GENE REGULATION INFERENCE WITH NASCENT RNA TRANSCRIPTION

by

RUTENDO FAITH SIGAUKE

A.S., Cottey College, 2011

B.S., Hamline University, 2013

M.S., University of Oregon, 2015

A thesis submitted to the

Faculty of the Graduate School of the University of Colorado in partial

fulfillment of the requirements for the degree of

Doctor of Philosophy

Computational Bioscience Program

2023

This thesis for the Doctor of Philosophy degree by

Rutendo Faith Sigauke

has been approved for the Computational Bioscience Program

by

Katerina Kechris, Chair

Robin D. Dowell, Advisor

Sonia Leach

Ryan Layer

Anthony Gerber

Date: May 19, 2023

Sigauke, Rutendo Faith (Ph.D., Computational Bioscience Program)

Gene regulation inference with nascent RNA transcription

Thesis directed by Professor Robin D. Dowell

ABSTRACT

Gene regulation is an intricate and precise process that determines when a gene is expressed. The process is controlled by proteins known as transcription factors (TFs), which bind to DNA at preferred sequences typically within enhancers and promoters. TFs then recruit the transcription machinery, known as RNA polymerase, to nearby regions, allowing for the region to be transcribed. This process is context-dependent and has been shown to be well-timed and orchestrated. Understanding how and when genes are regulated can aid us in understanding how the disruption of gene transcription affects downstream processes. Many experimental protocols have been developed to understand the gene regulation process. For example, RNA-seq measures steady state RNA levels and chromatin immunoprecipitation (ChIP-seq) determines physical interactions between DNA and TFs. Together these assays have been used to infer when a particular TF is responsible for changing transcription at a set of target genes. However, these assays are limited by their steady state nature, and for ChIP-seq by the inherent low throughput. An alternative protocol that addresses some of these limitations is nascent RNA sequencing, which measures RNAs that are actively being transcribed by RNA polymerase. This gives a unique view on the immediate transcriptional response to perturbations that is not available from assays such as RNA-seq. Additionally, this assay captures short, unstable, enhancer associated RNAs, often called eRNAs. These eRNAs have been shown to be markers of active enhancers and thus can be used, with sequence motif information, to infer TF activity profiles.

In this thesis, I conduct a large scale meta-analysis of published nascent transcription data sets. To achieve this goal, I first summarize and compare the different nascent RNA sequencing protocols and their limitations. I then introduce methods for processing these data in a uniform manner that ensures the recovery of biologically meaningful signals. I then apply these approaches to the development of a large scale repository of nascent RNA sequencing data. In the repository, all

data has been assessed for quality and processed using a standardized pipeline. Summarizing transcription across hundreds of samples, I show that enhancer activity is more tissue-specific than gene transcription. Finally, I develop a correlation based framework for linking enhancers to their gene targets. Correlated eRNA to gene pairs are enriched for disease-associated variants as well as experimentally validated pairs. Lastly, I use these interactions to build gene regulatory networks (GRNs). This work provides a new resource for understanding gene regulation using nascent RNA transcription. It offers putative gene regulatory mechanisms in a tissue-specific context. With more nascent RNA data being produced, we can continue to expand our understanding on how genes are regulated at transcription.

The form and content of this abstract are approved. We recommend its publication.

Approved: Robin D. Dowell

To advocates of open science everywhere.

"Ubuntu ngubuntu ngabantu" - Bantu philosophy in Zulu

Humanity is determined through other people - English translation

ACKNOWLEDGEMENTS

Firstly, I would like to thank my advisors Dr. Robin Dowell and Dr. Mary Allen who have supported and encouraged me throughout my research journey. They have fostered an environment that allowed me to grow as a scientist and a person. Words can not express my gratitude for their mentorship. Thank you, Robin, for insightful discussions on problems and for creating a space for me to gain confidence as a scientist. Thank you to Mary for always welcoming my ideas and results with great enthusiasm. I would like to thank Dr. Katerina Kechris, Dr. Sonia Leach, Dr. Ryan Layer, and Dr. Anthony Gerber for serving on my dissertation committee. I would also like to thank Dr. Larry Hunter and Dr. Fan Yang who also served on my comprehensive exam committee. They have all been a source of direction as I navigated my dissertation project and graduate school. I could not have undertaken this journey without their guidance and advice. This endeavor would not have been possible without the technical and resource support of the BioFrontiers IT department. Additionally, this research was funded in part by a grant from NIH, R01GM125871.

I would like to extend my sincere thanks to the Dowell and Allen (DnA) lab. My dissertation project was only possible with the dedication and time commitment of current and former DnA lab members. I am grateful to Zach Maas and Margaret Gruca for laying out the foundation of the processing pipeline and initial curation of published nascent RNA experiments. Many thanks to Dr. Lynn Sanford for the thousands of hours (compute and walk clock) put into pre-processing the database. Special thanks to Dr. Lynn Sanford and Taylor Jones for collaborating on the dbNascent and co-transcription project. Apart from the dbNascent project, I had the pleasure of collaborating on a couple of other projects. I am grateful to Jessica Westfall, Zach Maas, Dr. Samuel Hunter, Dr. Jonathan Rubin, and Dr. Jacob Stanley who have been great collaborators over the years. I would like to thank Dr. Jacob Stanley for his friendship, encouragement, and willingness to provide feedback on various projects. I have thoroughly enjoyed our discussions on life, the universe, and everything. Being part of the DnA lab has been a rewarding experience that has awarded me with great collaborators and friends. I feel lucky to have been part of the DnA lab.

I would like to express gratitude to the Computational Bioscience program. Namely, Caitlin Moloney, for making the navigation of the administrative side of being in graduate school manageable. I would also like to thank current and former students from the program. Especially, Jo Hendrix for being a supportive classmate and friend. To Dr. Mayla Boguslav, thank you for your constant support and advice through all the milestones of graduate school. Special thanks to Brook Santangelo, Katerina Cortes, Dr. Brian Ross, and Emily Mastej for their friendship and support.

Finally, I am eternally grateful to my family and friends for without their support I would not have made it this far in my education journey. I would like to thank my good friend Anna who has been a constant support over the years. I am grateful for my mother Erica whose dedication to education defied cultural norms by leading by example and going back to school as a young mother. She has been my biggest cheerleader and a steadfast supporter of my dreams. I am also grateful to my father Agrippa for encouraging me to pursue my goals. To my siblings, Tapiwa, Simbarashe, and Miranda, thank you for your continued support and encouragement over the years. To my nieces and nephew (Anika, Joshua, and Michaela) thank you for always being able to put a smile on my face :). I am also grateful to my in-laws Terry and Ed for their unwavering love and support. Lastly, I would like to express my deepest gratitude to my partner and best friend Luke, who believed in me and was my pillar of strength, he motivated me to keep moving even when I did not feel like it. I'm extremely grateful for the philosophical discussions, adventures, and emotional support.

TABLE OF CONTENTS

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURE

CHAPTER I

INTRODUCTION

The human body contains trillions of cells. Even though all these cells have the same genome, each of the cells is specialized for a specific function that is, ideally, in harmony with the whole organism. These cells follow instructions that govern when a gene is expressed and to what extent – allowing for the proper function of the cell at a specific time developmentally and in response to stimuli. Following the central dogma, these genes are translated into proteins that perform a lot of the critical work in a cell. Therefore, characterizing which genes are expressed and how they are regulated can aid in a better understanding of the proper functioning of organisms and the implications of the aberrant expression of certain genes.

The transcription process, which is conserved across eukaryotes, dictates which genes are expressed throughout development and controls expression of genes in specific cell types and cell states [1, 2]. The process involves an organized assembly of proteins called transcription factors (TFs) that physically bind to enhancer and promoter regions, followed by the recruitment of RNA polymerase machinery which then transcribes genes (Figure 1.1)[1, 3, 4]. Enhancers are cis-regulatory regions that drive the transcription rate of a gene, and they work in concert with promoters that are located near the transcription start site of a gene [5]. Furthermore, it has been shown that when an enhancer is regulating a gene, short bidirectional RNA transcripts are produced in the region generating enhancer RNAs (eRNAs) [6–8]. These eRNAs have been used to infer TF activity and identify active regulatory regions [8, 9]. Importantly, it has be shown that these enhancer regions are enriched for disease associated variants which disrupt the expression of target genes by hindering the proper recruitment of TFs and/or RNA polymerase [7, 10].

Transcription regulatory regions have been mapped throughout the genome. Using chromatin immunoprecipitation (ChIP) assays that pull down histone modifications associated with active enhancers and promoters, the ENCODE consortium has characterized cis-regulatory elements. Active enhancers are marked by the acetylation of histone H3 at residue lysine 27 (H3K27ac), and monomethylation of H3 lysine 4 (H3K4me1), while active promoters are

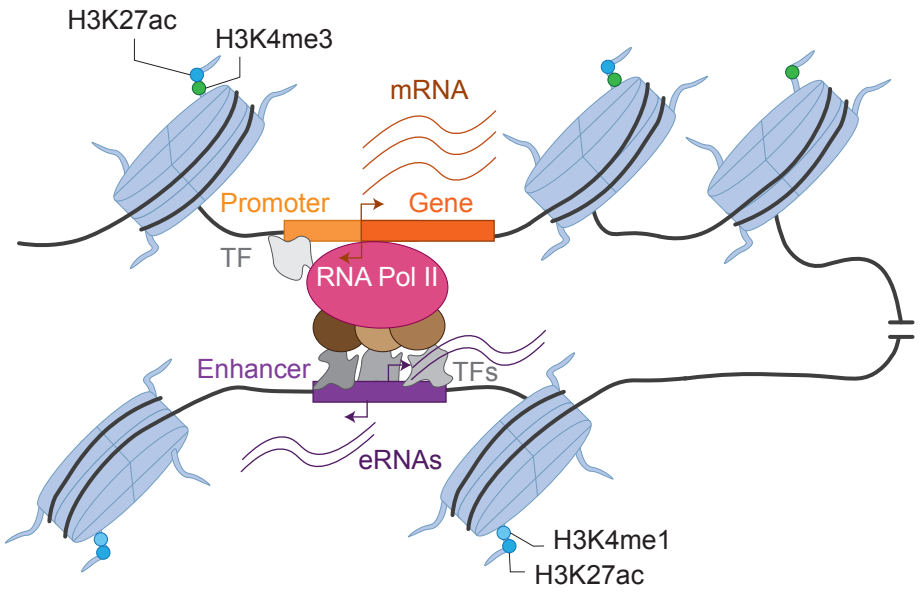


Figure 1.1: Schematic of the transcription process. The graphic shows some of the major components necessary for the transcription of genes. Transcription factors bind at enhancer and/or promoter regions where they recruit RNA polymerase to gene promoters. This allows for genes to be transcribed, often bidirectionally. There is also bidirectional transcription at the enhancer region giving rise to enhancer RNAs (eRNAs). Furthermore, there are histone markers that indicate active enhancer (H3K27ac and H3K4me3) and promoter regions (H3K27ac and H3K4me). [Adapted from Preissl et al. 2022] [11]

associated with the trimethylated form of H3 lysine 4 (H3K4me3) and H3K27ac (Figure 1.2) [12–16]. Since the annotation of these cis-regulatory elements relies on multiple ChIP-seq experiments, only 25 human and 15 mouse cell types have been completely annotated [12, 13, 17]. Another drawback of using ChIP-seq assays is that it measures physical binding only, yet only a subset of these events are actually actively engaged in regulation of RNA polymerase. To address the transcription activity of these regions, bidirectional transcripts at these regions are assayed using cap analysis gene expression (CAGE), which captures the 5' end of capped RNA molecules [18–20]. Since both enhancer and promoter regions yield these bidirectional transcripts, annotation of regulatory elements can be done in a single experiment. The FANTOM project has curated over 400 distinct cells and annotated both enhancer and promoter transcription using CAGE experiment [19, 20]. However, CAGE experiments capture only highly expressed transcripts, missing lowly expressed enhancers and promoters. Relatively newer protocols that measure nascent RNAs recover these lowly transcribed regions that are not detected in CAGE

assays [21]. Nascent transcription assays, such as global run-on sequencing (GRO-seq) and precision run-on sequencing (PRO-seq), measure RNA from actively engaged RNA polymerases.

Given that the assignment of enhancers to their target genes in various contexts (e.g. cell type, disease state, stimuli) is still an open problem [5, 22–27]. In this thesis, I use nascent transcription data to annotate transcribed regulatory elements, infer transcription factor activity, and link enhancers to the genes they regulate. To start, I optimize the annotation of the regions of nascent RNA transcription, I then develope a method to characterize the different nascent RNA protocols. Lastly, I make strides towards linking the regulatory elements to target genes using nascent RNA data and correlation based methods. The regulatory links measured here will provide scientists and doctors with a framework to discover the underlying causes of gene misregulation associated with human disease.

Background

One major outstanding obstacle in understanding the regulome is linking enhancers to their target gene promoters. Previously, gene regulatory networks (GRNs) have been used to elucidate transcription processes across cell lines and tissue types. GRN are networks whose nodes represent genes and/or their regulators (TFs, enhancers), with edges representing links between the nodes. For example, GRNs from gene expression data would have nodes representing the genes and edges representing coexpression (Figure 1.3 A), and the strength of the associations can be denoted by the weights of those edges (Figure 1.3 B). In cases where the expression of the TF is noted, a directed graph can be used to link TFs to target genes (Figure 1.3 C). Overall, the main goal of GRNs is to map out the key components that result in a disease phenotype, or predict possible outcomes in the event that the network is disrupted.

Various experimental techniques have been used to build GRNs, with most methods relying on gene expression (microarray or RNA-seq) and sometimes augmented by measurements of physical interactions (ChIP-seq or motif presence) [28, 29]. However, methods such as RNA-seq and ChIP-seq are steady state assays that lack temporal resolution and suffer from noise within the data. Thus GRNs are typically not representative of the underlying biological processes

since they do not directly account for all elements essential for the regulation of gene transcription such as enhancers. Accessibility assays such as DNase I digestion and sequencing (DNase-seq) and assay for transposase-accessible chromatin with high-throughput sequencing (ATAC-seq) have been used as a proxy for enhancer and promoter activity.

With the rise of single cell sequencing technologies, methods that combine single cell accessibility assays (scATAC-seq) and single cell RNA-seq experiments (scRNA-seq) have identified co-expressed gene and/or co-accessible enhancer regions [30–32]. These methods are motivated by the appreciation of tissue specific gene regulation, and use single cell data to identify putative enhancer and gene pairs from which to build GRNs. However, these methods face the same drawbacks as their bulk equivalent methods, namely they are steady-state assays. Additionally, single cell methods, particularly RNA-seq, are sampling based methods and therefore rarely return lowly expressed entities. Hence these data sets tend to be sparse. Moreover, since the accessibility data and the expression data often come from different cell populations, it is difficult to make concrete links between regulators and their targets [33]. It is important to note that these single cell methods are still rapidly developing, and methods to extract and quantify protein abundance and chromatin accessibility from the same cell are now emerging [33–35]. Regardless of the limitations posed by these data sets, the foundations have been laid, demonstrating their utility in inferring gene regulatory networks with the current sequencing technology landscape.

Additional information links enhancers to the genes they regulate. For example, chromatin interaction analysis by paired-end tag sequencing (ChIA-PET), where regions in close proximity to RNA polymerase II are pulled down, physically link enhancers to the promoter regions of genes they regulate [19, 36, 37]. Further support of these findings was also highlighted in time series data from CAGE where it was shown that enhancers and their target genes are correlated in transcription and, more revealing, the transcription of an enhancer precedes that of the target gene [38]. These findings showed that given the activation of an enhancer region and gene region, we can infer interaction by their coordinated change in signal.

Motivation

Since nascent RNA data is better at capturing lowly transcribed regions compared to CAGE, it is presumably more informative about enhancer to gene linkages than CAGE [21]. In the last decade, there has been an expansion of nascent RNA experiments from multiple nascent protocols. In this thesis, I set out to exploit this data to infer GRNs, I integrate co-transcription analyses in building these GRNs [39, 40]. The ultimate goal is a more accurate representation of gene regulation and its underlying mechanisms.

Currently, there are several technologies that capture newly transcribed RNAs or nascent RNAs. The methods can broadly be classified into two categories, nuclear run-on (NRO) techniques which capture RNAs as they are transcribed by cellular RNA polymerases and via metabolic RNA labelling of RNAs over longer periods of time. All of these approaches label a nucleotide and then use the label to pull down a specific collection of RNAs. Global run-on sequencing (GRO-seq) labels nascent transcripts with 5-bromouridine 5'-triphosphate (brUTP) whereas precision run-on sequencing (PRO-seq) uses biotin-NTPs as the label to identify newly created RNAs [41, 42]. In contrast, metabolic labelling experiments incubate cells in medium supplemented with a modified cell permeable nucleoside. Examples of metabolic labelling protocols include transient transcriptome sequencing (TT-seq) and TimeLapse-seq which use 4-thiouridine labelling [43, 44]. Unlike nuclear run-on methods, metabolic labelling of RNA is the only nascent RNA method that has been demonstrated to work in both cultured cells and also in living organisms [45]. However, due to the long incubation periods necessary for most of these metabolic protocols, they fail to capture transient RNAs such as most enhancer associated RNAs [43, 45].

Methods For Building Gene Regulatory Networks

The DREAM5 project benchmarked tools for the construction of GRNs using experimental and simulated data [46]. In their evaluations, they established the best methods for the construction of GRNs. The methods I highlight below are used for inferring GRNs with steady-state assays such as ChIP-seq, RNA-seq, ATAC-seq, or their single-cell alternatives

(scRNA-seq and scATAC-seq). Methods are split into four main categories, namely: regression, mutual information, Bayesian, and correlation networks. Here I briefly summarize each method, pointing out their strengths and weaknesses.

Regression networks

Given a large compendium of expression data from either microarray or RNA-seq data, one regression method called TIGRESS treats the problem as a feature selection problem. The goal is to identify TFs that best explain the expression of a given gene [46, 47]. TIGRESS uses a Lasso sparse regression approach combined with stability selection, which adds confidence to the TFs assigned to a gene [46–48]. The main benefit of this approach is it allows for the testing of multiple TFs working to regulate one or multiple genes, as each target gene is assessed independently to infer its direct regulators among all TFs. However, this method is limited to gene expression data as the method assumes that expression level of the TF itself is informative for predicting the expression level of its targets.

