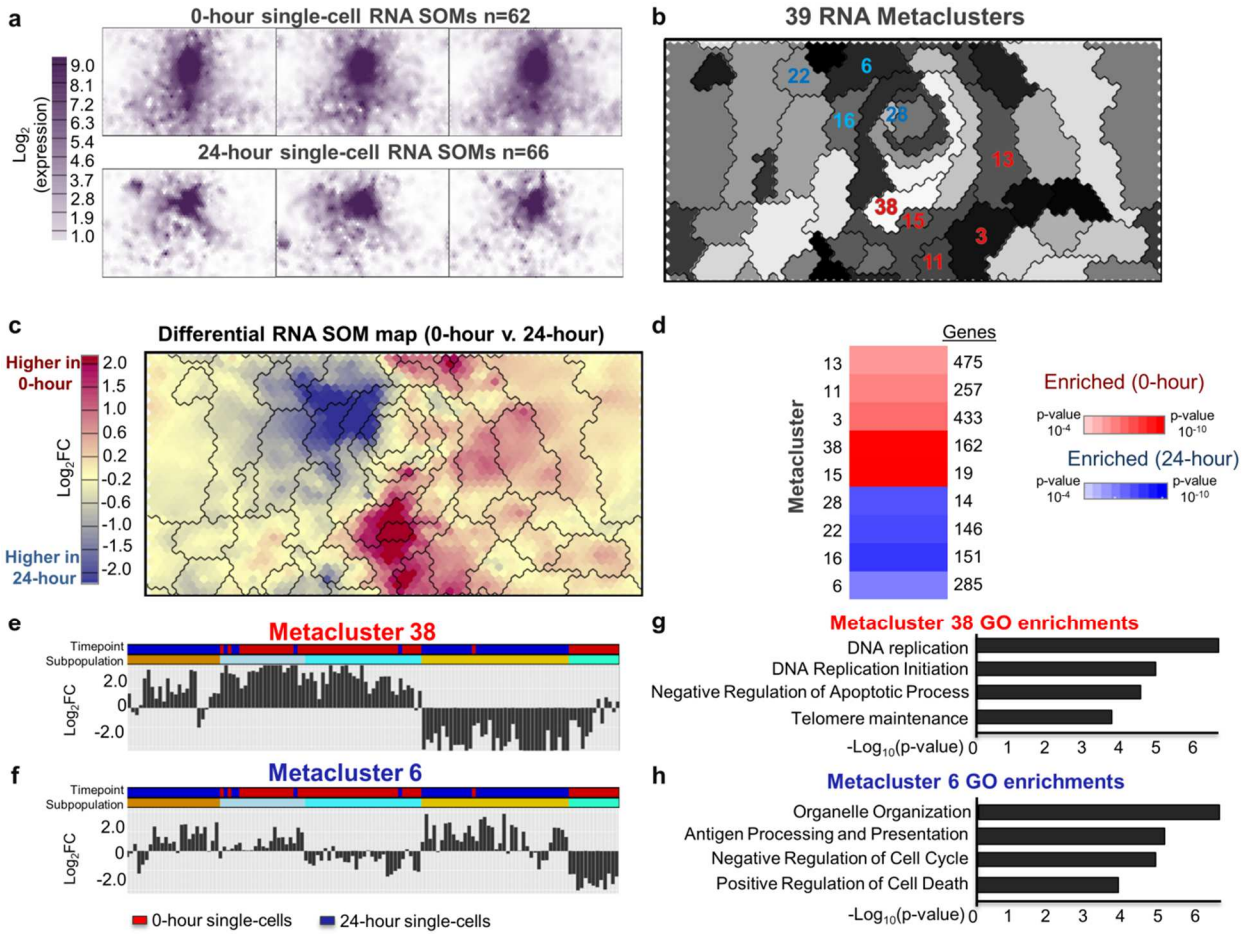


Figure 2.2. Single-cell gene expression patterns during cellular differentiation are profiled using SOMatic

(a) A SOM was generated for the single-cell RNA-seq dataset (0-hour 62 cells, 24-hour 66 cells). Maps for 3 cells from each time point were arbitrarily selected for display. (b) 39 metaclusters were identified using AIC scoring. Metacluster number and color were arbitrarily assigned for visualization purposes. (c) SOM difference map comparing 0-hour and 24-hour time-points. Maps for cells from 0 and 24-hour timepoints were averaged to generate a single map for each and then subtracted to create a map that represented gene expression fold change during pre-B cell development. Overlaid metacluster divisions generally follow contours of the map. (d) Trait enrichment analysis deployed on gene metaclusters revealed which are enriched in each time point. Metaclusters of interest are highlighted in panel b. (e-f) Summary showing the representative expression profile for metaclusters 38 and 6. Columns are individual cells color-coded for 0 and 24-hour time-points ordered by hierarchical clustering on every metacluster representative gene expression profile. Cell subpopulations are represented by a 40% cut on that clustering. (f-g) Top gene ontology terms for the 162 genes in metacluster 38 and the 151 genes in metacluster 16.

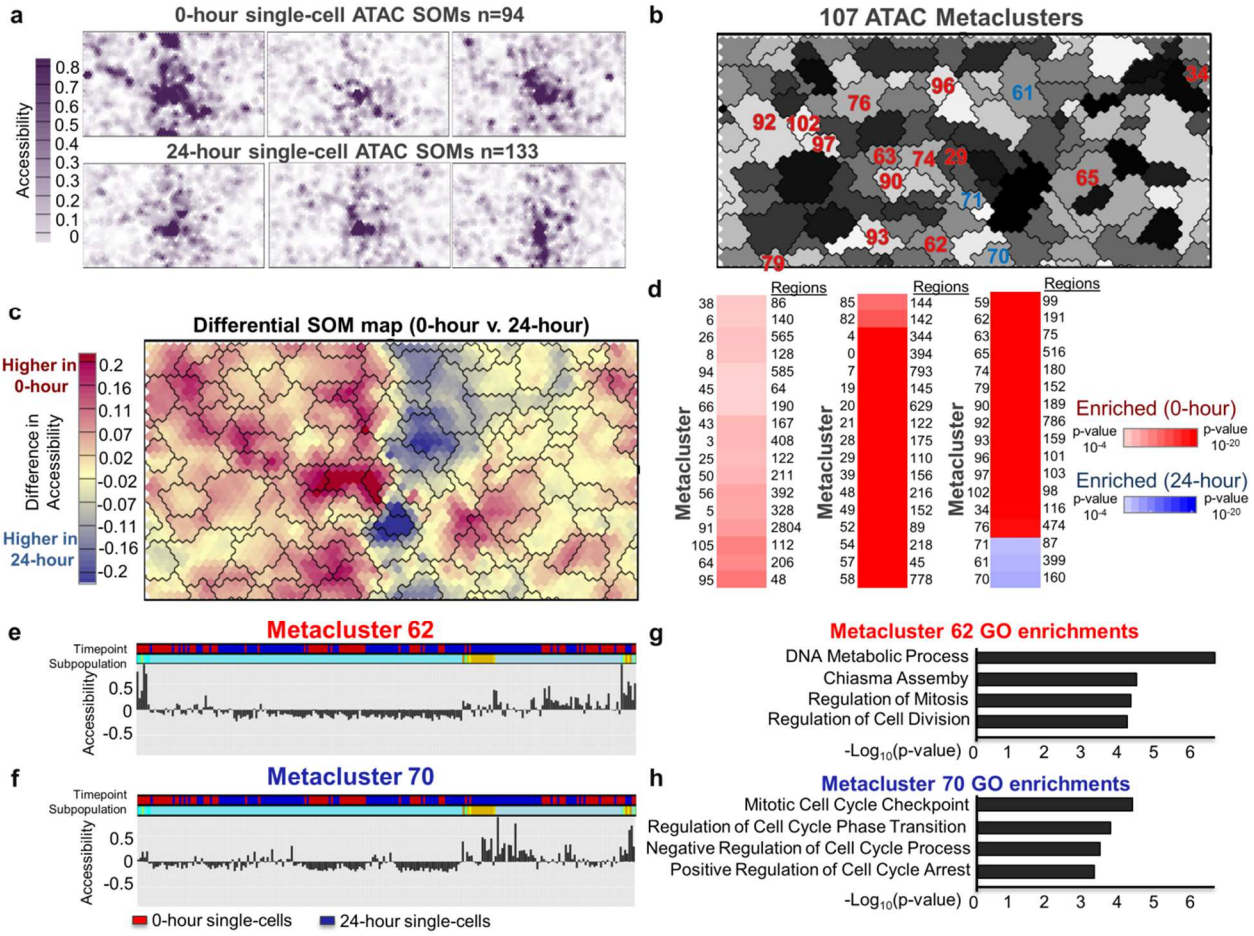


Figure 2.3. SOMatic reveals the dynamic chromatin landscape in single-cells
(a) A chromatin SOM was generated for the single-cell ATAC-seq dataset (0-hour 94 cells, 24-hour 133 cells). Maps for 3 cells from each timepoint were arbitrarily selected for display. (b) 107 metaclusters were identified using AIC scoring. Metacluster number and color were arbitrarily assigned for visualization purposes. (c) SOM difference map comparing 0-hour and 24-hour time-points. Maps for cells from 0 and 24-hour timepoints were averaged to generate a single map for each and then subtracted to create a map that represented chromatin accessibility fold change during pre-B cell development. Overlaid metacluster divisions generally follow contours of the map. (d) Trait enrichment analysis deployed on gene metaclusters revealed which are enriched in each time point. Metaclusters of interest are highlighted in panel b. (e-f) Summary showing the representative accessibility profile for SOM metaclusters 62 and 70. Columns are individual cells color-coded for 0 and 24-hour time-points ordered by hierarchical clustering on every metacluster representative gene expression profile. Cell subpopulations are represented by a 40% cut on that clustering. (f-e) Top gene ontology terms for genes associated to chromatin elements from SOM metaclusters 62 and 70. Association was determined through use of the GREAT algorithm (See methods).

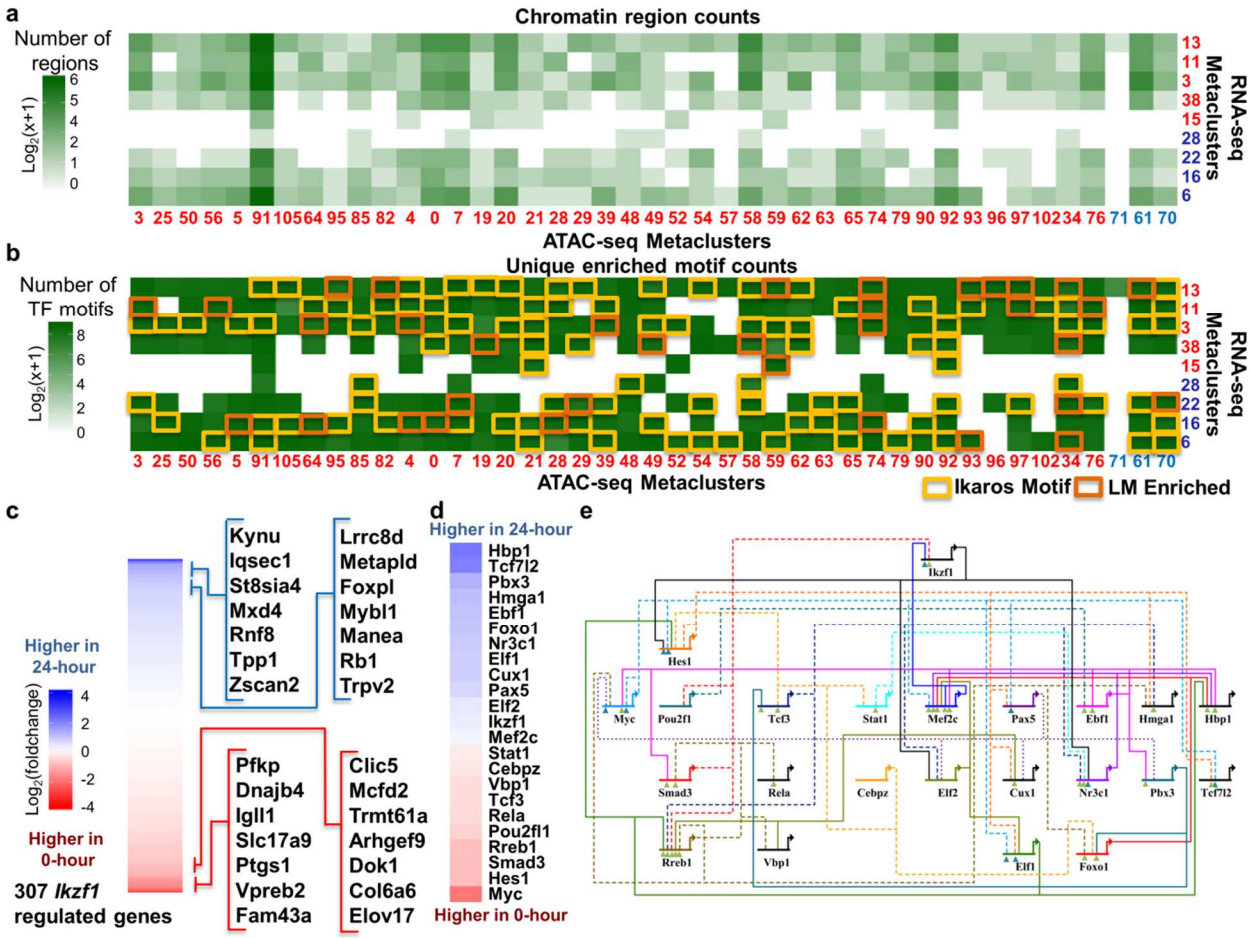


Figure 2.4. Transcriptional regulation by *Ikzf1* recovered using linked SOMs
(a) Size of pair-wise metaclusters that contain both differentially-expressed genes and differentially-accessible chromatin sites. Metaclusters of genes and regions with a higher enrichment at 24-hours are colored blue and are ordered by enrichment in the two time points. (b) Number of statistically-significant motifs found in each pair-wise metacluster from (a). Presence and enrichment of the *Ikzf1* motif in the pair-wise metacluster is noted. (c) Heatmap of expression fold change for genes predicted to be regulated by *Ikzf1*. Genes with the largest change between time points are noted. (d) Predicted downstream targets of *Ikzf1* with significant change over the time course. Each gene is labels with the fold change between time points with the same scale as 4c. (e) Predicted gene regulatory network downstream of *Ikzf1*. Genes are ordered left to right by their fold change over the time course. Connections are dashed if their signal is significantly lower at the 24-hour time point. Connections at each gene are labeled by level of evidence found in existing literature. Teal triangles indicate experimental evidence and green triangles indicate previous computational prediction.

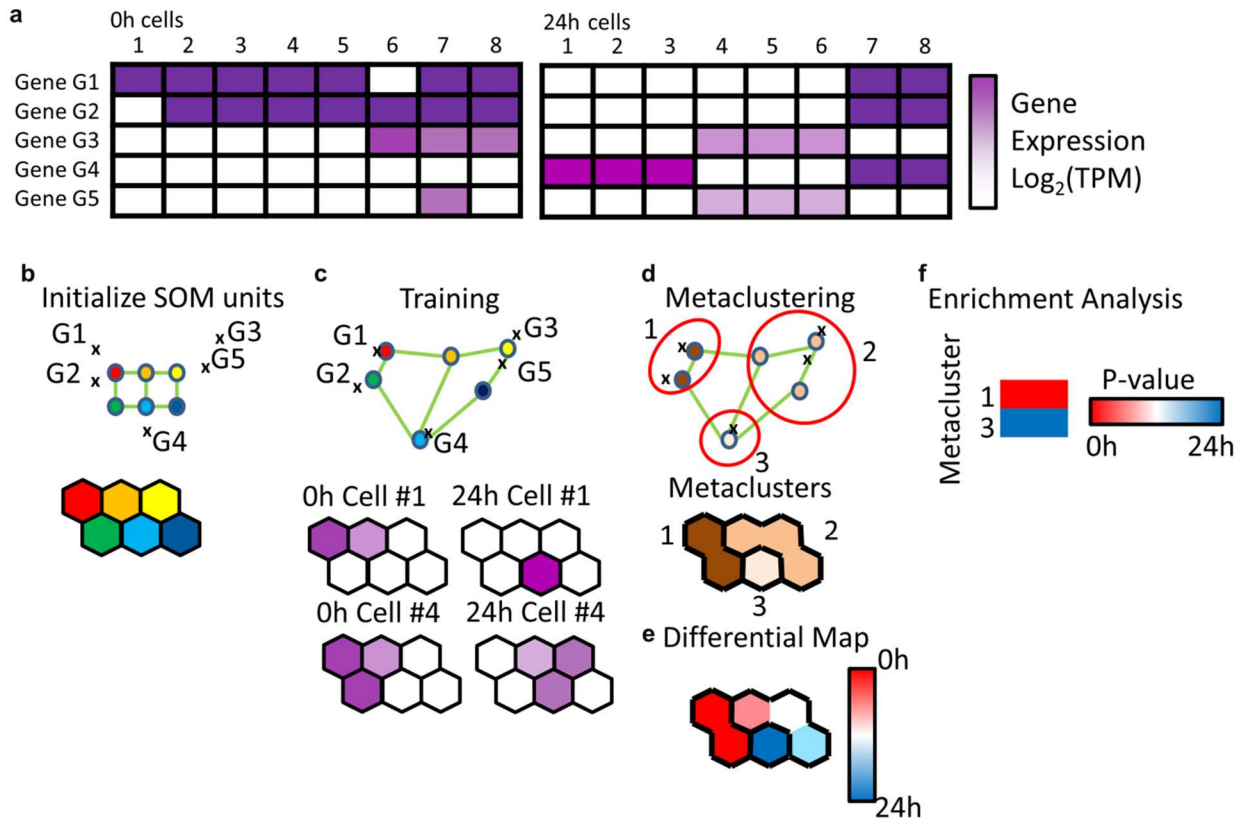


Figure 2.5. Self-Organizing Map Clustering Overview
(a) Example heatmap for 5 genes' expression in a typical single-cell RNA-seq with 2 time points. Genes G1 and G2 are enriched at 0h with two 0h cells missing that signal due to technical noise and gene G4 is enriched at 24hr. Genes G3 and G5 also have a similar expression pattern with two cells missing signal in G5 due to technical noise, but are not particularly enriched in either time point. (b) 2D representation of the genes' expression profile with an initial SOM scaffold. The colors in the scaffold correspond to those the map below. (c) 2D representation of the genes' expression profile with a typical trained SOM scaffold overlaid. The maps below represent the signal for each unit in the labeled experiment's dimension. For example, only gene G4 has signal in 24h Cell #1, and thus, only the unit near G4 has signal on the map. (d) Neighboring units with similar expression profiles are metaclustered to fix the overclustering of genes G1 and G2 into separate units. (e) Multiple individual maps can be combined into one through arithmetic. This map represents the average of each 24h map subtracted from the average of each 0h map. (f) Trait enrichment analysis can be applied on each metacluster to provide a p-value for enrichment in a particular time point. Here, metacluster 1, containing genes G1 and G2, is enriched in 0h, and metacluster 3, containing gene G3, is enriched in 24h.

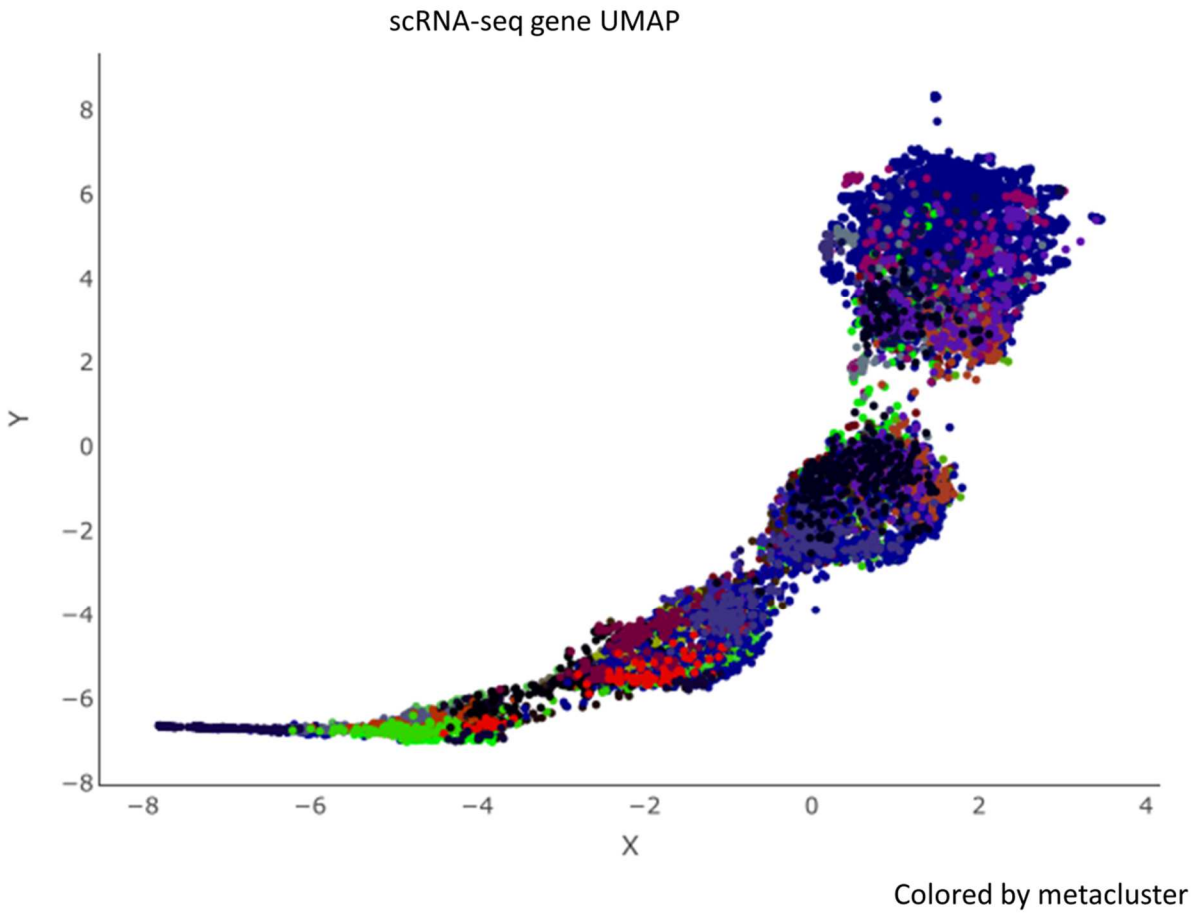


Figure 2.6. scRNA-seq gene UMAP
UMAP[30] generated using uwat[53] from scRNA-seq data with each point representing a gene's expression in each cell. The umap is separated into 4 large clusters, which provides a poor level of resolution for downstream analysis. Points were colored by RNA SOM metacluster, which divides the large clusters into many sub-clusters.

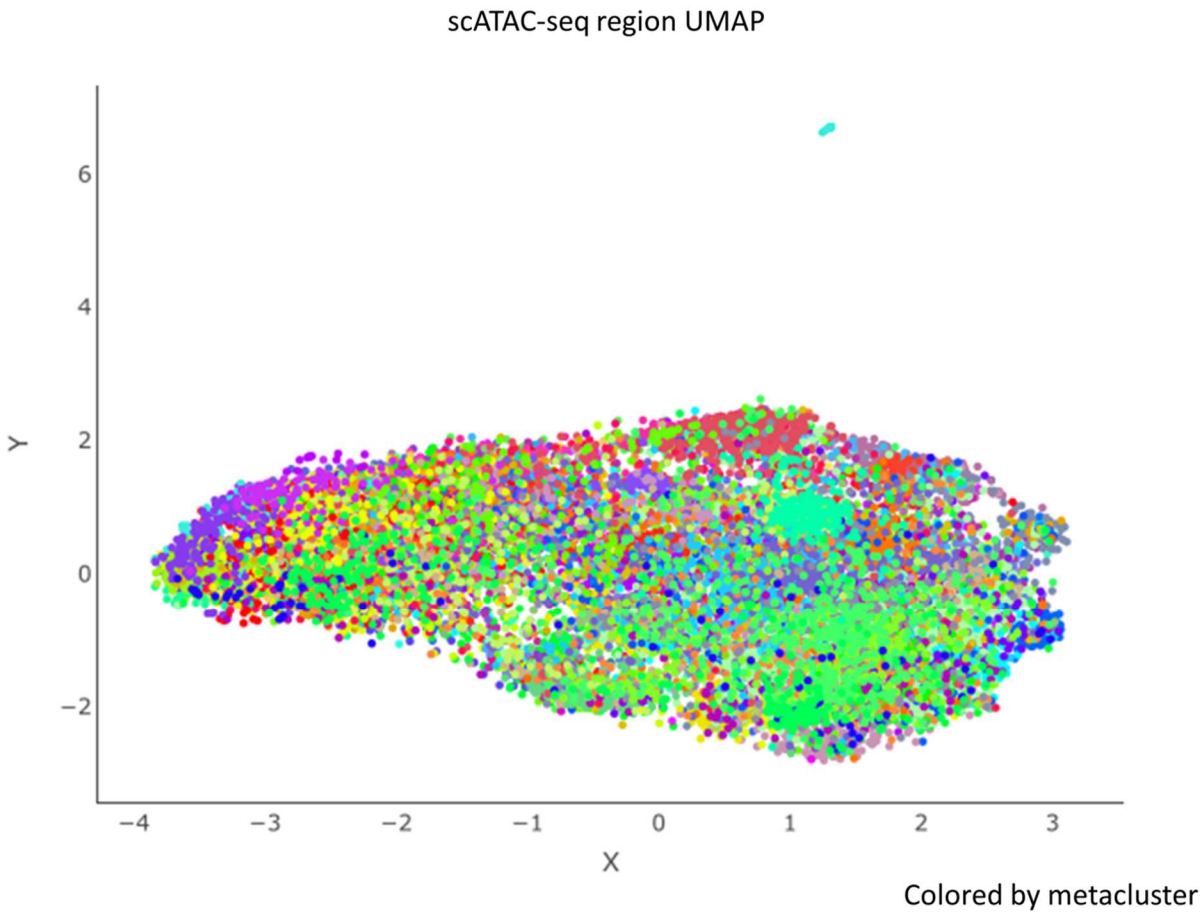


Figure 2.7. scATAC-seq region UMAP
UMAP[30] generated using uwat[53] from scATAC-seq data with each point representing a genome region's ATAC-seq signal in each cell. The umap could not be separated into any significant clusters. Points were colored by ATAC SOM metacluster, which divides the large cluster into many sub-clusters.

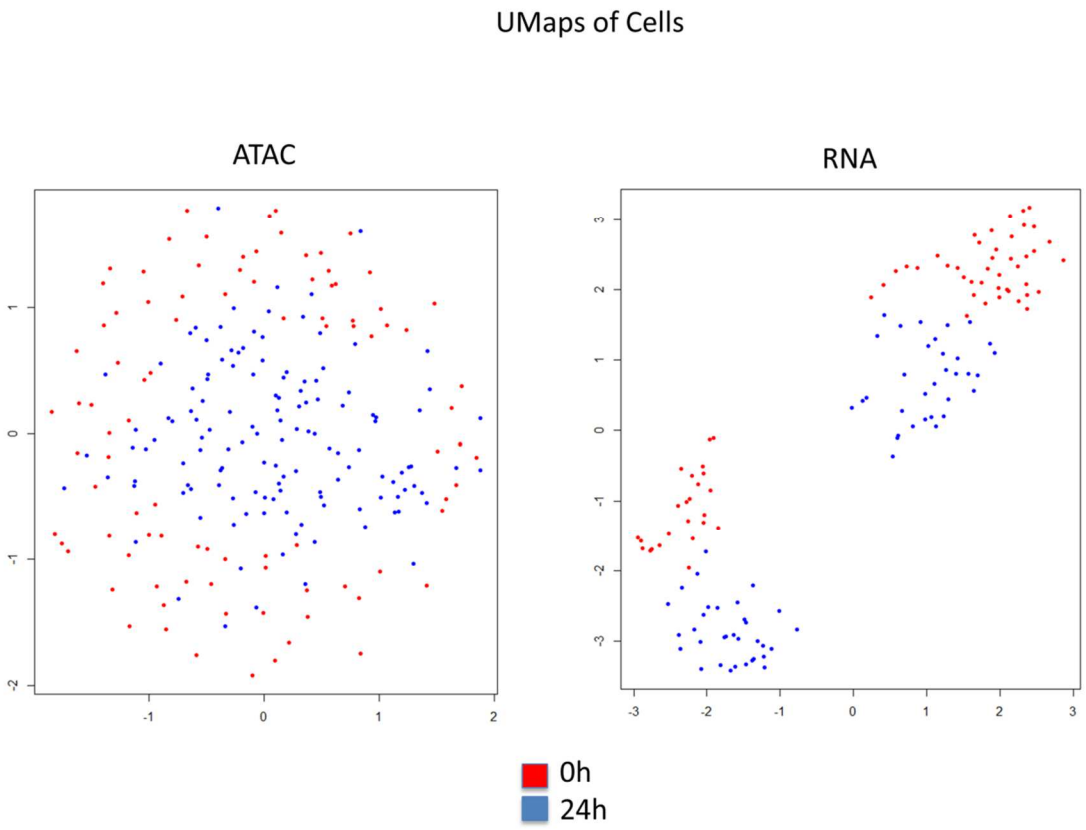


Figure 2.8. UMAPs of cells used in analysis
UMAP[30] generated using uwat[53] from both data types with each point representing a cell colored by timepoint.

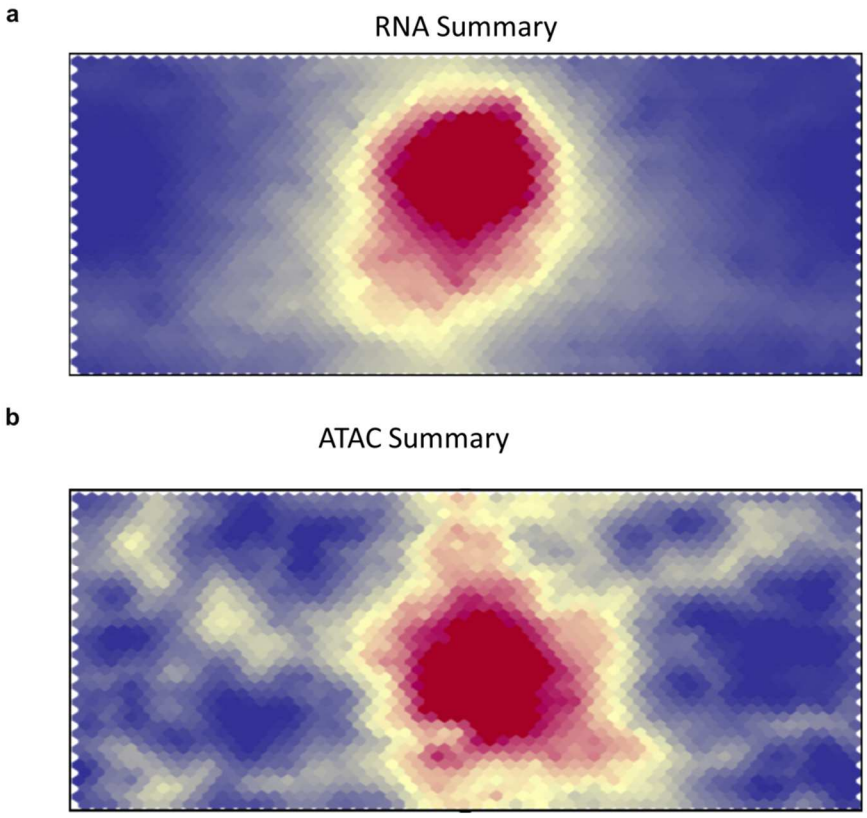


Figure 2.9. SOM summary maps (total signal in every cell)
(a-b) Summary maps for the (a) RNA and (b) ATAC SOMs. Each unit's value is generated by totaling the values in the full SOM unit's vector. A blue-white-red color spectrum was used. These graphs are mainly used to determine 'smoothness' of the SOM fit and to see if more timesteps or changes to the learning rate are needed.

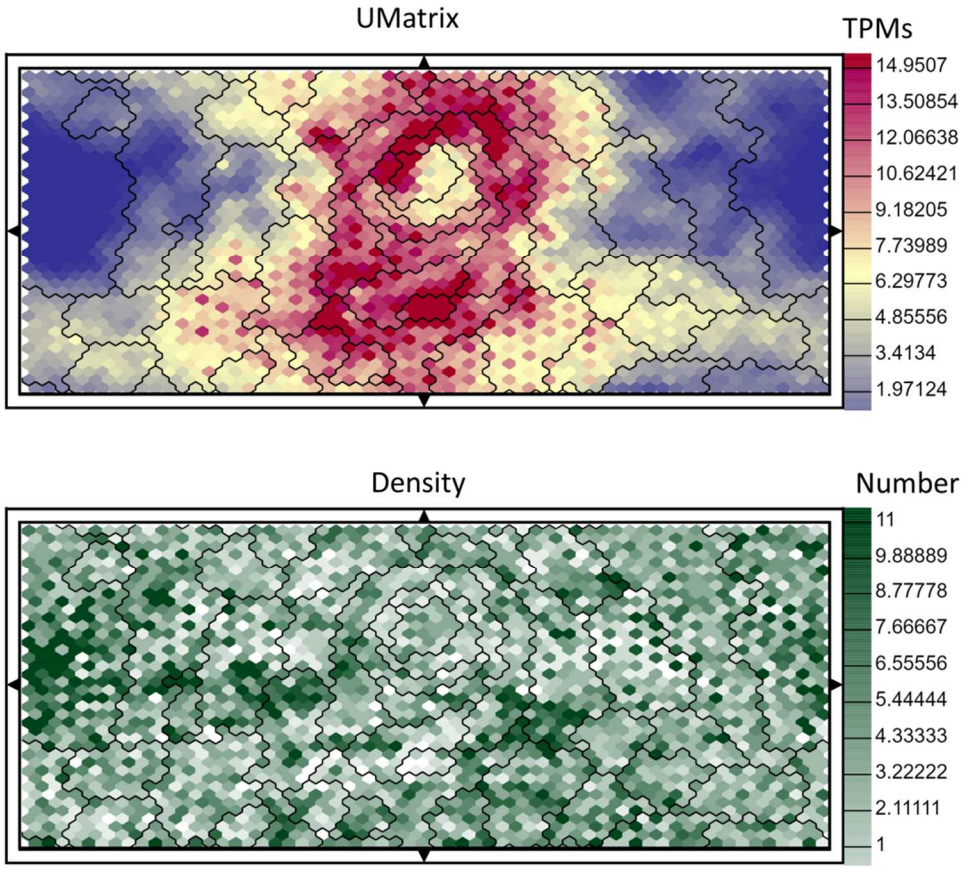


Figure 2.10. Statistic maps for scRNA-seq SOM
(a) U-Matrix for the SOM built with the single-cell RNA-seq dataset. Each unit contains the average of the distance to all neighboring units. Metacluster divisions are overlaid. Areas of high distance correspond primarily to a metacluster division. (b) Density map for the RNA-seq SOM. The color corresponds to the number of genes found in each unit. Metacluster divisions are overlaid. Most metaclusters are ruled by a few high density units.

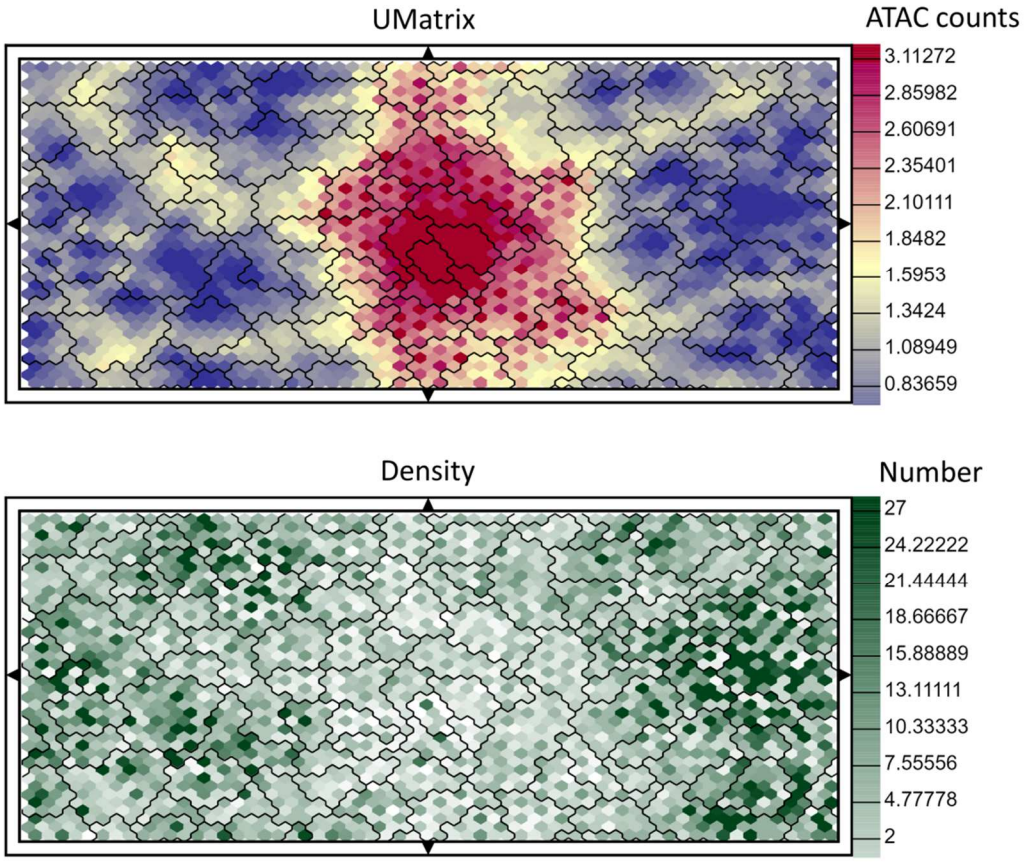


Figure 2.11. Statistic maps for scATAC-seq SOM
(a) U-Matrix for the SOM built with the single-cell ATAC-seq dataset. Each unit contains the average of the distance to all neighboring units. Metacluster divisions are overlaid. Areas of high distance correspond primarily to a metacluster division. (b) Density map for the ATAC-seq SOM. The color corresponds to the number of chromatin regions found in each unit. Metacluster divisions are overlaid. Most metaclusters are ruled by a few high density units.

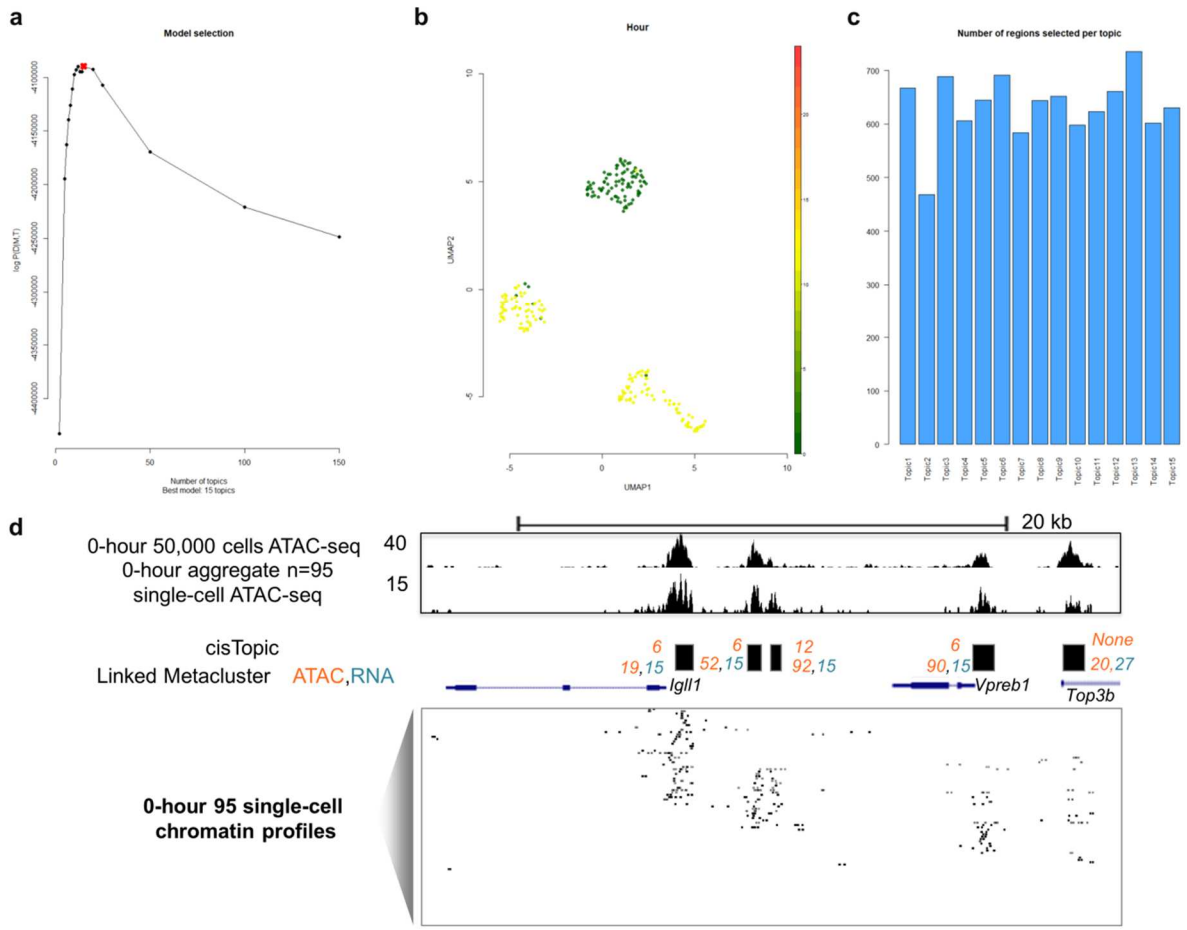
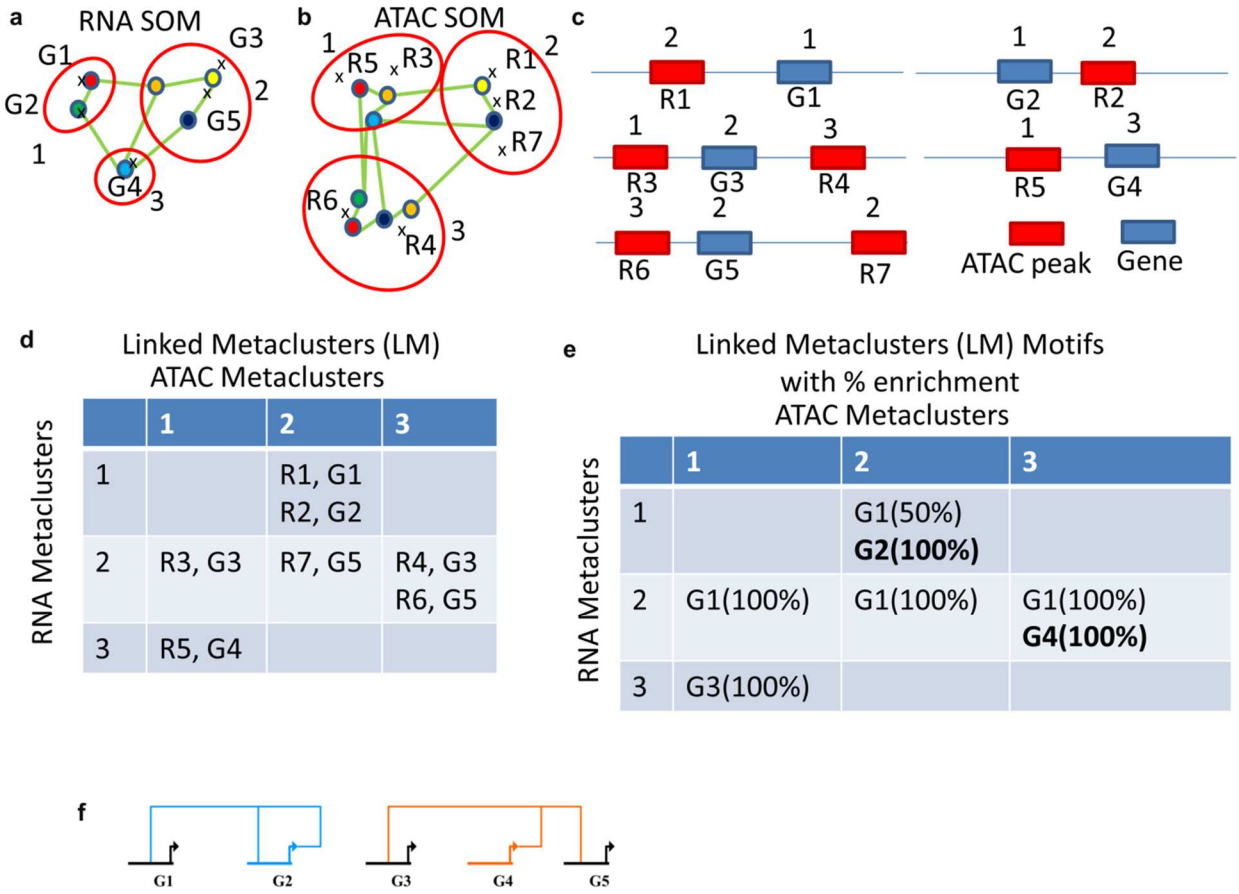


Figure 2.12. cisTopic Analysis of Pre-B cell ATAC-seq Data

a.) Graph detailing the score of various topics tried in cisTopic training. The best model had 15 topics. b.) T-sne output from cisTopic after training. Each point is a cell colored by timepoint (Yellow is 0 hr and green is 24 hr). c.) Bar graph detailing the number of regions in each called topic. d.) Comparison of cisTopic topics and SOM linked metaclusters. Several ATAC-seq peaks with very different profiles ended up in different ATAC-seq SOM metaclusters and the same cisTopic topic.



	1	2	3
1		R1, G1 R2, G2	
2	R3, G3	R7, G5	R4, G3 R6, G5
3	R5, G4		

	1	2	3
1		G1(50%) **G2(100%)**	
2	G1(100%)	G1(100%)	G1(100%) **G4(100%)**
3	G3(100%)		

Figure 2.13. SOM Linking Overview
(a) An example SOM after training on RNA-seq data. Metaclusters 1, 2, and 3 contain genes (G1, G2), (G3, G5), and (G4) respectively. (a) An example SOM after training on ATAC-seq data. Metaclusters 1, 2, and 3 contain genome regions (R3, R5), (R1, R2, R7), and (R4, R6) respectively. (c) An example of how the genes in (a) and the genome regions in (b) could be arranged with their respective metaclusters. (d) The final list of linked metaclusters (LM) that result from the above system. Note that Region 1 and 2 both end up in the same LM (ATAC 2, RNA 1) because they are both in ATAC metacluster 2 and their nearby genes, G1 and G2, are both in RNA metaclusters 1. (e) Example motif enrichments for each gene in (a) in each LM. Bolded genes have a significant enrichment over the background. G1 is found too highly in many LMs and might have an extremely permissive motif. In LM (ATAC 1, RNA 3), G3 motif is found, but would not be called significant due to it being only 1 observation. (f) An example gene regulatory network generated from (e).

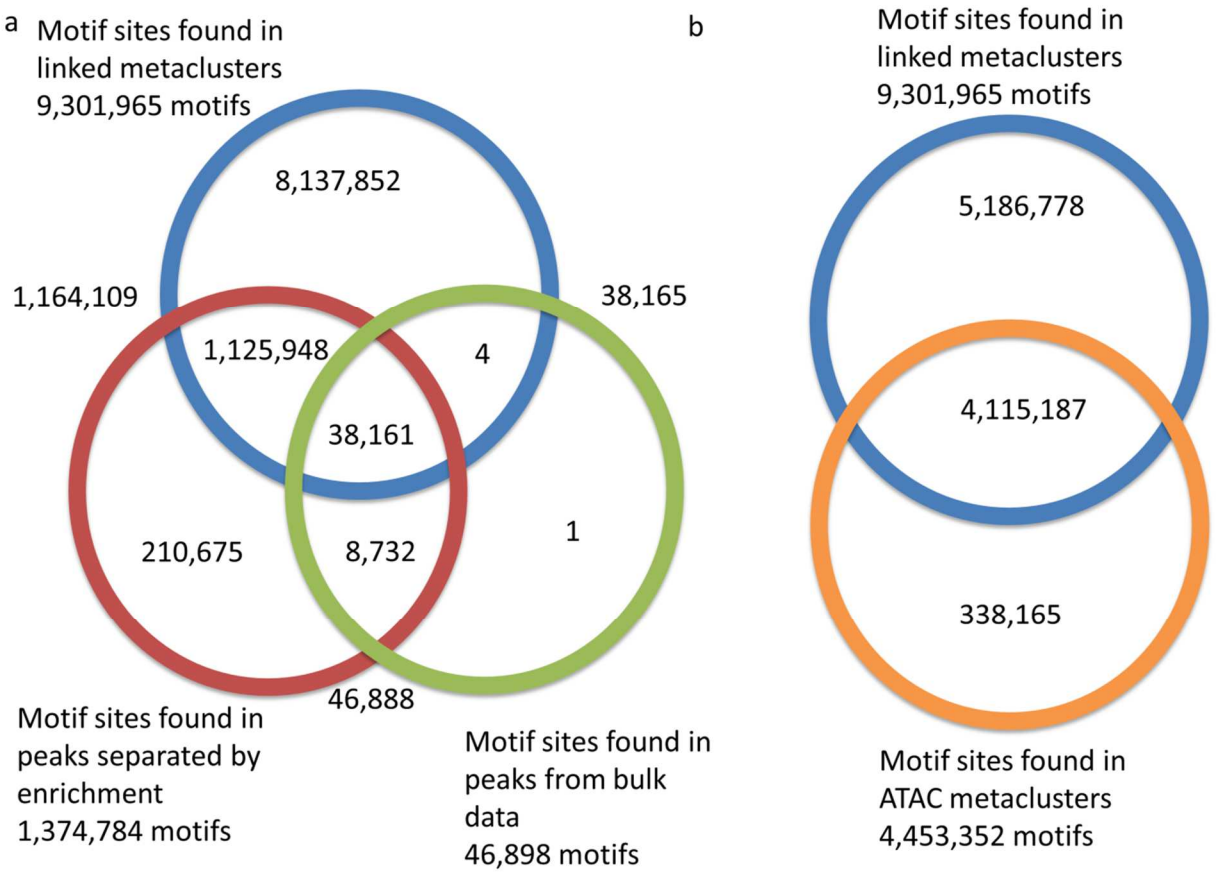


Figure 2.14. Motif mining efficiency using various techniques
a.) Graph detailing the number of motifs found using the same set of peaks with different groupings using the same q-value<.05 cutoff. B.) Graph detailing the number of motifs found using the same set of peaks with using the linked metacluster grouping and just the ATAC-seq SOM metaclusters grouping using the same q-value<.05 cutoff.

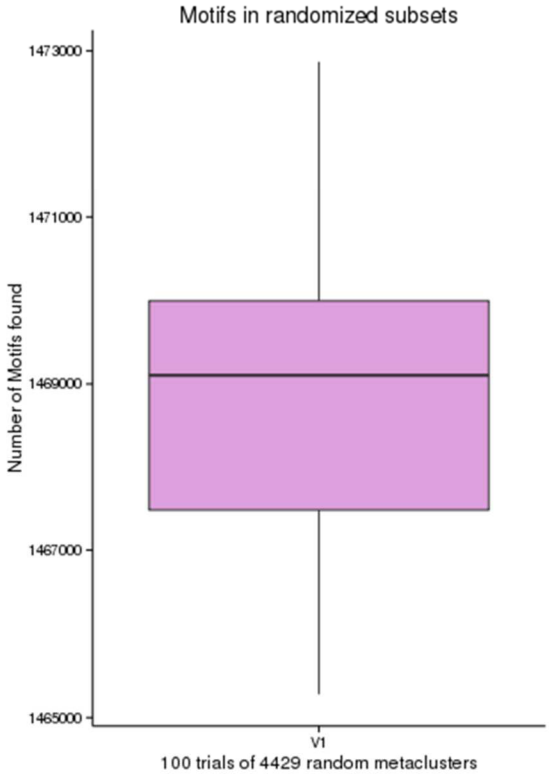


Figure 2.15. Motif scanning statistics for random separation validation
The distribution of motifs found by randomly splitting peaks into 4,429 synthetic linked metaclusters(LM). The mean was ~1,469,000 motifs which is significantly fewer than the ~9.3 million found in the real LMs.