Mutual information networks

Context likelihood relatedness (CLR) is a mutual information method that has been used to infer GRNs. Similar to the above-mentioned regression method, CLR assumes that the TF driving the expression of a gene are co-dependent in their expression level and therefore their mutual information (MI) is higher compared to non-interacting pairs. The input here are gene expression counts, where genes that encode for TFs are labeled as *regulators* and the other labeled as *targets*. A z-score of each regulatory interaction is calculated [46, 49]. This z-score depends on the distribution of MI scores of all possible regulators of a target gene (z_i) and on the distribution of MI score for all possible targets of the regulator gene (z_j). The final score is a product of both scores as shown below:

$$\mathrm{z_{i,j}} = \sqrt{z_i^2 + z_j^2} \tag{I.1}$$

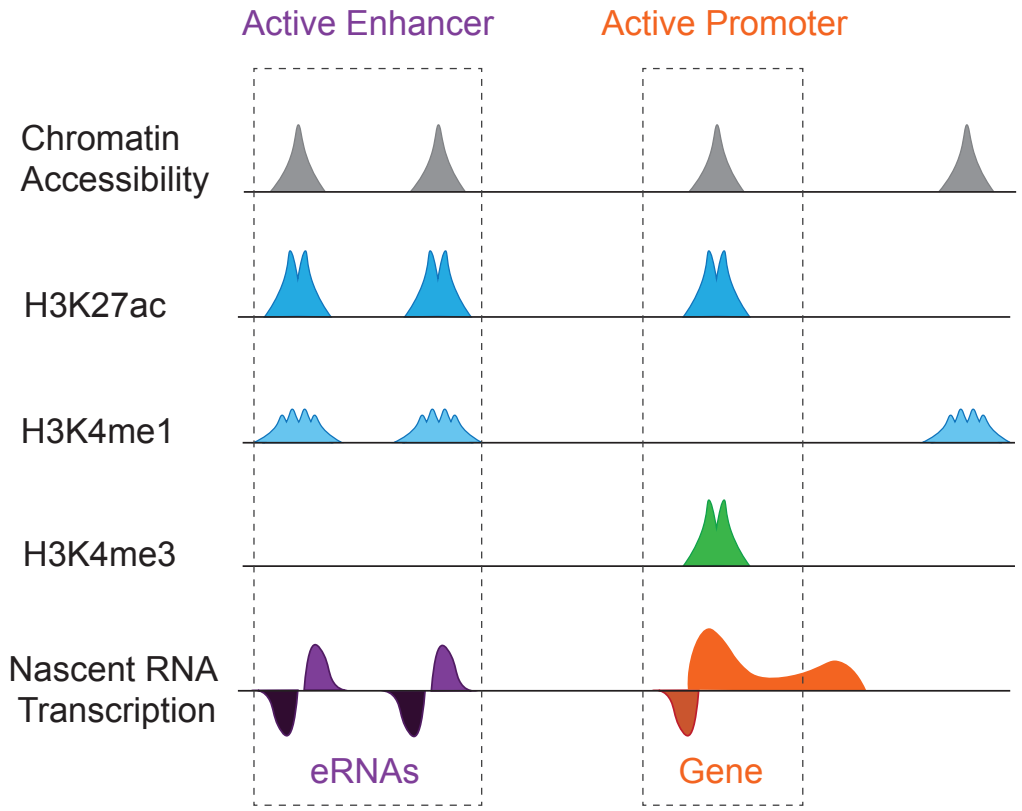


Figure 1.2: Profiles of sequencing methods and readout for genes and enhancers. Cartoon representation of genome tracks from accessibity, histone ChIP-seq and nascent RNA sequencing data. Read coverage from chromatin accessibility (measured by DNase-seq (DNase I digestion and sequencing) or ATAC-seq (assay for transposase-accessible chromatin with high-throughput sequencing)) overlaps regions of active enhancer and active promoters as indicated by H3K4me, H3K4me3, H3K27ac and Nascent RNA transcription. Note that accessibility and ChIP data are not stranded whereas nascent transcription data has strand information. [Adapted from Preissl et al. 2022] [11]

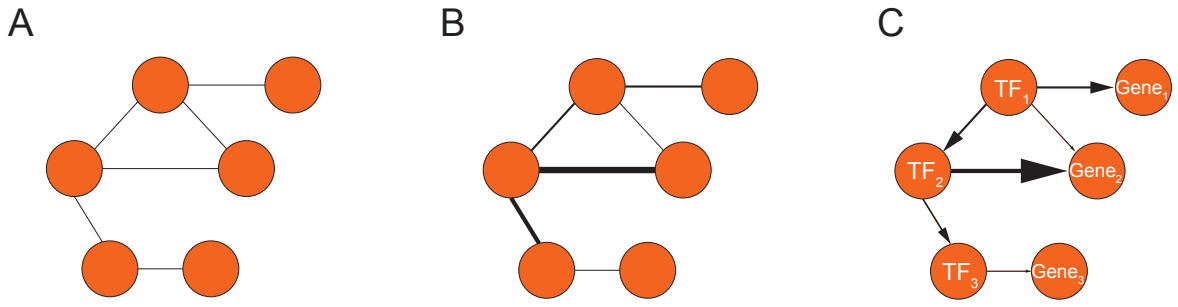


Figure 1.3: Example of gene regulatory networks. (A) An undirected GRN were the nodes represent genes and the edges are the links between the genes. (B) A weighted undirected GRN, where the weights are the strength of the interaction. (C) A weighted and directed GRN where directionality, when available, is informed by whether a gene is a TF.

Additionally, the method can be implemented using Pearson's correlation instead of MI [49]. When using MI, this method does not assume linearity, so complex non-linear relationships can be discovered.

Bayesian networks

Another method that has been used to build GRNs is based on Bayesian networks, due to their efficacy at incorporating prior knowledge. The gene expression levels represent the nodes of the network. In simple cases, the gene expression levels of a gene in its baseline state is compared to a perturbed state (e.g. treatment, knockout, over-expression), and used as the observed variables in the model. They are then compared to hidden variables, which are the magnitude of the influence on target gene expression. For each possible pair of TF to target gene relationships, the relative probability of each model is computed (no influence, A $\rightarrow$ B, B $\rightarrow$ A) given the observation data. In order to simplify the problem, conditional distributions are generally assumed to be Gaussian or discrete [46, 50, 51]. One main advantage of Bayesian networks is the relative ease of incorporating prior knowledge. However, even with the simplified methods and their benefits, the algorithm can be computationally intensive and does not perform as well as other less greedy algorithms [46, 51].

Correlation networks

An alternative method for GRNs argues that co-regulated genes have correlated expression across a large compendium of data. For example, the weighted correlation network analysis (WGCNA) method aims to find co-expressed genes from gene expression data [39]. The inputs to correlations methods are collections of gene expression counts. To meet the assumptions of these methods, count normalization for RNA-seq data is required.

Pearson's correlation

The Pearson's correlation coefficient ($r_{x,y}$) can be calculated between all TFs (x) and all target genes (y) as follows:

$$r_{x,y} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \quad \text{(I.2)}$$

where x are the the TFs and y are genes and n are all the samples [39]. The main assumptions of Pearson's correlations are linearity in the relationships as well as normality. Therefore, more complex non-linear relationships will not be easily identified. The benefit of Pearson's approach is its relative ease of use.

Spearman's correlation

Spearman's correlation (ρ) is obtained for all TFs (x) and all target genes (y) as follows:

$$\rho_{x,y} = 1 - \frac{6\sum d_i^2}{n(n^2-1)} \quad \text{(I.3)}$$

Where n is the number of samples from which x and y have been samples, and d is the difference in rank order between the TF x and gene y over the n conditions [52]. This approach can handle nonlinear, but monotonic, relationships.

Other

Tree based methods are a final alternative method for inferring GRN, one example is Gene Network Inference with Ensemble of Trees (GENIE3), which is based on variable selection with ensembles of regression trees. GENIE3 represents the prediction of a regulatory network between p genes into p different regression problems. In each of the regression problems, the expression pattern of one of the target genes is predicted from the expression patterns of all known TFs, using random forests [53, 54]. Based on the importance of the TF in the prediction of a target gene expression, the strength of the TF → gene link can be inferred. Integrating all interactions detected results in construction of a GRN. GENIE3 was the top-performing method in predicting GRNs in the DREAM challenge [46]. Therefore, GENIE3 has been integrated into a method for single-cell GRN inference called single-cell regulatory network inference and clustering (SCENIC). The first step in SCENIC is to identify co-expressed genes using GENIE3, followed by TF target site identification and then cell state enrichment [55]. The resulting GRN is a

combination of co-expression networks and TF regulons derived from TF motif binding, where a regulon is defined as genes coregulated by the same TF.

Methods For Building GRN With Enhancer To Gene Pairs

Using steady-state assays for GRNs

Attempts to create GRNs that include enhancers have used some of the methods mentioned above and incorporated steady-state high throughput experiments such as RNA-seq, ChIP-seq and ATAC-seq [23]. PreSTIGE uses a correlation based strategy with RNA-seq and H3K4me1 ChIP-seq to build GRNs with enhancers (based on histone marks)[24]. On the other hand, TargetFinder uses an ensemble of boosted decision trees with a variety of sequencing techniques (DNase-seq, FAIRE-seq, DNA methylation, RNA-seq, ChIP-seq for 32 histone marks, and TF information) [25]. JEME uses multiple linear regression and lasso shrinkage with DNase-seq, RNA-seq, ChIP-seq and three histone marks as input [26]. Lastly, GeneHancer uses a custom scoring method that includes weights and data transformations for each quantitative feature, using distance, TFs co-expression, eRNAs, eQTLs and 3D data [27].

Using Nascent RNA Sequencing Data To Infer GRNs

Nascent RNA sequencing gives a unique perspective, because both the cis-regulatory (enhancers and promoter) and gene regions are transcribed [41–45]. Furthermore, the transcription of a gene and its regulatory regions (enhancers linked in 3D) have been shown to be highly correlated, suggesting that we can identify enhancer targets using nascent RNAs [19, 38, 56]. This suggests that nascent RNA data contains signals for large-scale identification of enhancer targets via simple correlation analyses.

While working on this thesis, a different group (Lidschreider et al.) used correlation analysis to link enhancers to genes in nascent transcription. In that work, transient transcriptome sequencing (TT-seq) was used to identify putative enhancer – gene pairs in 14 cancer cell lines [57]. They identified about 40,000 putative pairs, many with known disease relevance and/or prior experimental validation. However, their work was limited by dependence on metabolic labeling,

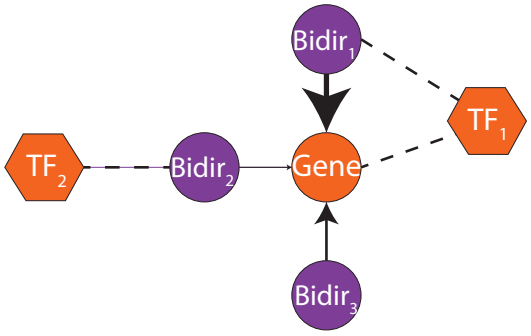


Figure 1.4: Example of gene regulatory networks from nascent RNA. Interactions from nascent transcription data allow inference of the directed linkage of enhancers to genes via correlation analysis ($Bidir_1 \rightarrow$ Gene, $Bidir_2 \rightarrow$ Gene and $Bidir_3 \rightarrow$ Gene). The presence of TF motif at transcribed gene promoters allows for the inference of direct TF to gene interactions ($TF_1 \rightarrow$ Gene). Lastly, TF motif found in bidirectional regions then link TFs indirectly to the genes they regulate (eg. TF_1 — $Bidir_1 \rightarrow$ Gene).

which is biased to more stable transcripts. Additionally, they failed to consider any assessment of false positives within their study.

In this thesis, I describe how we (the Dowell and Allen labs) manually curated 2880 published nascent RNA data sets. I use human and mouse PRO-seq and GRO-seq samples to annotate putative cis-regulatory regions, which are regions of bidirectional transcription. From this data I calculate tissue-specific bidirectional to gene correlation pairs. Previous work revealed that regions of bidirectional transcription are enriched for TF binding motifs [7, 8], so as an additional layer, I have added to the interactions the direct TF regulators upstream of gene transcription [7–9]. An example of GRNs from nascent RNA data includes enhancer → gene linkage, the TF → gene promoter and/or TF → enhancer interactions, where the TF interaction is derived from motif instances in the regions of bidirectional transcription (Figure 1.4).

Thesis Outline

Overview

This dissertation focuses on the use of nascent RNA data to understand the gene regulation process. Chapter 2 gives a broad overview of the different nascent RNA protocols and how the different techniques influence downstream analyses. The studies expose the protocol differences

and how to extract biologically meaningful signals regardless of these technical variations. In Chapter 3, I emphasize the standards for identifying regions of bidirectional transcription. This informs the standardized pipeline I built to identify transcribed regions from hundreds of nascent RNA datasets. In chapter 4, I present transcription factor enrichment analysis (TFEA), an algorithm for TF activity inference that allows us to infer active and repressed TF activity from nascent RNA sequencing experiments. Findings from this research informed how to incorporate TFs into GRNs using nascent data. Finally, in chapter 5, using GRO-seq and PRO-seq data from mouse and human data, I summarize regions of bidirectional transcription and build GRNs where I link cis-regulatory regions to genes across diverse human tissues. Put together, these works provide a resource for further exploration of the regulome and identification of previously unknown interactions. Chapter 6 is a summary of contributions and proposed future work to build GRNs using nascent RNA data. In what follows, I briefly summarize the major contributions of each chapter.

Chapter 2: Classifying nascent RNA datasets with wavelet transform analysis

In the last few years, there has been an expansion of protocols aimed at capturing nascent RNA. However, until this publication, there had not been a systematic comparison of these methods, and their implications for downstream algorithms and analyses. Here, in collaboration with Dr. Samuel Hunter, we characterize the different protocols along with the library preparation methods using data from the same biological system allowing for a direct comparison of technologies [58]. There were notable differences in the signatures from the PRO and GRO methods, despite the primary difference between the protocols being the identity of the tagged nucleotide. Notably, the 5′ end of transcripts yielded distinct signatures that influence the annotation of bidirectional transcripts downstream. In this work, I developed a wavelet transform analysis method that was able to distinguish GRO and PRO protocols from read coverage data alone. Importantly, despite the protocol differences, we were able to show that the underlying biological signal was not lost in downstream analyses. This chapter highlights the care needed to integrate data from these protocols into a meta-analysis study.

Chapter 3: Methods for identifying regions of nascent RNA transcription

In this chapter, I review methods for annotation-agnostic approaches to identifying regions of bidirectional transcription from nascent sequencing data. I detail the best practices for handling this unique data, from quality control assessments, to the identification of transcribed regions. In this methods review paper I give examples of input and output data and carefully walk a reader through the basic data husbandry. This work was written for a Methods in Enzymology issue that has yet to appear. In the larger scheme of the thesis, stepping through these methods informed the development of a workflow for the systematic processing of a large collection of nascent datasets to identify regions of bidirectional transcription in an annotation-agnostic manner.

Chapter 4: Transcription factor enrichment analysis with nascent RNA sequencing data

In this chapter, I present a method for inferring transcription factor activity from nascent transcription data. Earlier work from the Dowell lab showed that regions of bidirectional transcription are enriched for TF motif sequences and these can be used to infer which TFs respond to a perturbation [8]. However, the earlier work did not account for changes in the magnitude of bidirectional transcription. In a collaboration with Dr. Jonathan Rubin and Dr. Jacob Stanley, we developed transcription factor enrichment analysis (TFEA), which measures TF activity using co-localization of TF motifs with bidirectionals and the differential transcription of bidirectional regions in a rank-based method [59]. The method requires data from both a control and perturbation, from which the patterns of enhancer associated RNA changes is used to infer TF activity changes. In this effort we also introduced a tool (muMerge) for merging bidirectional regions across independent experiments in a systematic, probabilistic manner. My contribution to this work was that I tested TFEA on several experimental perturbations and showed that the algorithm does recover expected TFs. I also conducted an analysis to determine how the regions overlap TF ChIP-seq data. Since this publication, I have done extensive work on the benchmarking and debugging of muMerge, as it is an essential tool for a large scale meta-analysis.

Chapter 5: Building gene regulatory networks with nascent RNA sequencing data

Finally, I present dbNascent, a repository of published nascent RNA data. I summarize the quality of the data sets collected across organisms. Accounting for the sample quality, I present a consensus set of regions of bidirectional transcription across hundreds of human and mouse samples. I further assess the tissue specificity of gene and bidirectional transcripts, which both extends and supports previous findings that bidirectionals are more tissue-specific than genes or lncRNAs. I then use these data to generate GRNs that link bidirectional and gene transcripts using a correlation based framework. This work builds on the proper consideration of the protocol-specific differences pointed out in Chapter 2, systematic calling of bidirectionally transcribed regions (Chapter 3), and the regulatory linkages identified in Chapter 4.

CHAPTER II

CLASSIFYING NASCENT RNA DATASETS WITH WAVELET TRANSFORM ANALYSIS

This chapter is adapted from:

Samuel Hunter, **Rutendo F. Sigauke**, Jacob T. Stanley, Mary A. Allen, Robin D. Dowell. *Protocol variations in run-on transcription dataset preparation produce detectable signatures in sequencing libraries.* BMC Genomics. 2022 Mar 7;23(1):187. doi: 10.1186/s12864-022-08352-8. PMID: 35255806; PMCID: PMC8900324.

In this work, I developed a wavelet transform analysis method that was able to distinguish GRO and PRO protocols from read coverage data alone.

Abstract

Background:

A variety of protocols exist for producing whole genome run-on transcription datasets. However, little is known about how differences between these protocols affect the signal within the resulting libraries.

Results:

Using run-on transcription datasets generated from the same biological system, we show that a variety of GRO- and PRO-seq preparation methods leave identifiable signatures within each library. Specifically we show that the library preparation method results in differences in quality control metrics, as well as differences in the signal distribution at the 5′ end of transcribed regions. These shifts lead to disparities in eRNA identification, but do not impact analyses aimed at inferring the key regulators involved in changes to transcription.

Conclusions:

Run-on sequencing protocol variations result in technical signatures that can be used to identify both the enrichment and library preparation method of a particular data set. These technical signatures are batch effects that limit detailed comparisons of pausing ratios and eRNAs

identified across protocols. However, these batch effects have only limited impact on our ability to infer which regulators underlie the observed transcriptional changes.

Background

The transcriptome dictates much of a cell's identity and behavior. As such, tracking how transcription patterns change in response to a biological perturbation is a popular approach to understanding molecular regulatory mechanisms. In particular, newly transcribed RNAs provide a readout on the activity and regulation of cellular polymerases. Capturing and mapping these "nascent" transcripts provides a single base-pair resolution readout of the positions of all cellular RNA polymerases throughout the genome[41, 42, 60]. Changes in RNA polymerase behavior are associated with transcription factor activity[8, 59, 61], with a large portion of the changes occurring within enhancer regions. These enhancer RNAs (eRNAs) are unstable and thus not generally recovered by steady-state assays such as RNA-seq, which sample predominantly from the pool of stable transcripts such as mRNAs[62].

To capture all RNAs arising from cellular RNA polymerases, several run-on transcription capture protocols, such as global run-on sequencing (GRO-seq) and precision run-on sequencing (PRO-seq), have been developed[41, 60, 63–65]. These protocols, collectively known as RO-seq, follow roughly a two step process: first, the run-on RNA signal must be enriched above the background total RNA; second, the captured RNA is then converted into a sequencing-ready cDNA library[41]. For the first step, run-on protocols share the same basic strategy, namely use an enrichable nucleotide as a handle for distinguishing nascent RNA from previously produced RNA (Fig. 2.1A). Subsequently, sequencing adapters are added and the sample is reverse transcribed and amplified in preparation for sequencing. As these steps are somewhat modular, the process of enrichment is often interleaved with the various steps necessary for library preparation (Fig. 2.1B).

Similar to distinct RNA-seq library preparation methods, processing RNA through different RO-seq protocols is thought to leave technical artifacts within the library[66–68]; however, the extent to which these artifacts influence the resulting analysis has not been thoroughly explored. In this study, we sought to identify specific signatures and biases inherent to

the protocol (enrichment strategy) and library preparation methods typically employed in RO-seq methods. For this comparison, we generated data from HCT116 cells treated for 1 hour with the p53 activator Nutlin-3a or a DMSO control, a well studied perturbation[61, 69]. Using these matched datasets, we find specific and reproducible biases in each respective dataset that influence both the quality metrics and 5′ distribution of reads. However, we find that these protocol and library specific effects do not strongly impact the inference of which transcription factor is driving the observed perturbation induced changes in transcription. These protocol-specific signals could enable an agnostic detection program to identify the protocols used; such programs could then be utilized to increase the validity of online sequence databases.

Results

Quality metrics are influenced by RO-seq transcription capture protocols

The ultimate goal of run-on protocols is to produce a dataset that accurately reflects the distribution of actively transcribing RNA polymerase [41, 70] genome wide. However, success in this endeavor depends greatly on the sequencing depth, library complexity, quality of enrichment, and transcription strength of the cell line [71]. To control for cell line differences, we generated run-on libraries from HCT116 cells using a previously employed perturbation strategy[61, 69]. Namely, we used global run-on (GRO) sequencing[41] with a Br-tagged UTP, and precision run-on (PRO) sequencing[42] with a Biotin to mark CTP [60] (Fig. 2.1A) as enrichment protocols. We then combined these enrichment protocols with one of four library preparation techniques: RNA adapter ligation (LIG)[41], Circularization (CIRC)[63], Random Priming (RPR)[72], or Template-Switching Reverse Transcription (TSRT)[64] (Fig. 2.1B) after either 1 hr DMSO control or 1 hr treatment with Nutlin-3a. Nutlin-3a is a molecule which interrupts p53 inhibition and leads to rapid transcription of downstream p53 targets (see Materials and Methods). Samples were subsequently sequenced on an Illumina NextSeq 500 platform (RTA version: 2.4.11, Instrument ID: NB501447) using a single end strategy (37, 50 or 75 bp lengths) to variable depths (summarized in Supplemental Table 1, see Materials and Methods).

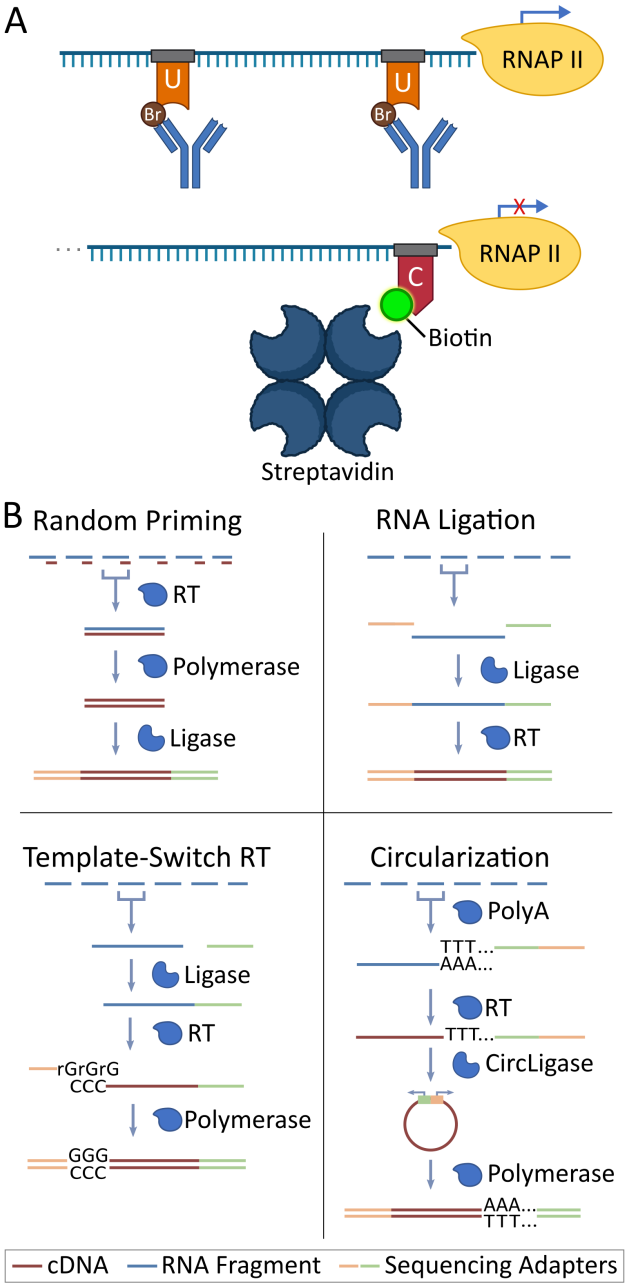


Figure 2.1: (A) Summary diagram indicating enrichment steps for Global Run-On (GRO-seq, top) and Precision Run-On (PRO-seq, bottom) reactions. (B) Summary diagram for library preparation reactions. Blue bars: RNA; brown bars: cDNA; yellow/green bars: sequencing adapters. Library preparation enzymes are labeled and represented by blue shapes at each step.

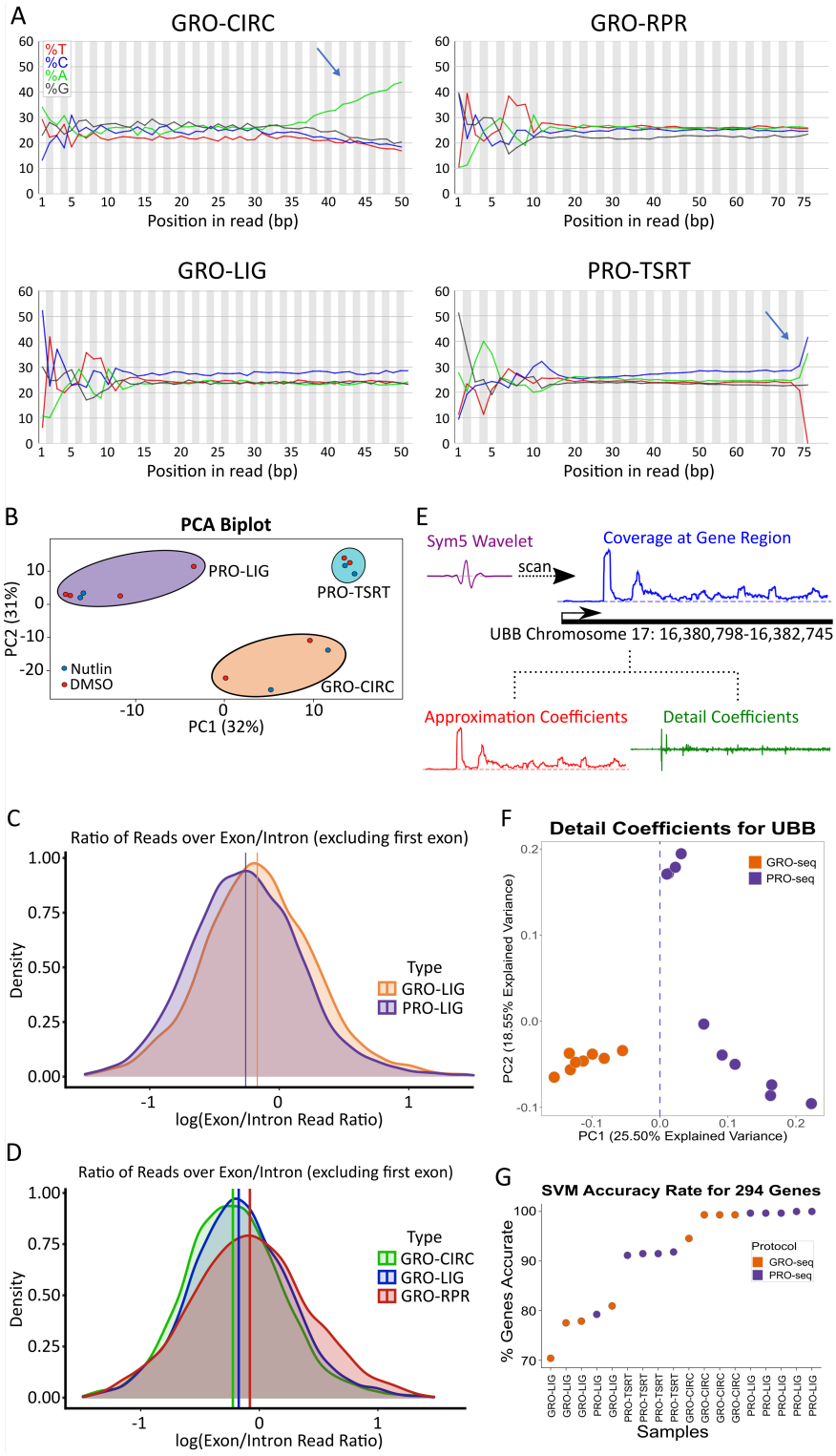
The first noticeable differences between any two datasets (even with the same protocol/library preparation) are depth of sequencing and complexity of the library. The depth of our samples range from 20 million to 170 million reads. We correct for the disparity in sequencing depth by combining the technical replicates of low-depth samples, and by subsampling deeply sequenced samples. As such, all subsequent comparisons were performed at equivalent depth (with a minimum of 75 million reads).

In contrast, library complexity reflects data quality and cannot be corrected for computationally and ideally would be similar between library preparations before comparison. We use two metrics to assess complexity, the number of unique reads relative to the depth of the sample and the number of unique bases covered within the genome (Supplemental Table 1). While most of our libraries were comparably complex, we found that our libraries generated with a random-priming library kit were generally of lower complexity. The random-priming strategy is rarely used and thus, it is unclear whether the tendency of reduced complexity is a consequence of the library preparation method or a fault of our handling. However, public random primed datasets exhibited similar 5′ read distributions to our datasets in spite of the differences in library complexity (Supplemental Figs. 2.1,2.2,2.3, Supplemental Table 1); therefore, we chose to include these libraries in our initial analyses to showcase possible technical signatures and potential biases, but refrained from using GRO-RPR libraries in further comparative analyses.

Notably, some library preparations result in clearly distinguishable sequence signatures within the acquired reads. In circularization (CIRC) libraries, regardless of the enrichment protocol, RNA is polyadenylated before reverse transcription, and the resulting cDNA is subsequently circularized via the enzyme circLigase[63]. As such, it is common to see many reads with long poly(A) tails before trimming (Fig. 2.2A). Additionally, the TSRT library preparation adds several C nucleotides to the end of each read[64]. Upon sequencing and adapter trimming, many read inserts showed an increased incidence of C nucleotides near the end of the read (Fig. 2.2A). In our samples, these sequence signals can effectively distinguish CIRC and TSRT libraries from the other library preparation methods. In contrast, LIG and RPR libraries

show similar nucleotide composition across the reads. Likewise, GRO and PRO datasets constructed with matched library preparation methods are not distinguishable from sequence content signatures alone.

However, principal component analysis (PCA) of the read counts over all genes tightly clusters based on library preparation and enrichment protocol, suggesting there are additional protocol-distinguishing features not evident in the average nucleotide composition of the dataset (Fig. 2.2B). Therefore, we next sought to identify whether enrichment quality metrics could be used to distinguish between the protocols. Quality control pipelines offer a way of quantifying steady-state RNA contamination by calculating the ratio of reads over exons and introns for each gene. While the specific value expected for this ratio depends on how reads are counted, a comparatively lower exon-intron ratio is indicative of less mRNA contamination[73]. But is this exon-intron ratio influenced by the choice of protocol? To answer this, we calculated log-normalized exon-intron ratios for every gene in each HCT116 control (DMSO) library. On average, PRO libraries showed a slightly lower amount of mRNA contamination across all genes relative to GRO libraries, consistent with the relative strength of the two enrichment strategies (Fig. 2.2C). Additionally, both CIRC and LIG libraries showed lower mRNA contamination relative to RPR libraries (Fig. 2.2D).



Quality Control metrics for varying library preparation and enrichment techniques.

Figure 2.2: Quality Control metrics for varying library preparation and enrichment techniques. (A) Nucleotide distribution of DMSO samples are plotted indicating the percent nucleotide representation (y-axis) versus the position within each read (x-axis). Library specific signatures are identifiable in CIRC and TSRT libraries (blue arrows). (B) Principal-Component Analysis of assorted library preparation and enrichment methods. Each library was prepped using HCT116 cells treated with either DMSO or Nutlin-3a for 1 hour. Log-normalized density plots of exon/intron ratios for each gene for each (C) enrichment method and (D) library preparation method (GRO-seq samples shown), (GRO-LIG vs PRO-LIG: $p < .001$; GRO-CIRC vs GRO-LIG: $p < .05$; GRO-CIRC vs GRO-RPR: $p < .001$; GRO-RPR vs GRO-LIG : $p < .001$, K-S Test, n=1795). Mean indicated by vertical line for each respective distribution. (E) Schematic showing the wavelet transformation approach at the UBB locus. (F) Detail coefficients at UBB locus separates PRO and GRO libraries on PC1 (Low-biotin PRO-seq samples omitted, see Supplemental Table 1). (G) SVM classifier results for each tested library.

Sequence composition (Fig. 2.2A) can be utilized to identify CIRC and TSRT library preparation protocols with high confidence, while LIG and RPR libraries were more similar in sequence composition, albeit with some differences in complexity and quality metrics (Fig. 2.2D, Supplemental Table 1). However, the differences between the enrichment protocols (GRO vs PRO) is less readily apparent from sequence composition or quality metrics alone (Fig. 2.2A,C, Supplemental Table 1). Yet, we wondered whether systematic signals exist within the data that could distinguish between the protocols. To this end, we applied a discrete wavelet transform (DWT) approach to the normalized coverage of each library (Fig. 2.2E). The DWT decomposes the signal in a region into low frequency signals (approximation coefficients) that capture consistent RNA polymerase signatures and high frequency signals (detail coefficients) that contain noise. The noise component captures both random noise and systematic noise. Because protocol specific signatures are a systematic source of noise, we reasoned that the high frequency signals may be able to distinguish between the protocols.

To test this hypothesis, we sought to evaluate the DWT on a set of genes where RNA polymerase signatures are the least influenced by library depth or complexity. Thus we identified a set of 294 highly transcribed genes that also had a low coefficient of variation across our datasets. Using the PyWavelets package in python, a symlet wavelet was scanned over the normalized coverage of each gene, effectively decomposing the signal into the two components (see Materials and Methods) (Fig. 2.2E)[74, 75]. Subsequently, we used principal component analysis (PCA) to cluster the detail coefficients. Overall, 117 genes (39.8%) separated the protocols (GRO vs PRO) directly on the first principle component whereas an additional 162 (55.1%) genes separated the protocols on a different plane within the PC1 and PC2 space (Fig. 2.2F, Supplemental Fig. 2.4, 2.5). These results suggested that the data sets contain a readily identifiable protocol signature. To confirm, we built a simple support vector machine classifier to determine whether the principle components of the wavelet analysis could be used to identify the protocol directly from the data (see Materials and Methods) (Supplemental Fig. 2.6). Using leave-one-out cross validation at the individual gene level, the classifier correctly identified the

protocol >70% of the time (Fig. 2.2G, Supplemental Fig. 2.7). Furthermore, applying a simple majority rules voting scheme to the classifier results identified the protocol every time (100%), further confirming that each data set contains identifiable protocol specific signatures.

Enrichment and Library Preparation Methods Significantly Shift 5' Distribution

To better understand the protocol specific signatures within the data sets, we next examined annotated, protein-coding genes for systematic differences in their read distributions. At protein-coding genes, the behavior of RNA polymerase II is well characterized[76] which leads to repeatable patterns of read distribution throughout the gene (Fig 2.3A). Therefore, we sought to determine whether the protocol (GRO vs PRO) led to systematic differences in the detected 5′ initiation region or the elongation region. Counts across gene body regions suggested that elongation regions correlated well between protocol and library preparation differences (Supplemental Fig. 2.8, see also Materials and Methods); therefore, we subsequently focused our attention on the 5′ regions of genes.

To assess the differences in the 5′ distribution across protocols, we examined the read distribution of GRO and PRO libraries prepped from DMSO-treated HCT116 cells, with an otherwise similar library preparation protocol (LIG). Metagenes revealed a shift in coverage near many transcription start sites (TSS) in PRO libraries that is not present in GRO libraries (Fig. 2.3B, Supplemental Fig. 2.9). GRO and PRO libraries differ in the nucleotide analog used to enrich for nascent RNA. In GRO-seq, bromouridine-triphosphate is used to mark newly transcribing RNAs which can then be detected by anti-BrdU antibodies. In contrast, PRO-seq uses Biotin-NTPs which also terminate transcription upon their incorporation into the nascent RNA. Streptavidin then efficiently isolates newly transcribed RNAs. The original PRO-seq strategy marked all four nucleotides to maximize precision[41], but for cost efficiency, subsequent efforts only marked a single nucleotide[60]. Notably, both the efficiency of pull down and the termination of transcription results in PRO-seq giving a more precise readout on the position of RNA polymerases relative to GRO-seq[42]. However, at the 5′ end this precision also results in short unmappable reads, leading to gaps in coverage near the TSSs[60]. In an attempt to mitigate

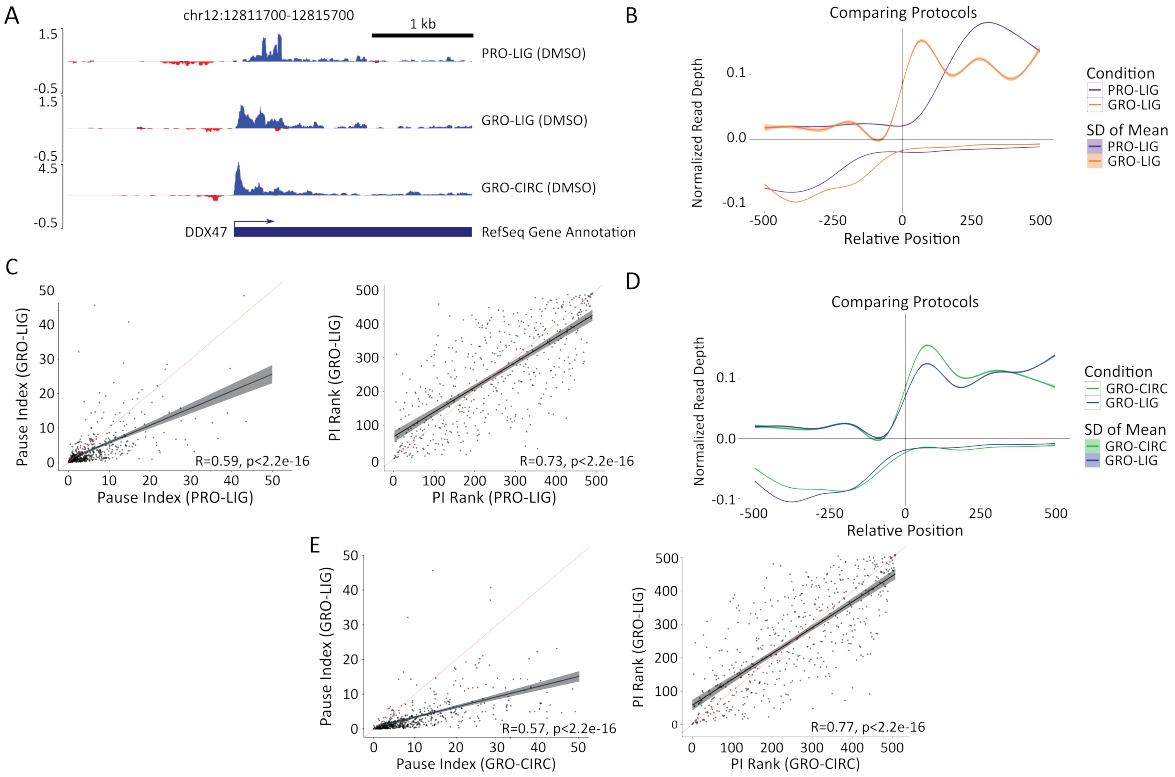


Figure 2.3: Analysis of gene transcription start sites among different protocols and library preparations. (A) Genome viewer screenshot of 5′ end distribution among various library preparation and enrichment methods. Negative read depth represents reads found on the minus strand. (B) Metagenes constructed from GRO-seq (orange) and PRO-seq (blue) libraries (Ligation based library preparation, HCT116, DMSO 1hr). Genes shorter than 2000 bp were removed, genes with significant signal 2 kb upstream (>1% of upstream bases covered), and genes with low coverage (TPM < .01) were removed (n=2527). Vertical line indicates TSS annotated in RefSeq database. Distance from TSS is in bp, read depth was normalized by counts-per-million (CPM). (C) Pausing index calculations for top 500 most transcribed genes in GRO-seq and PRO-seq libraries, presented with Pearson (left) and Spearman (right) correlations (red line: y=x, black line: best fit). Pausing region is defined as -50 bp to 250 bp from annotated TSS (See Materials and Methods). (D) Metagenes constructed from GRO-seq Ligation (blue), and Circularization-based (green) libraries (HCT116, DMSO 1 hr). Genes shorter than 2000 bp, genes with significant signal 1 kb upstream (>1% of upstream bases covered), and genes with low coverage (TPM < .01) were removed (n=2527). Vertical line indicates TSS annotated in RefSeq database. Distance from TSS is in bp, read depth was normalized by counts-per-million (CPM). (E) Pausing index calculations for Circularization and Ligation based libraries (GRO-seq, HCT116, DMSO 1 hr), graphed as in (C).

these 5′ read coverage gaps, subsequent variations in the PRO-seq protocol include a ratio of Biotin-NTP/NTP to the run-on mixture [60].

We theorized that the shift in the 5′ region observed in our PRO libraries arose from early incorporation of Biotin-NTP near the TSS which leads to short, truncated reads that are not well mapped. As such, we reasoned that generating new libraries with a different ratio of Biotin-NTP/NTP in the initial run-on mixture would result in more reads captured around the 5′ end (Supplemental Fig 2.10). Metagenes indeed show a smaller shift with lowered Biotin-NTP concentration, although GRO-LIG libraries continued to show more signal in these regions than any PRO library.

To ensure that our findings generalize to other data sets, we next examined publicly available datasets. While these data sets likely have larger batch effects arising from their preparation in distinct laboratories and cell types, we reasoned that the overall trend in 5′ end patterns should still be noticeable, albeit subject to more variance. GRO and PRO libraries obtained from other labs showed that the peak of PRO-seq libraries was noticeably further downstream than their GRO-seq counterparts; however, this comparison (using a consistent mapping and analysis strategy, see Materials and Methods) uncovered a broad range of peak positions (from +40 bps to +250 bps) with seemingly no linear relationship between the Biotin-NTP/NTP ratio and peak position (Fig 2.3B Supplemental Fig 2.11, 2.12, 2.10).

Therefore, we reasoned that there must be further underlying protocol influences on the 5′ read distribution. Differences in size selection, read fragmentation, and gene filtering criteria were all hypothesized to influence the distribution. To evaluate these criteria, we took an *in silico* approach and simulated reads arising near a TSS from each protocol configuration (see Materials and Methods). Briefly, positions of potential polymerase occupancy were sampled from a simulated gene, including both initiation and elongation regions. For each polymerase position, we extended the hypothetical RNA based on the gene template downstream of the polymerase position, with the designated probability of incorporating a Biotin-NTP and halting extension. The subsequent read was then filtered by size selection and plotted to generate simulated

metagene traces (Supplemental Figure 2.13). Using these simulations, we found that the 5′ peak position was influenced by both the Biotin-NTP run-on ratio and the size selection criteria.

To validate our *in silico* findings, we returned to the data and examined the distribution of short reads (less than 30 bps) relative to transcription start sites. We reasoned that short fragments would consist of a combination of TSS associated fragments truncated by Biotin-NTP incorporation and small fragments arising from sample handling, which should be randomly distributed throughout the genome. Hence the ratio of short reads near TSS relative to all short reads should be indicative of the ratio of labeled and unlabeled NTPs used in the run-on reaction. Indeed, the short read ratio does shift along the Biotin-NTP ratio, but not as a monotonically increasing function (Supplemental Fig. 2.14). Consistent with our simulations, intermediate Biotin-NTP/NTP ratios returned the highest fraction of mappable TSS associated short reads. Our results indicate that several library preparation elements, such as size selection, Biotin-NTP run-on ratios, and mappability strongly influence the 5′ distribution. Importantly, this work also suggests that the ideal run-on scenario is a balance between producing reads that are long enough to escape size selection and map effectively; yet remain short enough to accurately report on the position of RNA polymerase.