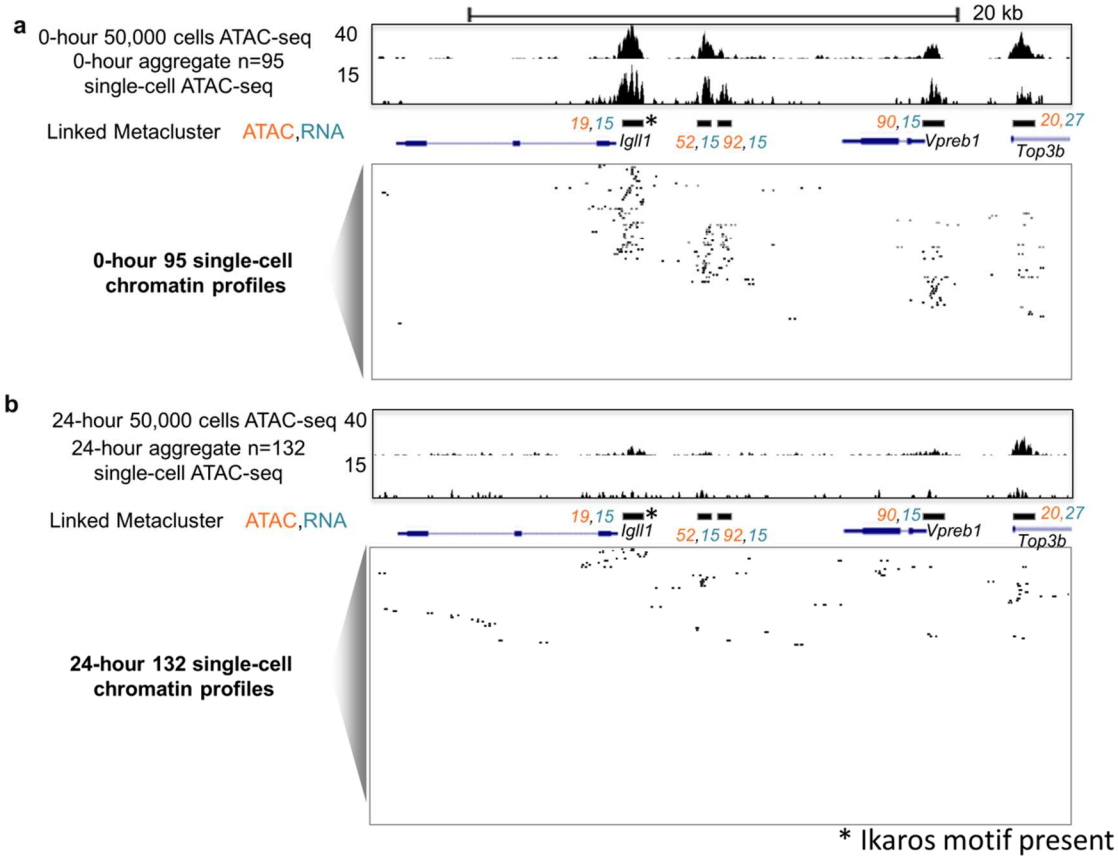


Figure 2.16. Chromatin accessibility patterns around Igll1 and Vpreb1 locus revealed by scATAC-seq labeled by SOMatic

(a-b) UCSC genome browser screenshots of the *Igll1* and *Vpreb1* loci with bulk (50,000 cells), aggregate (94 single-cells averaged) and single-cell ATAC-seq for 0 (a; 94 single-cells) and 24-hour (b;133 single-cells) pre-B cells. Linked SOM ids (ATAC, RNA) are depicted for all chromatin elements.

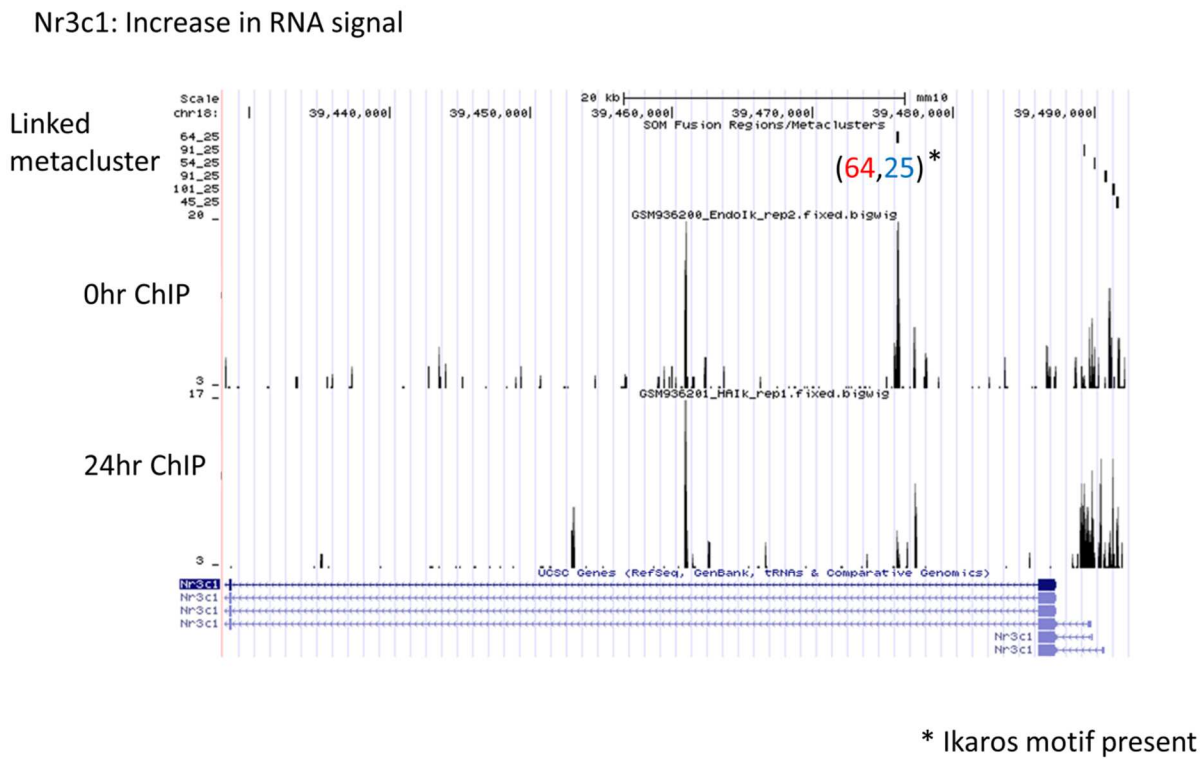


Figure 2.17. ChIP-seq validation of Ikaraos binding near Nr3c1
UCSC genome browser snapshots of Ikaros ChIP data taken at the 0-hour and 24-hour timepoints near *Nr3c1*. The location of the predicted motif is noted along with its linked metacluster ID. The marked location has a significant change in binding at the marked location over the time course.

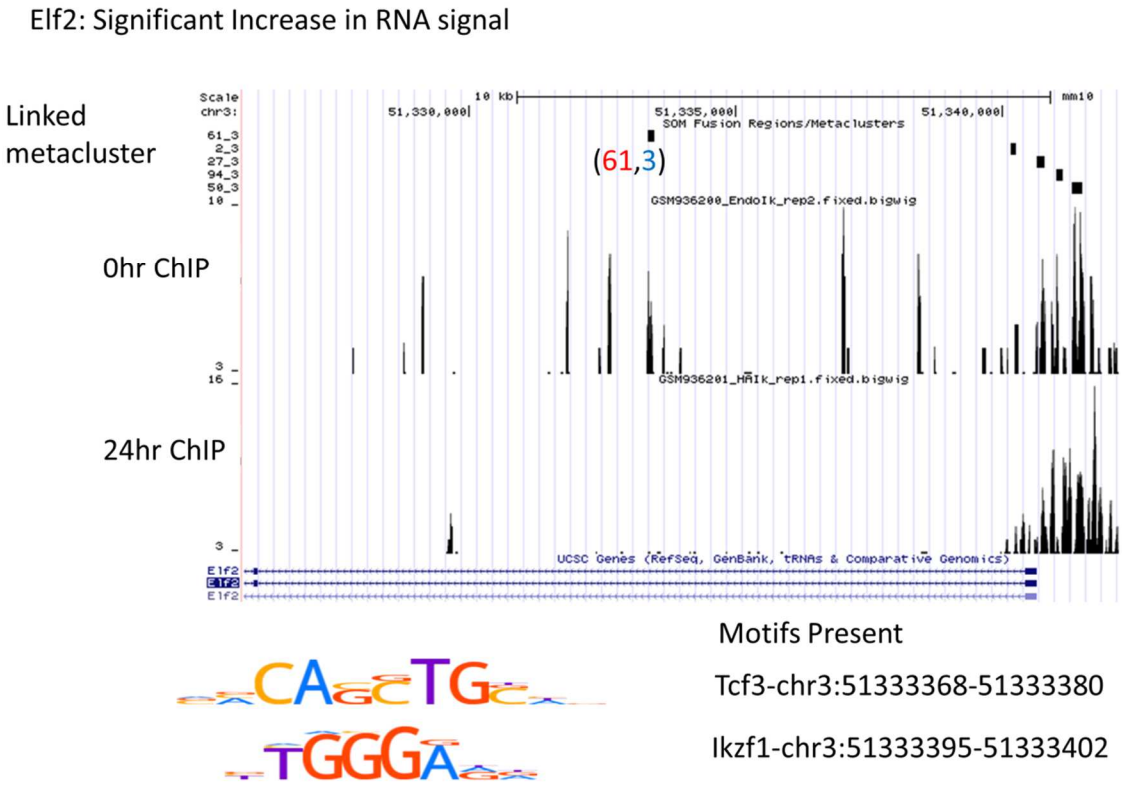


Figure 2.18. ChIP-seq validation of Ikaraos binding near *Elf2*
UCSC genome browser snapshots of Ikaros ChIP data taken at the 0-hour and 24-hour timepoints near *Elf2*. There were 2 predicted motifs in this metacluster, Ikaros and Tcf3.

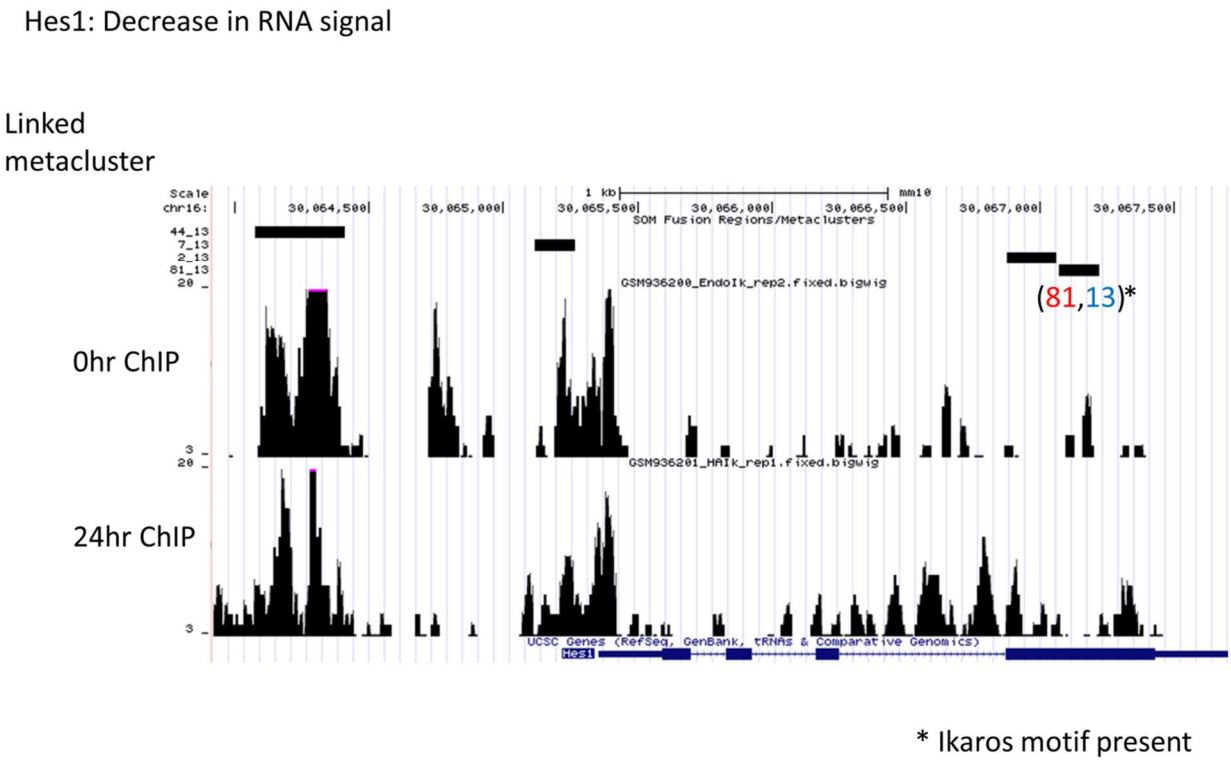


Figure 2.19. ChIP-seq validation of Ikaraos binding near *Hes1*
UCSC genome browser snapshots of Ikaros ChIP data taken at the 0-hour and 24-hour timepoints near *Hes1*. The location of the predicted motif is noted along with its linked metacluster ID. The marked location has a significant change in binding at the marked location over the time course.

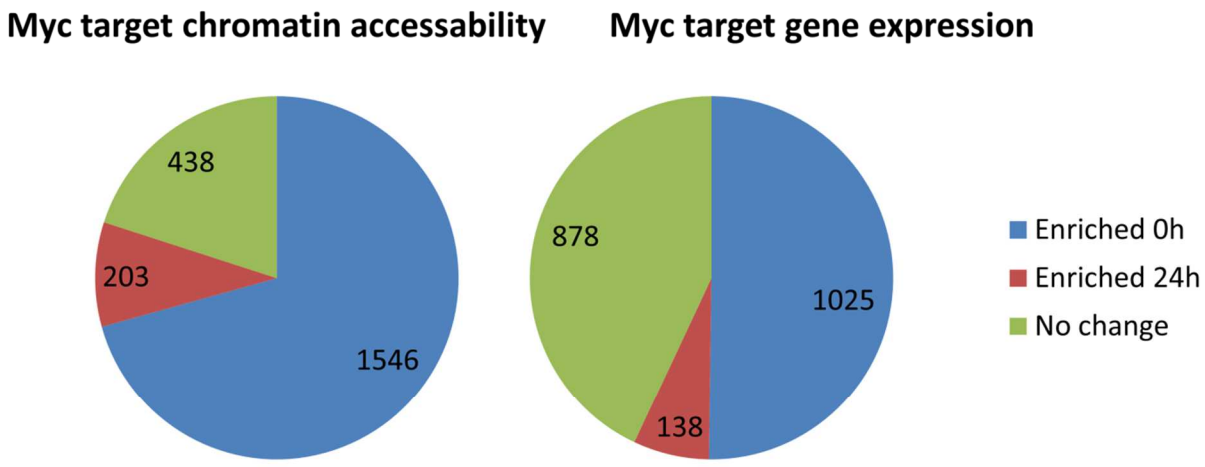


Figure 2.20. Downstream *Myc* target gene expression and chromatin accessibility dynamics
Myc (whose signal drops dramatically from 0- to 24- hour) downstream targets were predicted in a method similar to that in Figure 4. Around half of these react with a drop in signal with a small portion reacting with an increase. This is similar to the change in chromatin accessibility at the predicted binding sites near these genes.

Regulator	Targets w/ levels of evidence													
Myc	Mef2c	2[43]	Pax5	1[38]	Myc	1[39]	Tcf7l2	1[39]	Elf1	1[39]	Hes1	1[39]		
Hes1	Hes1	2[44]	Pax5	3	Cux1	2[32]	Elf1	1[37]	Tcf7l2	2[44]	Hbp1	3		
Smad3	Pou2f1	3	Rreb1	2[44]	Ikzf1	2[44]								
Rreb1	Myc	3	Smad3	2[44]	Vbp1	3	Foxo1	2[44]	Hmga1	3	Rela	2[45]	Rreb1	2[44]
Pou2f1	Ebf1	2[43]												
Tcf3	Nr3c1	3	Elf2	3	Nr3c1	2[43]	Hmga1	2[43]	Rreb1	2[43]				
Cebpz	Stat1	3	Foxo1	3	Tcf3	2[43]	Hes1	3						
Stat1	Nr3c1	2[43]	Stat1	2[43]										
Mef2c	Ikzf1	1[40]	Mef2c	2[44]										
Ikzf1	Nr3c1	1[34]	Hes1	1[39]	Elf2	3								
Elf2	Mef2c	3	Vbp1	3	Elf1	3	Cux1	3	Rreb1	2[44]				
Pax5	Pbx3	3	Elf2	3	Myc	1[37]								
Elf1	Hbp1	3	Rela	2[43]	Rreb1	2[44]	Cux1	2[43]						
Nr3c1	Hbp1	3	Mef2c	2[43]										
Foxo1	Hbp1	2[43]	Mef2c	2[44]										
Ebf1	Pbx3	2[43]	Hbp1	2[43]	Smad3	2[43]	Ebf1	2[43]	Mef2c	2[43]	Myc	2[43]		
Pbx3	Foxo1	2[43]	Tcf3	2[43]										

1 – Direct ChIP/KO evidence
2 – Predicted
3 – New Connection

Figure 2.21. Gene regulatory connections downstream of Ikaros with levels of known evidence

A list of transcription factors with significant changes over the time course and the transcription factors were predicted to regulate. Each regulated gene is followed by a label for the level of existing evidence and reference number if relevant.

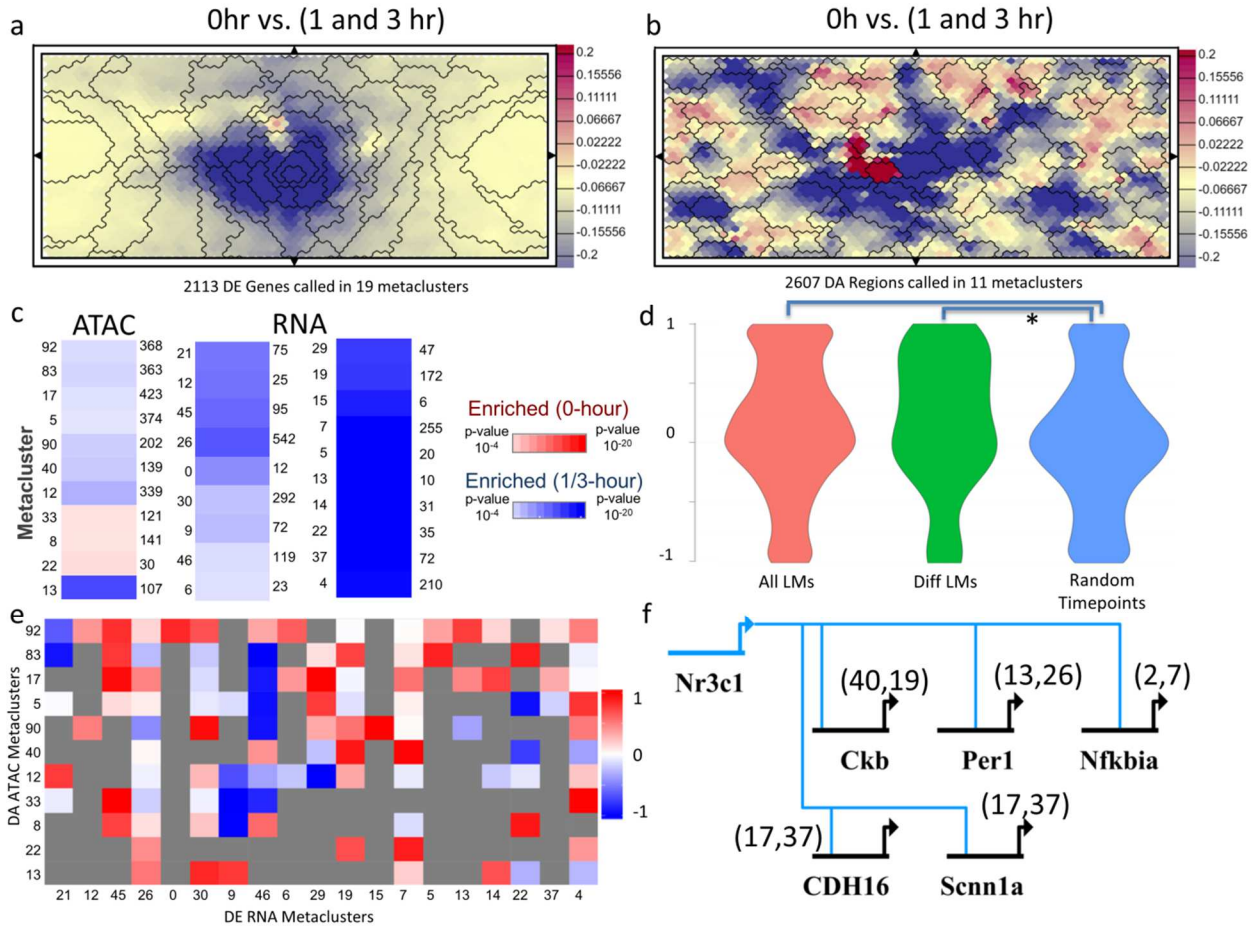


Figure 2.22. Application of linked metaclusters on sciCar data

(a-b) Difference maps displaying the areas of temporal enrichment after training on sciCar data. (c) RNA data was differential in 19 metaclusters that represent 2113 genes. The ATAC data was differential in 11 metaclusters representing 2607 genome regions. (d) Violin plots describing the distribution of average temporal correlations between linked ATAC-seq peaks and genes. The differential metaclusters (in green) have fewer combinations with no correlation and more with negative correlations than the distributions from all LMs (in red). Both distributions are significantly (pvalue < .05) different than when the timepoints of the cells are scrambled (in blue). (e) A heatmap of the average temporal correlations from the differential linked metaclusters. (f) Known targets of Nr3c1 (GR receptor) recovered during motif and network analysis. These downstream genes all appeared in differential RNA metaclusters.

2.6 Methods

2.6.1 Pre-B cell differentiation

ERt2-Ikaros inducible B3 cells were cultured in Iscove's Modified Dulbecco's Medium (IMDM) supplemented with 10% FBS. Differentiation was induced as previously shown[32]. Briefly, cells were induced with 20mM of 4-hydroxytamoxifen (4OHT), over the course of 24 hours. Prior to performing single-cell experiments, cells were washed twice with cold 1X PBS.

2.6.2 Single-cell RNA-seq

Single cells were isolated using the Fluidigm C1 System. Single cell C1 runs were completed using the smallest IFC (5-10 um) based on the estimated size of B3 cells. Briefly, cells were collected for 0 (1 batch) and 24-hour (2 batches) time-points at a concentration of 400 cells/µl in a total of 50 µl. To optimize cell capture rates on the C1, buoyancy estimates were optimized prior to each run. Each individual C1 capture site was visually inspected to ensure single-cell capture and cell viability. After visualization, the IFC was loaded with Clontech SMARTer kit lysis, RT, and PCR amplification reagents. After harvesting, cDNA was normalized across all libraries from 0.1-0.3 ng/µl and libraries were constructed using Illumina's Nextera XT library prep kit per Fluidigm's protocol. Constructed libraries were multiplexed and purified using AMPure beads. The final multiplexed single-cell library was analyzed on an Agilent 2100 Bioanalyzer for fragment distribution and quantified using Kapa Biosystem's universal library quantification kit. The library was normalized to 2 nM and sequenced as 75bp paired-end dual indexed reads using Illumina's NextSeq 500 system at a depth of ~1.0-2.0 million reads per library.

2.6.3 Single-cell ATAC-seq

Single-cell ATAC-seq was performed using the Fluidigm C1 system as done previously[8]. Briefly, cells were collected for 0 and 24-hours post treatment with tamoxifen, at a concentration of 500 cells/μl in a total of 30-50 μl. Additionally, 3 biological replicates of ~50,000 cells were collected for each measured time-point to generate bulk ATAC-seq measurements. Bulk ATAC-seq was performed as previously described[54]. ATAC-seq peak calling was performed using bulk ATAC-seq samples. ATAC-seq peaks were then used to estimate single-cell ATAC-seq signal. Our C1 single-cell capture efficiency was ~70-80% for our pre-B system. Each individual C1 capture site was visually inspected to ensure single-cell capture. In brief, amplified transposed DNA was collected from all captured single-cells and dual-indexing library preparation was performed. After PCR amplification of single-cell libraries, all subsequent libraries were pooled and purified using a single MinElute PCR purification (Qiagen). The pooled library was run on a Bioanalyzer and normalized using Kappa library quantification kit prior to sequencing. A single pooled library was sequenced as 40bp paired-end dual indexed reads using the high-output (75 cycle) kit on the NextSeq 500 from Illumina. Two C1 runs were performed for 0 and 24-hour single-cell ATAC-seq experiments.

2.6.4 Single-cell RNA-seq data processing

Single-cell RNA-seq libraries were mapped with Salmon[55] to the mouse Ensembl gene annotations and mm10 reference genome. Single-cell libraries with a mapping rate

less than 50% and less than 450,000 mapped reads were excluded from any downstream analysis. Analysis was performed using 0 and 24-hour single-cells.

2.6.5 Bulk and single-cell ATAC-seq data processing

Single-cell libraries were mapped with Bowtie2[56] to the mm10 reference genome using the following parameters (bowtie2 -S -p 10 --trim3 10 -X 2000). Duplicate fragments were removed using Picard (http://picard.sourceforge.net) as previously performed[8]. We considered single-cell libraries that recovered > 5k fragments after mapping and duplication removal. Bulk ATAC-seq replicates were mapped to the mm10 reference genome using the following parameters (bowtie2 -S --trim3 10 -p 32 -X 2000). Peak calling was performed on bulk replicates using HOMER with the following parameters (findPeaks <tags> -o <output> -localSize 50000 -size 150 -minDist 50 –fragLength 0). The intersection of peaks in three biological replicates was performed. A consolidated list of peaks was generated from the union of peaks from 0 and 24 hour time-points.

2.6.6 ChIP-seq analysis

Ikzf1 ChIP-seq data for 0 and 24-hour pre-B cells[29] was mapped to the mm10 reference genome using Bowtie2[56]. For all samples, we filtered duplicated reads and those with a mapping quality score below 20. To identify peaks, we used the CLCbio Peak Finder software ENREF 38[57] with default parameters and control input libraries. We defined significant peaks with an adjusted p-value <0.01 also using biological replicates.

2.6.7 Training and Metaclustering of the individual RNA and ATAC SOMs

We use the SOMatic package, which is a combination of tools written in C++ and R designed for the analysis and visualization of multidimensional genomic or gene expression data, to train our individual SOMs. The SOMatic package also builds a customized, optional javascript viewer to mine the results visually. Installation information for this package can be found at https://github.com/csjansen/SOMatic.