We next reasoned that the observed differences in the detected 5′ read distribution at genes would commensurately affect the pausing index (PI), measured as the ratio of reads in the initiation region relative to the gene body[77]. We defined the initiation region as 50 bp upstream from the annotated TSS to 250 bp downstream of the TSS; gene body regions were defined as 251 bp downstream of the TSS to the annotated cleavage site. Using these regions, we calculated the PI for the longest isoform of each gene in both libraries. Consistent with our findings above, PI for individual genes were reasonably consistent across replicates (Supplemental Fig. 2.15) but showed significant disparities between GRO and PRO libraries (Fig. 2.3C, $R = 0.59$, $p < 2.2e\text{-}16$). Spearman rank correlations for PI in both libraries were marginally higher ($R = 0.73$, $p < 2.2e\text{-}16$). These overall trends were also observed within PI distributions when we extended this analysis to publicly available data (Supplemental Fig. 2.12). While the PI is known to depend on the method

used to define the paused region[70], we found that the trends across protocols remained consistent even with different pause windows and read counting software (Supplemental Fig. 2.16).

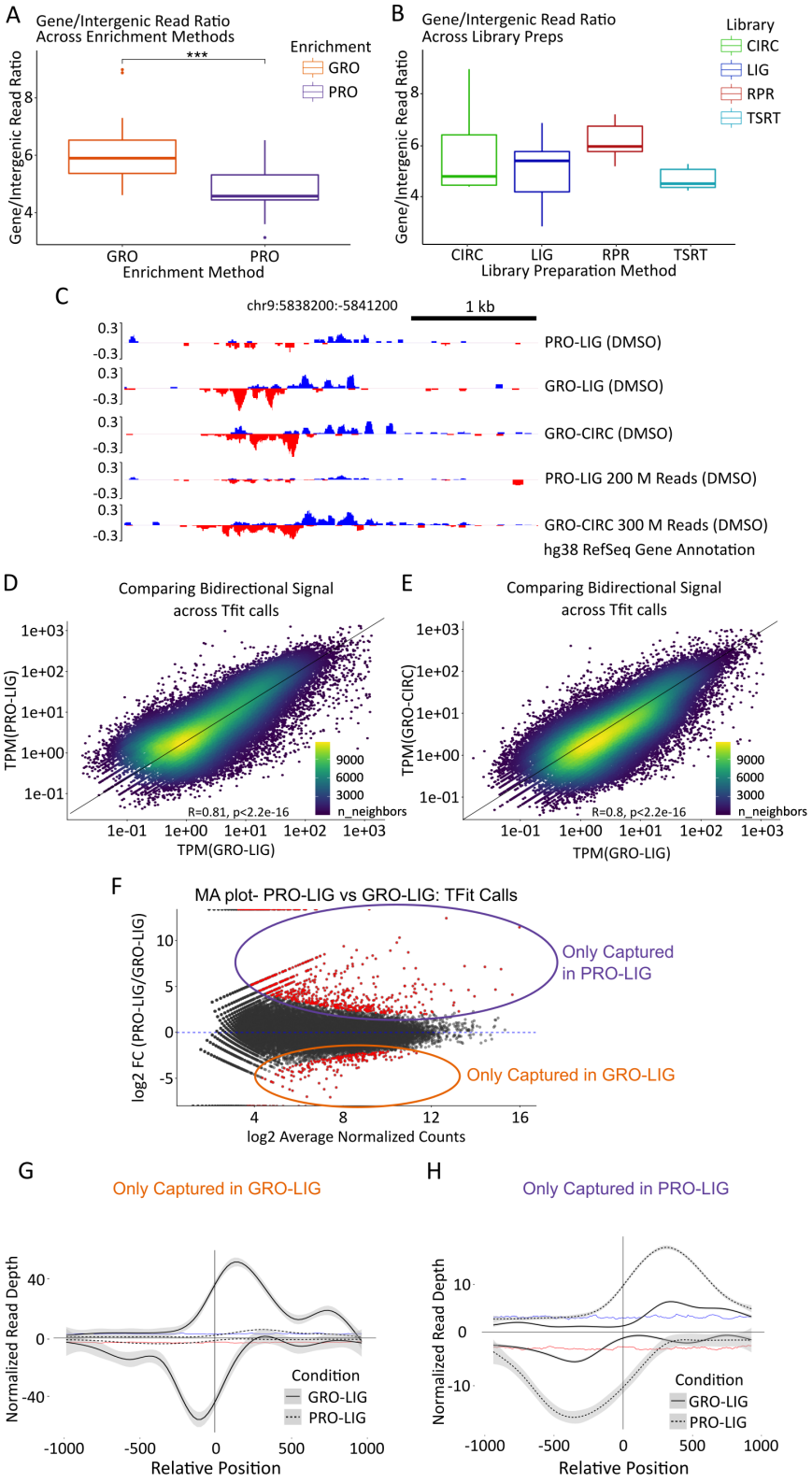
Next, we evaluated the effects of library preparation on the 5′ end. To accomplish this, we constructed metagene summaries of our GRO-CIRC, GRO-LIG, and GRO-RPR libraries (Fig 2.3D). While CIRC and LIG libraries showed a similar distribution near the 5′ end, GRO-RPR libraries show a shift in coverage that leaves a significant gap near the annotated start site (Supplemental Fig. 2.2). While it is unknown what leads to this shift, we theorize that random priming has a length bias that is a contributing factor (i.e. the longer a RNA is the more likely a primer is to anneal to it).

Additionally, we found that the pause ratio is sensitive to which method is used to prepare the RNA. We compared pause index calculations for GRO-CIRC and GRO-LIG libraries. We found that, for each gene, pause indices tended to be larger for GRO-CIRC libraries compared to GRO-LIG libraries (Fig. 2.3E, $R = 0.57$, $p < 2.2e\text{-}16$). To assay whether this shift was systematic, we also computed the Spearman rank-correlation for these indices. Rank correlation between GRO-LIG and GRO-CIRC libraries was stronger than Pearson correlation; however, there were still many genes that showed disparate rankings across our datasets (Fig. 2.3E, $R = 0.77$, $p < 2.2e\text{-}16$).

Changing library enrichment methods shifts intergenic read distributions and active enhancer detection

The bidirectional transcription typical of RNA polymerase initiation regions at the 5′ end of genes is also present at enhancers[6], albeit typically at much lower transcription levels. Therefore, we asked whether the patterns of enhancer transcription varied across protocols or library preparations. As a first pass inquiry that avoids reliance on enhancer annotations, we first compare the fraction of reads recovered from RefSeq annotated gene regions to reads recovered in intergenic regions for each data set. To ensure more statistical rigor, we included several publicly available datasets of different cell lines, along with six libraries we previously generated from MCF10A cells prepped with PRO-TSRT (See Supplemental Table 1). When comparing GRO and

PRO libraries (irrespective of cell type or library preparation method), we found that GRO libraries showed significantly more reads over gene regions compared to PRO libraries (Fig. 2.4A, $p < .01$, See Materials and Methods). Conversely, we found no significant differences when comparing library preparation methods (Fig. 2.4B).



Analysis of enhancer elements in multiple datasets.

Figure 2.4: Analysis of enhancer elements in multiple datasets. (A,B) Number of reads counted over RefSeq annotated gene regions divided by the number of reads counted over intergenic (unannotated) regions, for each dataset analyzed. The datasets represented here are all those listed in Supplemental Table 1, including public datasets. Datasets were first analyzed by enrichment method (GRO-seq (n=23) vs. PRO-seq (n=21), $p < .01$), then by library preparation method (LIG (n=17) vs CIRC (n=10) vs TSRT (n=10) vs RPR (n=7), $p > .05$). We note that the RPR boxplot includes 3 of our lower quality datasets; however, we chose to include them here owing to the scarcity of RPR datasets in the RO-seq database. These are otherwise excluded from further analysis. (C) Example section representing disparate representation of reads from our in-house datasets over an enhancer, even at high depths. (D, E) Scatterplots representing reads over Tfit (enhancer) calls (calls combined by MuMerge, counts normalized by TPM). (F) MA plot of calls found in (D). Red dots are significant ($p < .05$). (G, H) Metagenes of significant hits found in (F). Vertical line indicates the approximated center of the bidirectional transcripts as determined by Tfit. Distance from the center of the bidirectional is in bp, read depth was normalized by counts-per-million (CPM). (G): Calls that were differentially captured in GRO-LIG (n=1350). Background signal on the plus strand is indicated by the blue trendline, while background signal on the minus strand is indicated by the red trendline. (H): Calls that were differentially captured in PRO-LIG (n=3050), with the background signal depicted as in panel G.

The disparity in the gene-to-intergenic reads ratio in GRO and PRO libraries suggest their respective enrichment strategies may capture signal in unannotated regions at different rates. In particular, we were curious whether the capture of eRNAs would be affected by the choice of protocol. To investigate this possibility, we first examined annotated enhancers in the HCT116 cell line acquired from the FANTOM database (converted to hg38 coordinates using the online UCSC tool liftOver)[78]. The level of transcription between these enhancers was largely consistent between our datasets (Supplemental Fig 2.17). However, FANTOM annotated enhancers represent the comparatively stable enhancer transcripts arising from Cap Analysis Gene Expression (CAGE) data[79].

Therefore, we next sought to identify enhancers directly from the data using their characteristic bidirectional transcription signal[80]. Two algorithms have been developed to identify transcribed regulatory elements based on their bidirectional signal, dREG[7] and Tfit[81]. We employed both methods to annotate sites of bidirectional transcription in our GRO-CIRC, GRO-LIG, and PRO-LIG libraries. Strikingly, the identified regions varied substantially across protocol and library preparation for both algorithms (Supplemental Fig. 2.18). We hypothesized that these differences may be exaggerated by the sequencing depth, as eRNAs are lowly transcribed and therefore these regions are only consistently detectable at high sequencing depth. To this end, we combined replicates for PRO-LIG libraries to an effective depth of approximately 200 million reads, and replicates of GRO-CIRC libraries to an effective depth of approximately 300 million reads. Transcribed regions identified in these combined libraries remained inconsistent; while many strong enhancers were called in both of these two deep data sets, other regions were exclusively found in only one (Fig. 2.4C, Supplemental Fig. 2.19).

This suggested the existence of transcribed regions whose signal is strongly dependent on the underlying experimental protocol. To confirm this possibility, we next sought to identify the set of transcribed regions with apparent differential transcription across protocols or library preparations. To compare enrichment protocols, we combined Tfit regions from PRO-LIG and GRO-LIG libraries (Fig 2.4D, Supplemental Fig. 2.17, see Materials and Methods), while library

preparation methods were compared by combining Tfit regions from GRO-LIG and GRO-CIRC libraries (Fig 2.4E). In every case, regions were combined using *muMerge*[59] and differential read signal was assessed with DESeq1 analysis (Fig. 2.4F). We then constructed metagenes from set of regions with differential signal (Fig. 2.4G, H, Supplemental Fig. 2.20) and observed strong bidirectional signal in only one of the two datasets, while the other dataset showed signal only slightly above background. Manual inspection confirmed that these transcribed regions were only effectively captured by one library, even at high depths (Fig. 2.4C).

Biological response to p53 activation is preserved across run-on transcription capture protocols

The protocol-specific nature of both pausing ratios and eRNA recovery led to concerns about whether the choice of experimental preparation influences commonly conducted downstream analyses, such as identifying which genes respond to a perturbation[61] and which transcription factors drive those changes[8, 59, 82, 83]. As such, we used the competitive MDM2 inhibitor Nutlin-3a, which has a known, specific, robust transcription response in human cells induced by the subsequent activation of the transcription factor p53[61, 69, 84].

First, we sought to determine the reproducibility of detecting differential gene transcription within our libraries. The precise identity of which genes respond to 1 hour of p53 activation is expected to vary across protocols and library preparations – as similar batch effects have been observed for RNA-seq libraries[85]. Thus, we focused specifically on whether the core p53 response program, i.e. the known targets of p53, was captured efficiently in each dataset. To this end we utilize the Gene Set Enrichment Analysis (GSEA) - Preranked[86, 87] tool on ranked, signed p-values obtained from DESeq2[88] (See Materials and Methods). Additionally, we expected that a substantial amount of variation between two libraries generated from different protocols would arise from the gene initiation region (Fig. 2.3). To confirm this, we subsequently examined two distinct methods of calculating differential gene transcription: the commonly used elongation-region-only approach and the full annotated gene region (Fig. 2.5A). Across all libraries and counting methods, the p53 pathway was the top hit in the GSEA-Preranked module

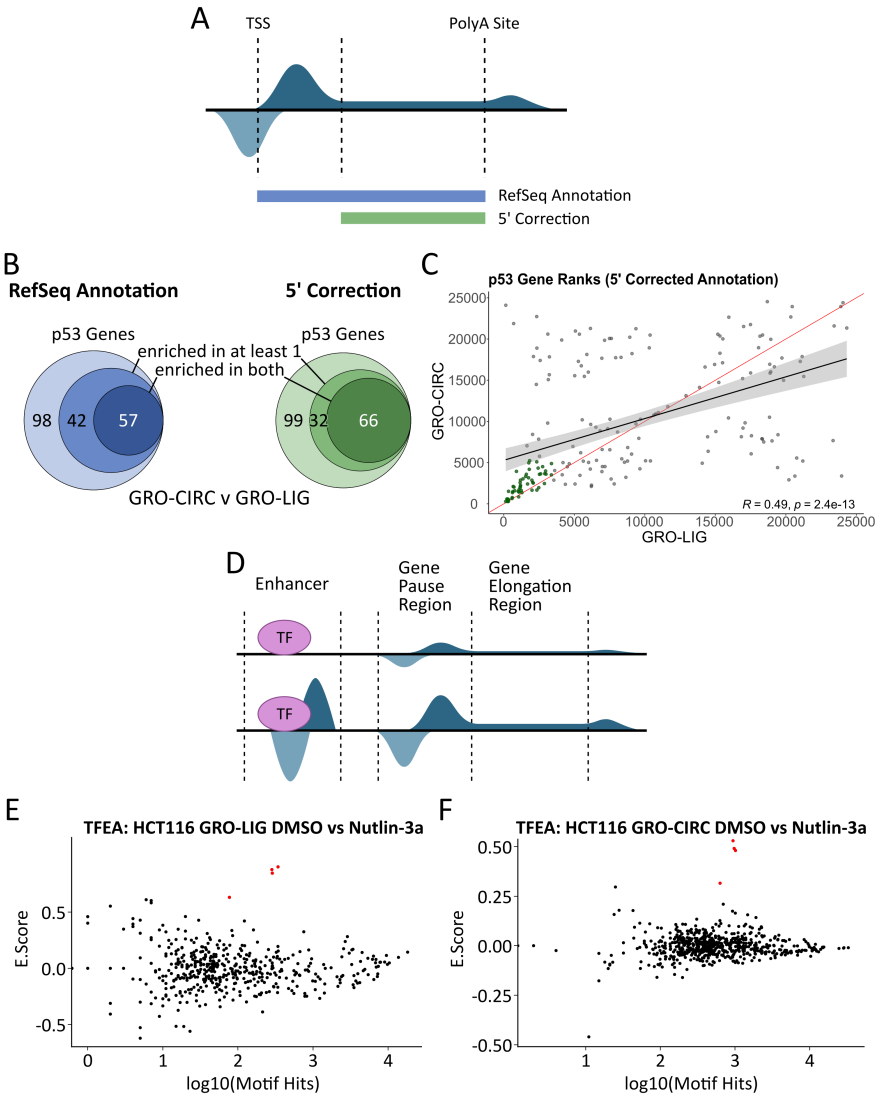


Figure 2.5: TFEA and DESeq2 analyses of library preparation methods. (A) Cartoon schematic demonstrating uncorrected (RefSeq Annotation) and 5′ corrected counting methods. (B) GSEA gene rank comparison of HALLMARK_P53 Gene set. Overlap is shown as genes that enrich in both datasets, genes that enrich in only one dataset, and genes that do not enrich in either dataset (Left: Uncorrected annotation, hypergeometric test p-value=4.32e-15; Right: Corrected annotation, hypergeometric test p-value=9.03e-22). (C) Scatterplot of comparative gene ranks for all p53 genes. Points in green indicate significant enrichment, as in (B). (Red line: y=x trendline, black line:line of best fit). (D) Representation of nascent transcription data set. Bidirectional transcripts occur at active enhancer sites and gene start sites. Enhancer transcription co-occurs with upregulated gene transcription, indicating transcription factor activation. (E) TFEA results for GRO-LIG (Left) and GRO-CIRC (Right). p53 family (p53, p63, p73) highlighted by red dots.

(FDR q-val < 0.001, Fig. 2.5B, Supplemental Fig. 2.21), suggesting that each protocol, library preparation and counting method was capable of detecting the underlying biological perturbation in spite of technical signals introduced by protocol differences.

Next, we compared the correlation of the ranks of the genes in the Hallmark p53 pathway used by GSEA. We found that the majority of enriched genes were common between each of the libraries (58.3% in GRO-LIG vs GRO-CIRC, 57.1% in GRO-LIG vs PRO-LIG) (Fig. 2.5B,C, Supplemental Fig. 2.22). However, there remained several genes that were only enriched in one of libraries. When only the elongation region was considered, the overlap improved (68.3% in GRO-LIG vs GRO-CIRC, 58.9% in GRO-LIG vs PRO-LIG), consistent with the 5′ initiation regions being the most variable portion of the gene between protocols. These results add further support to the most common method of assessing differential transcription from run-on sequencing protocols, namely excluding the 5′ initiation regions[89–92].

The second typical use of run-on sequencing data is to infer which regulators are driving observed patterns of differential transcription[6–8]. Alterations in transcription factor activity can be detected by changes in the locations and levels of sites of bidirectional transcription[8, 59], the majority of which reside at enhancers[80]. Therefore we next sought to determine whether the alterations observed in eRNA detection (Fig. 2.4) impacted TF activity inference[59].

To this end, we used the Transcription Factor Enrichment Analysis (TFEA) tool to evaluate which transcription factor motifs are enriched at transcription initiation sites with altered transcription levels in response to Nutlin-3a[59]. In all cases, TFEA correctly identifies the p53 family (TP53, TP63, and TP73) as significantly upregulated, independent of the protocol and library prep used to generate the dataset (Fig. 2.5E and F, Supplemental Fig. 2.23). Upon closer inspection, 94.59% of p53-responsive enhancers responded similarly across protocols, but 5.41% of p53-responsive enhancers were unique to a particular protocol (Supplemental Fig. 2.24, 2.25).

Discussion

We used multiple protocols and library preparations on HCT116 cells exposed to Nutlin-3a and determined that these experimental choices influence the signal of run-on sequencing libraries

in systematic and often predictable ways. The shape of the characteristic gene initiation peak is strongly influenced by the underlying protocol, while the signal at gene elongation regions remain largely consistent across protocols. Likewise, the recovery of many intergenic regions was protocol specific, even when at high sequencing depths. Despite these differences, the ability to detect p53 activation was unaffected by the choice of enrichment or library preparation protocol.

Promoter proximal pausing is a pervasive feature of RNA polymerase II activity[70]. Pausing is often quantified through calculations of the pausing index, the ratio of reads within the initiation region relative to the elongation region. While PI values are known to depend on the choices of windows used to define these regions[70], our work demonstrates that they also depend on the underlying protocol even when the details of the PI index calculation are held constant. Furthermore, genes sometimes appear to have an additional pause site downstream of the annotated TSS (Fig. 2.3E)[93]. However, we have found that these second pause sites are protocol dependent; as changes in the library preparation method shift or ablate the signal of this second peak. While more work is necessary to fully characterize how protocol choices influence the precise location of the 5′ peak, it is clear that care must be taken when comparing 5′ distributions across experiments, as batch effects strongly influence this region.

Given the uniform activity of RNA polymerase II[94], the 5′ end protocol specific patterns we observed at genes should also impact enhancer associated transcripts. The most highly transcribed eRNAs (e.g. those annotated by FANTOM) are detected equally well by each protocol, but many eRNAs are lowly transcribed. Indeed, we observe that some enhancers with relatively high read coverage in one library are not detectable using a different protocol. We were surprised that increased depth did not resolve many of these protocol specific eRNAs. The variability in eRNA detection has likely hampered efforts to answer an outstanding question in the field; namely, how many eRNAs exist throughout the genome? Combining results from many different protocols and cell types may help alleviate this issue.

This disparity in eRNA signal raises an intriguing question: which aspects of the protocols and resulting libraries contribute to the difference in eRNA capture rates? The slightly higher

exon to intron ratio (Fig. 2.2D) of GRO-seq suggests this protocol contains a higher level of contaminating mRNA[95], consistent with Br-UTP antibody enrichment being a less efficient pull down method than Biotin-streptavidin enrichment. This bias also explains why GRO-seq has a higher gene to intergenic ratio compared to PRO-seq (Fig. 2.4A). These features may lead to some lowly transcribed eRNAs being more readily detectable with PRO-seq. In contrast, the use of Biotin halts polymerase elongation in PRO-seq, giving it a higher precision on RNA polymerase position[42]. However, this also results in short, unmappable fragments near the 5′ end of transcripts, which may limit the ability of PRO-seq to capture some shorter eRNAs. This phenomenon would explain why certain eRNAs are only captured in GRO-seq. Likewise, other factors probably contribute to the recovery of eRNAs[96], including sequence composition and biological variability.