For the RNA-seq SOM, we built a matrix of 12,380 expressed genes in 128 single cells and we used half the genes (6190) to train a self-organizing map with a toroid topology with size 40x60 with 6,190,000 million time steps (1000 epochs) as previously described[25] to select the best of 100 trials based on lowest fitting error. The entire matrix was used for scoring this best trial to generate the final SOM. The SOMatic website for this SOM can be viewed at http://crick.bio.uci.edu/STATegra/RNASOM/

Similarly, the ATAC-seq data was organized into a matrix consisting of scATAC signal in 227 cells at 25,466 ATAC-seq peaks (from pooled data) and half of the peaks were used to train a SOM with a toroid topology with size 40x60 using 12,733,000 time steps (1000 epochs) as previously described[25]. The best of 100 trials based on lowest fitting error was selected and the entire matrix was used for scoring the final SOM. The SOMatic website for this SOM can be viewed at http://crick.bio.uci.edu/STATegra/ATACSOM/

SOM units with similar profiles across cells were grouped into metaclusters[24, 25] using SOMatic. Briefly, continuity-constrained[23] metaclustering was performed using k-means clustering to determine centroids for groups of units. Metaclusters were built around these centroids so that each cluster is in one piece to maintain the SOM topology. SOMatic's metaclustering function attempts all metacluster numbers within a range given and scores them based on Akaike information criterion (AIC)[58]. The penalty term for this

score is calculated using a parameter called the "dimensionality," which is the number of independent dimensions in the data. We performed a hierarchical clustering on the SOM unti vectors and counted the number of clusters that were present at a height level equal to 30% of the total distance in the clustering. For the ATAC-seq SOM, the dimensionality was calculated to be 35, and for the RNA-seq SOM, the dimensionality was calculated to be 128.

For the RNA metaclustering, we tried all metaclusters numbers (k) between 20 and 50, whereas for the ATAC metaclustering we tried all k between 80 and 120. The k with the lowest AIC score was the one chosen for each SOM. For ATAC-seq, 107 metaclusters had the best score, and for RNA-seq, 39 metaclusters had the best score. R scripts for generating metacluster reports are provided in the SOMatic package. Metatcluster/Trait correlation and hypothesis testing analysis were done as previously described[24].

2.6.8 Hyperparameter Variation

There are inherent trade-offs that have to be kept in mind when choosing SOM parameters for training and metaclustering. For example, the size of a SOM is typically one of the most important decisions to be made in analyses of this type. A smaller SOM may group elements together that do not belong together and will reduce the statistical power of down-stream analysis, and a larger SOM may separate elements that belong in the same cluster but are separated due to noise, causing down-stream analysis to miss patterns that may exist. Similarly, the number of timesteps and the learning rate will change the chances of under and over-clustering by changing how the SOM scaffold morphs into the topology of the data. Proper metaclustering can improve the robustness of the SOM by easily revealing improper training due to poor parameters.

The scRNA-seq SOM was built with additional sizes 20x30 and 80x120 with little change to the calculated number of metaclusters, with 36 and 43 respectively. The 20x30 SOM was not chosen for the final analysis due to the occurrence of multiple 1-unit metaclusters, which indicates an underclustering. The 80x120 SOM was not chosen due to having a metacluster that contained a unit in each row which indicated a possible overclustering. The number of timesteps and learning rate chosen were determined to be sufficient due to the smoothness of the final summary map (Fig. 9a). An insufficient value in either of these parameters would cause the summary to have large breaks in total signal between neighboring units, indicating under-training.

The scATAC-seq SOM was also built with sizes 20x30, and 80x120 with little change to the calculated number of metaclusters, with 98 and 109 respectively. The 80x120 SOM was not chosen due to the map focusing too much on regions that were unique to each cell, indicating overclustering. The 40x60 size was chosen over the 20x30 due to it having a better score. The number of timesteps and learning rate chosen were determined to be sufficient due to the smoothness of the final summary map (Fig. 9b).

2.6.8 Linked SOMs

In order to define this, a few preliminary definitions are required. For a set A of data vectors, it is possible to define a set of n vectors, B, indexed on a 2D lattice to partition A into n subsets with each vector assigned to the subset i iff B_i is the closest element of B to that vector. Due to the 2D indexing lattice that they are placed on, each vector in B is adjacent to its closest member in B, with "closest" defined by an unsupervised neural network. The set of vectors, B, is the set of SOM units.

Similarly, it is possible to compute a set of m vectors, M, to partition B into m subsets, S, with each vector assigned to the subset i iff M_i is the closest element in M to that vector such that a path can be drawn on the lattice using only elements of S_i. This path requirement is in place to maintain the SOM topology calculated in training of the neural network. The subsets, S, are the metaclusters defined previously.

Let G be the set of gene vectors from a number of RNA-seq experiments and let R be the set of genome region vectors defined by ATAC-seq peaks. Using the procedure above, it is possible to segment these sets into metaclusters, named N and M respectively. Between these two metacluster partitions, we can define a linker mapping, h, from R to G. Using a linker mapping designed to link the individual SOM datatypes, we can define a set of partitions, $F^{M,N,h}$, where $(r,g) \in (R,G)$ is an element of $F^{M,N,h}{}_{ij}$ iff $h(r)=g$, $g \in N_j$, and $r \in M_i$. In this case, the linker mapping that we use to link RNA and open chromatin data is an implementation of the GREAT[33] OneClosest algorithm with a cutoff of 50kb to build regulatory regions around transcription start sites for each gene and check if these regions overlap with the ATAC-seq peaks. The resulting Linked SOM metaclusters (LMs) contain clusters of similar genome regions such that their linked genes are also similar.

2.6.9 Motif Analysis

The regulatory regions, including repeat regions, in each Linked SOM metacluster were separately scanned for motifs from the HOCOMOCOv11 mouse motif database[59] with FIMO v4.12.0[60] using a q-value threshold of .05. The background for FIMO was calculated using the entire mm10 reference genome. Then, for each transcription factor in the

database, the percentage of regions in each LM with a motif for that factor was calculated. To determine enrichment, the percentages for each transcript factor were separately compared in a one-tailed z-score analysis. LMs with a percentage that was significantly (pvalue < .05) enriched over the baseline, the average percentage across all LMs for that transcription factor motif, was reported for each transcription factor. Finally, transcription factors with a statistically significant number of motifs were mapped to the gene fused to the regulatory region the motif was found within. The full list of these potential connections can be found here: http://crick.bio.uci.edu/STATegra/LinkedMotifMappings.txt.

2.6.10 sciCar Scalability Analysis and cisTopic scATAC-seq Analysis

Processed sciCar data was downloaded from NCBI GEO (GSE117089) and reformatted into data matrices for both the scRNA-seq and scATAC-seq data. Of the ~4825 cells with measurements in both experiments, we kept those (3,234 cells) with more than 5% of genes detected in the RNA signal and more than 1000 mapped fragments detected in the ATAC signal. Additionally, we removed genes that were detected in less than 5% of cells and peaks with less than 100 total reads in all cells that passed the above filter. In total, the final RNA matrix was 17,751 genes x 3,234 cells and the final ATAC matrix was 18,638 peaks x 3,234 cells. These matrices were both trained into 40x60 SOMs over 1000 epochs with 100 replicates (best score taken) as above. We then performed the entire Linked SOMs pipeline above using the hg19 reference genome and Homo_sapiens.GRCh37.87 gene annotations.

For the cisTopic analysis, we input the pre-B cell ATAC-seq training matrix into cisTopic v0.2.0 and followed the vignette on their github.

(https://github.com/aertslab/cisTopic)

2.7 References

1. Dasgupta, S., G. Bader, and S. Goyal, *Single-Cell RNA Sequencing: A New Window into Cell Scale Dynamics.* Biophys J., 2018.
2. Nguyen, A., et al., *Single Cell RNA Sequencing of Rare Immune Cell Populations.* Front Immunol., 2018. **9**: p. 1553.
3. Ortiz, V. and M. Yu, *Analyzing Circulating Tumor Cells One at a Time.* Trends Cell Biol, 2018.
4. Davie, K., et al., *A Single-Cell Transcriptome Atlas of the Aging Drosophila Brain.* Cell, 2018.
5. Han, X., et al., *Mapping the Mouse Cell Atlas by Microwell-Seq.* Cell, 2018. **172**(5): p. 1091-1107.
6. Chappell, L., A.J.C. Russell, and T. Voet, *Single-Cell (Multi)omics Technologies.* Annual Review of Genomics and Human Genetics, 2018. **19**.
7. Cusanovich, D.A., et al., *Multiplex single-cell profiling of chromatin accessibility by combinatorial cellular indexing. .* Science, 2015. **248**: p. 910-914.
8. Buenrostro, J.D., et al., *Single-cell chromatin accessibility reveals principles of regulatory variation. .* Nature, 2015: p. doi:10.1038/nature14590.
9. Jin, W., et al., *Genome-wide detection of DNase I hypersensitive sites in single cells and FFPE tissue samples. .* Nature 2015. **528**: p. 142-146.
10. Consortium, T.E.P., *An integrated encyclopedia of DNA elements in the human genome. .* Nature 2012. **489**(57-74).
11. Maurano, M.T. and J.A. Stamatoyannopoulos, *Previews Taking Stock of Regulatory Variation.* Cell Systems 2015. **1**: p. 18-21.
12. Pott, S. and J.D. Lieb, *Single-cell ATAC-seq: strength in numbers.* Genome Biology 2015. **16**: p. 172.
13. Zamanighomi, M., et al., *Unsupervised clustering and epigenetic classification of single cells.* Nature Communications, 2018. **9**.
14. Boer, C.G.d. and A. Regev, *BROCKMAN: deciphering variance in epigenomic regulators by k-mer factorization.* BMC Bioinformatics, 2018. **19**(253).
15. González-Blas, C.B., et al., *Cis-topic modelling of single-cell epigenomes.* bioRxiv, 2018.
16. Deerwester, S., et al, *Improving Information Retrieval with Latent Semantic Indexing.* Proceedings of the 51st Annual Meeting of the American Society for Information Science, 1988. **25**: p. 36-40.

17. Cusanovich, D.A., et al., *A Single-Cell Atlas of In Vivo Mammalian Chromatin Accessibility.* Cell, 2018. **174**: p. 1309-1324.
18. Trapnell, C., et al., *The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. .* Nature Biotechnology 2014. **32**: p. 381-386.
19. Pliner, H., et al., *Chromatin accessibility dynamics of myogenesis at single cell resolution.* bioRxiv, 2017.
20. Buenrostro, J.D., et al., *Integrated Single-Cell Analysis Maps the Continuous Regulatory Landscape of Human Hematopoietic Differentiation.* Cell, 2018. **173**(6): p. 1535-1548.
21. Kohonen, T., *SELF-ORGANIZED FORMATION OF FEATURE MAPS. .* Cybern Syst, Recognit, Learn, Self-Organ, 1984: p. 3-12.
22. Kohonen, T., *Self-organized formation of topologically correct feature maps. .* Biological Cybernetics 1982. **43**: p. 59-69.
23. Kiang, M.Y. and A. Kumar, *An Evaluation of Self-Organizing Map Networks as a Robust Alternative to Factor Analysis in Data Mining Applications.* Information Systems Research, 2001. **12**(2): p. 177-194.
24. Longabaugh, W.J.R., et al., *Bcl11b and combinatorial resolution of cell fate in the T-cell gene regulatory network. .* PNAS in press 2017.
25. Mortazavi, A., et al., *Integrating and mining the chromatin landscape of cell-type specificity using self-organizing maps. .* Genome Research 2013. **23**: p. 2136-2148.
26. Cheng, Y., et al., *Principles of regulatory information conservation between mouse and human.* Nature 2014. **515**: p. 371-375.
27. Yue, F., et al., *A comparative encyclopedia of DNA elements in the mouse genome. .* Nature 2014. **515**: p. 355-364.
28. Kim, D.H., et al., *Single-Cell Transcriptome Analysis Reveals Dynamic Changes in lncRNA Expression during Reprogramming. .* 16, 2015(88-101).
29. Ferreir´ os-Vidal, I.e.a., *Genome-wide identification of Ikaros targets elucidates its contribution to mouse B-cell lineage specification and pre-B-cell differentiation.* Blood, 2013. **121**(1769-1782).
30. Leland McInnes, J.H., *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.* 2018.
31. Patzelt, T., et al., *Foxp1 controls mature B cell survival and the development of follicular and B-1 B cells.* PNAS, 2018(March 5, 2018).
32. Ferreirós-Vidal, I., et al., *Genome-wide identification of Ikaros targets elucidates its contribution to mouse B-cell lineage specification and pre-B-cell differentiation.* Blood, 2013. **121**: p. 1769-1782.
33. McLean, C.Y., et al., *GREAT improves functional interpretation of cis-regulatory regions.* Nature Biotechnology, 2010. **28**: p. 495-501.
34. Sellars, M., P. Kastner, and S. Chan, *Ikaros in B cell development and function.* World J Biol Chem., 2011: p. 132-139.
35. Marke, R., F.N.v. Leeuwen, and B. Scheijen, *The Many Faces Of IKZF1 In B-Cell Precursor Acute Lymphoblastic Leukemia. .* Haematologica, 2018. **103**.
36. Escamilla-Powers, J.R., et al., *The Tumor Suppressor Protein HBP1 Is a Novel c-Myc-binding Protein That Negatively Regulates c-Myc Transcriptional Activity. .* JBC, 2010. **285**.

37. Bansod, S., R. Kageyama, and T. Ohtsuka, *Hes5 regulates the transition timing of neurogenesis and gliogenesis in mammalian neocortical development.* The Company of Biologists Ltd | Development, 2017. **144**: p. 3156-3167.
38. Neph, S., et al., *Circuitry and dynamics of human transcription factor regulatory networks.* Cell, 2012. **150**(6): p. 1274-1286.
39. Zhang, Y., et al., *High Throughput Determination of TGFβ1/SMAD3 Targets in A549 Lung Epithelial Cells.* PLOS One, 2011.
40. Ross, J., et al., *GATA-1 Utilizes Ikaros and Polycomb Repressive Complex 2 To Suppress Hes1 and To Promote Erythropoiesis.* Molecular and Cellular Biology, 2012. **32**(18): p. 3624-3638.
41. Yoshida, T., et al., *Transcriptional regulation of the Ikzf1 locus.* Blood, 2013. **122**(18): p. 3149-3159.
42. Liu, G.J., et al., *Pax5 loss imposes a reversible differentiation block in B-progenitor acute lymphoblastic leukemia.* Genes & Development, 2018. **32**: p. 15-16.
43. Yu, D., et al., *Oscillation between B-lymphoid and myeloid lineages in Myc-induced hematopoietic tumors following spontaneous silencing/reactivation of the EBF/Pax5 pathway.* Blood, 2003. **101**: p. 1950-1955.
44. AD, R., et al., *The harmonizome: a collection of processed datasets gathered to serve and mine knowledge about genes and proteins.* Database(Oxford), 2016. **Jul 3;2016**.
45. Huang, M., et al., *dbCoRC: a database of core transcriptional regulatory circuitries modeled by H3K27ac ChIP-seq signals.* Nucleic Acids Research, 2018. **46**(D1): p. D71-D77.
46. CY, P., et al., *Tissue-aware data integration approach for the inference of pathway interactions in metazoan organisms.* Bioinformatics, 2015.
47. Cao, J., et al., *Joint profiling of chromatin accessibility and gene expression in thousands of single cells.* Science, 2018. **361**(6409): p. 1380-1385.
48. Faridani, O.R., et al., *Single-cell sequencing of the small-RNA transcriptome.* Nature Biotechnology, 2016. **1-5**.
49. Angermueller, C., et al., *Parallel single-cell sequencing links transcriptional and epigenetic heterogeneity.* Nature Methods, 2016. **13**: p. 229-232.
50. Genshaft, A.S., et al., *Multiplexed, targeted profiling of single-cell proteomes and transcriptomes in a single reaction.* Genome Biology, 2016. **17**.
51. Hou, Y., et al., *Single-cell triple omics sequencing reveals genetic, epigenetic, and transcriptomic heterogeneity in hepatocellular carcinomas.* Cell Research, 2016. **26**: p. 304-319.
52. Macaulay, I.C., et al., *G{&}T-seq : parallel sequencing of single- cell genomes and transcriptomes. .* Nature Methods, 2015. **12**.
53. *https://github.com/jlmelville/uwot.*
54. Ramirez, R.N., et al., *Dynamic Gene Regulatory Networks of Human Myeloid Differentiation.* Cell Systems, 2017. **4**: p. 416-429.
55. Patro, R., et al., *Salmon provides fast and bias-aware quantification of transcript expression.* Nature Methods., 2017.
56. Langmead, B. and S.L. Salzberg, *Fast gapped-read alignment with Bowtie 2.* Nat Methods, 2012. **9**: p. 357-359.
57. Strino, F. and M. Lappe, *Identifying peaks in *-seq data using shape information.* BMC Bioinformatics, 2016. **17 Suppl 5**: p. 206.

58. Akaike, H., *Information theory and an extension of the maximum likelihood principle.* International Symposium on Information Theory, 1973: p. 267-281.
59. Kulakovskiy, I.V., et al., *HOCOMOCO: Expansion and enhancement of the collection of transcription factor binding sites models.* Nucleic Acids Research, 2016. **44**(D116–D125).
60. Grant, C.E., T.L. Bailey, and W.S. Noble, *FIMO: scanning for occurrences of a given motif.* Bioinformatics (Oxford, England), 2011. **27**(1017-1018).

CHAPTER 3

Mapping the Developmental Gene Regulatory Networks of *Xenopus tropicalis* Mesendoderm Development using Self-Organizing Maps

Note: (1) Dr. Kitt Paraiso from the Cho lab at UCI performed the RNA-seq and ChIP/ATAC-seq data collection, mapping, gene quantifications, and peak calling.
(2) Dr. Ken Cho and Dr. Ira Blitz used their considerable expertise to help interpret the results.

Chapter 3

Building Developmental Gene Regulatory Networks for *Xenopus tropicalis* Mesendoderm Development with Self-Organizing Maps

3.1 Abstract

Deciphering developmental gene regulatory networks (GRNs) is one of the long-term challenges of regulatory biology. GRNs are traditionally built piecemeal with each connection representing months or years of work. The recent availability of multiple highly-dimensional functional genomic data sets should allow us to build GRNs directly. Here, we focus on building the mesoendoderm specification GRN in *Xenopus tropicalis* using ChIP-seq for multiple different TFs and histone marks combined with RNA-seq collected at multiple embryonic stages and conditions during gastrulation. We used the Linked Self-Organizing Maps to identify gene and chromatin modules that change in a coordinated manner followed by a foxh1-centric network analysis which recovers most of the known GRN connections as well as a novel set of predicted linkages that are candidates for validation and incorporation in the mesoendoderm specification GRN.

3.2 Introduction

Gastrulation is one of the most important times during the development of most animals, as the single-layered blastula develops into a multilayered organism. The transition from a single sheet of cells to an embryo with distinct cell lineages, multiple basic axes, and internalized cell types over the course of a few hours is an incredibly complex process that requires a set of very tightly controlled set of regulatory genes working in

concert across multiple cells[1]. A comprehensive gene regulatory network (GRN) that explains the process of germ layer specification is an important open question in developmental biology. All gene regulation programs from cell differentiation to tissue/organ development to final cellular states depend on inputs from the previous state, and gastrulation sits at a critical junction of germ layer specification.

Gastrulation has been extensively studied across many different animals, but *Xenopus tropicalis* has proven to be an ideal model system for its study in vertebrates[2] due to a number of factors. First, large broodsizes of between 500 to 3000+ synchronized embryos becoming available per lay provides plenty of cells for bulk functional genomics. Second, the embryos are large, transparent, and accessible to manipulation, such as morpholino knock-downs. Third, unlike the better-known *Xenopus laevis*, *X. tropicalis* is diploid, which further simplifies genomic analyses. Finally, amphibians occupy a good middle distance evolutionarily from human compared to fish or invertebrates and therefore the obtained results remain relevant for human health.

There have been multiple previous efforts to compile the available molecular and genomic data from Xenopus into GRNs that describe mesendoderm development during gastrulation in a bottom-up manner as a core GRN with 23 transcription factors (TFs) and 12 growth factors for a total of 96 validated network connections, each of which is the result of one or more publications[3, 4]. One of these maternal transcription factors in particular, foxh1, has been shown[5] to be essential in mesendodermal development in Xenopus through its co-binding with the smad2/3, a nodal signaling pathway phosphorylation target. In addition, foxh1 was also shown to act independently of this interaction and regulate many non-nodal targets and, therefore, has additional regulatory

functions[5]. While many genes are regulated by foxh1, several known targets from other systems have not yet been found, including negative regulation of nodal5, nodal6, and flk1(kdr) in *X. laevis*[6] and zebrafish[7]. Thus, a more in-depth network analysis for foxh1 would be helpful in determining its function and targets during gastrulation

However, to complete a mesendoderm specification GRN in *X. tropicalis* at the same level of detail as the dorsal-ventral determination in Drosphila[8] would require many more years of work. Recently, there have been numerous improvements in building GRNs in a "top-down" approach through either analysis of putative TF footprints from deeply sequenced DNase/ATAC-seq data[9] or un-supervised learning of highly-dimensional data from multiple data sources to find potentially co-regulated regions as was described in Chapter 2. Large-scale integration methods such as Self-organizing maps (SOM) normally need hundreds of datasets to work well. Given the number of available datasets in *X. tropicalis*, we felt that we had sufficient dataset complexity to properly separate genes and chromatin regions into gene regulatory modules. In this chapter, we applied our linked SOM method to map the mesoendoderm GRN, which we validate computationally by characterizing the recovery of known connections as well as to identify potential additional network connections for future validation.

3.3 Results

3.3.1 Integration of highly-dimensional genomic bulk data types using SOMs

To investigate the *Xenopus tropicalis* mesendoderm GRN, our collaborators in the Cho lab assembled a highly dimensional data set of 95 RNA-seq and 63 ChIP-seq as well as ATAC-seq experiments taken during embryonic stages 8-12 using various conditions such

as morpholinos or special dissections (Fig. 1a, Fig. 6, Fig. 7). The goal of this project was to integratively analyze the data from these experiments to identify both known as well as novel TF-gene interactions in the core mesendoderm GRN beginning with foxh1, which is a key transcription factor in this process by using our linked SOM the computational pipeline rom Chapter 2 to connect the gene expression and chromatin data first analyzed separately as shown in Fig. 1b.

We metaclustered each SOM separately to identify distinct sets of genes or chromatin regions that have a similar experimental profile and represent groups of similarly-regulated genes and chromatin regions. To further improve our ability to detect co-regulation, the metaclusters from each set of experiments are convolved through the SOM Linking algorithm into linked metaclusters (LM). These LM contain sets of chromosomal regions that have similar chromatin signals in the DNA experiments and are near genes that also share a similar RNA-seq profile. LMs can be further mined for motifs that can be built into gene regulatory networks.

3.3.2 Identification of mesendoderm development-specific gene expression modules using a RNA SOM

To identify the different gene modules present in *Xenopus tropicalis* mesendoderm development, we began by training a 60x90 SOM on the collected RNA-seq data using the 31,399 genes using the v9 gene annotations. We recovered 84 distinct SOM metaclusters that capture the different gene expression profiles present in the included data. Of these, 13 contain genes that were present in the core mesendodermal network (Fig. 2a). Plotting the signal in the wild-type experiments for these clusters ordered in developmental time

revealed that each has a distinct temporal profile as expected (also see Fig. 12). For example, metacluster 11, which contained the key mesendodermal TFs hhex and gsc, began to show signal in embryonic stage 8, peaks in stage 9 and slowly curves down in stages 10 and on. Meanwhile, metacluster 76, containing foxa2, did not appear until stage 10 and steadily increased until the end of the experimental measurements.

Each metacluster contained genes that have similar profiles across all experiments (Fig. 2b, Fig. 10). For example, the 110 genes in metacluster 11 had similar responses to multiple morpholino and spatial conditions. However, genes in different metaclusters had different responses to the knockdown experiments (Fig. 2c). Some metaclusters, such as 11 and 16, were depleted in stage 10.5 of the foxh1 morpholino experiment while others, such as 23 and 58, were enriched. Additionally, each metacluster appeared to have unique functions after performing GO enrichment analysis on the separate gene lists (Fig. 2d, Fig. 8). For example, metacluster 11 contained genes with functions related to Dorsal/ventral patterning and cell fate, whereas metacluster 16 had genes related to morphogenesis and development of tissues and structure. Each of these is consistent with activities that cells at these stages must complete to perform mesendoderm development.

3.3.3 Investigating co-regulation and histone codes around cis-regulatory elements in *Xenopus tropicalis* mesendoderm development using a chromatin SOM

In order to explore the role of the chromatin landscape during mesendoderm development in *Xenopus tropicalis*, we also trained a 40x60 SOM on the RPKMs of each ChIP/ATAC-seq experiment over the 731,726 partitioned genome segments, and SOM metaclustering identified 88 distinct DNA profiles present in the data (Fig. 3). We analysed

adjacent TF ChIP-seq experiments on the hierarchical clustering to explore whether the SOM accurately captured known protein co-regulation. We found that eomes and vegt ChIP-seq experiments from stage 8 clustered close together (boxed in brown), and it has been shown that zygotic vegt and eomes cobind in Xenopus[10]. Similarly, otx1 and vegt (boxed in blue) are both enriched in the vegetal pole during embryonic development[3] and thus should regulate a similar set of genes. In addition, gsc is known to be activated by smad2/3 and are found in very similar embryonic programs[11]. There are additional adjacent experiments with known co-binding in other species such as Brachyury and sox17, which are known to co-bind frequently in mouse[12] (boxed in red), and foxh1 and smad1 (boxed in green), which have a highly interactive relationship in many species including human, mouse, zebrafish, and Xenopus[5, 13]. We were unable to find clear results showing the co-regulation of sia and ventx2 (boxed in orange), even though both of these genes are important for dorsal-ventral patterning and should regulate similar targets.

Given that foxh1 was the central player in mesendoderm development processes we focused on, we collapsed the above heatmap to only focus on those metaclusters that were sufficiently enriched in foxh1 (Fig. 4a, Fig. 11). Each had a unique combination of histone and TF ChIP signal, which was very valuable for classification. For example, metacluster 51 has a strong H3K27me3 and H3K4me1 signal, which indicated that this metacluster contains inactive promoters, whereas metacluster 77 replaced the H3K27me3 signal with H3K27ac, which indicated active promoters. Metaclusters 71 and 58 had a strong H3K4me1 without any of the promoter histone codes indicating they contained active enhancers. Metacluster 62 appeared contain regions that refused to open even with foxh1 acting as a pioneer factor.

The metaclusters had functional differences as well (Fig. 4b, Fig. 9), which could be seen by the GO enrichment of the closest genes to these foxh1-enriched regions. Metacluster 62 and 51 contained terms associated with later developmental stages, which is consistent with the lack of activity in the other experiments from stages 8-12 for 62 and the strong H3K27me3 signal in 51. Metaclusters 77 and 45 have terms that should be strongly unregulated in the embryonic stages present, with metacluster 77 having terms more related to pattern formation and regionalization and metacluster 45 more related to developmental processes.

In order to identify the TFs that may control these different profiles, we performed motif analysis on each set of regions with the Hocomoco v11 human motif database[14]. After we removed all overlapping motifs, we recovered 63 unique to metacluster 45 including several important mesendoderm development TFs such as smad2/3, sox7, and ventx. Similarly, we found 56 unique TFs in metacluster 77 including regionalization and patterning TFs such as foxa2, creb3, and tcf3. Finally, we found 37 unique TFs in metacluster 51 such as gata6, irx2, and tead1 (See Fig. 13 for full list). The recovery of the Tead1 motif was interesting because it is a known repressor in stem cells[15] that may cause these regions to maintain their H3K27me3 signal. With this analysis, we believe that the DNA SOM metaclusters captured the structure of the data very well and could be linked to the RNA SOM and for further network analysis.

3.3.4 Application of Linked SOMs to perform multi-omic data integration and generate developmental gene regulatory networks

In Chapter 2, we developed the Linked SOM method for multi-clustering data sets containing multiple types of genome experiments. The idea behind this algorithm is to improve motif capture by increasing the density of these motifs in the genomic regions analyzed. To improve the density, the regions should be as similarly regulated as possible. As such, we convolved the metaclusters of a scRNA-seq SOM and a scATAC-seq SOM to build sets of genome regions that had similar scATAC-seq profiles that were near genes with similar scRNA-seq profiles. We believed that this method would work similarly with this current collection of Xenopus RNA-seq and ChIP/ATAC-seq data, and therefore, we applied the Linked SOMs algorithm to the two SOMs described in the previous sections.

The algorithm created a set of 88x84=7,392 Linked Metaclusters (LM) such that each contained regions from the same DNA metacluster and the nearby genes for these regions were in the same RNA metacluster. A summary of the number of detected regions can be found in Fig. 14. We utilized the ChIP experiments to build a database of Xenopus-specific motifs for tcf3, eomes, foxa2, foxh1, gsc, mix1, otx1, otx2, smad2/3, sox17, sox7, and vegt. These motifs were used to scan each LM with a high q-value (for motif work) of .1 (Fig. 15), and the detected motifs were subject to the same LM enrichment as in Chapter 2. In all, we detected 30,634 unique network connections (79,410 motifs) for the 12 different TFs scanned. Of these, the largest portion belonged to foxh1 with 46,531 detected motifs near 12,831 genes. Of these motifs, 41,960 (~90%) overlap foxh1 ChIP-seq peaks. Of note, the missing negative regulatory targets from previous works, nodal5, nodal6, and kdr, where among this set. Of these, nodal5 and nodal6 were in metaclusters (55 and 41

respectively) that showed significant enrichment in the foxh1 morpholino experiment during stage 10.5. Additionally, 159 TFs (1,788 genes) were in the same RNA metaclusters as genes from the core network (Fig 5a), only 4 of which do not overlap foxh1 ChIP-seq peaks (starred). 35 of these TFs were from the core network (Fig 5b), with all of the known connections recovered (in black).

3.4 Discussion

In this study, we used self-organizing maps (SOMs) to analyze integratively highly-dimensional genomic data sets of gene expression and chromatin state during mesendoderm development in *Xenopus tropicalis*. Our analysis of the collected gene expression data combined morpholino, wild-type, and spatial data to group genes into developmental time-specific clusters without using any knowledge of when the time points were taken. Additionally, the SOM clustered the genes based on the effect that various morpholinos had on them, thus capturing similar regulation. Finally, the groups had distinct functional and/or developmental differences suggesting that they may be co-regulated.

The SOM analysis of the ChIP-seq and ATAC-seq data as the metaclustering was able to re-capture many known co-binding/co-regulation interactions from Xenopus or other vertebrates. When we focused on foxh1, we found 12 different metacluster profiles including particular histone modification combinations that allowed us to classify these regions by their nearby gene's transcriptional activity. These different signal profiles had different functional and motif enrichments, which provided additional evidence that they were true genomic modules.

Combining the gene and genomic region modules into one analysis, we were able to achieve our goal of recovering the known Xenopus developmental gene regulatory network for foxh1 by grouping genome regions/gene pairs through the Linked SOMs method. Through the inheritance of each SOM's properties into the joint analysis, the Linked Metaclusters (LMs) contain regions that should be very similarly regulated and contain a higher density of the same motifs. This, in turn, allows for an in-depth motif analysis with multiple rounds of Type 1 error correction. This allowed us to expand the foxh1-centric GRN for Xenopus mesendoderm development in a "top-down" manner and provide multiple new targets for ongoing validation and study.

3.5 Figures

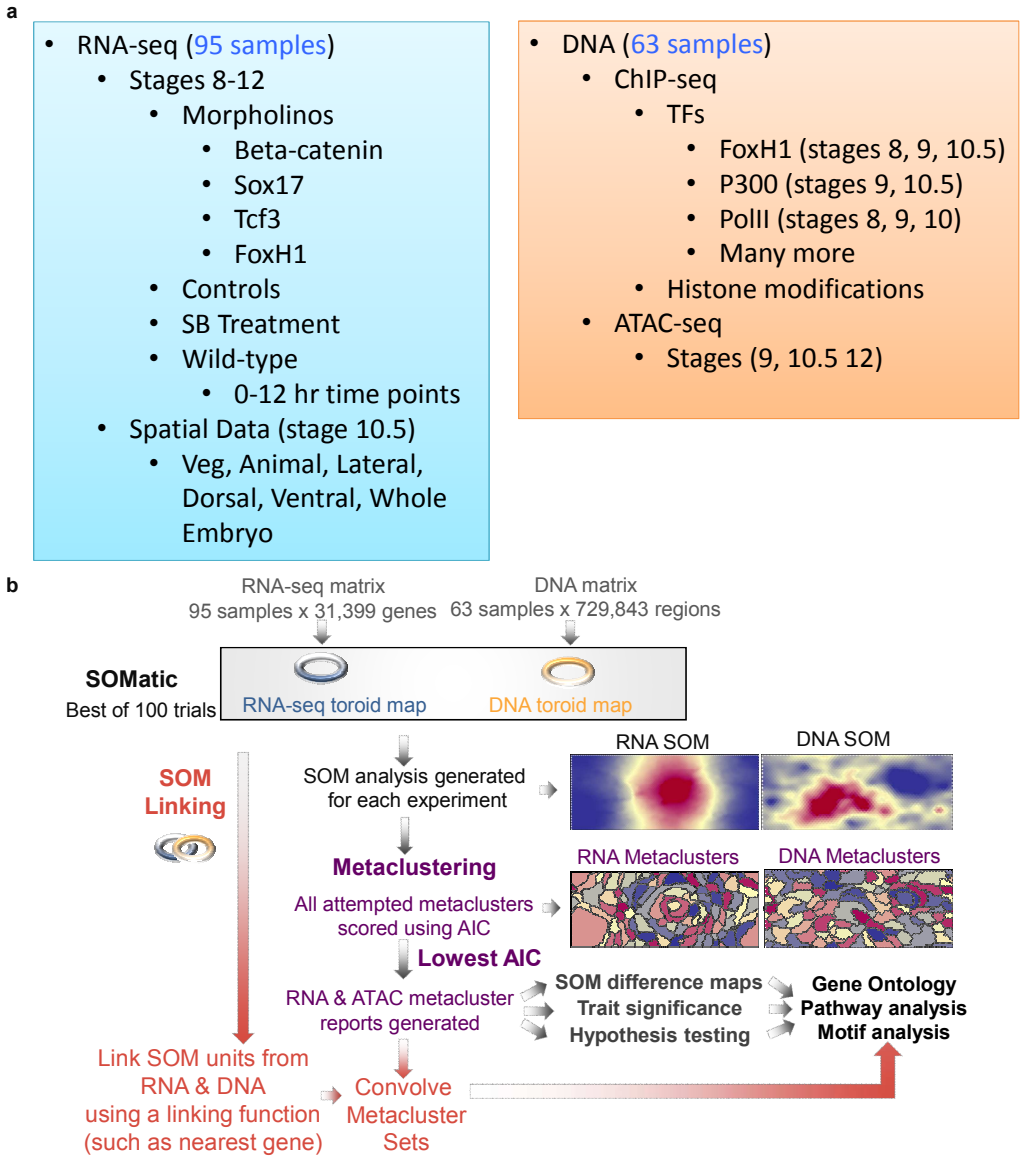


Figure 3.1. Bulk multi-data integration through SOM Linking
(a) The *Xenopus tropicalis* genomic data sets used for SOM analysis in this chapter. **(b)** These data sets were converted into training matrices and had SOMs built using SOMatic. This was followed by metaclustering and SOM Linking. The pair-wise linked metaclusters (LM) were mined for regulatory connections and built into networks.

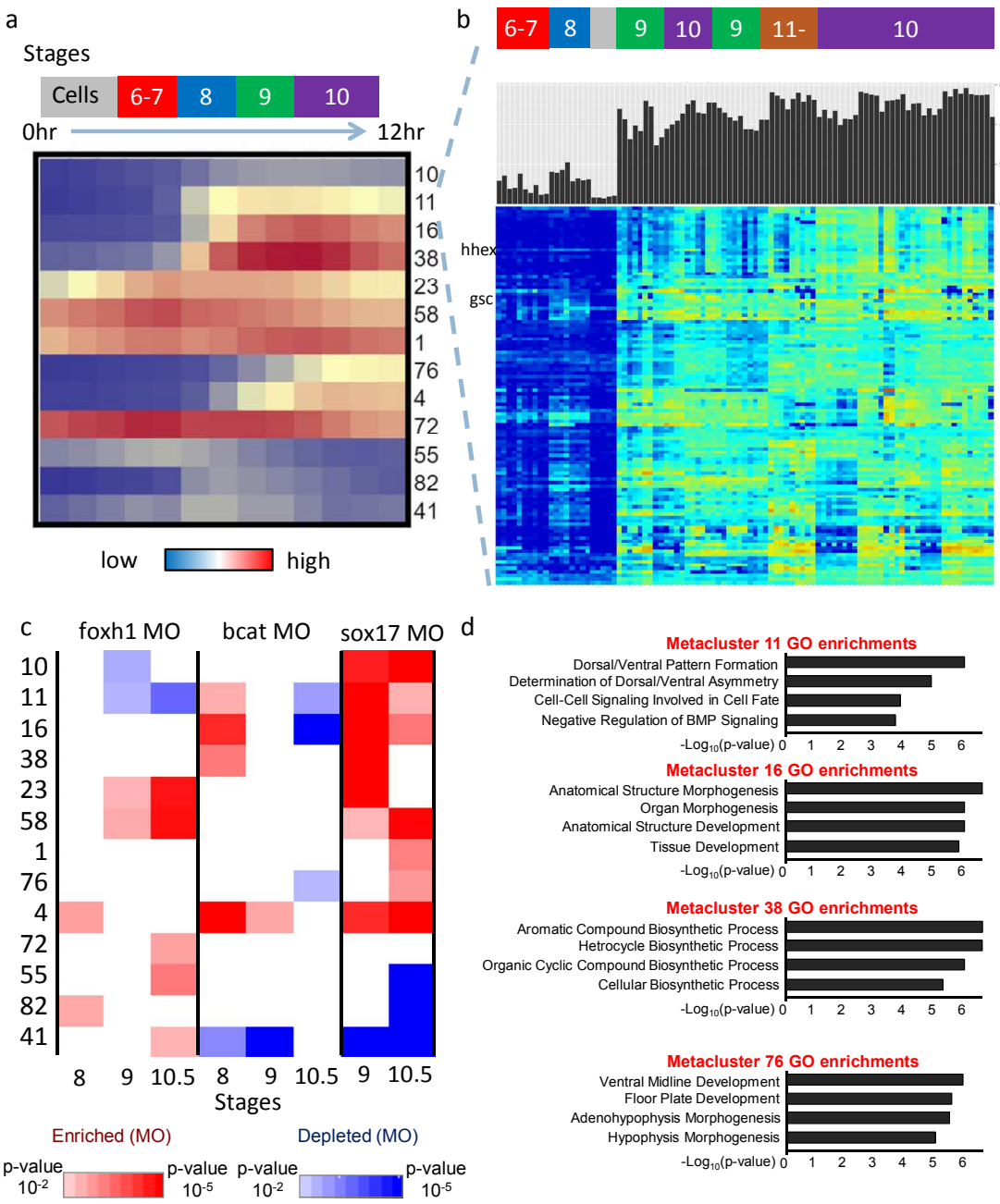


Figure 3.2. RNA SOM metaclustering reveals developmental gene modules
(a) Metaclusters containing genes from the core mesendoderm network show unique temporal dynamics during development. **(b)** Example of the genes within metacluster 11 such as hhex and gsc generally follow the eigen-profile shown at the top. They begin to come on in stage 8 and remain past stage 11 at a medium level. **(c)** Two-tailed hypothesis analysis applied on gene metaclusters after subtracting out control experiments. Each metacluster responded to each morpholino experiment differently at different time points. **(d)** Each metacluster had unique functional enrichments supporting the coherence of these clusters.

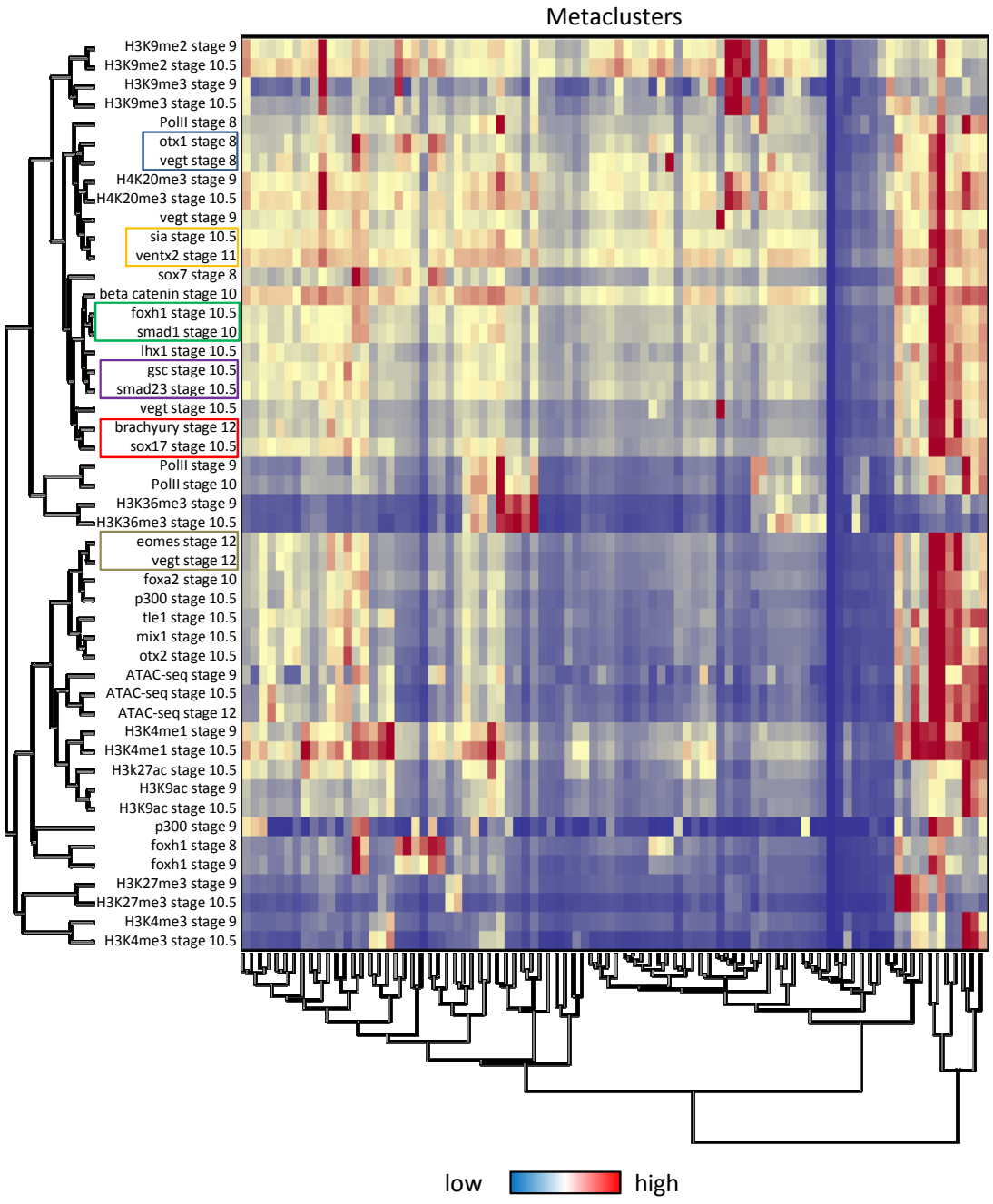


Figure 3.3. Full DNA metacluster heatmap captures known co-regulatory interactions
The full set of eigenprofiles revealed that several experiments had very similar results on the collected genome region clusters. Some of these, marked by colored boxes, are known co-regulatory interactions in Xenopus or vertebrae in general.

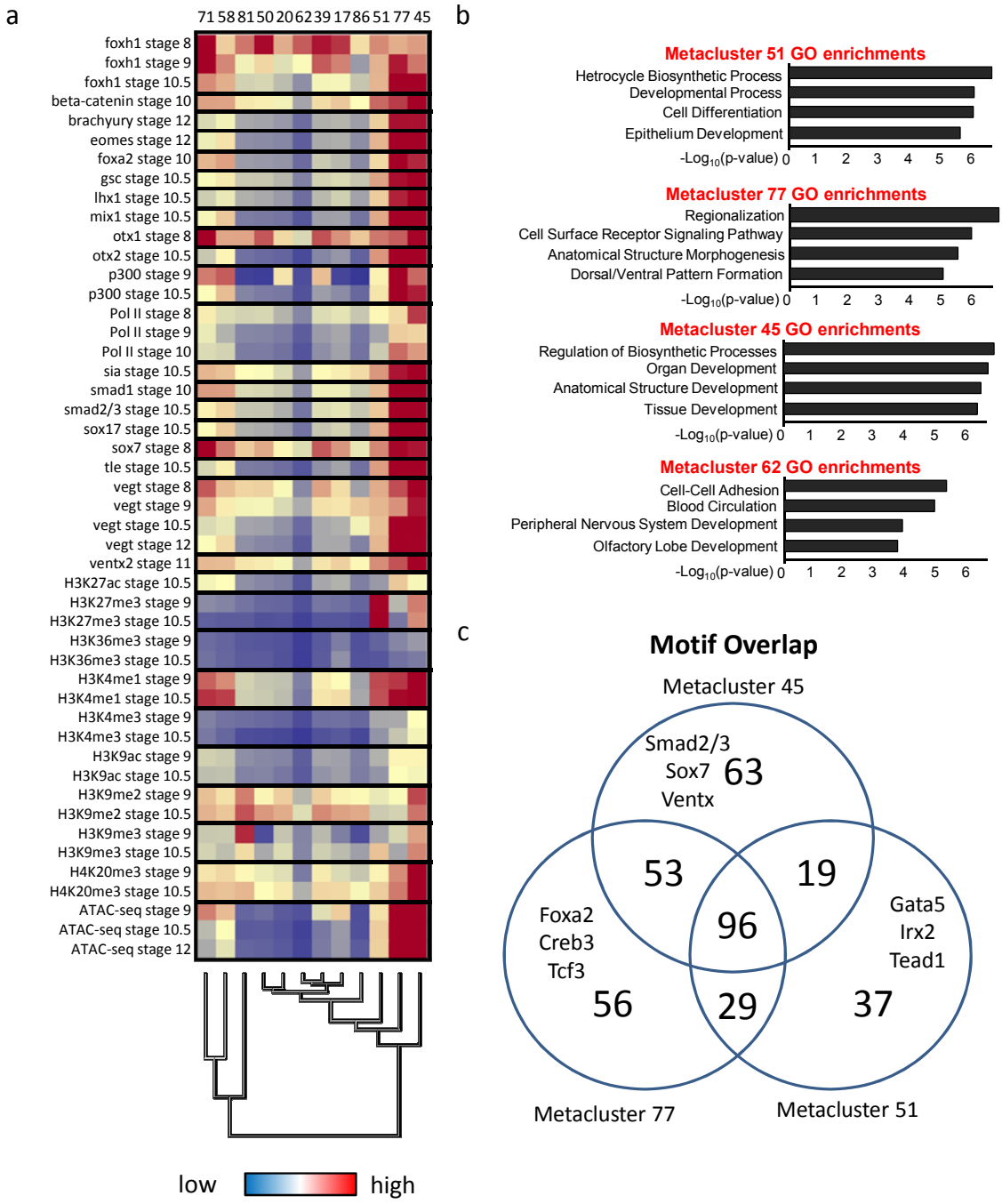


Figure 3.4. Detailed analysis on foxh1 ChIP-enriched metaclusters reveals different methods of action

(a) The collapsed heatmap of foxh1 ChIP-enriched metaclusters shows many different patterns of co-regulation present in foxh1-bound CRMs. **(b)** Genes near these genomic modules have distinct functional enrichments, which gives evidence that these are true regulatory modules. **(c)** The metaclusters also have a distinct motif signature with some extremely important developmental TFs specifically enriched in a metacluster.

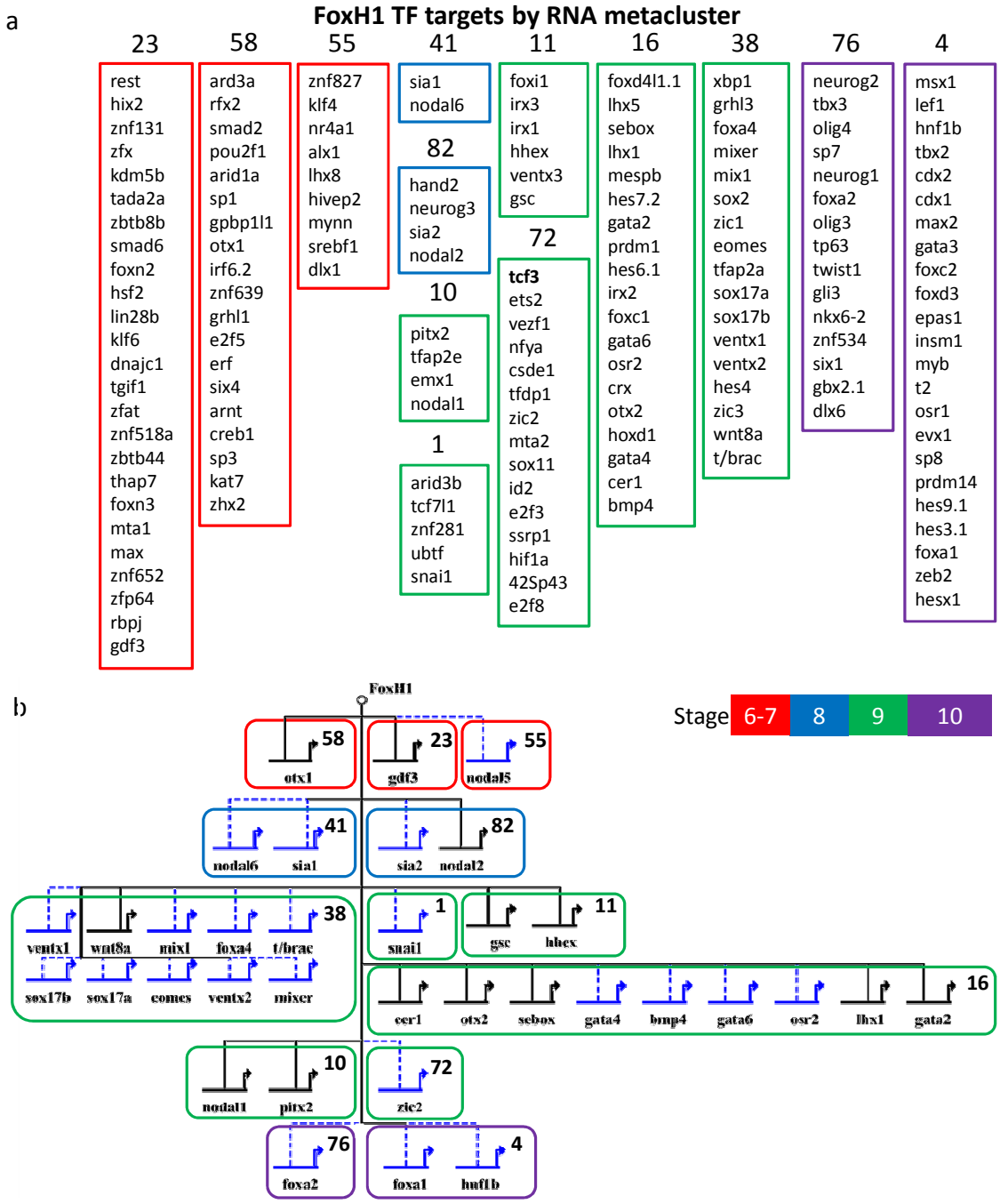


Figure 3.5. Foxh1-focused network analysis re-computes core network and provides additional potential connections

(a) This list of predicted TF foxh1 targets is displayed by RNA metacluster, which in turn, also, sorts them by developmental time. This list contains over 120 new predicted foxh1 TF targets, each of which have nearby foxh1 ChIP signal and are active during mesendoderm development. Each of these could potentially appear in a core network in future works. **(b)** The new foxh1-targeted core network with 22 new connections adding additional evidence that foxh1 has a central role in the regulation of mesendoderm development as predicted.

Stages/Timepoints	Treatment	Spacial	Source	Reps
7, 8, 9.1, 9.2, 10, 10.5, 11, 12	Beta-Catenin MO	Whole Embryo	Zorn	1
7, 8, 9.1, 9.2, 10, 10.5, 11, 12	Beta-Catenin Uninj Ctrl	Whole Embryo	Zorn	1
8, 9, 10.5	DMSO Treatment	Whole Embryo	Cho	A, B
8, 9, 10.5	SB Treatment	Whole Embryo	Cho	A, B
8, 9, 10.5	FoxH1 MO	Whole Embryo	Cho	A, B
8, 9, 10.5	FoxH1 Uninj Ctrl	Whole Embryo	Cho	A, B
9.1, 9.2, 10, 10.5, 11, 12	Sox17 MO	Whole Embryo	Cho	1, 2
9.5, 10, 10.5	Tcf3 MO	Whole Embryo	Zorn	1, 2
0-12 hr (Each hour)	Wild Type	Whole Embryo	Cho/Khokha	A, B
10.5	Wild Type	Animal Cap	Cho	1, 2
10.5	Wild Type	Dorsal	Cho	1, 2
10.5	Wild Type	Lateral	Cho	1, 2
10.5	Wild Type	Ventral	Cho	1, 2
10.5	Wild Type	Vegetal	Cho	1, 2
10.5	Wild Type	Whole Embryo	Cho	1, 2

Figure 3.6. Detailed list of RNA-seq data collected for SOM analysis
This table contains details of the collected RNA-seq data for this work. The list has 95 total experiments from the Cho, Zorn, and Khokha labs from a variety of stages, timepoints, treatments, and special dissections.