Despite the observed protocol specific differences, our downstream analysis was consistent in detecting the underlying p53 perturbation. At genes, it is customary to exclude the initiation peak from differential gene transcription analysis[89–92], and our work indicates this is a wise choice, as counting reads only over elongation regions gave more consistent results across the protocols. Yet even when using only elongation regions, protocol specific batch effects determine which exact genes appear to respond, a problem also seen with RNA-seq[67, 97]. Likewise, detection of enhancer associated RNAs showed similar protocol specific batch effects. Importantly, despite the specifics of individual genes (and eRNAs) being not fully consistent, the large scale conclusion (p53 is activated by Nutlin-3a) remained consistent. Thus nascent transcription remains a powerful approach for understanding the immediate responses to perturbations including compounds and drug activity[8, 59, 91, 98].

Conclusion

Protocol and platform differences have long been recognized as batch effect variables that introduce non-trivial experiment specific signals within high throughput sequencing data[99, 100]. Numerous efforts have focused on correcting batch effects, but it is always difficult to do so without some loss of biological signal[101, 102]. On the other hand, the distinct signals we detect

raise an intriguing possibility that protocol and library preparation information can be inferred directly from the data itself. The noise component of the data can reliably differentiate between GRO- and PRO-seq datasets with remarkable accuracy, while sequence and quality signatures can often identify the library preparation methods used to prepare the dataset. Thus an automatic detection approach could be built to confirm or correct experimental information within the short read archive, at least for run-on assays[103]. Regardless, knowing the experimental details and managing associated batch effects is necessary when comparing in house data to previously published data sets.

Materials And Methods

Cell Culture Conditions

HCT116 and MCF10A[104] cells were cultured in DMEM media supplemented with 10% FBS, 100 units/mL penicillin and 100 μg/mL streptomycin, at 37°C with 5% CO2. Cells were grown to a confluency of 60-70% in 15 cm culture dishes before passaging. Cells were passaged twice before harvesting, using PBS to wash and 0.05% w/v trypsin to detach the cells from the plate. Cells were aspirated and treated with media containing 10 μM Nutlin-3a (or DMSO) for 1 hour before harvest.

Nuclei Isolation

Post-treatment, cells were placed on ice and washed three times with ice-cold PBS. Cells were incubated on ice in 10 mL ice-cold Lysis Buffer (10 mM Tris-HCl pH 7.5, 2 mM $MgCl_2$, 3 mM $CaCl_2$, 0.5% IGEPAL, 10% Glycerol, 2 U/mL SUPERase-IN, brought to volume with 0.1% DEPC DI-water, filtered before use) for 10 minutes. Cells were scraped and collected into 50 mL Falcon tubes, and centrifuged with a fixed-angle rotor at 1000 x g for 10 minutes at 4°C. Cells were resuspended with Lysis buffer with a wide-opening P1000 tip, and washed twice with 10 mL Lysis buffer (centrifuged at 1000 x g for 5 minutes at 4°C). After the second Lysis buffer wash, the samples were resuspended with 1 mL Freezing Buffer (50 mM Tris-HCl pH 8.3, 5 mM $MgCl_2$, 40% Glycerol, 0.1 mM EDTA pH 8.0, brought to volume with 0.1% DEPC DI-water, filtered before use). Nuclei were centrifuged at 1000 x g for 5 minutes at 4°C, and resuspended with 500

μL Freezing Buffer. Nuclei were then centrifuged for 2 minutes at 2000 x g, 4°C, and resuspended in 110 μL Freezing Buffer. 10 μL was retained for counting nuclei, while the remaining sample was snap-frozen in liquid nitrogen and stored at -80°C until use.

GRO-seq and Library Preparation Methods

Ligation (LIG)

Run-on reactions were performed as in [41]. In brief, ice-cold isolated nuclei (100 μL) were added to 37°C 100 μL reaction buffer (Final Concentration: 5 mM Tris-Cl pH 8.0, 2.5 mM $MgCl_2$, 0.5 mM DTT, 150 mM KCl, 10 units of SUPERase In, 0.5% sarkosyl, 500 μM rATP, rGTP, and Br-UTP, 2 μM rCTP). The reaction was allowed to proceed for 5 min at 37°C, followed by the addition of 23 μL of 10X DNAseI buffer, and 10 μL RNase free DNase I (Promega). RNA was extracted twice with Trizol, washed once with chloroform, and precipitated with 3 volumes of ice-cold ethanol and 1-2 μL GlycoBlue. The pellet was washed in 75% ethanol before resuspending in 20 μL of DEPC-treated water. Libraries were prepared as in [41]. In brief, nascent RNA was extracted and fragmented by base hydrolysis in 0.2 N NaOH on ice for 10–12 min, and neutralized by adding a 1× volume of 1 M Tris-HCl pH 6.8. Fragmented nascent RNA was purified using Anti-BrdU beads and ligated with reverse 3′ RNA adaptor (/5Phos/GAUCGUCGGACUGUAGAACUCUGAAC/3InvdT/), and BrdU-labeled products were enriched by a second round of Anti-BrdU bead binding and extraction. For 5′ end repair, the RNA products were treated with tobacco acid pyrophosphatase (Epicenter) and T4 polynucleotide kinase (NEB). 5′ repaired RNA was ligated to reverse 5′ RNA adaptor (5′ UGGAAUUCUCGGGUGCCAAGG) before being purified by a final round of Anti-BrdU bead binding and extraction. RNA was reverse transcribed using 25 pmol of RP1 primer (5′AATGATACGGCGACCACCGAGATCTACACGTTCAGAGTTCTACAGTCCGA). The product was amplified 15 ± 3 cycles and products >150 bp (insert > 70 bp) were size selected with 1X AMPure XP beads (Beckman) before being sequenced.

Random Priming (RPR)

Run-on reactions were performed as in [41]. In brief, ice-cold isolated nuclei (100 μL) were added to 37°C 100 μL reaction buffer (10mM Tris-Cl pH 8.0, 5 mM $MgCl_2$, 1 mM DTT, 300 mM KCl, 20 units of SUPERase In, 1% sarkosyl, 500 μM ATP, GTP, and Br-UTP, 2 μM CTP). The reaction was allowed to proceed for 5 min at 30°C, followed by the addition of 23 μL of 10X DNAseI buffer, and 10 μL RNase free DNase I (Promega). RNA was extracted twice with Trizol, washed once with chloroform, and precipitated with 3 volumes of ice-cold ethanol and 1-2 μL GlycoBlue. The pellet was washed in 75% ethanol before resuspending in 20 μL of DEPC-treated water. Libraries were prepared based on the NEBNext Ultra II Directional Library Preparation Kit. In brief, nascent RNA was extracted and fragmented by base hydrolysis in 0.2 N NaOH on ice for 10–12 min, and neutralized by adding a 1× volume of 1 M Tris-HCl pH 6.8. Fragmented nascent RNA was purified using Anti-BrdU beads (Santa Cruz Biotech, Santa Cruz, CA) 3 times. Samples were reverse-transcribed using random hexamers, and sequencing adapters added by PCR. The product was amplified 15 ± 3 cycles and products >150 bp (insert > 70 bp) were size selected with 1X AMPure XP beads (Beckman) before being sequenced.

PRO-seq and Library Preparation Methods

Ligation (LIG)

Run-on reactions were adapted from [60]. In brief, ice-cold isolated nuclei (100 μL) were added to 37°C 100 μL reaction buffer (Final Concentration: 5 mM Tris-Cl pH 8.0, 2.5 mM $MgCl_2$, 0.5 mM DTT, 150 mM KCl, 10 units of SUPERase In, 0.5% sarkosyl, 125 μM rATP, 125 μM rGTP, 125 μM rUTP, 25 μM biotin-11-CTP (additionally, two libraries generated with 25 μM biotin-11-CTP, 250 μM rCTP, see Supplemental Table 1). The reaction was allowed to proceed for 5 min at 37°C. RNA was extracted twice with Trizol, washed once with chloroform, and precipitated with 3 volumes of ice-cold ethanol and 1-2 μL GlycoBlue. The pellet was washed in 75% ethanol before resuspending in 20 μL of DEPC-treated water. Nascent RNA was extracted and fragmented by base hydrolysis in 0.2 N NaOH on ice for 10–12 min, and neutralized by adding a 1× volume of 1 M Tris-HCl pH 6.8. Fragmented nascent RNA was purified using

streptavidin beads and ligated with reverse 3′ RNA adaptor (/5Phos/GAUCGUCGGACUGUAGAACUCUGAAC/3InvdT/), and biotin-labeled products were enriched by a second round of streptavidin bead binding and extraction. For 5′ end repair, the RNA products were treated with tobacco acid pyrophosphatase (Epicenter) and T4 polynucleotide kinase (NEB). 5′ repaired RNA was ligated to reverse 5′ RNA adaptor (5′ UGGAAUUCUCGGGUGCCAAGG) before being purified by a final round of streptavidin bead binding and extraction. RNA was reverse transcribed using 25 pmol of RP1 primer (5′AATGATACGGCGACCACCGAGATCTACACGTTCAGAGTTCTACAGTCCGA). The product was amplified 15 ± 3 cycles and products >150 bp (insert > 70 bp) were size selected with 1X AMPure XP beads (Beckman) before being sequenced.

Template-Switch Reverse Transcription (TSRT)

Template-Switch Reverse Transcription protocol (also known as uPRO), was adapted from [64]. Nuclei were incubated in the nuclear run-on reaction condition (5 mM Tris-HCl pH 8.0, 2.5 mM $MgCl_2$, 0.5 mM DTT, 150 mM KCl, 0.5% Sarkosyl, 0.4 units / l of SUPERase-In) along with biotin-NTPs and rNTPs (125 μM rATP, 125 μM rGTP, 125 μM rUTP, and 25 μM biotin-11-CTP) for 5 min at 37°C. Run-On RNA was extracted using TRIzol, and fragmented with 0.2 N NaOH for 10-12 min on ice. Fragmented RNA was neutralized with 1 M Tris-HCl pH 6.8, and buffer exchanged by passing through P-30 columns (Biorad). 3′ RNA adaptor (/5Phos/GAUCGUCGGACUGUAGAACUCUGAAC/3InvdT/) is ligated at 5 μM concentration for 1 hour at room temperature using T4 RNA ligase (NEB), and nascent RNA was enriched twice with streptavidin beads. Extracted RNA was converted to cDNA using template switch reverse transcription with 1 μM RP1-short RT primer (5′ GTTCAGAGTTCTACAGTCCGA), 3.75 M RTP-Template Switch Oligo (5′ GCCTTGGCACCCGAGAATTCCArGrGrG), 1x Template Switch Enzyme and Buffer (NEB) at 42°C for 30 min. Resulting product was size selected with AMPure XP beads, and the cDNA was PCR amplified using primers compatible with Illumina Small RNA sequencing (TruSeq Small RNA primers RP1 and RPIn).

Trimming, Mapping, Visualization, Quality Control

Resulting FASTQ files were trimmed and mapped to the GRCh38/hg38 reference genome and prepared for analysis and visualization through our in-house pipeline. In short, resulting FASTQ read files were first trimmed using bbduk (v38.05) to remove adapter sequences, as well as short or low quality reads. Reads were mapped with HISAT2 (v2.1.0), and resulting SAM files converted to BAM files using Samtools (v1.8). Reads with a mapping quality less than 5 were removed, which consequently also removed multi-mapping reads. BedGraph files were generated using Bedtools (v2.25.0), and converted to TDF files for visualization using IGVtools (v2.3.75). Quality metrics were generated with FastQC (v0.11.8), Preseq (v2.0.3), RSeQC (v3.0.0), with figures generated through MultiQC (v1.6). For further version information and specific input information, see NextFlow pipeline found at https://github.com/Dowell-Lab/Nascent-Flow.git.

Exon/Intron Ratio

RefSeq annotations were used to define exonic and intronic boundaries for each gene. The first exon of each gene was excluded (to avoid the initiation peak signal) in each calculation. Reads were counted using featureCounts from the R-Subread package (v1.6.0). Exonic and intronic reads were summed and normalized by RPKM, and a ratio for each gene is calculated. These ratios were log-normalized and the median ratio calculated for each set of libraries analyzed.

Discrete Wavelet Transform

Samples with high coverage were used for this analysis. This included samples from the GRO-LIG, PRO-LIG, GRO-CIRC and PRO-TSRT libraries. The coverage over a gene transcript was normalized to 0-1 scale as show below:

$$c_i = \frac{x_i - min(x)}{max(x) - min(x)}$$

Where $x = (x_i, ..., x_n)$ represents read counts over a genomic location n, and c_i is the normalized coverage per genomic location. As we sought to identify protocol influences independent of biological gene variability, we limited our analysis to ubiquitously transcribed genes with low coefficient of variation (CV) across all samples. Thus, a total of 294 genes with a

CV less than 0.55 and average transcripts per million (TPM) greater than 150 were selected. Using the PyWavelet (version 1.0.3) API in python (version 3.6.3), the symlet 5 mother wavelet was scanned across the 294 genes, returning wavelet coefficients (approximation coefficient and detail coefficients) (Fig. 2.2E) [74, 75, 105]. After the first pass of wavelet transform, the detail coefficients were used as input for principal component analysis (PCA) using scikit-learn (version 0.20.2) [106]. So, for each gene and each sample, PC1 and PC2 values were returned. Genes were split into categories based on whether the protocols could be split on PC1 and PC2 or whether the gene could not separate the protocols in PC space. The above process was then repeated for a larger set of 669 genes (CV less than 0.85 and average TPM greater than 100). Plots were generated with matplotlib (version 3.3.4), ggplot2 (version 3.3.3) and cowplot (version 1.1.1) [107–109]. Code for the DWT analysis can be found on github (https://github.com/Dowell-Lab/Protocol-Comparisons).

Support Vector Machine

Principal component analysis values (from PC1 and PC2) derived from the wavelet transform analysis pipeline were used as input to a support vector machine (SVM). In order to verify the performance of the classification, the leave-one-out cross validation (LOOCV) criteria was used (Supplemental Fig. 2.6). A linear kernel was chosen for the SVM using the e1071 (version 1.7-4) package in R (R version 3.6.0) [110, 111]. The folds for the LOOCV were created with the caret package (version 6.0-86) in R (version 3.6.0) and accuracy for each fold and gene was calculated [112]. A total of 18 folds were created, where each of the 18 samples was held out one at a time as the test sample in the SVM, while the remaining samples were used as a training set. This was done for all the genes analysed and the evaluation determined the number of genes accurately predicting the protocol for each of the 18 samples. Plots were made using ggplot2 (version 3.3.3) and cowplot (version 1.1.1) [108, 109]. The jupyter notebook for the SVM LOOCV analysis can be found on github (https://github.com/Dowell-Lab/Protocol-Comparisons).

Pause Index Calculations

Refseq annotations were used as the basis for pause index calculations. Counts were generated either from bedtools multicov (v2.28.0). The paused region was defined as -50 bp to 250 bp from the annotated TSS [77], and the elongation region was defined as 251 bp from the TSS to the annotated PolyA site. Reads from the same strand as the annotated gene were counted for the paused and elongation region, and calculated the index as follows:

$$\text{pausing index}(pi) = \frac{ReadCount(Pausing\ Region)/L1}{ReadCount\,(Gene\ Body)\,/L2}$$

Where L1 is the length of the pausing region (300 bp) and L2 is the length of the elongation region, measured from 251 bp past the TSS to the annotated cleavage site found in RefSeq. Only pause index values from a gene's longest isoform were considered. Genes shorter than 2000 bp were removed.

The above analysis was repeated using featureCounts (v1.6.2) in the R-Subread package (v1.6.0), where the paused region was defined as -20 to +80 from the annotated TSS, and the elongation region as +81 from the TSS to -1000 from the annotated PolyA site. Genes shorter than 2000 bp were filtered out. These results are available in Supplemental Fig. 2.16.

Simulation of reads near transcription start sites

We generated 2000 base gene template with equal proportions of A, C, G, and T. Using these templates, we then simulated RNA polymerase activity similar to a previously established mathematical framework[81]. Briefly, the model assumes a position for reads to start (the transcription start site) and a polymerase distribution around the TSS determined by a normal distribution. We sampled 10,000 initiation polymerases and 5,000 elongating polymerases randomly. Each polymerase was then allowed to run-on with a random change to terminate transcription based on the sequence identity and biotin-NTP/NTP ratio specified. Transcript lengths, e.g. reads, were then determined using the difference between the TSS and the termninated location of the polymerase. To mimic Ampure bead size selection, reads were then subjected to a size selection cutoff determined by an exponential distribution proportional to their

length, resulting in an average cutoff of approximately 25 bases. The resulting read pool was subsequently used to generate metaplots of our synthetic template (Python v. 3.6.3, Numpy v.1.15.4, Pandas v. 0.23.4. Jupyter Notebook available at https://github.com/Dowell-Lab/Protocol-Comparisons).

Short Read Ratio Comparison

All reads greater than 30bp were filtered out of PRO-seq libraries to analyze the location of short reads within the genome. Each library was first assigned an Unlabeled/Labeled NTP ratio based on the run-on reaction concentrations of biotin-NTP relative to unlabeled NTPs reported by the authors for each dataset. GRO samples SRR14355674, SRR14355673, SRR14355662, SRR14355655 were included as a reference point. All PRO-seq libraries indicated in Supplemental Table 1 were considered for this analysis. Public samples SRR8033049, SRR8033050, SRR8033051, SRR8033052, SRR8033053, SRR8033054, SRR8033055, SRR8033056, SRR8033057, SRR8033058, SRR6205688, SRR6205689, SRR4041365, SRR4041366, SRR4041367, SRR4041368, SRR4041369, SRR4041370, SRR4041371, SRR4041372, SRR4041373, SRR5364303, and SRR5364304 were also included in this analysis, but were excluded from Supplemental Table 1 as they were not part of other analyses within this study.

Reads within 20 bp of the RefSeq TSS were considered to be near the TSS; we then calculated the ratio of these reads relative to all small reads found throughout the genome. The resulting ratio was plotted relative to the run-on reaction NTP ratio using R (version 3.6.3). Plots were made using ggplot2 (version 3.3.3) and cowplot (version 1.1.1) [108, 109].

Gene/Intergenic Reads Ratio Calculation

Genic and intergenic regions were determined by RefSeq (hg38, release number 109, downloaded August 14, 2019 from UCSC genome browser) annotation. Genic and intergenic read proportions were calculated by RSeQC (v3.0.0) read_distribution.py. Genic regions were defined as those overlapping a RefSeq annotation, including introns and untranslated regions. Intergenic

regions were calculated as the remainder of reads not mapping to a gene region. The reads ratio of genic and intergenic regions can be found for each sample in Supplemental Table 1.

Tfit

Tfit was used to identify regions of bidirectional transcription in each of our run-on sequencing libraries. Resultant BedGraph files from our samples were used as the input for the –bedgraph flag of the Tfit prelim module. The resultant preliminary region file was used as the –segment flag input for the Tfit model module, resulting in the final bidirectional calls used for analysis (see also https://github.com/Dowell-Lab/Tfit.git). Calls between replicates and treatments were combined using *muMerge*, generating a set of combined calls for each set of conditions (GRO-LIG, PRO-LIG, and GRO-CIRC). To compare library preparation methods, the above GRO-CIRC and GRO-LIG sets were combined together through bedtools merge (v2.28.0). Likewise, to compare enrichment methods, PRO-LIG and GRO-LIG sets were combined via bedtools merge (v2.28.0).

dREG

We used dREG to identify regions of bidirectional transcription in each of our run-on sequencing libraries. Resultant BAM files from our samples were first converted to BigWig files compatible with dREG (see https://github.com/Danko-Lab/RunOnBamToBigWig.git). Using the online dREG portal, these files were used to generate dREG calls for bidirectional regions (https://django.dreg.scigap.org). Calls between replicates and treatments were combined using *muMerge*, generating a set of combined calls for each set of conditions (GRO-LIG, PRO-LIG, and GRO-CIRC). For comparative analyses between any of these sets, each set combined by *muMerge* was concatenated and used as the input for bedtools merge (v2.28.0), generating a consensus set of regions for those two sets.

Differential Transcription Analysis

Differential transcription was performed using the DESeq2 (v1.26.0) R package (R version 3.6.3). DESeq2 no longer allows differential calls without replicates; thus, when comparing libraries where treatments and replicates were combined, the DESeq (v. 1.38.0) R package was

used instead. Gene counts were generated using featureCounts (v1.6.2) from the R Subread package (v1.6.0), counting over the entire gene body from RefSeq Annotations (release number 109, downloaded August 14, 2019 from UCSC genome browser). For featureCounts, BED6 region files were converted to SAF format with the following command: awk -F "\t" -v OFS="\t" 'print{$4, $1, $2, $3, $6}' region.bed > region.saf. Only the highest transcribed isoform of each gene was considered. Counts over Tfit, dREG, or FANTOM calls were generated with featureCounts.

GSEA

DESeq2 gene results were ranked based on -log(P-value)/sign(Fold-Change). These ranked lists were used as the input for GSEA-preranked module (v4.1.0). The Hallmark v7.4 gene sets were used as the input database. Results were generated using 1000 permutations. Gene symbols were not collapsed.

TFEA

Resulting Tfit bidirectional calls were used as the input for TFEA for each experiment (summarized in Supplemental Table 1). Calls were combined using *muMerge*. Transcription factor motifs were identified using FIMO (MEME Suite v5.1.1), using full human HOCOMOCO (version 11) motifs.

Abbreviations

RO-seq: Run-On sequencing. PRO-seq: Precision Run-On sequencing. GRO-seq: Global Run-On sequencing. CIRC: Circularization based library preparation. LIG: Ligation based library preparation. RPR: Random Priming based library preparation. TSRT: Template Switching Reverse Transcriptase based library preparation. DWT: Discrete Wavelet Transform. PCA: Principal Component Analysis. SVM: Support Vector Machine. LOOCV: Leave-One-Out Cross Validation. TSS: Transcription Start Site. eRNA: Enhancer RNA. GSEA: Gene Set Enrichment Analysis. TFEA: Transcription Factor Enrichment Analysis.

Declarations

Competing interests

Dr. Dowell is founder of Arpeggio Biosciences, the other authors declare that they have no competing interests.

Acknowledgements

We thank artist David Deen for figure composition and refinement assistance. We thank Chi Zhang and Nuria Morral for their contributions to PRO-LIG library generation. We also thank the BioFrontiers Institute Next-Gen Sequencing Core and the Biochemistry Shared Cell Culture Facility for their invaluable contributions to this study.

Author's contributions

This study was conceived by RDD, MAA and SH. Discrete wavelet transform analyses was conducted by RFS with guidance from JTS. GRO-seq libraries were generated by MAA. PRO-seq libraries were generated by SH and MAA. The scripts for *in silico* read generation and metaplot formation were written by MAA. All other analyses and initial manuscript was written by SH. All authors reviewed and revised the manuscript.

Funding

This work was funded by a National Science Foundation (NSF) ABI grant number 1759949. We acknowledge the BioFrontiers Computing Core at the University of Colorado Boulder for providing high-performance computing resources (NIH 1S10OD012300) supported by BioFrontiers' IT staff.

Availability of data and materials

The datasets used in this study are summarized in Supplemental Table 1. Datesets generated for this study are available through the Sequence Read Archive, under the accession PRJNA722106.

CHAPTER III

APPROACHES TO IDENTIFY REGIONS OF BIDIRECTIONAL TRANSCRIPTION

This chapter are adapted from:

Rutendo F. Sigauke, Margaret A. Gruca, Michael A. Gohde, Robin D. Dowell. *Annotation Agnostic Approaches to Nascent Transcription Analysis.* Methods in Enzymology (To appear)

Abstract

Nascent transcription analysis provides insight into the mechanisms of gene regulation by capturing RNA transcripts pre-splicing and maturation. However, genome annotations are curated relative to mature, stable RNA transcripts and therefore they fail to capture the transcriptional regulatory dynamics that are observed using nascent sequencing methods. These dynamics include 5′-end initiation, RNA polymerase pausing, 3′-end transcriptional run-on, and bidirectional transcription signatures indicative of RNA polymerase loading positions. Furthermore, changes in RNA polymerase activity can be leveraged to infer participation by key transcriptional regulatory proteins. In this chapter, we describe annotation agnostic tools including Fast Read Stitcher (FStitch) and Transcription fit (Tfit), which can be used to analyze run-on sequencing data in order to annotate and describe genome-wide RNA polymerase transcriptional activity. The output from these tools can be used for differential transcription analysis, using muMerge and DEseq2, as well as to provide insight into transcription regulatory dynamics, using tools such as transcription factor enrichment analysis (TFEA).

Introduction

To capture transcription dynamics across time in a broad range of cells, a number of protocols have been developed that provide information on nascent transcripts including global run-on sequencing (GRO-seq) [113], precision run-on sequencing (PRO-seq)[42], mammalian native elongating transcript sequencing (mNET-seq) [114], and chromatin run-on sequencing (ChRO-seq)[115]. These methods utilize a variety of techniques to capture transcripts as they are being produced by cellular RNA polymerases (RNAPs)[58]. By focusing on RNA production,

these assays are particularly well suited to studies of RNA polymerase activity[21, 70, 76, 96], novel non-coding RNAs[116, 117] and transcriptional regulation[7–9].

Nascent transcripts are typically captured pre-splicing and maturation, and therefore are markedly distinct from mature, stable RNA[41]. Genome annotations are curated relative to mature, processed transcripts. Consequently, typical genome annotations do not accurately reflect the region of active transcription, as cellular polymerases may initiate upstream of the annotated 5′-end start site and often continue thousands of kilobases downstream of the annotated 3′-end of a gene. Furthermore, annotations are not sufficient to capture the large number of unannotated transcripts that occur outside of genes and are often unstable, including enhancer associated RNAs (eRNAs). Thus one goal of nascent transcription data analysis is the identification of transcribed regions[56, 118, 119] and how they compare to annotation.

A large fraction of a mammalian genome is transcribed, and most of this transcription arises from RNA polymerase II (RNAP). Luckily, RNAP has a well studied activity cycle[120] that leaves identifiable patterns within nascent transcription data[8]. Most sites of RNAP initiation result in distinct bidirectional transcription signatures, which coincide with sites of transcription regulatory elements[7]. Because of the tremendous interest in these regulatory regions and how they function, a number of approaches exist for identifying them from nascent transcription data[81, 121–123]. In fact, motif co-occurrence with changes in nascent transcription between conditions can be used to infer which transcription factors are the key regulators eliciting the observed changes[8].

In this chapter, we describe our annotation agnostic toolbox for analysis of data from nascent protocols. Fast Read Stitcher (FStitch) identifies the bounds (5′ and 3′ ends) of all transcribed regions, even when they differ significantly from annotation[121]. Notably, FStitch works on any data where the desired outcome is regions of interest, including ChIP-seq[124]. Transcription fit (Tfit) uses a mathematical model of RNA polymerase II (RNAPII) to dissect active regions within nascent transcription data into individual transcripts[81], originating from sites of RNAPII loading[80]. Because sites of RNAPII loading and initiation are identified

directly from data, they can change between data sets. Consequently, we provide muMerge, a statistically principled method of generating a consensus list of regions from multiple replicates and conditions[9]. The consensus regions of transcription can then be used across conditions to infer transcription factor activity[8] using Transcription Factor Enrichment Analysis (TFEA)[9].

Materials: Data And Software Requirements

In this chapter we discuss both the FStitch and Tfit workflows: how to go from mapped read files to model output, including a brief discussion on how to leverage those outputs in downstream analysis focusing on transcription factor enrichment analysis (TFEA). For all examples, we utilize the nascent transcription data from the Allen paper where Nutlin induced p53 activity is compared to a control[61]. We describe how to run our tools primarily using Linux/Unix command line (prefaced with $) and assume the reader has access to adequate compute resources.

Software Requirements

The following is a list of software that we will be using throughout this tutorial. We refer the reader to each package's documentation for installation requirements and instructions. In addition to standard Linux/Unix tools (awk, grep) we will be using the following software:

1. Fast Read Stitcher (FStitch): rapidly identifies regions of active transcription within nascent transcription data
   (`https://github.com/Dowell-Lab/FStitch`)

2. Transcription Fit (Tfit): A tool for modeling and annotating RNA polymerase II activity
   (`https://github.com/Dowell-Lab/Tfit`)

3. muMerge: A module for merging genomic coordinates across replicates and conditions. muMerge is also part of TFEA, but can also be installed separately.
   (`https://github.com/Dowell-Lab/mumerge`)

4. Transcription Factor Enrichment Analysis (TFEA): A method for identifying enriched TF motifs relative to changes in transcription between conditions. (`https://github.com/Dowell-Lab/TFEA`)

5. BEDTools: A suite of tools used to perform coordinate math (`https://bedtools.readthedocs.io`)

6. SAMtools: Utilities for the Sequence Alignment/Map (SAM) format (`http://www.htslib.org/doc/samtools.html`)

7. preseq: A package used to calculate sample complexity and estimate future yields if the sample were sequenced to increased depth, available in R or for command line usage (`http://smithlabresearch.org/software/preseq/`)

8. BBMap Suite: A suite of various bioinformatics tools including utilities such as trimming, mapping, and post-mapping quality control (QC) (`https://github.com/BioInfoTools/BBMap`)

9. Integrative Genomics Browser (IGV): A genome browser application developed by The Broad Institute which includes a suite of visualization utilities. (`http://software.broadinstitute.org/software/igv/`)

Some analysis software is computationally intensive – requiring cluster compute resources – whereas other steps are rather lightweight in compute needs and can be run on a typical laptop. To guide the reader, we will note when an analysis step is CPU or memory intensive. At the time of this writing, the commands presented in this chapter utilized the following versions of these software: bedtools v2.30, samtools v1.12 (using htslib 1.12), BBMap v38.93, preseq v3.1.2, IGV v2.11.1, and python 3.9.5 on a Thinkpad Intel iCore i7 1.90 GHz running Fedora Linux v34. All compute-intensive processes were run on a compute cluster (referred to as Fiji in this chapter) running CentOS Linux v7 with software versions: bedtools v2.28.0, samtools v1.8, BBMap v38.05, preseq v2.0.3, IGV v2.4.10 and python v3.6.3. Our in house software (FStitch, Tfit,

muMerge, and TFEA) were all the latest repository versions (October 2021). In general, all of the above software is constantly changing and improving. Therefore, we encourage the user to refer to each package for any updates and/or bug fixes that may have transpired since this chapter was written.

Pre-Analysis: Quality Control

The goal of nascent protocols is data that accurately reflects the distribution of actively transcribing cellular RNA polymerases genome wide[41, 58, 70]. However, analysis results are strongly influenced by the quality of the data obtained from the sequencer. Consequently we recommend assessing the quality of the input data before proceeding beyond sequence alignment. There are a number of excellent software packages for assessing data quality on short read sequencing data[125], including the recently developed nascent RNA sequencing specific PEPPRO[73]. In the following section, we describe only a minimal quality control analysis and provide recommendations for the quality of data needed for best results.

Unfortunately, loss of data quality strongly influences the patterns observed within nascent data. For example, both the distance between regions (whether or not reads overlap) and the read density over these regions (level of coverage) vary based on read depth (total reads sequenced) and library complexity (number of unique reads in sample). Therefore we recommend accessing both depth and complexity prior to proceeding with any post-mapping analysis. To this end, we describe here two tools: BBMap's pileup.sh [126] and preseq [127], which calculate sample coverage and complexity, respectively.

After read mapping, the first step is to create a mapped read file in binary format (BAM), sorted and indexed using SAMtools[128] as follows:

```
$ samtools view -S -b -o SRR.unsorted.bam SRR.sam

$ samtools sort SRR.unsorted.bam SRR.sorted.bam
```

```
$ samtools index SRR.sorted.bam SRR.sorted.bam.bai
```

Once the BAM file has been generated, sample coverage can be assessed using BBMap's pileup.sh with the following arguments:

```
$ pileup.sh in=SRR.sorted.bam out=SRR.coverage_stats.txt
```

We note that pileup.sh is a Java program that requires sufficient memory , related to the size of your genome and data file. For example, for our human dataset, we utilized the '-Xmx6G' option to provide 6 Gb of heap memory. With sufficient memory, pileup.sh runs in a few minutes on our laptop.

The output file (SRR.coverage_stats.txt) is a statistics file containing key information on the distribution and depth of reads, including average fold coverage (read density per chromosome length), chromosome length, total percent covered, the GC percentage of mapped reads, and plus/minus strand read coverage. Statistics are reported on a per chromosome basis. While all of these are valuable sample quality metrics, of particular interest for nascent data is chromosome coverage. Generally, higher coverage is preferred, as a large fraction (> 20%, often around 40%) of the genome is actively transcribed. Lower coverage suggests insufficient sequencing depth or low quality data. We argue that a minimum average coverage >3% (omitting mitochondrial reads) is necessary for any subsequent analysis. Coverage can easily be visualized graphically, as well as in a genome browser (Figure 3.1).

Increasing sequencing depth can sometimes improve overall genome coverage, but not always. To assess whether further sequencing a sample with low coverage will yield more unique reads, e.g. increased sample complexity, we also recommend running preseq to calculate both current sample complexity and future yields if the sample were to be sequenced to a greater depth. Preseq was designed to run on BED files but, if compiled with HTlib support, can also analyze BAM files (the -B option). Requiring only a few minutes per sample, preseq can be run as follows:

```
$ preseq c_curve -B -o SRR_c_curve.txt SRR.sorted.bam

$ preseq lc_extrap -B -o SRR.lc_extrap.txt SRR.sorted.bam
```

The c_curve module calculates the current sample complexity and lc_extrap calculates predicted future yield, if further sequencing were performed. The predicted number of unique reads can be plotted against the total number of reads to visualize sample complexity (Figure 3.1B). It is recommended that at least 10% of 50M sequenced reads are unique for running both FStitch and Tfit. Sequencing to a greater depth such that at least 15% of 100M reads are unique in a given sample will provide the best results from both algorithms.

Data Husbandry: Formatting the coverage file

After quality assessment, the next step in analysis is to generate the expected input files for subsequent analysis software. Downstream analysis programs can sometimes be mislead by low quality reads as well as multi-mapped reads. Therefore we recommend removing these using the following:

```
$ samtools view -h -q 1 SRR.sorted.bam | \
grep -P '(NH:i:1|^@)' | \
samtools view -h -b > \
SRR.mmfilt.sorted.bam
```

where we first use samtools view to filter all reads with low quality scores (-q 1). We then search, using grep (a built in Unix command) with a Perl regular expression to select lines with either uniquely mapped reads (lines with the NH:i:1 tag) or the SAM header section (lines that start with @ symbol). Finally, samtools view converts the resulting intermediate into a BAM formatted output.

A common input file format, used by both FStitch and Tfit, is the bedGraph. A bedGraph file is tab-delimited binned representation of the data, e.g. a histogram. It is therefore a more

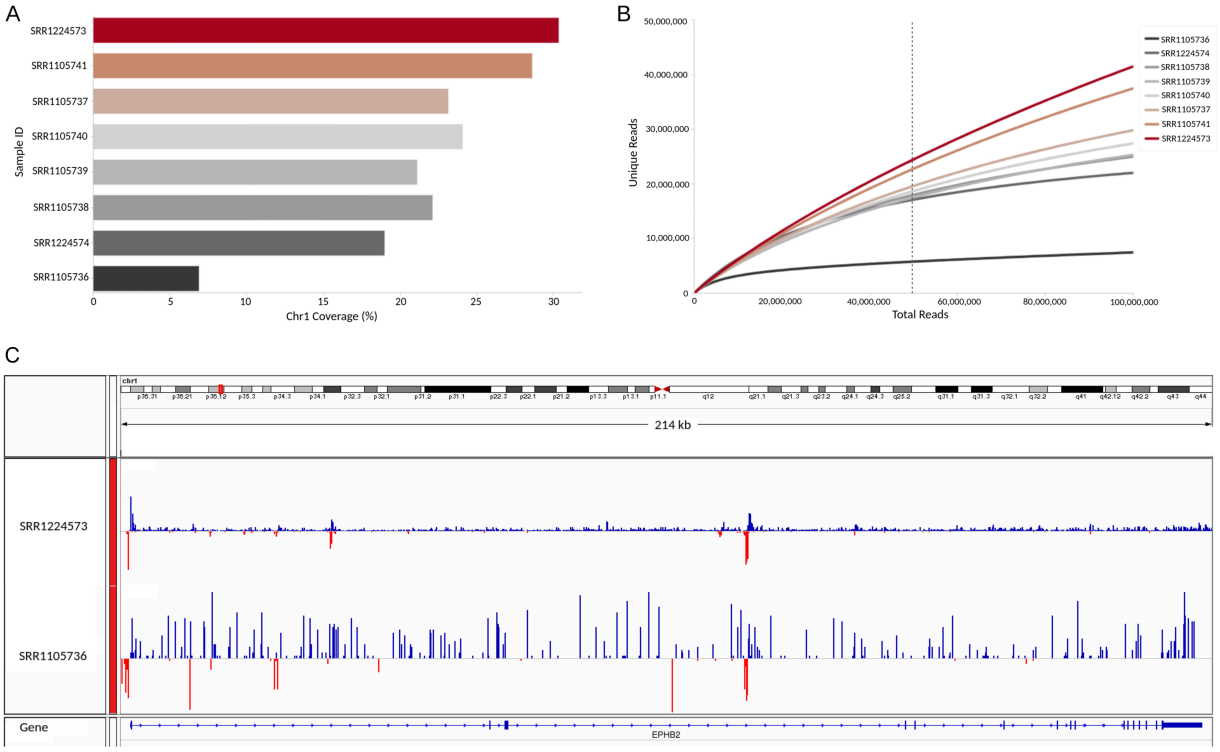


Figure 3.1: Depth and complexity as quality measures. (A) Fraction of chromosome 1 (as percentage) covered in multiple samples[61], graphed using BBMap's pileup.sh output. (B) Comparative sample complexity curves generated from the preseq lc_extrap module, where the y-axis represents the predicted number of unique reads at corresponding sampling depth (total reads) on x-axis. We recommend a 50M read sampling depth with at least 10% unique reads (dashed line). (C) Samples SRR1105736 and SRR1224573 have 6.4% and 30.9% coverage over chromosome 1 and 5.8M and 24.5M unique reads per 50M reads, respectively. SRR1105736 has noticeably lower coverage and complexity relative to the other samples in this experiment (hg38; group auto-scaled; chr1:22,708,838-22,923,500; y-axis[-0.405 - 0.49]; blue: positive strand reads, red: negative strand reads).

compact file format than SAM/BAM files. Both programs specifically require a bedGraph that is a non-normalized, high fidelity representation of the data, without zeros. In both cases, we assume the depth is signed, with negative numbers indicating the negative strand data. While both the deepTools [129] and the BEDTools [130] suite have options for generating a bedGraph, deepTools by default smooths the data through generation of discrete bin sizes and includes regions of zero coverage and is therefore not recommended. As such, we recommend using the BEDTools genomecov tool for generating bedGraphs using the following command:

```
$ bedtools genomecov -ibam SRR.sorted.bam  \
-bg -strand + \
> SRR.pos.bedgraph
```

The argument -ibam specifies the input BAM file, -bg specifies the output bedGraph file, and -strand specifies the strand for summarizing data, in this case positive. The command should be repeated for the negative strand as follows:

```
$ bedtools genomecov -ibam SRR.sorted.bam \
-bg -strand - \
> SRR.neg.bedGraph
```

To obtain the required signed output on negative strand reads, we negate the fourth column (coverage values) of the negative strand bedGraph using the Linux/Unix tool awk:

```
$ awk 'BEGIN {FS = OFS = "\t"} \
{$4 = -$4} {print}' SRR.neg.bedGraph > \
SRR.neg.formatted.bedGraph
```

Both FStitch and Tfit have arguments to provide separate positive and negative strand files, so at this point the necessary processing is complete for the bedGraph files. However, for visualization in Integrative Genomics Viewer (IGV) [131] we recommend that the two stranded data files be concatenated into one bedGraph containing reads on both the positive and negative strands as follows:

```
$ grep -v '^@' SRR.neg.formatted.bedGraph | \
cat - SRR.pos.bedGraph > SRR.cat.bedGraph

$ bedtools sort SRR.cat.bedGraph > SRR.sorted.bedGraph
```

which removes any header present on the negative strand bedGraph, concatenates the positive and negative strand coverage information, and then sorts the bedGraph for optimal downstream processing. Importantly, bedtools sort is both the most time and memory intensive step in the process of creating the required bedGraph files. The Unix sort command:

```
$ sort -k 1,1 -k2,2n SRR.cat.bedGraph > SRR.sorted.bedGraph
```

is an alternative to bedtools sort that uses more CPUs and less memory. This concatenated, sorted file is also accepted as input by both FStitch and Tfit in place of the individual strand files.

Methods

Both Tfit and FStitch are stand-alone applications that take as input a coverage file (in bedGraph format) and output annotations of nascent transcription data. FStitch outputs regions of active and inactive transcription on a per strand basis whereas Tfit outputs the locations of polymerase loading as well as key characteristics (model parameters) for each unique loading position. Both FStitch and Tfit produce information in an annotation agnostic fashion, relying on the data only to identify desired features. Both data quality and biological variability influence the results obtained from these tools. Consequently, the precise bounds of these regions may vary. Yet assessing changes between samples requires consistent coordinates for regions of interest. Thus, we developed muMerge as a method of combining calls across datasets for subsequent analysis with tools such as DEseq2. Regions of interest can also be examined by TFEA to identify which transcription factor motifs are enriched adjacent to changes in transcription. Here we focus on using TFEA on sites of RNA polymerase initiation, most of which are also sites of bidirectional transcription. In the sections that follow, we describe the motivations for each algorithm and outline the steps necessary for data analysis.