Assay/Factor	Stages	Source	Peak Number
ATAC-seq	9, 10.5, 12	Veenstra	29349, 82011, 80965
beta-catenin	10	Hoppler	
beta-catenin Ctrl	10	Hoppler	
brachyury	12, 20		8842, 12598
eomes	12		16199
foxa2	10		33340
foxh1	8, 9, 10.5		90784, 54359, 2728
gsc	10.5		5013
H3K27ac	10.5		65563
H3K27me3	11		1181
H3K27me3 Ctrl	11		
H3K36me3	9, 10.5		24157, 39868
H3K4me3	9, 10.5, 11		12528, 19998, 12550
H3K4me3 Ctrl	11		
H3K9ac	9, 10.5		15597, 15876
H3K9me2	9, 10.5		2817, 3
H3K9me3	9, 10.5		3575, 20995
H4K20me3	9, 10.5		192, 60
lhx1	10.5		270
mix1	10.5		57933
otx1	8	Cho	4296
otx2	10.5		29153
p300	9, 10.5, 11		15559, 23633, 194
p300 Ctrl	11		
Pol II	8, 9, 10, 10.5		4728, 19853, 36526
sia	10.5		4421
smad1	10		3158
smad2/3	10.5		1607
sox17	10.5		14628
sox7	8	Cho	23973
Input	8,9,10.5		
tle	10.5		70991
vegt	8, 9, 10.5,12		551, 0, 30112, 27686
ventx2	11	Cho	140

Figure 3.7. Detailed list of ATAC-seq/ChIP-seq experiments collected for the DNA SOM

This table contains details of the collected ATAC-seq/ChIP-seq data for this work. The list has 63 experiments from the Cho, Veenstra, and Hoppler labs from a variety of stages, timepoints, and ChIP factors. The number of called peaks for each set of experiments is also listed.

Metacluster 10	
regulation of calcium ion transport into cytosol	0.002625
negative regulation of calcium ion transport into cytosol	0.002625
early neuron differentiation in forebrain [GO:0021862]	0.002625
regulation of release of sequestered calcium ion into cytosol	0.002625
negative regulation of release of sequestered calcium ion into cytosol	0.002625
negative regulation of calcium ion transport	0.002625
Metacluster 11	
dorsal/ventral pattern formation	2.186638e-6
determination of dorsal/ventral asymmetry	2.326149e-5
cell-cell signaling involved in cell fate	1.500731e-4
negative regulation of BMP signaling pathway	3.904300e-4
Spemann organizer formation	4.073744e-4
developmental induction	4.500731e-4
Metacluster 16	
anatomical structure morphogenesis	4.085410e-8
organ morphogenesis	1.105024e-6
anatomical structure development	1.712032e-6
tissue development	2.316352e-6
organ development	2.597125e-6
epithelium development	7.785168e-6
Metacluster 38	
aromatic compound biosynthetic process	1.867651e-6
heterocycle biosynthetic process	1.931965e-6
organic cyclic compound biosynthetic process	2.637073e-6
cellular biosynthetic process	2.768408e-6
regulation of macromolecule metabolic process	3.468496e-6
regulation of primary metabolic process	4.022989e-6
Metacluster 23	
nucleic acid metabolic process	1.246455e-6
primary metabolic process	2.063619e-6
organic cyclic compound metabolic process	3.743976e-6
RNA metabolic process	6.292219e-6
cellular nitrogen compound metabolic process	6.637755e-6
macromolecule modification	7.454182e-6
Metacluster 58	
phosphate-containing compound metabolic process	2.572503e-5
phosphorus metabolic process	3.607997e-5
protein metabolic process	3.658698e-5
RNA metabolic process	6.037699e-5
phosphorylation	7.309570e-5
protein phosphorylation	1.225857e-4
Metacluster 1	
proteasomal ubiquitin-independent protein catabolic process	9.588302e-5
nucleoside triphosphate metabolic process	1.002809e-4
organonitrogen compound metabolic process	1.646930e-4
regulation of protein complex assembly	3.238906e-4
purine nucleoside triphosphate metabolic process	3.577242e-4
purine ribonucleoside triphosphate metabolic process	3.577242e-4
Metacluster 76	
ventral midline development	5.993062e-7
floor plate development	3.499960e-6
adenohypophysis morphogenesis	6.152375e-6
hypophysis morphogenesis	1.525018e-5
diencephalon morphogenesis	1.525018e-5
floor plate formation	3.024119e-5
Metacluster 4	
anatomical structure formation involved in morphogenesis	4.936150e-7
otic vesicle formation	1.036038e-6
formation of primary germ layer	1.749520e-6
otic vesicle morphogenesis	4.881844e-6
columnar/cuboidal epithelial cell differentiation	9.173287e-6
otic vesicle development	1.090777e-5
Metacluster 72	
mitotic nuclear division	2.911910e-7
cell division	8.226029e-7
protein localization	1.384106e-6
macromolecule localization	1.539786e-6
organic substance transport	1.558890e-6
protein transport	1.779921e-6
Metacluster 55	
inactivation of MAPK activity	6.296554e-6
negative regulation of MAP kinase activity	1.693008e-5
negative regulation of MAPK cascade	3.700710e-5
negative regulation of protein serine/threonine kinase activity	4.648310e-5
negative regulation of protein kinase activity	4.903869e-4
negative regulation of kinase activity	5.490462e-4
Metacluster 82	
embryo development	3.127272e-4
embryo development ending in birth or egg hatching	6.088309e-4
chordate embryonic development	6.088309e-4
cardiogenic plate morphogenesis	7.159905e-4
embryonic heart tube formation	7.159905e-4
determination of intestine left/right asymmetry	0.001432
Metacluster 41	
positive regulation of pathway-restricted SMAD protein phosphorylation	8.222003e-7
SMAD protein signal transduction	8.222003e-7
transmembrane receptor protein serine kinase signaling pathway	2.668077e-5
regulation of MAPK cascade	3.827523e-5
positive regulation of phosphorylation	4.048251e-5
regulation of cell death	4.569667e-4

Figure 3.8. List of GO enrichments for 2nd half of RNA metaclusters in Figure 3.2
The above table lists the top 6 GO enrichments for the second half of the metaclusters from Figure 3.2 and their associated p-value. Each metacluster appears to have functional differences from the others.

Metacluster 71	
cell surface receptor signaling pathway	4.022739e-6
multicellular organismal development	6.524134e-6
enzyme linked receptor protein signaling pathway	1.258082e-5
anatomical structure development	3.210449e-5
regulation of developmental process	3.395734e-4
cellular response to growth factor stimulus	3.540328e-4
Metacluster 58	
anatomical structure development	8.999373e-11
tissue development	1.182942e-9
cell surface receptor signaling pathway	1.200572e-9
cell differentiation	4.775218e-9
anatomical structure morphogenesis	4.908902e-9
embryo development	7.478200e-9
Metacluster 81	
cell-cell adhesion	5.744243e-7
synaptic transmission	2.846910e-6
cell surface receptor signaling pathway	8.961430e-6
developmental process	1.595695e-5
single-organism developmental process	1.874529e-5
anatomical structure development	2.427965e-5
Metacluster 50	
cell surface receptor signaling pathway	1.212578e-6
single-organism developmental process	2.037885e-6
epithelium development	2.346471e-6
anatomical structure morphogenesis	2.409038e-6
regulation of multicellular organismal development	7.146743e-6
cardiovascular system development	7.151621e-6
Metacluster 20	
Wnt signaling pathway	1.517214e-4
regulation of biosynthetic process	1.696950e-4
negative regulation of nitrogen compound metabolic process	1.776099e-4
single-organism developmental process	1.797274e-4
cell-cell adhesion via plasma-membrane adhesion molecules	1.800672e-4
neuron projection morphogenesis	1.991242e-4
Metacluster 62	
cell-cell adhesion	6.455865e-4
cell morphogenesis involved in differentiation	6.824969e-4
regulation of blood pressure	7.018037e-4
olfactory bulb development	9.984525e-4
olfactory lobe development	9.984525e-4
biological regulation	0.001103

Metacluster 39	
regulation of nervous system development	9.254623e-6
regulation of neurogenesis	1.003801e-5
regulation of multicellular organismal development	1.192273e-5
enzyme linked receptor protein signaling pathway	2.646321e-5
regulation of cell differentiation	3.169157e-5
anatomical structure morphogenesis	3.374588e-5
Metacluster 17	
protein phosphorylation	6.291082e-7
multicellular organismal development	9.559791e-7
phosphate-containing compound metabolic process	1.329916e-6
macromolecule metabolic process	1.979958e-6
phosphorus metabolic process	2.162585e-6
neuron projection morphogenesis	2.580655e-6
Metacluster 86	
cell communication	2.351931e-8
nervous system development	2.436024e-8
signaling	2.800220e-8
single organism signaling	3.285711e-8
synaptic transmission	2.679522e-7
cell-cell signaling	1.165174e-6
Metacluster 51	
heterocycle metabolic process	1.336538e-8
developmental process	1.637837e-8
cell differentiation	1.535474e-7
epithelium development	4.687741e-7
embryo development	3.521240e-6
organ morphogenesis	5.351063e-6
Metacluster 77	
regionalization	3.962452e-8
cell surface receptor signaling pathway	1.825943e-6
anatomical structure morphogenesis	4.204158e-6
dorsal/ventral pattern formation	5.474042e-6
head development	3.828283e-5
brain development	2.761862e-4
Metacluster 45	
regulation of biosynthetic process	1.772537e-8
organ development	2.322214e-8
aromatic compound biosynthetic process	3.786947e-8
tissue development	2.422149e-7
tube development	1.540700e-6
biological regulation	2.205765e-6

Figure 3.9. List of GO enrichments for 2nd half of DNA metaclusters in Figure 3.4
The above table lists the top 6 GO enrichments for the closest genes to the genomic regions in the second half of metaclusters from Figure 3.4 and their associated p-value. Each of the metaclusters from that figure had foxh1 ChIP-seq enrichment. The functional differences shown in this and the previous table suggest that foxh1 works together with different factors to activate different developmental modules.

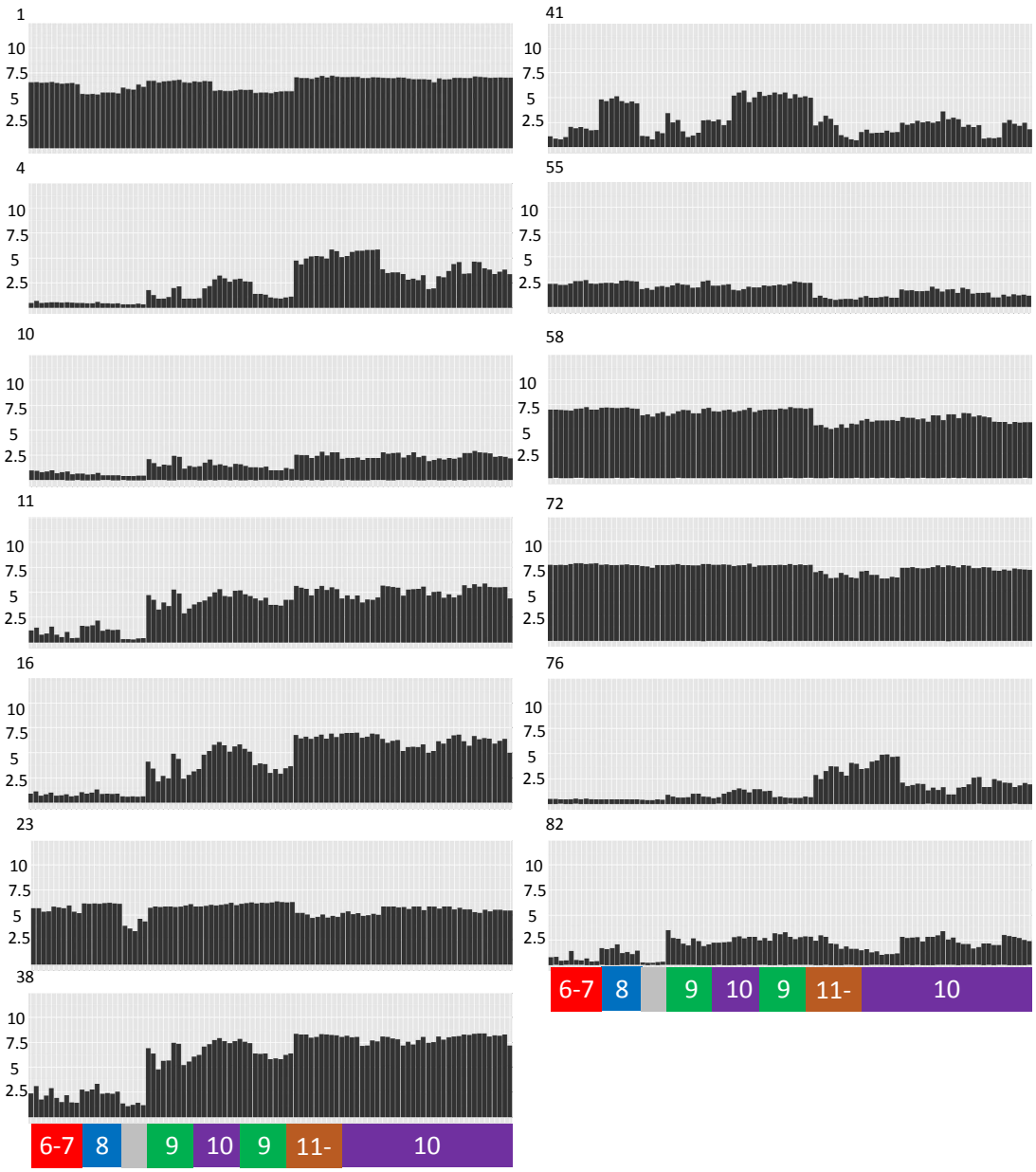


Figure 3.10. Eigen-profiles for RNA metaclusters from Figure 3.2
The above plots show the eigenprofiles for each of the RNA-seq metaclusters from Figure 3.2. Each plot is on the same scale with the metacluster number above it. The stages for each experiment are shown and are the reverse order as the heatmap in Fig. 3.13. Each metacluster contains genes that are active in the same developmental stages and thus should be similarly regulated.

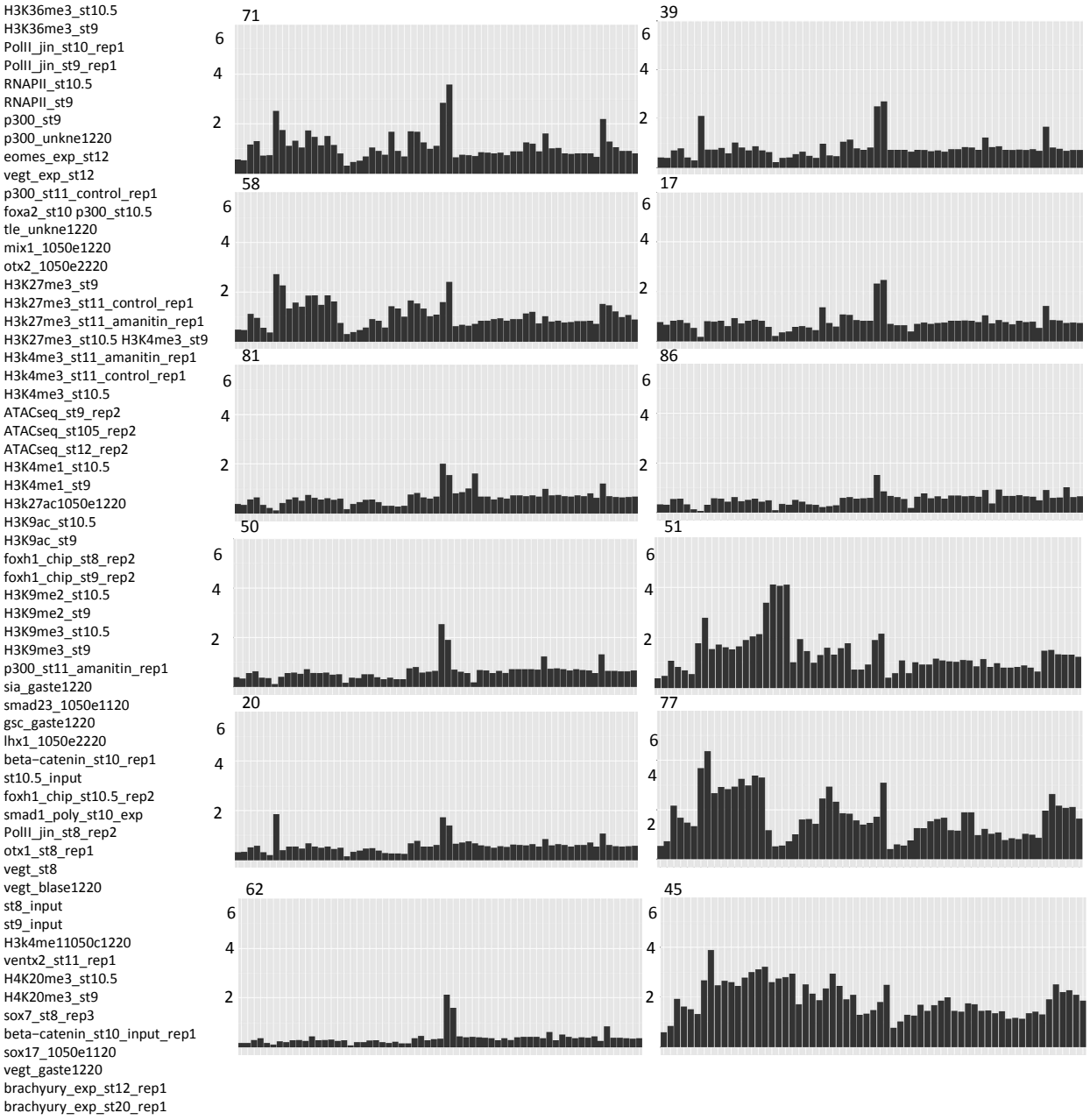


Figure 3.11. Eigen-profiles for DNA metaclusters from Figure 3.4
The above plots show the eigenprofiles for each of the DNA metaclusters from Figure 3.4. Each plot is on the same scale with the metacluster number above it. The order of each plot is on the left. Each metacluster contains regions that have similar co-binding profiles, and thus, should be similarly regulated.

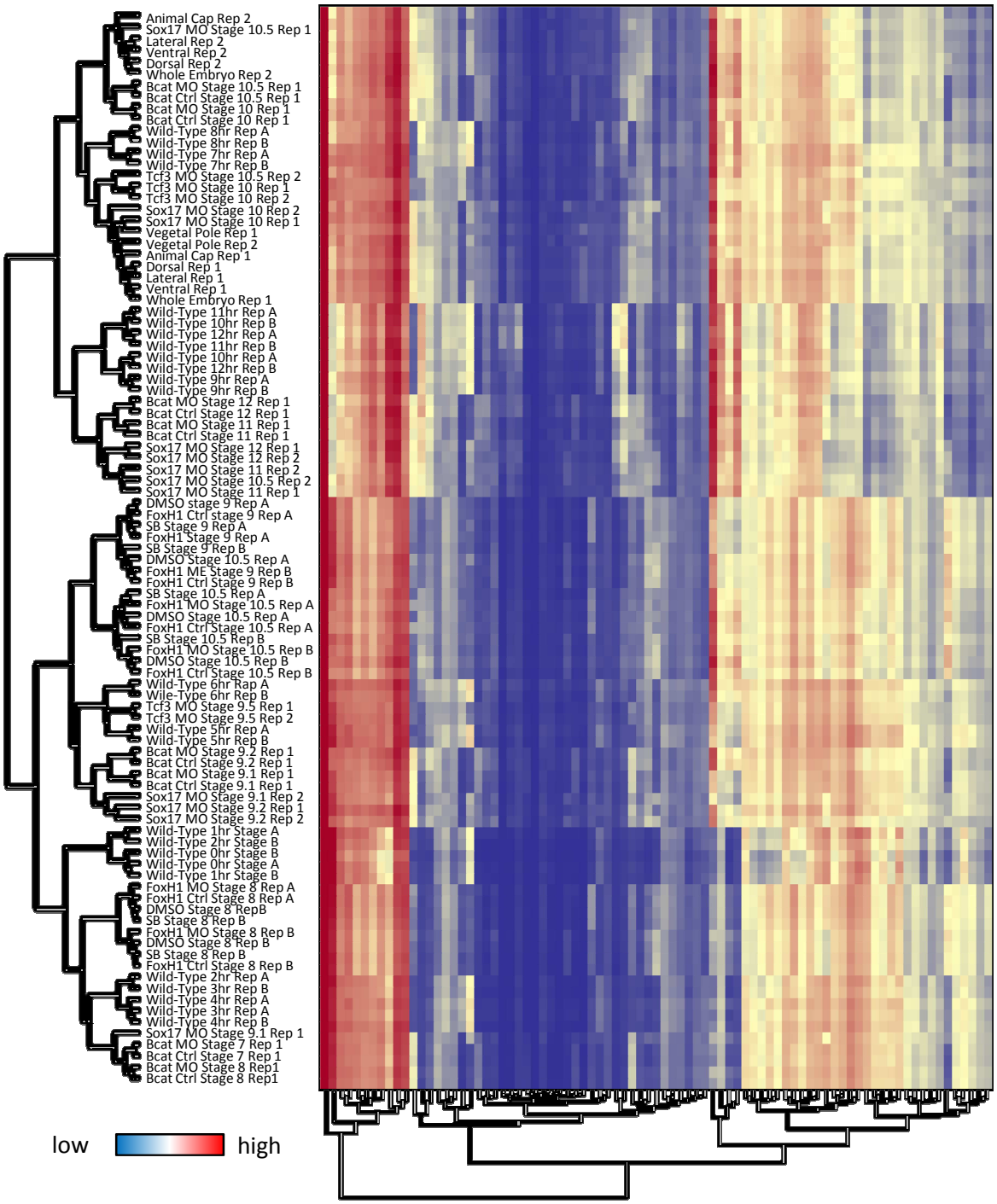


Figure 3.12. Full RNA-seq metacluster heatmap
This heatmap shows the entirety of the structure that was determined by the RNA-seq SOM metaclusters. Each experiment and metacluster was hierarchically clustered, and each row was normalized.