Using FStitch: Identifying expanse of transcription

Nascent transcription reflects the position and levels of all cellular RNA polymerases. Therefore the simplest first question to ask is, “which regions are transcribed?” Unfortunately, genome annotation is insufficient to describe which regions of the genome are transcribed. First, RNA polymerase loads at many unannotated regions genome wide, producing short, unstable,

often bidirectional sites of low transcription. The same bidirectional signal is seen at most genes, consistent with a uniform model of RNA polymerase II activity[21]. Furthermore, at genes transcription extends beyond the annotated cleavage site. In fact, in some conditions the extent of 3′-end transcriptional run-on is extensive[132]. The goal of FStitch is to identify regions of transcription directly from the data.

FStitch is comprised of two modules: train and segment. Fundamentally, the core algorithm of FStitch is a two state Hidden Markov model (HMM) that aims to distinguish between "active" and "inactive" transcription regions [56, 121]. Because the characteristics of "active" regions are influenced by the underlying protocol, sequencing depth, and library complexity, FStitch utilizes a user defined training file to learn the key characteristics of these regions. The user defined "active" regions should show typical characteristics of active transcription: read dense, high-coverage contigs that span a minimum of several hundred base pairs. Once trained, FStitch can then be utilized to segment (e.g. label) each strand's data into "active" and "inactive" regions. These labeled regions can be compared to genome annotations to assess whether the full extent of a gene is transcribed[121] and the extent of 3′ run-on observed[56]. Additionally, FStitch identified regions can be used as a pre-filter for subsequent Tfit analysis (described in Section)[81]. In the original FStitch paper[56], regions of overlap at the 5′ end on opposite strands were used an a estimation for regions of bidirectional transcription[56]. However, as FStitch is not guaranteed to identify individual transcripts in transcription dense regions, we recommend identifying bidirectionals using methods specifically aimed at identifying this signal such as Tfit (described below) or dREG[122].

FStitch Train Module

The FStitch train module requires as input two files: the data (in sorted bedGraph format) and the training file (in BED4 format). In the training file, each row is a tab-delimited instance of a single region: chromosome, start, end, and status. The status column is an indicator (0/1) as to whether the region in question is inactive or active, respectively. The quality of the final FStitch model is strongly influenced by the number of regions and the accuracy of their labeling within

the training data, consequently we will describe multiple methods of creating a training file. For best results, a custom training file should be created to capture the unique characteristics of the specific dataset.

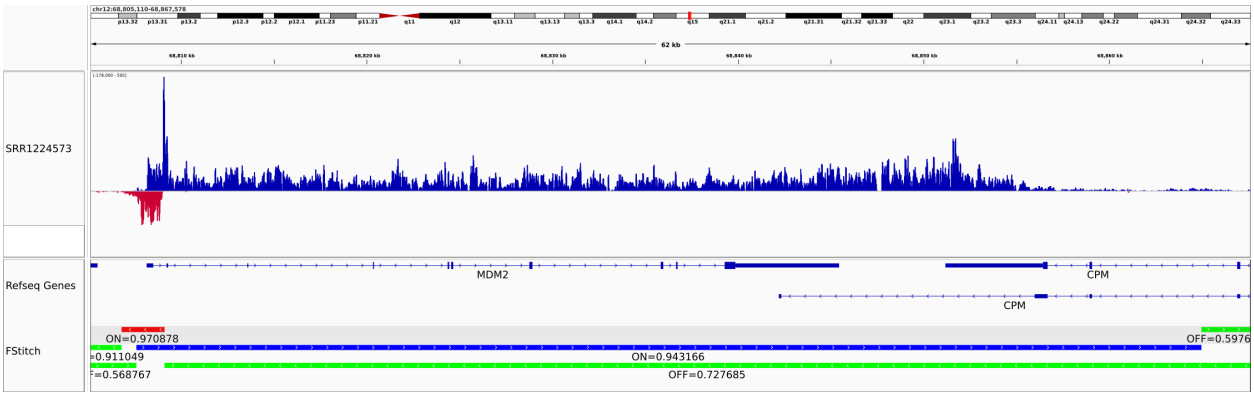


Figure 3.2: Using FStitch to identify expanse of transcription. FStitch segment can be used to capture data driven full gene-level transcription annotation. Note that RNAP transcription continues past the annotated 3′ end of MDM2 (hg38; chr12:68,806,508-68,855,728; y-axis [-1.172 - 3.58]; blue: positive strand reads, red: negative strand reads), in this case overlapping the 3′ end of the CPM gene. FStitch output show as blue boxes for positive strand transcribed region (labeled "ON"), red boxes for negative strand transcribed region (labeled "ON"), green boxes for regions without transcription (labeled "OFF") on either stand. Each labeled region has an associated probability score.

Pre-configured training file The simplest approach to training is to utilize the pre-configured training file provided within the train directory of the FStitch distribution. The pre-configured training file, available for the human genome version hg38 and mouse version mm10, contains twenty ubiquitously expressed genes [133] and twenty intergenic regions on the positive (+) strand. In our experience, these regions provide a reasonable initial training for most high quality (e.g. sufficiently complex and sequenced to appropriate depth) data sets. However, custom training data (described below) typically improves FStitch performance.

When using the pre-configured training data, it is recommended that the user first check that the default regions have adequate coverage in their specific dataset using BEDTools multicov:

```
$ bedtools multicov -bams [SAMPLE BAMS] \
-bed hg38_annotations.bed \
> sampleCoverage.bed
```

Note that here [SAMPLE BAMS] refers to the list of file names of your datasets, as BAM files. Regions of zero read coverage should be removed, but care should be taken not to remove too many regions as this dramatically reduces training effectiveness. We recommend that the training data, after customization, always contain a minimum of 15 inactive (e.g. labeled 0) and 15 active (e.g. labeled 1), for a total of at least 30 regions. In our experience, the pre-configured training data is effective when all active regions have coverage and the overall read depth over active and inactive regions is over 10:1 after normalizing for bin size ($total\ reads/(end - start)$). If the pre-configured training file fails these standards, the sample should be assessed for quality (see Section), as the data may have insufficient sample depth or complexity. Alternatively, a custom training file must be constructed.

Custom training file Generating good training data from scratch requires a certain degree of trial and error, however there are a few key points to keep in mind that will expedite the process. First, the BED4 format of training data does not contain strand information. Therefore all regions within the list should originate from the same strand of data (either positive or negative). We recommend naming the training file in a manner that indicates the strand to which it corresponds. Second, regions of zero coverage are not terribly informative for either label. Most inactive regions will contain some noise, therefore it is better for the training data to reflect this expectation. As a consequence, we advise not to pick regions with zero coverage. Third, picking regions that are too small (rule of thumb, roughly <1 kilobase) does not accurately reflect that many transcribed regions (e.g. annotated genes) are long. Consequently, we recommend picking a mixture of regions between 1 and 200 kilobases to reflect the diversity in active transcription lengths typical in a mammalian genome. Lastly, do not stress over the precise boundaries of training regions. In other words, the start location that is annotated as active does not need to be at the precise base that signal began. With these recommendations in mind, the training data file can be generated from scratch or built by augmenting the provided pre-configured training set. We will describe both methods and provide recommendations regarding its construction.

Creating a custom training dataset from scratch requires manually identifying regions of both active and inactive signal within the data. For this, we recommend utilizing the Broad's Integrative Genomics Viewer (IGV). While you can import mapped read files (typically BAM or bedGraph files) directly into IGV, numerous large files can decrease IGV's performance. As such, we recommend that the user convert the bedGraph file to TDF format, which is a binary form of the bedGraph tailored for faster access. To convert the bedGraph to a TDF, utilize IGV tools with the following command:

```
$ igvtools toTDF SRR.cat.bedGraph SRR.cat.tdf genome.chrom.sizes
```

where the file genome.chrom.sizes is a text file containing chromosome size information that corresponds to the genome to which your samples were mapped, obtained either from UCSC or provided with IGV tools. Alternatively, you may convert the bedGraph within the IGV browser by selecting:

```
Tools $\rightarrow$ Run igvtools ...
```

from the top drop-down menu and specifying the same minimum arguments used in the command above.

Once the samples have been converted to TDF format and loaded into IGV, it is best practice to begin by looking at annotated genes that are highly expressed in the data of interest to become familiar with the typical read distribution patterns of active regions compared to inactive regions. It is recommended that the user select minimally 20 inactive regions and 20 active regions. Generally speaking, annotating more regions improves FStitch training, however the regions must be representative of the diverse characteristics (length, depth) expected in active and inactive regions.

While the user can build the required BED4 file manually, it is also possible to take advantage of IGV's built-in capacity for collecting a table of regions. The coordinates for the current field of view within IGV can be added to an ongoing list using the top down menu:

```
Regions $\rightarrow$ Region Navigator ...
```

which will open a table with the necessary four columns: chromosome, start, end, and description. By clicking the "Add" button at the top, the current region shown will be added (chromosome, start and end) with the "Description" column remaining empty. In this description column, the user must add the status of the region, either a 0 or 1 for inactive or active transcription, respectively. Once multiple regions have been added to the table, the set of annotations can be exported by selecting:

```
Regions $\rightarrow$ Export Regions ...
```

and choosing both where to save the training file and what to name it (it must end in .bed). The file will be saved in the required BED4 format, so no further editing is required before running FStitch train. Likewise, the pre-configured training regions file can be imported into IGV by going to the top drop-down menu in the program and selecting:

```
Regions $\rightarrow$ Import Regions ...
```

Regions can then be added, edited, or removed from the list to tailor the training file to your specific data using the Region Navigator, as described previously.

Training the Model Using the data (as bedGraph) and training file (as BED4), the train module can be invoked using the following minimum arguments:

```
$ FStitch train --bedgraph SRR.cat.bedGraph \
--strand + --train hg38_train.pos.bed \
--output PROJECTNAME.hmminfo
```

where, in this case, the model is trained on the positive (+) strand of data. Alternatively the model may be trained on the negative (-) strand, as is appropriate for the annotations provided by the training file. Multi-threading is also available using the -n/–threads argument, however is not typically necessary as the train module typically takes less than five minutes on a single core.

The output file, which must have the .hmminfo extension for use in the FStitch segment module, contains information relevant to the effectiveness of training. The header of the output file contains information pertinent to its creation including the configuration of the model, command line input, data and time the file was generated. The first line below the header indicates whether training converged and will display 'True' if the run was successful and 'False' if it was not. While it is rare for the model not to converge, it can occur if there is not enough training data or if the input regions are inconsistent (active regions resemble inactive regions too closely). The other relevant value for assessing the training is the line marked 'HMM Transition Parameters'. If your first and last values are equal to 1, this indicates that the training has converged to a single state and subsequent segmentation will fail. If this occurs, check that the training file follows the outlined requirements or include more regions in your training.

FStitch learns from the training data what are the typical statistical characteristics of active and inactive regions. Consequently, the quality of subsequent segmentation (labeling; discussed in the next section) is highly sensitive to the specifics of the training data. When considering a collection of experiments, questions arise concerning how best to train and segment across the set. If the samples within the set have similar coverage and complexity, then we encourage using the same modeling parameters (hmminfo file) to segment all samples, particularly when the plan is to subsequently compare transcription levels across samples. However, if the samples have very different complexities and coverage, FStitch may not be able to resolve the samples using a single trained model. In this scenario, one can train each sample individually but care must be taken to ensure that any downstream results do not arise simply from the sample specific training. Consequently, it is often best in this scenario to either sequence the lower coverage sample to a greater depth (if greater complexity can be achieved), discard the sample from the analysis, or obtain a new, better quality sample.

FStitch Segment Module

The FStitch segment module uses the output parameter file obtained from the train module to annotate (a.k.a. label) regions of active and inactive transcription across an entire dataset. The minimum arguments necessary to run the segment module are as follows:

```
$ FStitch segment --bedgraph SRR.cat.bedGraph \
--strand (+/-) --params PROJECTNAME.hmminfo \
--output SRR.fstitch.{pos,neg}.bed
```

Each strand must be segmented separately, however the outputs can be concatenated such that it appears as one track in the genome browser as follows:

```
$ cat SRR.fstitch.pos.bed SRR.fstitch.neg.bed |\
sortBed > SRR.cat.fstitch.bed
```

If these results are imported into IGV, be aware that you will need to right-click on the track and select 'Expanded' to view the full annotations for both the positive and negative strands (Figure 3.2). Notice that each strand is labeled with "ON" and "OFF" segments corresponding to active and inactive regions of transcription, respectively. The quality of the segmentation is highly sensitive to the training process. Thus is not uncommon for the training and segmentation steps to be done repeatedly in order to refine the training process.

Using Tfit: Inferring polymerase activity

Most sites of transcription are the result of the activity of RNA polymerase II (RNAP). The activity of RNAP is regulated at a number of steps within the well-characterized transcription cycle[120]. As several steps of the transcription cycle give rise to RNA, these steps leave distinct shapes within nascent transcription data. Tfit is a probabilistic, generative mixture model of RNAPII behavior that leverage the patterns within nascent transcription data to annotate and characterize RNAP activity throughout the entire genome [81]. Because Tfit seeks to capture and

model specific data distributions, data quality can significantly impact its ability to model regions of RNAP activity.

The output of Tfit provides a quantification on RNA polymerase II behavior within a particular dataset. RNA polymerase II loads at a number of locations genome wide and cycles through three distinct phases: initiation, elongation, and termination [134]. Each Tfit output is a single fit to a mathematical description of RNA polymerase II activity (the model) and has a number of model parameters associated (see Figure 3.3A-C). Tfit outputs can be leveraged in downstream analyses including differential transcription analysis, changes in RNAPII behavior (e.g. loading, pausing and elongation), examination of motif displacements relative to polymerase loading, evaluation of pausing ratios and investigations into transcription factor activity analysis (see Section , Figure 3.4).

Practically, the process of determining the number of loading locations is computationally expensive. Consequently, the first step in using Tfit is to identify regions of interest that are small (for compute efficiency) but cohesive (e.g. don't break up signal from a single transcript). There are two primary annotation agnostic options available for region identification (pre-filtering) prior to running Tfit: 1) FStitch and 2) template matching (i.e. the Tfit bidir module).

In principle, FStitch can be trained to behave as a rigorous pre-filter to Tfit. When utilized in this fashion, it is the recommended approach to eliminating non-transcribed and noisy regions of the genome from Tfit's consideration. Once the regions of interest have been generated, the user can run the Tfit model module to produce a set of annotated RNAP model fits for each region. The Tfit model is highly tunable, and advanced users can modify the configuration file to adjust the behavior of Tfit and its EM algorithm in order to obtain the full RNA polymerase model fits[81]. In practice, this can be extremely computationally intensive process.

Alternatively, the most common use of Tfit is identifying sites of bidirectional transcription. For this purpose, the Tfit template matching pre-filter, provided in the Tfit distribution, is a rapid method of identifying regions of interest. In this scenario, the Tfit model is subsequently run with the default configuration file provided with the Tfit distribution, provided

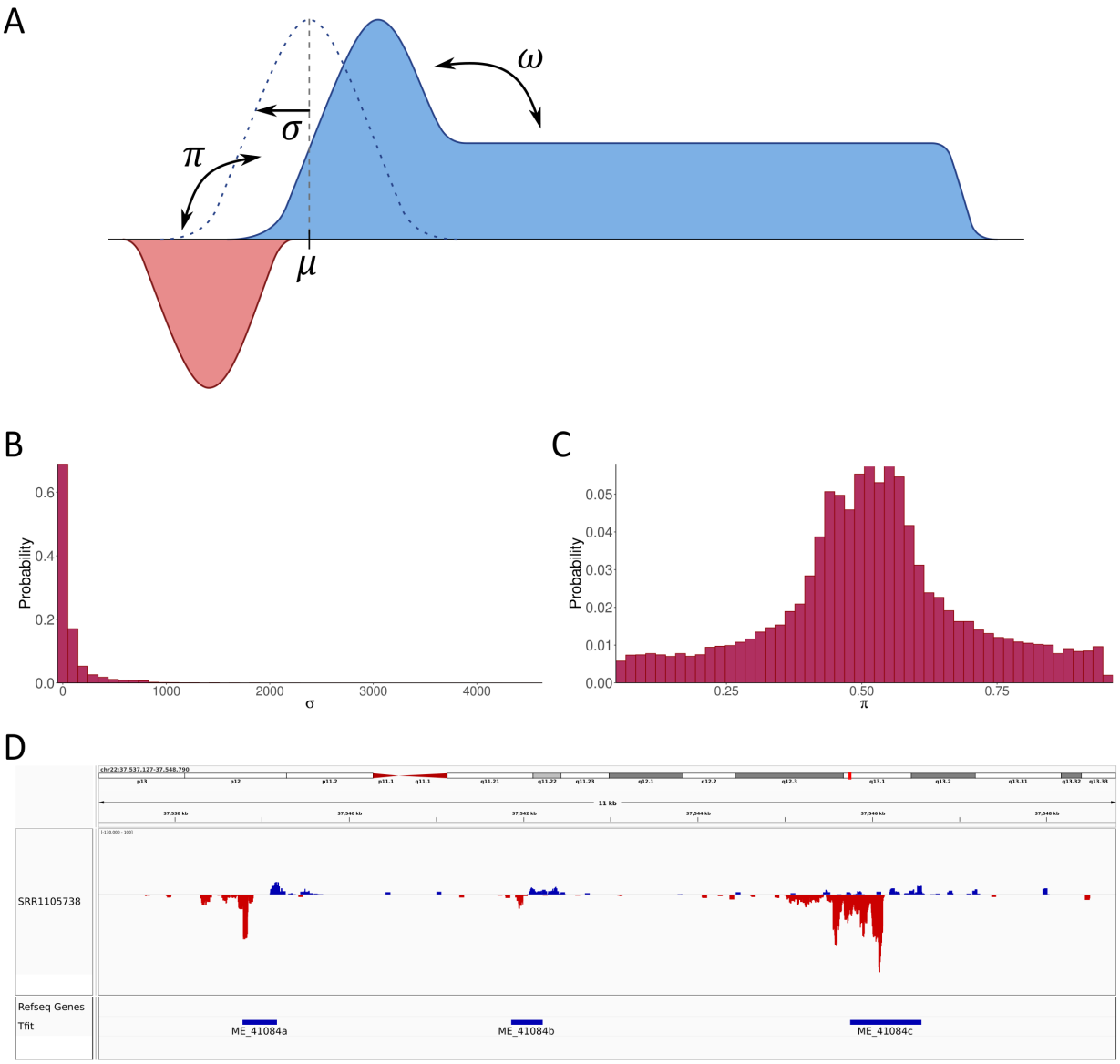


Figure 3.3: The Tfit model identifies sites of bidirectional transcription. (A) Cartoon representation of the Tfit model parameters highlighting the position of polymerase loading (μ), the variance on loading (σ), the strand bias (π) and pausing probability (ω) parameters. See [81] for a full description of the model. Histograms representing variance in RNAP (B) loading position (σ) and (C) the strand bias (π) for SRR1105738. (D) An example of Tfit output on the same sample (hg38; chr22:37537127-375548790; y-axis [-130 - 100]; blue: positive strand reads, red: negative strand reads). This scenario used the bidir preliminary filter and the model was run on our laptop configuration.

for this purpose. The regions output by the default Tfit model correspond to the inferred loading and initiation region of bidirectional, having length 2*(σ + λ) centered at the inferred location of RNA polymerase II loading (μ).

In either scenario, Tfit is not guaranteed to return model fits for all input regions. Regions are dropped if the model fails to converge or the fits are too poor. These scenarios may arise from poor quality data in the region, tight restrictions on the EM algorithm (as specified in the configuration file), or potentially from biology – as the Tfit model reflects only RNA polymerase II activity whereas most nascent assays capture RNAs from all cellular polymerases. Additionally, Tfit does not currently contain a model of termination, therefore we recommend utilizing FStitch to identify the 3′ end of transcribed genes. By intersecting FStitch called regions of transcription with annotations, the extent to which an elongation region extends beyond the annotated cleavage site can be readily determined.

Finding preliminary regions of interest

For efficiency, the data should be pre-processed to identify regions of interest containing one or more RNA polymerase loading locations.

Annotated genes The simplest method of preprocessing the genome is to focus exclusively on promoters at annotated genes. In this scenario, the coordinates should be padded to account for un-annotated upstream antisense RNAs (e.g. the bidirectional nature of initiation regions). However, most sites of RNAP loading and initiation are not at annotated genes. Therefore it is preferred that the data be used to identify regions of interest, either from FStitch (for all transcribed regions) or using the Tfit template matching pre-filter (if focusing on bidirectionals).