45

ALX1_HUMAN.H11MO.0.B
ALX3_HUMAN.H11MO.0.D
ARI5B_HUMAN.H11MO.0.C
BARH2_HUMAN.H11MO.0.D
COE1_HUMAN.H11MO.0.A
DMBX1_HUMAN.H11MO.0.D
DRGX_HUMAN.H11MO.0.D
E2F1_HUMAN.H11MO.0.A
E2F4_HUMAN.H11MO.0.A
EMX1_HUMAN.H11MO.0.D
EMX2_HUMAN.H11MO.0.D
ETV3_HUMAN.H11MO.0.D
FOXC1_HUMAN.H11MO.0.C
FOXD1_HUMAN.H11MO.0.D
FOXD2_HUMAN.H11MO.0.D
GLI2_HUMAN.H11MO.0.D
HMGA2_HUMAN.H11MO.0.D
HMX1_HUMAN.H11MO.0.D
HMX3_HUMAN.H11MO.0.D
HXC10_HUMAN.H11MO.0.D
IRF4_HUMAN.H11MO.0.A
ISL2_HUMAN.H11MO.0.D
MAFK_HUMAN.H11MO.1.A
MBD2_HUMAN.H11MO.0.B
MEIS3_HUMAN.H11MO.0.D
MESP1_HUMAN.H11MO.0.D
NANOG_HUMAN.H11MO.1.B
NFAT5_HUMAN.H11MO.0.D
P53_HUMAN.H11MO.1.A
P73_HUMAN.H11MO.1.A
PBX3_HUMAN.H11MO.0.A
PHX2A_HUMAN.H11MO.0.D
PHX2B_HUMAN.H11MO.0.D
PKNX1_HUMAN.H11MO.0.B
PLAL1_HUMAN.H11MO.0.D
PO4F1_HUMAN.H11MO.0.D
PPARG_HUMAN.H11MO.0.A
PROP1_HUMAN.H11MO.0.D
RFX1_HUMAN.H11MO.1.B
RFX3_HUMAN.H11MO.0.B
RFX5_HUMAN.H11MO.1.A
RORA_HUMAN.H11MO.0.C
SCRT1_HUMAN.H11MO.0.D
SCRT2_HUMAN.H11MO.0.D
SMAD2_HUMAN.H11MO.0.A
SMAD4_HUMAN.H11MO.0.B
SOX10_HUMAN.H11MO.0.B
SOX21_HUMAN.H11MO.0.D
SOX7_HUMAN.H11MO.0.D
TF65_HUMAN.H11MO.0.A
TFCP2_HUMAN.H11MO.0.D
TGIF2_HUMAN.H11MO.0.D
UNC4_HUMAN.H11MO.0.D
VENTX_HUMAN.H11MO.0.D
ZBT49_HUMAN.H11MO.0.D
ZEB1_HUMAN.H11MO.0.A
ZN317_HUMAN.H11MO.0.C
ZN423_HUMAN.H11MO.0.D
ZN554_HUMAN.H11MO.0.C
ZN586_HUMAN.H11MO.0.C
ZN708_HUMAN.H11MO.0.C
ZNF41_HUMAN.H11MO.1.C
ZSC16_HUMAN.H11MO.0.D

51

ANDR_HUMAN.H11MO.2.A
DLX1_HUMAN.H11MO.0.D
EVX1_HUMAN.H11MO.0.D
FEZF1_HUMAN.H11MO.0.C
FOXF1_HUMAN.H11MO.0.D
GATA5_HUMAN.H11MO.0.D
HAND1_HUMAN.H11MO.0.D
HIC2_HUMAN.H11MO.0.D
HXB3_HUMAN.H11MO.0.D
IRF5_HUMAN.H11MO.0.D
IRX2_HUMAN.H11MO.0.D
ITF2_HUMAN.H11MO.0.C
MEOX1_HUMAN.H11MO.0.D
P63_HUMAN.H11MO.0.A
P73_HUMAN.H11MO.0.A
PAX2_HUMAN.H11MO.0.D
PAX6_HUMAN.H11MO.0.C
PRDM1_HUMAN.H11MO.0.A
SHOX_HUMAN.H11MO.0.D
SMCA1_HUMAN.H11MO.0.C
SOX3_HUMAN.H11MO.0.B
SOX8_HUMAN.H11MO.0.D
STA5B_HUMAN.H11MO.0.A
TEAD1_HUMAN.H11MO.0.A
VSX2_HUMAN.H11MO.0.D
Z324A_HUMAN.H11MO.0.C
ZBT48_HUMAN.H11MO.0.C
ZKSC1_HUMAN.H11MO.0.B
ZN121_HUMAN.H11MO.0.C
ZN274_HUMAN.H11MO.0.A
ZN329_HUMAN.H11MO.0.C
ZN410_HUMAN.H11MO.0.D
ZN490_HUMAN.H11MO.0.C
ZN502_HUMAN.H11MO.0.C
ZN563_HUMAN.H11MO.0.C
ZN582_HUMAN.H11MO.0.C
ZNF18_HUMAN.H11MO.0.C

71

AIRE_HUMAN.H11MO.0.C
AP2A_HUMAN.H11MO.0.A
AP2C_HUMAN.H11MO.0.A
BATF3_HUMAN.H11MO.0.B
BPTF_HUMAN.H11MO.0.D
COT2_HUMAN.H11MO.1.A
CR3L2_HUMAN.H11MO.0.D
CREB3_HUMAN.H11MO.0.D
CRX_HUMAN.H11MO.0.B
EHF_HUMAN.H11MO.0.B
ELF3_HUMAN.H11MO.0.A
ELF5_HUMAN.H11MO.0.A
ESX1_HUMAN.H11MO.0.D
ETV2_HUMAN.H11MO.0.B
EVI1_HUMAN.H11MO.0.B
FOXA2_HUMAN.H11MO.0.A
FOXA3_HUMAN.H11MO.0.B
FOXJ2_HUMAN.H11MO.0.C
HXC6_HUMAN.H11MO.0.D
KAISO_HUMAN.H11MO.0.A
KLF4_HUMAN.H11MO.0.A
LEF1_HUMAN.H11MO.0.A
LHX9_HUMAN.H11MO.0.D
MEF2B_HUMAN.H11MO.0.A
MEF2C_HUMAN.H11MO.0.A
MGAP_HUMAN.H11MO.0.D
MTF1_HUMAN.H11MO.0.C
NFIA_HUMAN.H11MO.0.C
NFIC_HUMAN.H11MO.0.A
NR1H4_HUMAN.H11MO.0.B
NR2C1_HUMAN.H11MO.0.C
NR2F6_HUMAN.H11MO.0.D
ONEC2_HUMAN.H11MO.0.D
PO2F3_HUMAN.H11MO.0.D
PO3F1_HUMAN.H11MO.0.C
PO3F4_HUMAN.H11MO.0.D
PO4F2_HUMAN.H11MO.0.D
PO6F1_HUMAN.H11MO.0.D
PPARD_HUMAN.H11MO.0.D
PRDM4_HUMAN.H11MO.0.D
RUNX2_HUMAN.H11MO.0.A
RXRG_HUMAN.H11MO.0.B
SIX1_HUMAN.H11MO.0.A
SOX1_HUMAN.H11MO.0.D
TBR1_HUMAN.H11MO.0.D
TFE2_HUMAN.H11MO.0.A
THA_HUMAN.H11MO.0.C
THB_HUMAN.H11MO.0.C
VDR_HUMAN.H11MO.0.A
Z354A_HUMAN.H11MO.0.C
ZEP2_HUMAN.H11MO.0.D
ZN134_HUMAN.H11MO.1.C
ZN260_HUMAN.H11MO.0.C
ZN528_HUMAN.H11MO.0.C
ZN816_HUMAN.H11MO.0.C
ZNF85_HUMAN.H11MO.1.C

Figure 3.13. List of motif IDs in the subtractions between metaclusters 45, 51, and 71
Full list of HOCOMOCO v11 motif IDs in the Venn diagram subtractions from Figure 4c.

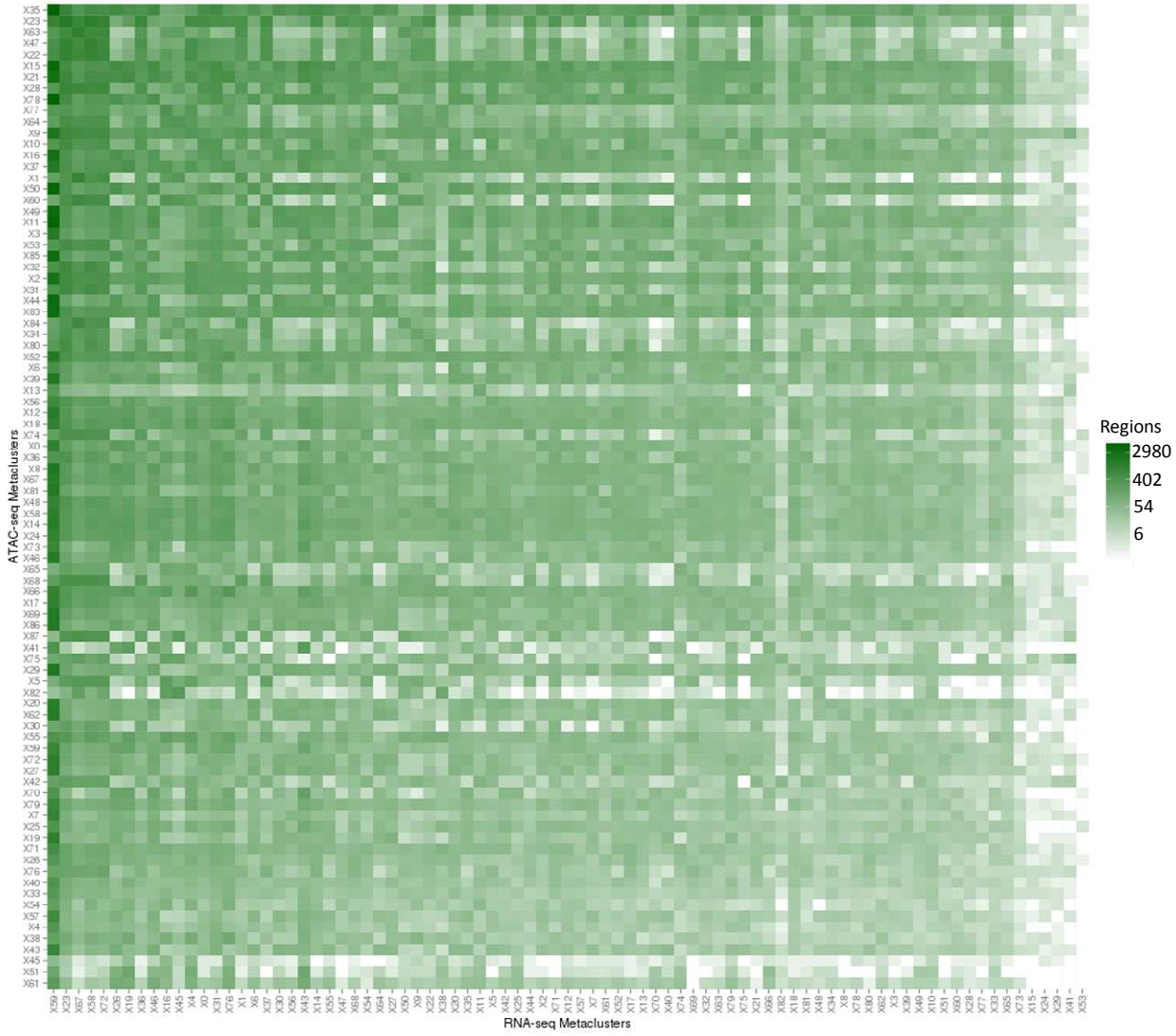


Figure 3.14. Linked metacluster region magnitude confusion heatmap
The above heatmap portrays the number of genomic regions in each linked metacluster with the largest overlaps oriented in the top left.

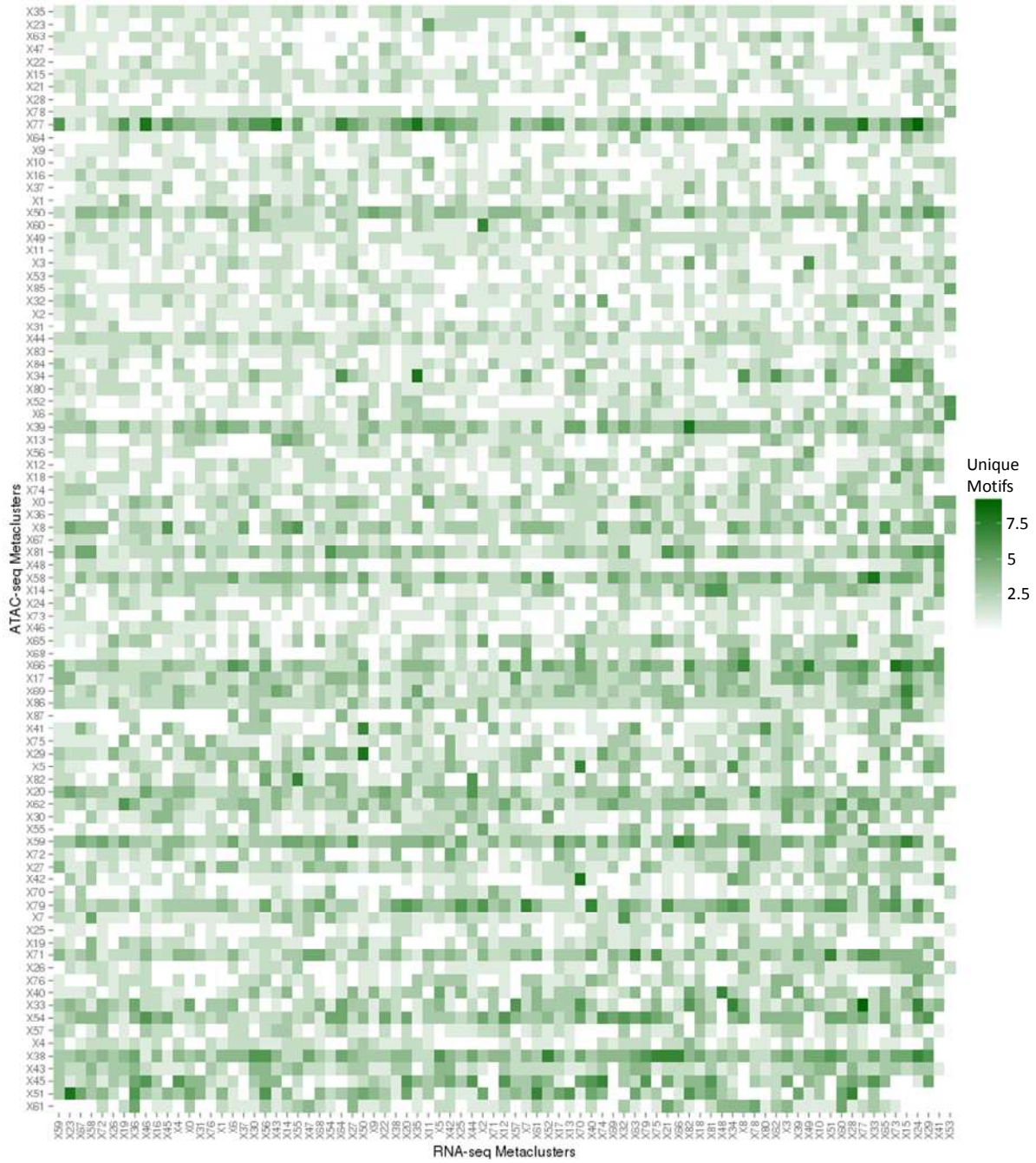


Figure 3.15. Linked metacluster unique motif magnitude heatmap
This heatmap displays the number of unique *Xenopus tropicalis* motifs detected in each linked metacluster. It is interesting to note that the size of the linked metaclusters is not correlated with the number of motifs found. For example, (59,35) in the top left corner is the largest linked metacluster, and it only contained the foxa2 motif.

3.6 Methods

3.6.1 Data processing

RSem output files were specifically provided for this analysis and TPMs from these were collected into a large training matrix. This matrix was log scaled and gene names were added to the gene ids for readability in downstream analysis steps. Similarly, ChIP/ATAC-seq data was provided in the form of sam files and called peaks in the form of bed files. These peak files were used to partition the genome using the partition tool of SOMatic. This tool concatenates the peak starts and ends into a genomic position list and sorts them. Then, for each chromosome, partitions are built from position 1 or the previous partitions's end point to the next position on the list. Partitions are given a minimum size by skipping positions from the list that are too close together. In this work, partitions were set to a minimum of 200 bp. Then, the RPKMs of each experiment were calculated over these partitions using the regionCounts tool from SOMatic.

3.6.3 Training and metaclustering of the individual RNA and DNA SOMs

With the 2 training matrices in hand, we were free to build SOMs with SOMatic. For the RNA SOM, we ran the 31,399 genes through 100 epochs for 100 trials. SOMatic automatically splits the training matrix 50/50 into a training set and a scoring set. This allows for the best trial to be taken without causing over-clustering. Then, we calculated the dimensionality of the SOM for metaclustering at 45, and checked every metacluster number from 50 to 150 for 100 trials a piece and took the trial/metacluster number with the best AIC score. 84 metaclusters had the best score. SOMatic created all of the heatmaps

for the figures including the hypothesis analysis. We used Xenbase's GO term tool to get functional enrichments.

For the DNA SOM, we ran the 731,726 partitioned genome segments through 100 epochs for 100 trials. Again, we calculated the DNA SOM's dimensionality to be 13 and metaclustered looking at every metacluster number from 50 to 150 for 100 trials. 88 metaclusters had the best AIC score. For functional enrichments, we calculated the closest gene TSS to each region (within 1 Mb) and built a unique list of regulated genes for each metacluster. Finally, we, again, used the Xenbase GO term tool for the GO term enrichments.

3.6.4 RNA-seq SOM Sample Choice and Hyperparameter Variation

We originally built the RNA SOM with a significantly larger data set. Each of the treatments had their own sets of controls, and the wild-type data contained time points at every half-hour. In all, there were 181 data sets. However, due to the inclusion of many control and wild-type datasets, the SOM was not properly clustering on the features that we were interested in. Instead, it was separating the data on differences between experiments. To solve this issue, we cut several control experiments and half of the wild-type time course. Due to this change, the final SOM successfully clustered genes into developmental modules. Thus, it is important to not use data sets that are too close together when building SOMs.

When building the SOMs above, we used a few additional sizes in a process similar to Chapter 2.6.8. For the RNA-seq SOM, we used 20x30, 40x60, and 60x90. The 60x90 SOM had the best score at the end of training, and the 20x30 SOM had single unit metaclusters

indicating underclustering. For the ATAC/ChIP-seq SOM, we used 20x30, 40x60, and 60x90. The 40x60 SOM was chosen due to the 60x90 SOM having a metacluster that contained a unit in every row, indicating overclustering, and the 20x30 SOM, again, containing single unit metaclusters.

3.6.5 Linked SOMs

To convolve the 2 SOMs' metaclusters, we used the linking tool in SOMatic. This tool is detailed in Section 2.6.4. Again, we chose to look for the nearest gene within 1Mb for each region within a DNA metacluster and connected it to the RNA metacluster with that gene. We had to use a specific xenopus option (-Xeno) due to the fact that their gtf file is a non-standard format. Other than that, the default options were used.

3.6.6 Motif Analysis

For the initial ChIP/ATAC-seq SOM, the regions, including repeat regions, in each metacluster were scanned for motifs using the HOCOMOCOv11 human motif database with FIMO v4.12.0 using a q-value threshold of .1. For the further network analysis, each linked metacluster (LM) was scanned using motifs calculated from the ChIP experiments (provided to us) using FIMO v4.12.0 using a q-value threshold of .1. The background for both analyses was calculated using the entire *Xenopus tropicalis* v9 reference genome. For each of the 12 provided TFs, we calculated the percentage of regions in each LM with that motif. Then, we calculated LM enrichment using a one-tailed z-score analysis with a q-value of .05. These significant TF motif locations were mapped to the linked gene.

3.7 References

1. Heasman, J., *Patterning the early Xenopus embryo.* Development, 2006. **133**: p. 1205-1217.
2. GN, W. and B. AW, *Simple vertebrate models for chemical genetics and drug discovery screens: lessons from zebrafish and Xenopus.* Dev Dyn, 2009. **238**(6): p. 1287-1308.
3. Charney, R.M., et al., *A gene regulatory program controlling early Xenopus mesendoderm formation: network conservation and motifs.* Semin Cell Dev Biol, 2017. **66**: p. 12-24.
4. M, L. and P. R, *A genetic regulatory network for Xenopus mesendoderm formation.* Dev Biol., 2004. **271**(2): p. 467-478.
5. Chiu, W.T., et al., *Genome-wide view of TGFβ/Foxh1 regulation of the early mesendoderm program.* The Company of Biologists, 2014. **141**: p. 4537-4547.
6. Kofron, M., et al., *New roles for FoxH1 in patterning the early embryo.* Development, 2004. **131**: p. 5065-5078.
7. Choi, J., et al., *FoxH1 negatively modulates flk1 gene expression and vascular formation in zebrafish.* Dev Biol., 2007. **304**: p. 735-744.
8. Levine, M. and E.H. Davidson, *Gene regulatory networks for development.* PNAS, 2005. **102**(14): p. 4936-4942.
9. RN, R., et al., *Dynamic Gene Regulatory Networks of Human Myeloid Differentiation.* Cell Systems, 2017. **4**(4): p. 416-429.
10. M, F., et al., *Zygotic VegT is required for Xenopus paraxial mesoderm formation and is regulated by Nodal signaling and Eomesodermin.* Int J Dev Biol, 2010. **54**(1): p. 81-92s.
11. Morikawa, M., et al., *Genome-wide mechanisms of Smad binding.* Oncogene, 2013. **32**(13): p. 1609-1615.
12. Lolas, M., et al., *Charting Brachyury-mediated developmental pathways during early mouse embryogenesis.* PNAS, 2014. **111**(12): p. 4478-4483.
13. Attisano, L., et al., *The transcriptional role of Smads and FAST (FoxH1) in TGFβ and activin signalling.* Molecular and Cellular Endocrinology, 2001. **180**(1-2): p. 3-11.
14. Kulakovskiy, I.V., et al., *HOCOMOCO: towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis.* Nucleic Acids Research, 2017. **46**(D1): p. D252-D259.
15. Maeda, T., D.L. Chapman, and A.F.R. Stewart, *Mammalian Vestigial-like 2, a Cofactor of TEF-1 and MEF2 Transcription Factors That Promotes Skeletal Muscle Differentiation.* Journal of Biological Chemistry, 2002. **277**: p. 48889-48898.

CHAPTER 4

Progressive clustering and characterization of increasingly higher dimensional datasets with Living Self-Organizing Maps

Chapter 4

Progressive clustering and characterization of increasingly higher dimensional datasets with Living Self-Organizing Maps

4.1 Abstract

Long-lived consortiums in genomics generate massive highly-dimensional datasets over the course of many months or years with substantial blocks of data added over time. Algorithms designed to characterize and cluster this data are designed to run once on a dataset in its entirety, and thus, any analysis of these collections must be entirely re-done from scratch every time a new block of data is added. We describe a novel progressive clustering approach using a variation of the self-organizing map (SOM) algorithm, which we call the Living SOM. Our software package is capable of clustering highly-dimensional data with all of the power of regular SOMs with the added benefit of incorporating additional datasets as they become available while maintaining the initial structure as much as possible. This allows us to evaluate the impact of the new datasets on previous analyses with the potential to keep classifications intact if appropriate. We demonstrate the power of this technique on a collection of gene expression experiments done in an embryonic time course of development for mouse from the ENCODE consortium.

4.2 Introduction

Self-Organizing Maps (SOMs)[1] and further metaclustering[2] have been shown to effectively cluster highly-dimensional data for characterization[3,4]. However, like other unsupervised learning algorithms, they were designed to be run on a set of data in its

entirety and must be re-trained every time new data is available. In the field of genomics, it is common for large consortiums such as the Encyclopedia of DNA Elements (ENCODE) to generate huge collections of highly-dimensional data such as new RNA-seq experiments on the same genes in new tissues or ChIP-seq experiments on the same genome regions with new transcription factors over the course of many years with big blocks being released over time. The nature of this data might be interpreted as additional dimensions added to a fixed number of points in the existing dataset rather than new data points added in a similarly sized N-dimensional space. Unfortunately, unsupervised learning algorithms do not typically support dynamic datasets that change in dimensionality, as all of the down-stream classification has to be re-done after each release.

After each data release, it would be ideal to be able to use the previous analysis to help train a new clustering. However, simply using the previous SOM unit locations as the initialization point is problematic. For example, adding a dimension can potentially disassociate clustered data points. This can cause the units sitting in those clusters to settle halfway between their original associated genes, becoming stuck in local minima, and generating a sub-optimal clustering. (Fig 1)

Here, we present a novel method that we call the Living SOM (LSOM) that allows dimensions to be inserted, one at a time, into an already trained SOM while maintaining the original topology as much as possible. Its purpose is not only to speed up learning over complete re-training, but also to allow for the possible preservation of the down-stream classification. This method for data insertion is fast and highly reactive to SOM units becoming stuck in local minima during the addition of dimensions. We present a full comparison between this algorithm and the standard Kohonen SOM in terms of clustering

reproducibility at both the SOM unit and metacluster level using a highly-dimensional genomic dataset from the ENCODE consortium. Finally, we show that drops in reproducibility after certain data insertions reveal structural novelty in that data and could be used to detect either biological novelty or erroneous data.

The results of this paper are organized as follows: Sect. 3 introduces the LSOM algorithm. Sect. 4 contains comparisons between Kohonen SOMs and the LSOM, both in regular usage and after simulating a data release. Sect. 5 details an exhaustive analysis of the importance of data insertion order. Finally, Sect. 6 contains a discussion of the results.

4.3 The Living SOM

There have been many modifications to the standard Kohonen SOM to cluster various modalities of data over the years. For example, modifications have been developed for streaming data where data points are added at inconsistent time steps to the overall pool, such as the Ubiquitous SOM[5]. In that algorithm, a more-aggressive organizing step is added to the standard SOM to solve a similar issue to the one described in Fig. 1, in which SOM units would get stuck between several data points as the streaming data would move to another part of data space. We began development of the LSOM from this algorithm because many of the issues with adding dimensions are similar to those from streaming data, and we use similar metrics for triggering the organizing step and computing the learning rate and radius. As such, dimensions are added one at a time in a similar manner as observations being added by streaming data.

The main metric for triggering the organizing step is when the "average drift," which is a weighted average between the average quantization error and the average neuron utility, exceeds a limit for a number of time steps. This limit is set to the average drift at the end of the organizing step or at the beginning of training and is kept between insertions. This step lasts for 1 epoch, or one pass through each data point, allowing them to influence the new position of the units. Afterwards, the LSOM returns to the beginning of regular learning. The organizing step is rare in practice, with 0-1 occurring in the LSOMs built in Section 3.

There are two sets of learning parameters in the LSOM that are active during the different states. During the ordering state, the learning rate and radius are set to an aggressive level dependent only on ordering time[5]. Conversely, in the default learning state, the learning parameters are dependent on learning time (to force convergence) and the current drift level compared to the drift limit (Equations 1 and 2 below). See the Algorithm 1 and the Parameters and Formulae section at the end for details.

4.4 Clustering Comparisons to Kohonen SOMs

The LSOM was designed with datasets from genomic consortiums such as ENCODE in mind. Thus, to test the performance of LSOMs compared to Kohonen SOMs, we selected a set of gene expression data from a developmental time course done in mouse by the ENCODE consortium (Fig 2). In this context, the data points correspond to the genes and the dimensions correspond to the experimental tissue-timepoint combinations. This time course was chosen due to its high quality and the high variety of the biological samples.

Algorithm 1: Living SOM Algorithm (per insertion)

1: Input previous set of SOM units. $\mathbf{U}_{\text{old}} \in \boldsymbol{R}^{(k-1) \times (n \times m)}$, where $\boldsymbol{n}$ and $\boldsymbol{m}$ are the SOM rows and columns respectively and $\boldsymbol{k}$ is the new total number of dimensions.
2: Input previous training matrix. $\boldsymbol{M_{old}} \in \boldsymbol{R}^{(k-1) \times o}$, where $\boldsymbol{o}$ is the number of data points.
3: Input new vector of observations. $\boldsymbol{v}$.
4: Input previous drift limit, $\boldsymbol{d}_0$.
5: Input previous number of faults, $\boldsymbol{f}$.
5: Create new training matrix $\boldsymbol{M} \in \boldsymbol{R}^{k \times o}$ by combining $\boldsymbol{M_{old}}$ and $\boldsymbol{v}$ by row.
6: Create new set of SOM units, $\boldsymbol{U} \in R^{k \times (n \times m)}$, by adding a 0 to each unit in $\boldsymbol{U_{old}}$.
7: Set variables, $\boldsymbol{g}$ = *true*, $\boldsymbol{i}$ = 1
8: Randomly reorder the rows in $\boldsymbol{M}$.
9: **while** $\boldsymbol{g}$ **do**
10: **if**($\boldsymbol{f}$>=$\boldsymbol{o}$) **do**
11: Perform standard SOM algorithm on organizing parameters for 1 epoch **(Organizing Step)**
12: set $\boldsymbol{i}$ = 1
13: Calculate drift, $\boldsymbol{d}$
14: Set drift limit, $\boldsymbol{d}_0 = \boldsymbol{d}$
15: Find closest unit $\boldsymbol{u}$ in $\boldsymbol{U}$ to $\boldsymbol{M}$[,$\boldsymbol{i}$]
16: Calculate drift, $\boldsymbol{d}$
17: Calculate learning radius and learning rate
18: **if**(current drift > $\boldsymbol{d}_0$) **then**
19: $\boldsymbol{f}$++
20: Perform update step on unit $\boldsymbol{u}$
21: **if**(learning radius < 1) **then**
22: $\boldsymbol{g}$=*false*
23: $\boldsymbol{i}$++

Also, all of these experiments were done by a single lab, so batch effects should be less prevalent. The gene expression values for the first replicate of each experiment were downloaded from the ENCODE portal [6] and built into a large training matrix containing 69,691 gene expression measurements per experiment.

To ensure that the LSOM generates comparable clusterings to Kohonen SOMs, we trained a control 40x60 Kohonen SOM on all 78 data sets over 100 trials (full individual runs) with 1000 epochs. Then, we built 100 Kohonen SOMs and 100 Living SOMs on the same data. Afterwards, metaclusters were called on each of these SOMs[2]. Finally, we

calculated the Jaccard indexes[7] between the Control SOM and each of the experiment SOMs (Fig 3A). Comparison of the distributions of these indexes did not find any significant difference between the two types (Fig 3B). Thus, the changes made to the standard algorithm did not affect its ability to cluster highly-dimensional genomic data at the unit or metacluster level.

Next, we analyzed whether the scaffold of the LSOM would maintain its structure, and thus, have a higher reproducibility, during a simulated data release. For this analysis, we built 100 LSOMs on the datasets, each with a random one removed, which is then added into the LSOMs. Again, metaclusters were called on each of these SOMs. Finally, we calculated the Jaccard indexes for the clusterings done before and after the data insertion (Fig. 4). These indexes were significantly higher than those from Kohonen SOMs at both the unit and metaclusters level. This provides evidence that the LSOM is leveraging the prior training and that the LSOM scaffold is maintaining its structure after a data insertion as intended.

4.5 Reproducibility is Affected by Data Insertion Order

In the previous section, LSOMs were built by inserting genomic data one at a time in a random order. In the Kohonen SOM, the order of the columns in the training matrix does not matter, and we therefore wished to determine what effect, if any, data insertion has on the reproducibility of the LSOM. We built 2 sub-collections - one with six of the heart data sets and a second subset with six of the Day 10.5 data sets and built control Kohonen SOMs for each (Fig. 5A). We then built 720 LSOMs for each sub-collection, exhaustively testing every possible data insertion order and we calculated Jaccard indexes between the LSOMs

and the control SOM. The distributions of these indexes show that the data insertion order does not have a significant effect on reproducibility most of the time, but we found a few low-scoring clusterings that we inspected further (Fig. 5B).

Displaying the data insertion order of the 5 clusterings of the Day 10.5 data with the worst Jaccard indexes reveals that adding the heart data set last has the potential to create a detectable decrease in the reproducibility of the LSOM (Fig. 5C). This is interesting as the heart dataset is not the most distant data when analyzing the PCA of the training matrix (Fig. 5D) with hindbrain's sample accounting for 55.8% of the variance to heart's 32.7%. However, it may signify that at those points where heart differs from the other datasets, the dataset splits a substantial number of otherwise clustered points. This suggests that the information in the heart dataset adds more novelty to the analysis.

4.6 Discussion

In this work, we have presented a novel method, the Living SOM, for clustering datasets that grow over time without requiring a complete re-clustering on each release. LSOMs do this by using a previous SOM's units as the initialization with a gentle learning rate based on the current "drift," a weighted average between average error and neuron usage. If the average drift goes over a predetermined limit, it indicates that the LSOM has settled into local minima, and the LSOM will switch into a more aggressive re-organization mode for 1 epoch and set a new limit. Datasets are added one at a time until the data release is fully inserted.

This algorithm produces similar clusterings to the classical SOM trained on the same collection of highly-dimensional genomic datasets. Additionally, when simulating the

subsequent addition of new datasets, the LSOM leverages the previous analysis to maintain the structure of the scaffold, thus generating a significantly higher reproducibility to the previous iteration compared to clustering de novo. The metaclusters in particular see a very large improvement. Finally, we showed that the order of data insertion can affect the reproducibility if the final dataset is very structurally different from the previous data. To combat this issue, LSOMs could ideally be run with the most different datasets first (as calculated by hierarchical clustering), and thus, the reproducibility would never drop below acceptable values.

It may be possible to use this property of the LSOM as an advantage. By virtue of computing the reproducibility of the LSOM as we add datasets, it is possible to measure this drop. Datasets that result in a substantial drop could be inspected to assess whether they are improperly labeled or extremely error-prone data as the clustering is done. An alternative view is that monitoring the reproducibility also provides us with a metric for measuring how much "novelty" a new dataset adds to existing analyses.

4.7 Figures

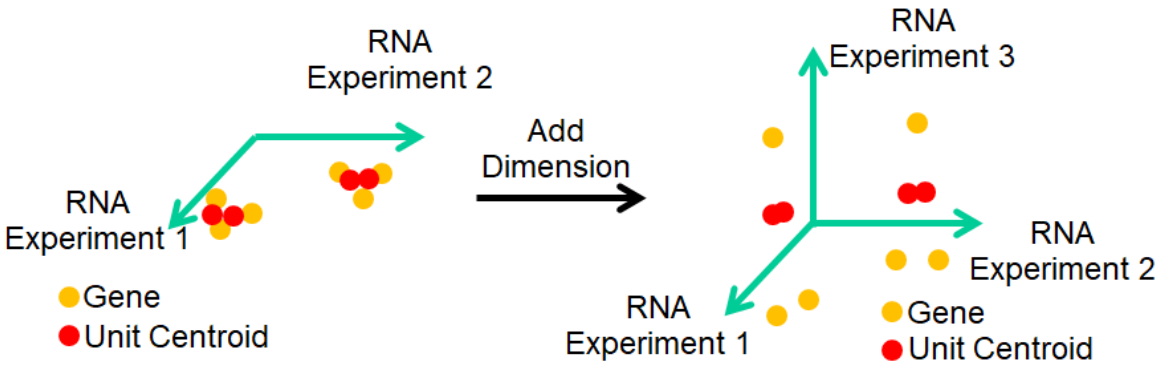


Figure 4.1. An illustration of a potential effect of adding a dimension to an existing analysis.
Adding a dimension can possibly disassociate clustered data points, which can cause the units sitting in those clusters to settle halfway between their original associated genes and become stuck.

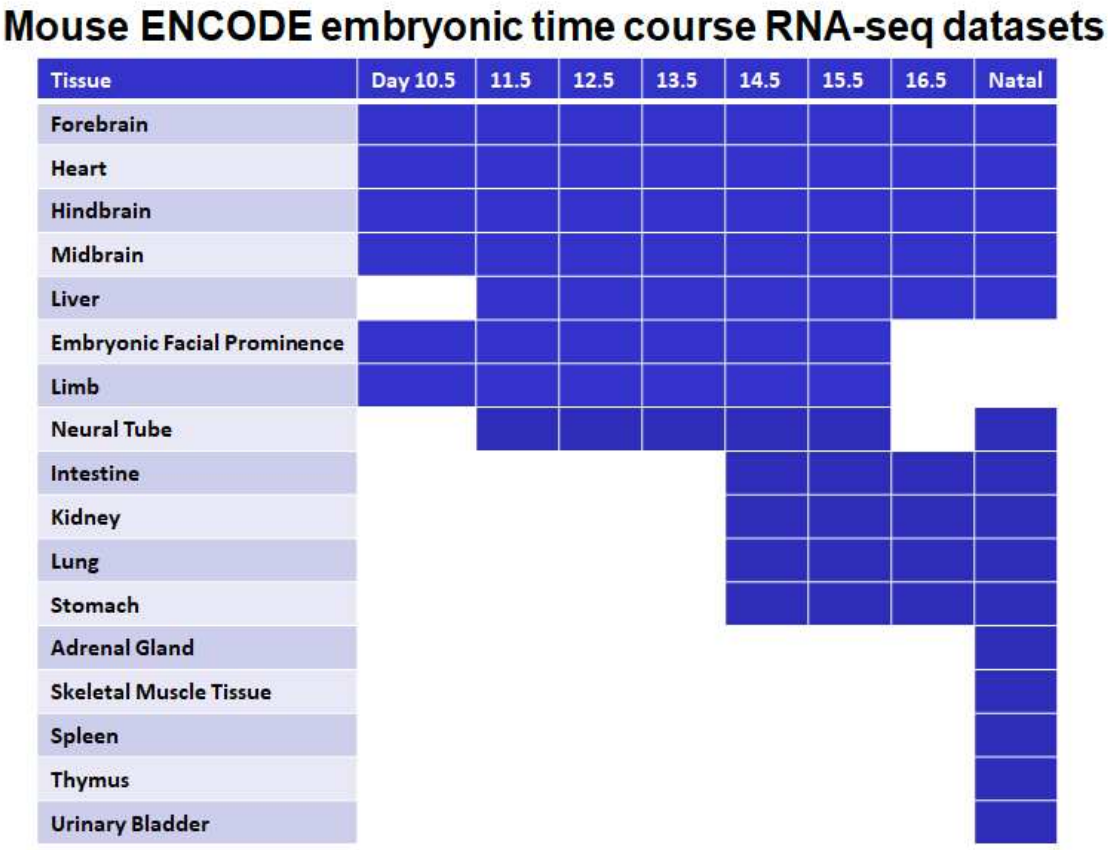


Figure 4.2. Mouse ENCODE embryonic time course RNA-seq datasets
78 datasets chosen to test the LSOM's clustering reproducibility because of their high quality and these samples are part of a time course of related samples. The 6 Day 10.5 datasets and 6 of the 8 Heart datasets (skipping Day 11.5 and Day 13.5) were also used to test the effect of data insertion order.

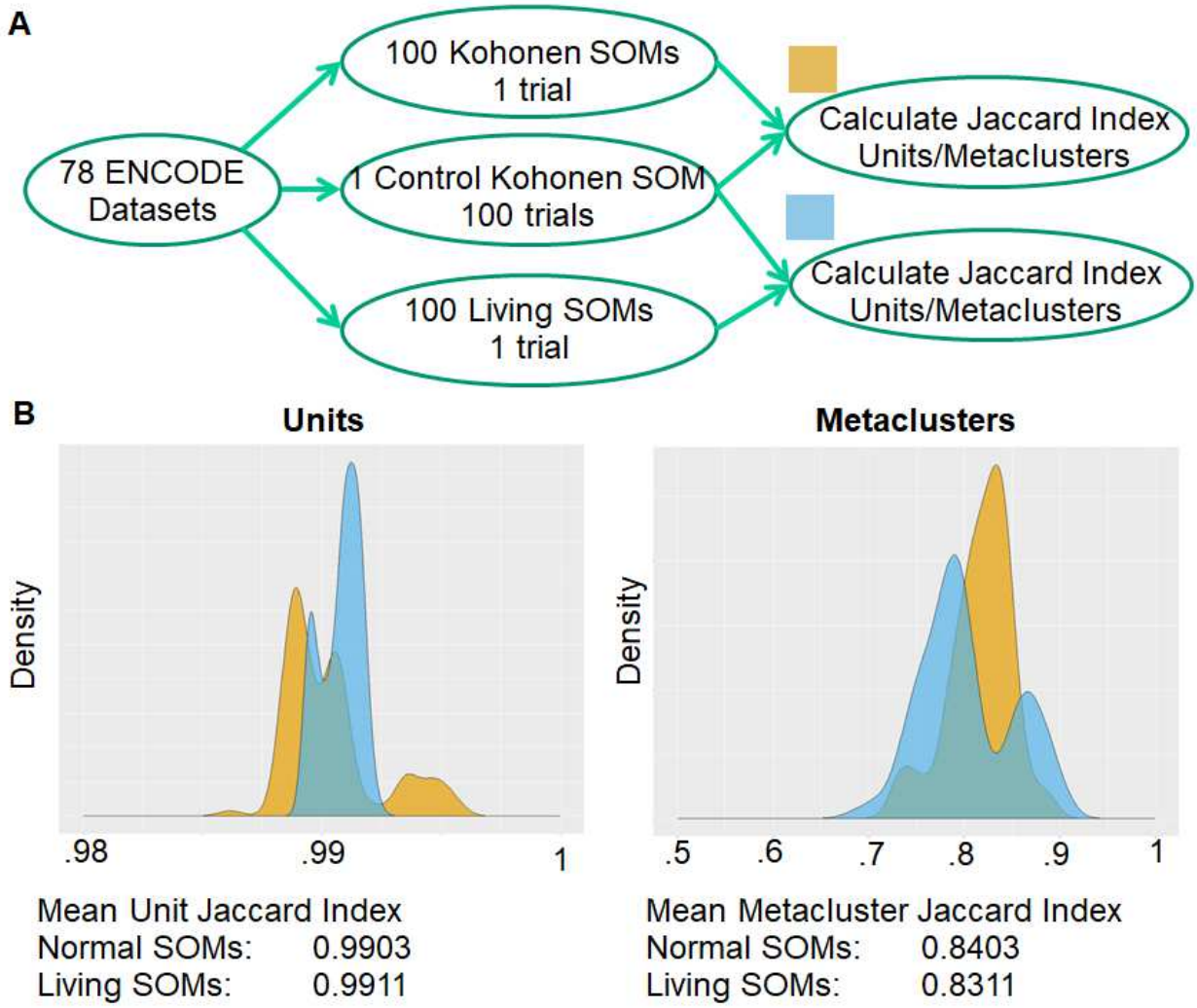


Figure 4.3. Living/Kohonan SOM comparison
(A) In order to determine whether LSOMs create similar clusterings to normal SOMs, we trained 100 Kohonen and LSOMs on the same set of data in random orders and calculated the Jaccard Index, or reproducibility, of these clusterings at the unit and meatcluster scales. **(B)** The Jaccard indexes were very similar, indicating that LSOMs generate similar clusterings to normal SOMs.

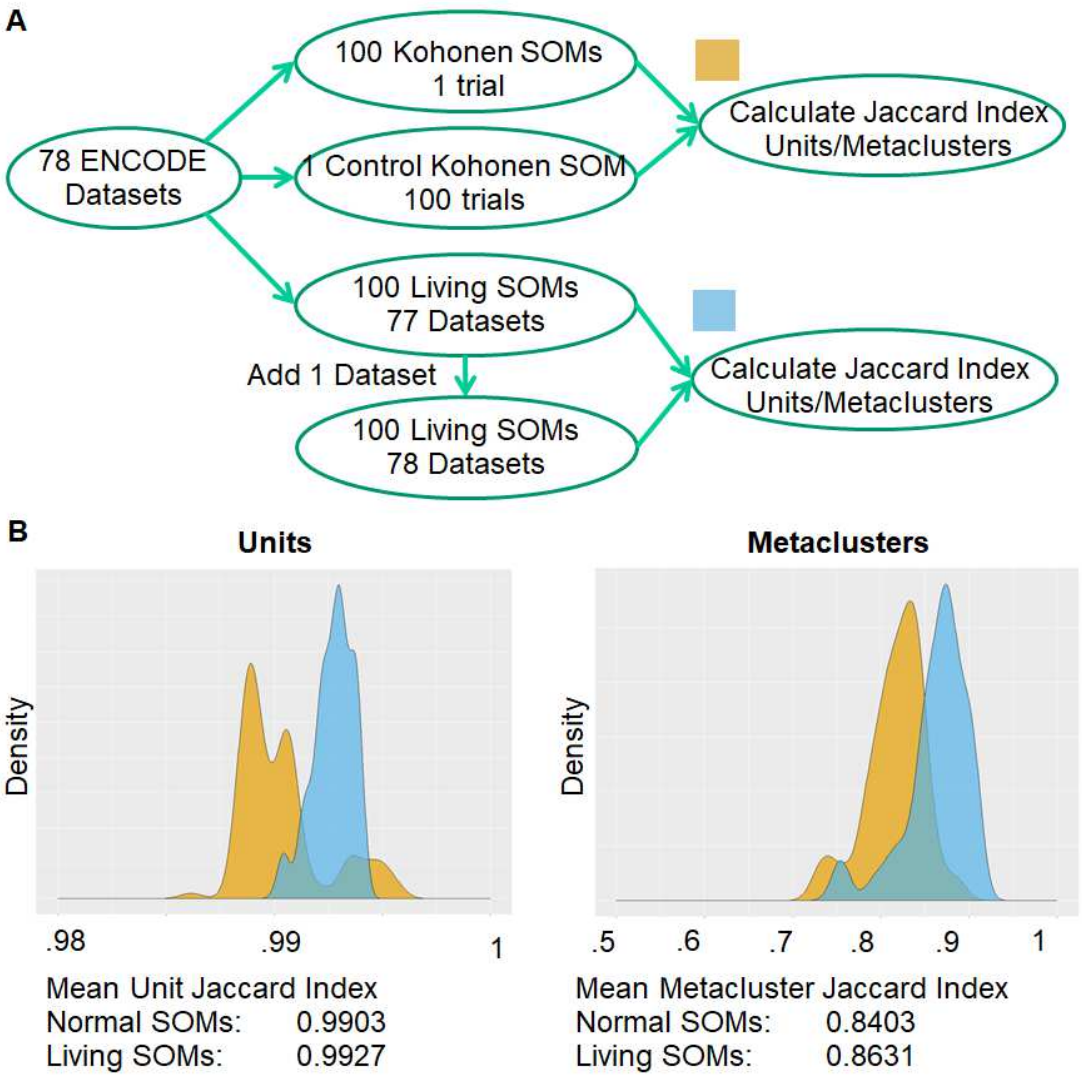


Figure 4.4. Simulated data release comparison between Living and Kohonan SOM (A) In order to simulate a data release, we also trained 100 living SOMs on 77 of the data sets. Then, we inserted 1 data set and compared the reproducibility at the unit and metacluster level to re-training the SOM from scratch. (**B)** Adding one dimension to the Living SOM was not only significantly faster than re-training a normal SOM, but the following clustering was very similar to the previous analysis, more so than re-training from scratch.

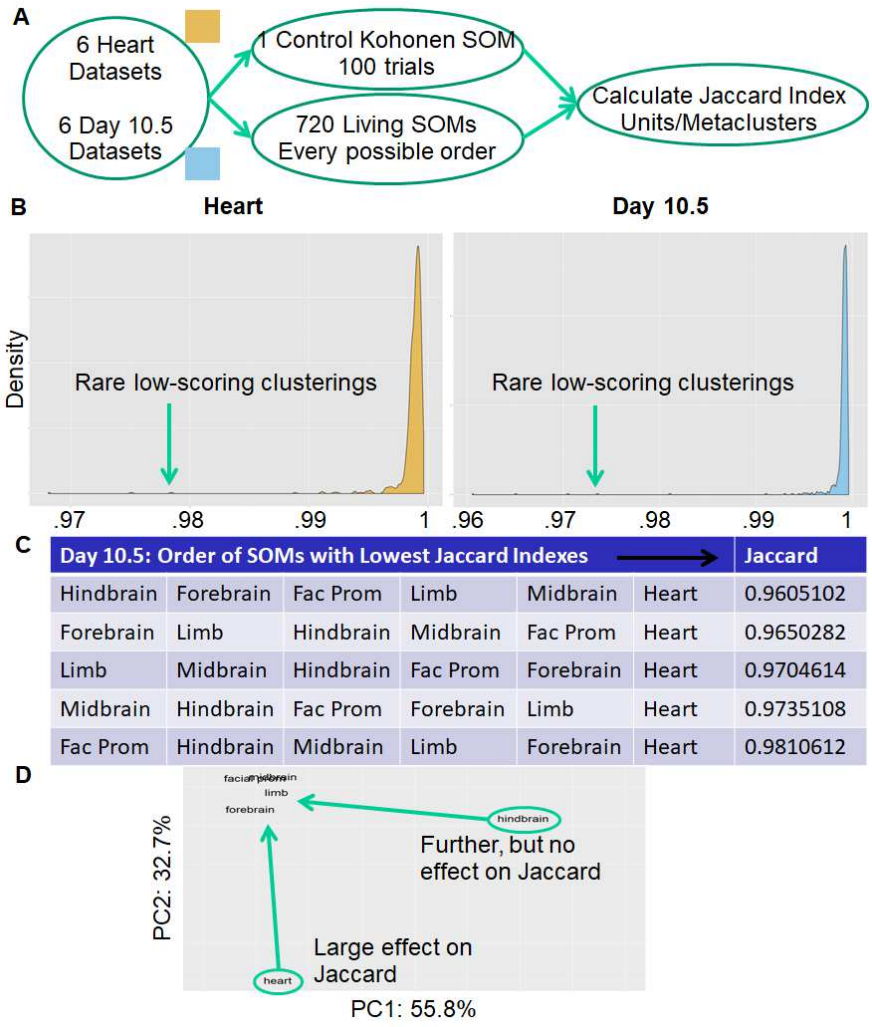


Figure 4.5. Effect of data insertion order on reproducibility
(A) In order to test the effect of data insertion order on clustering, we chose (1) six datasets from heart and (2) six datasets from embryonic Day 10.5, built SOMs using every possible insertion order, and compared those to a regular SOM. **(B)** While the majority of LSOM runs resulted in good Jaccard indexes, there were a few orderings that were lower-scoring. **(C)** Visual inspection of the bottom 5 SOMs using Day 10.5 data revealed that the heart 10.5 dataset, when added last, caused this effect. **(D)** In a PCA of the six Day 10.5 datasets, hindbrain, not heart is the most Euclidian-distant, but the heart is the most biologically-distinct.

4.6 Parameters and Formulae

Table 1. Parameters used in analysis

Parameter	Symbol	Value	Parameter	Symbol	Value
Rows		40	Radius Factor Initial	σ_i	0.8
Columns		60	Radius Factor Final	σ_i	0.2
Learning Rate Initial	η_i	0.2	Beta Factor	β	0.7
Learning Rate Final	η_f	0.08			

Formulae: Most of the formulae in this work come from [5], except for the following which have been edited in this work.

Learning State - Learning Rate η, Radius Factor σ

$$\eta(t) = \begin{cases} \left(\frac{\eta_f}{d(t_f)} d(t)\right)^{\left(1-\frac{t}{t_f}\right)}, & d(t) < d(t_f) \\ \eta_f, & otherwise \end{cases} \quad (1)$$

$$\sigma(t) = \begin{cases} \left(\frac{\sigma_f}{d(t_f)} d(t)\right)^{\left(1-\frac{t}{t_f}\right)}, & d(t) < d(t_f) \\ \sigma_f, & otherwise \end{cases} \quad (2)$$

4.7 References

1. Kohonen, T. (2001) Self-Organizing Maps. Springer, 3rd edition.
2. Alhoniemi, E. (2000) Clustering of the Self-Organizing Map. IEEE Transactions On Neural Networks, Vol. 11, No. 3
3. Mortazavi, A. et al. (2013) Integrating and mining the chromatin landscape of cell-type specificity using self-organizing maps. Genome Research, 23: 2136-2148.
4. Tamayo, P. et al. (1999) Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. PNAS, 96 (6) 2907-2912.
5. Silva, B. and Marques, N. (2015) The ubiquitous self-organizing map for non-stationary data streams. Journal of Big Data, 2:27.
6. Link to ENCODE datasets, https://bit.ly/2FGKWnx, last accessed 2019/01/17.
7. Jaccard P. (1912) The distribution of the flora of the alpine zone. New Phytologist, 11 (1912), pp. 37-50

CHAPTER 5

Future directions

Chapter 5

Future directions

Thus far in this work, I have introduced the Linked SOMs method for building gene regulatory networks (GRNs) from highly-dimensional functional multi-nomics data and displayed its application on both single cell and bulk data in two model systems. I have also demonstrated the Living SOM method of analyzing a growing dataset without constant re-classification.

In Chapter 2, I showed that functional data from scRNA-seq and scATAC-seq was sufficient to build a gene regulatory network that recovered previously known interactions and add many more potential connections for further study. When that work was completed, the state-of-the-art in motif scanning involved statistical over-representation tests which use obsolete views on transcription factor binding sites. An improved method would use neural networks built for each transcription factor in order to find and to score transcription factor binding sites rather than rely on pre-defined PWMs. These neural networks could even include input nodes that incorporate each region's linked metacluster to improve the motif classification. I predict that the GRNs from this refined analysis would be a substantial improvement over current methods.

Additionally, when I linked genomic regions from the DNA metaclusters to the genes in the RNA metaclusters, I only considered the closest gene, which is suboptmal. There are enhancers that are known to regulate genes over 1 megabase away of important developmental genes such as Sonic Hedgehog. Any errors in multi-clustering the RNA and ATAC data would reduce the motif density as multiple biological programs clustered

together would water down the regulation of any given program. There are assays to discover long-range interactions genome-wide such as HiC. Incorporating this type of data into the regulatory network linking would also improve its performance and lead to significantly more accurate GRNs.

However, simply building gene regulatory networks is only the first step in understanding the regulation of gene expression. Even if we were able to use this information to determine if a gene were expressed or not, it is usually the level of expression that is important for down-stream functions. Many GRN studies attempt to create and to simulate networks only using boolean networks, which ignore the weight of regulation. However, detailed studies have shown that the logic of gene regulation cannot be described with Boolean functions alone (Teif. 2010). My proposed solution to this problem involves leveraging the similarity of topologies between neural networks and gene regulatory networks. Each neural network contains three layers of units that are connected to the other layers: (1) an input layer containing all of the inputs to the network, (2) a hidden layer, which is determined by summation functions applied on the input layer, and (3) an output layer, which is similarly determined by the hidden layer. After constructing these networks separately, they can be stacked on top of each other and can be slightly adjusted to fit the total output of a larger system. The regulation of genes has a similar pattern. Transcription factor expression levels acting as input nodes building up the activation of regulatory modules modified by chromatin accessibility that, in turn, act as hidden nodes activating gene expression, which themselves feed into input nodes elsewhere in the network. Thus, I propose building draft neural networks for a set of transcription factors using high-resolution gene expression and chromatin accessibility

data taken from the same cell using previously built GRNs. Then, stacking these neural networks and incorporating multiple pseudo-time points simultaneously will allow me to closely fit the data. The goal would be an entire stacked neural network representation of a model system's transcriptional regulation network. Due to it being a neural network and not just a binary network, I should be able to simulate gene expression changes with respect to perturbation experiments.

In Chapter 3, I used the Linked SOMs method to build a mesendodermal development GRN. This shows the power of the technique on developmental time course data. Unfortunately, the experiments in that analysis are out-dated, limiting the results as they required many pooled embryos. Single-embryo RNA-seq would vastly improve the information content of the dataset, especially in the morpholino experiments. Additionally, ChIP-seq has problems with resolution that can be solved by using the new "ChIP-nexus" technique to find the exact positions of the transcription factor binding sites. Reducing the width of the peaks would vastly improve the statistics of motif scanning. Finally, single-cell ChIP would also improve the resolution of the histone ChIP experiments. All of these state-of-the-art techniques could add to the resolution of the developmental data set which would improve gene regulatory network detection.

In Chapter 4, I developed a new technique for analyzing a growing data set that I termed the Living SOM. A follow up to this project would be to expand this technique further downstream and develop a set of analysis tools to properly make use of the improved reproducibility. After data insertion, the metaclustering classification could

change if the new data is topologically different than the previous set. This means that standard classification tools would require a full run-through anyway. I propose a follow-up software tool that can use previous classifications to quickly re-classify after a data insertion.