FStitch FStitch can be used as a rigorous pre-filter to identify transcribed regions in a data driven fashion within nascent transcription data. As FStitch is sensitive to its training data, care must be taken to train FStitch in a manner that identifies longer, contiguous regions of one or more bidirectionals. All FStitch labeled "ON" regions should be provided as input to Tfit, created typically by merging the positive and negative strand segmentation outputs. This is the preferred

approach for advanced users interested predominantly in the full RNA polymerase II mathematical model, where customization of the Tfit model configuration is required. We note that often transcription dense regions, such as super enhancers, cannot be parsed into individual bidirectional regions by FStitch, but can be modeled and annotated as discrete RNAP loading events (e.g. broken into distinct sub units) using Tfit. At the other extreme, poor training, low quality data, or shallow sequencing can cause FStitch to break individual transcripts into distinct regions. Appropriate use of bedtools merge with padding (the -d option) can combine adjacent regions and sometimes overcome this issue.

Template matching When the focus is predominantly on sites of RNA polymerase initiation, most of which are sites of bidirectional transcription, an alternative pre-filter approach known as template matching is recommended. The remainder of this chapter will focus on this application of Tfit. Encoded directly into the Tfit package, the bidir module scans the genome to identify regions that loosely match the expected 5′ model.

Importantly, Tfit is computationally expensive. We recommend that multiple processors be used to reduce overall runtime. Tfit uses the message passing interface (MPI) framework for taking advantage of multiple threads, processors, or nodes (for simplicity, we will refer collectively to these as number of processors, np). Check your specific machine's architecture and MPI implementation for details on how to run Tfit with np > 1. While we recommend using multiple processors (we typically use np = 16 or 32), we describe here the minimum arguments (e.g. np = 1) needed to run the bidir module:

```
$ Tfit bidir -config config_file.txt \
-ij SRR.cat.bedGraph -N SRR -o outdir
```

which takes as input a bedGraph file (SRR.cat.bedGraph in this example) and a configuration file (called config_file.txt). The Tfit github repository provides a default configuration file that is typically well suited to finding sites of RNA polymerase loading and initiation. The -N flag is the

prefix name to assign to the output files (e.g. SRR). The above command returns two output files: preliminary regions of interest (SRR_prelim_bidir_hits.bed) and a log file. The regions file is a BED4 file (chr, start, stop, id) that can be subsequently used for fitting the full Tfit model. The log file includes run information (input file name and parameters used) and the total number of predicted preliminary regions. See the Tfit Github page for more detailed documentation.

Tfit Model Module

Armed with regions of interest (ROI), we subsequently use the Tfit model in order to fit zero or more instances of the RNAP model in each region. The model will attempt to find the best set of parameters for μ (inferred position of polymerase loading), σ (variance in the loading position), π (strand bias), λ (processivity of initial loaded polymerase), and ω (pausing probability, e.g. fraction of bidirectional signal to elongation/noise signal). The model can be run using the following commands:

```
$ Tfit model –config config_file.txt \
–ij SRR.cat.bedGraph –k SRR_prelim_bidir_hits.bed \
–N SRR –o outdir
```

which takes as input a coverage file (SRR.cat.bedGraph), the regions of interest (SRR_prelim_bidir_hits.bed, see Section), and returns as output a BED4 file containing annotated bidirectionals of the model fit (SRR_bidir_predictions.bed). Using the default configuration file, each output region is an individual bidirectional centered at the best estimate for μ and whose width is 2*(σ + λ), reflecting the loading zone[80]. These regions can be readily viewed in IGV (Figure 3.3A).

For advanced users, the full model parameter estimates are also written to a separate text file (SRR_K_models_MLE.tsv) which gives a detailed account for every possible number of components fit to each region of interest. Using the default configuration file, we evaluate 1 to 10 models for each region of interest. The parameters for each of these fits is given in condensed form in the K_models file. Importantly, the bidir_hits bed file corresponds to only the best case

scenario (i.e. the selected optimal number of components). Full details of the input and output files are included in the Tfit GitHub documentation.

Tfit model is compute intensive. While individual regions can be run easily on our laptop configuration (see Figure D), whole genome datasets and collections of data are preferentially run on compute cluster resources. For example, most of the datasets used here took roughly 20 hours on 32 threads of Fiji.

Differential Transcription Analysis with muMerge.

There have been numerous methods developed for assessing differential expression of mature RNA from read count data [135, 136]. However, nascent transcription has unique properties relative to steady state mature messenger RNA. Notably, nascent transcripts are pre-splicing, have a distinct RNA polymerase initiation peaks, and terminate far beyond the canonical cleavage site[121]. When assessing gene level differential transcription from nascent data, the most popular approach uses fixed windows to ignore the 5′ initiation peak based on annotated gene coordinates, and our experience suggests this approach is the most consistent across nascent protocols[58]. Given the desired regions, read counts can be gathered and differential transcription assessed using the program of your choice such as DESeq/DESeq2 or edgeR [88, 137, 138].

In some scenarios, however, it may be of interest to also assess patterns of change for eRNAs or the full region of gene transcription, which often extends well beyond the annotated cleavage site. As described above, Tfit and FStitch can be utilized to delineate these distinct regions. Another excellent option for identifying transcribed regulatory regions is dREG[7, 81, 121].

Importantly, the regions output from from these tools are derived directly from the data and therefore will not necessarily be identical across sample replicates and conditions (see Figure 3.4A). Consequently, it is necessary to first identify the consensus regions of interest (ROI) that most accurately reflect the calls across replicates and conditions. It is on these consensus ROI that differential signal is assessed, for example using DEseq2. For identifying consensus ROI, we

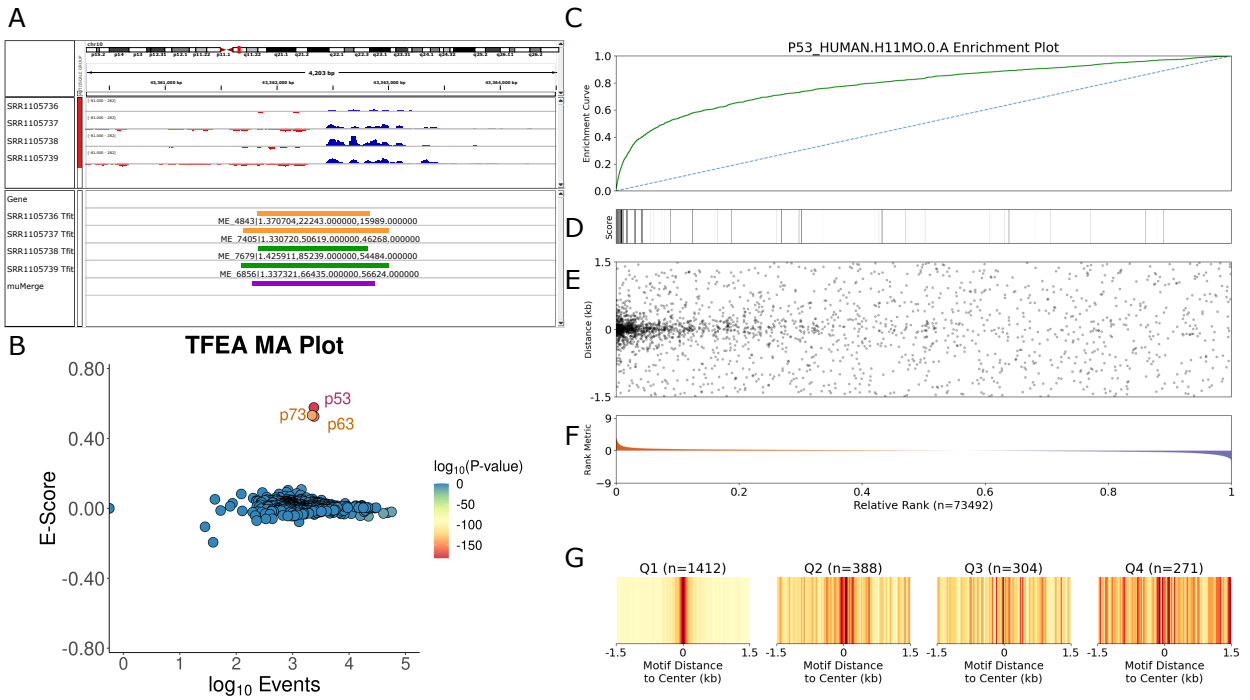


Figure 3.4: Output examples from muMerge and TFEA. (A) Example region highlighting individual Tfit model calls for four samples across two conditions DMSO (orange) and Nultin (green). The bottom track (purple) shows the final muMerge consensus region derived from each of the samples Tfit calls. (B) Enrichment scores (E-scores) for 401 TFs from the HOCOMOCO database v11 [139]. Each point represents a single TF and the points are colored based on the log_{10}(adjusted P-values). The x-axis shows the number of motif hits for a given TF and the y-axis shows the E-scores. (C) Enrichment plot with the running sum (green) for all ROIs (x-axis). The E-score is the area under the curve. (D) Scores for all ROIs represented as a heatmap. The darker the line, the greater the score. (E) A scatter plot where each point is a p53 TF motif instance within the ROI, and the distances (y-axis) are relative to the center of each ROI. (F) The ROI are ranked based on differential transcription signal. Up-regulated regions are shown in red and down-regulated regions are blue. (G) Heatmaps depicting the motif displacement distribution of each quartile along the ranked ordered list of regions of interest. Here all samples and TFEA were run on our compute cluster (Fiji).

recommend leveraging muMerge, a statistically principled way of combining regions[9]. Given a set of samples, muMerge treats the regions as probability distributions reflecting confidence in the position of RNA polymerase loading and initiation (e.g. μ). The regions are combined across samples taking into account the replicate and the condition information to produce a joint probability distribution that highlights the most likely consensus region. muMerge can combine and split regions as necessary to maximize the informative nature of the replicates and conditions. In this manner, muMerge ensures that regions are not counted twice and potential bidirectional regions are captured within the dataset (see Figure 3.4A).

The basic command used to run muMerge is:

```
$ python mumerge.py -i file_with_sample_info.txt -o SRR
```

The file_with_sample_info.txt is a tab-delimited file that, after the first line, contains three columns: paths to files with regions of interest in BED file format, the sample identifier, and the group name sample information for each of the samples and the -o is the prefix assigned to the output files. Notably, the first line is required to be "#file \t sampid \t group" (e.g. tab delimited). The sample identifier column specifies identifiers for each replicate, typically SRR names. The group name specifies an identifier for each condition, which typically has multiple replicates associated. After a few minutes (precise timing depends on number of files and their size), muMerge produces an output file (SRR_MUMERGE.bed) that is the final muMerge identified regions of interest as a BED3 file. This file can be used as input to subsequent comparative analyses between conditions.

Inferring Transcription Factor Activity using TFEA

Regulatory regions, including enhancers, are dense with transcription factor (TF) binding sites [7, 81]. When a transcription factor binds and regulates transcription nearby, there is an increase of nascent transcription proximal to the TF's motif instance[61, 140]. The resulting pattern can be leveraged to identify which transcription factors are altered in response to a perturbation[8] (Figure 3.4B).

Over time, the techniques for inferring TF activity have improved. The original method, the Motif displacement (MD) approach, looked at TF motif colocalization with sites of RNA polymerase II loading (μ from Tfit). The original implementation of the MD-score took the ratio of TF motif instances located within a 150bp window around μ relative to the larger local background (a 1500bp window) [8, 141]. As the primary focus is on changes in TF activity across conditions, a modified version of the MD-score called differential motif displacement (MDD), was developed to compare the MD-score between a set of differential transcribed regions to a background set of regions not changing in transcription between the conditions [142]. While the MDD method quantified TF enrichment, it relied on an arbitrary cutoff to specify differential

transcription (typically a DEseq2 p-value). Transcription Factor Enrichment Analysis (TFEA) was subsequently developed as a refinement of the MDD method and eliminates arbitrary cutoffs in differential transcription[9]. As such, TFEA is now the recommended method for identifying changes in TF activity, e.g. changes in motif co-localization with sites of RNA polymerase initiation across conditions.

TFEA quantifies positional TF motif enrichment that is associated with changes observed between conditions. In order to calculate an enrichment score (E-score), TFEA first ranks regions based on the differential p-values derived from DESeq as well as the direction of fold change (Figure 3.4F). All the regions being compared then contribute to the E-score in a weighted manner. The weights are based on the distance of the motif instance to the center of the region (which is typically the center of the bidirectional) using an exponential function, favoring closer motifs (Figure 3.4E). The resulting E-score is the difference between the enrichment and the background random curves (Figure 3.4C). E-scores are calculated for every motif provided within a meme formatted database file (Figure 3.4B).

TFEA takes as input: regions to compare in BED file format, alignment files as BAM files, genome file in FASTA format and a TF motif database; executed as follows:

```
$ TFEA --output output_folder \
--bed1 condition1_rep1.bed condition1_rep2.bed \
--bed2 condition2_rep1.bed condition2_rep2.bed \
--bam1 condition1_rep1.bam condition1_rep2.bam \
--bam2 condition2_rep1.bam condition2_rep2.bam  \
--label1 condition1 --label2 condition2 \
--genomefasta genome.fa \
--fimo_motifs tf_database.meme
```

By default, the input regions are combined with muMerge giving consensus regions to assess. By default, regions are ranked and ordered by signed DEseq2 p-values (assuming replicates are

available). The HOCOMOCO v11 database of motifs[139] is provided with TFEA. It is important to note, however, that TFEA has multiple run options which allows the user to customize these choices. For example, a user can opt to run TFEA starting with BAM files and ROI BED files (as shown in the above run example), in which case TFEA will rank the ROI use the ranked ROI for the subsequent TF enrichment calculation step. Alternatively, a preranked ROI file can be give as input, in which case TFEA moves straight to the enrichment calculation step. Running TFEA can be done through the command line or, for more advanced experimental setups, with a configuration file (see GitHub documentation for full details).

The output files from TFEA include a results.txt file that contains E-scores and statistical significance for the calculated E-scores. Furthermore, an HTML document is created that has summary figures for the significantly changed TFs (p53 plots shown in Figures 3.4C-G) and a scatter plot with E-scores and motif hits for all TFs (Figure 3.4B). We found that TFEA drastically improves the signal of enriched TFs compared to both the MD-score and MDD method[9]. As shown in Figure 3.4, TFEA accurately identified p53 (and TFs with similar motifs) as activated in Nutlin treatment (Figure 3.4B). Overall TFEA is memory efficient and relatively fast, depending on the number of ROIs. Case in point, the example in Figure 3.4 took roughly 6 hours on 8 threads.

Conclusions

Nascent RNA sequencing offers a detailed look at transcription and RNA polymerase behavior. Since most transcription occurs at non-coding regions, methods that are annotation agnostic are instrumental in identifying the regions being transcribed. In this chapter we describe our software for nascent transcription data analysis, providing general guidelines and highlighting the importance of data quality on algorithm outputs. With the methods and tools discussed, users can investigate questions about the transcription process (initiation, elongation, termination) or its regulation.

Funding

This work was funded in part by a National Science Foundation (NSF) ABI grant number 1759949 and a National Institutes of Health (NIH) grant RO1 GM125871. We acknowledge the

BioFrontiers Computing Core at the University of Colorado Boulder for providing high-performance computing resources (NIH 1S10OD012300) supported by BioFrontiers' IT.

Acknowledgements

We would like to thank Mary A. Allen whose guidance and contributions to all aspects of this toolkit are invaluable. Additionally we thank Joseph G. Azofeifa for developing FStitch and Tfit, Jonathan D. Rubin for developing TFEA and Jacob T. Stanley for developing muMerge.

Author Attributions

All software was developed with supervision from R.D.D, who currently maintains these packages. M.A. Gruca developed and wrote the sections on Materials (Section 2) and FStitch (Section 3.1). R.F.S developed and wrote the sections on Tfit (Section 3.2), muMerge (Section 3.3), and TFEA (Section 3.4). M.A. Gohde. contributed to Tfit software maintenance. All authors revised the manuscript for clarity.

Declarations

Dr. Robin Dowell is a founder of Arpeggio Biosciences and holds a patent on leveraging eRNAs towards transcription factor activity inference.

Availability

Our software (FStitch, Tfit, muMerge, TFEA) are publicly available and open source. Additional documentation as well as installation guides are available on the lab's GitHub repository pages: `https://github.com/Dowell-Lab/`, where each package has an individual repository. In every case, the software can be cloned directly from these sources. Specific installation instructions are provided for each package. FStitch and Tfit are C programs that require the GCC compiler version >5.1 and Tfit additionally requires MPI support. TFEA (and muMerge) can be pip installed along with all their dependencies. Alternatively, FStitch, Tfit and TFEA are each available as Docker containers, which can be found on the respective GitHub repositories.

CHAPTER IV

TRANSCRIPTION FACTOR ENRICHMENT ANALYSIS WITH NASCENT RNA SEQUENCING DATA

This chapter are adapted from:

Jonathan D. Rubin, Jacob T. Stanley, **Rutendo F. Sigauke**, Cecilia B. Levandowski, Zachary L. Maas, Jessica Westfall, Dylan J. Taatjes, Robin D. Dowell. *Transcription factor enrichment analysis (TFEA) quantifies the activity of multiple transcription factors from a single experiment.* Commun Biol. 2021 Jun 2;4(1):661. doi: 10.1038/s42003-021-02153-7. PMID: 34079046; PMCID: PMC8172830.

My contribution to this work was that I tested TFEA on several experimental perturbations and showed that the algorithm does recover expected TFs. I also worked on how the regions overlap TF ChIP-seq data. Since this publication, I have done extensive work on the benchmarking and debugging of muMerge as it is an essential tool for a large scale meta-analysis (more detail in Chapter V).

Abstract

Detecting changes in the activity of a transcription factor (TF) in response to a perturbation provides insights into the underlying cellular process. Transcription Factor Enrichment Analysis (TFEA) is a robust and reliable computational method that detects positional motif enrichment associated with changes in transcription observed in response to perturbation. TFEA detects positional motif enrichment within a list of ranked regions of interest (ROIs), typically sites of RNA polymerase initiation inferred from regulatory data such as nascent transcription. Therefore, we also introduce *muMerge*, a statistically principled method of generating a consensus list of ROIs from multiple replicates and conditions. TFEA is broadly applicable to data that informs on transcriptional regulation including nascent transcription (eg. PRO-Seq), CAGE, histone ChIP-Seq, and accessibility (e.g. ATAC-Seq). TFEA not only identifies the key regulators responding to a perturbation, but also temporally unravels regulatory networks with time series

data. Consequently, TFEA serves as a hypothesis-generating tool that provides an easy, rigorous, and cost-effective means to broadly assess TF activity yielding new biological insights.

Introduction

The cellular response to everything from environmental stimuli to development is orchestrated by transcription factors (TFs). Therefore, when transcription changes, one important objective is to infer which transcription factors are causally responsible for the observed changes. Transcription factors bind to DNA at preferred sequence specific recognition motifs and ultimately alter transcription nearby. Extensive DNA-protein binding has been measured by chromatin immunoprecipitation (ChIP)[13, 143, 144], leading to large collections of high quality sequence recognition motifs[144–146]. Unfortunately, acquisition of protein-DNA binding is not sufficient for understanding regulation, as many binding sites do not lead to altered transcription nearby[147–149].

In an effort to causally link a TF to observed transcription changes, binding data is often combined with expression, typically measured by RNA-seq[150–152]. However, the success of this approach is limited. Fundamentally, the difficulty lies not in the binding data, but rather in the use of steady state RNA-seq to assay expression. RNA-seq levels reflect both transcription and degradation[153–155], e.g. both newly created and long-lived RNAs contribute to the measurement[156, 157]. Hence RNA-seq is, at best, only an indirect measure on transcription. Additionally, RNA-seq data is dominated by the most abundant RNAs, rather than those that are most recently made. Thus, after ribosomal RNAs, the dominant signal is protein-coding genes, which are highly stable processed transcripts. Additionally, to infer TF activity, one must solve the assignment problem[150] – namely linking TF binding sites to stable gene transcripts, which are often both positionally (in the genome) and temporally (RNA processing) distant[158].

Nascent transcription[42, 113] circumvents the assignment problem. Nascent transcription assays measure *bona fide* transcription, prior to RNA processing. Thus, changes in transcription induced by transcription factors can be detected within minutes[76]. Conveniently, a TF's regulatory activity has been shown to alter RNA polymerase initiation immediately proximal to

sites of TF binding[61, 83, 159]. The majority of altered RNA polymerase initiation sites are at transcription regulatory regions (e.g. enhancers) –not at genes[41]. Thus, by using all polymerase initiation sites (both at enhancers and genes) rather than just the target gene, the assignment problem is sidestepped[80]. Enhancer RNAs (eRNAs) are highly transient unstable transcripts that are essentially undetectable in RNA-seq, yet effectively serve as markers of TF activity. Consequently, our previous work demonstrated the ability to directly infer causal TF activity from changes in RNA polymerase initiation observed in nascent transcription assays[8].

Despite recent improvements to the protocols[64, 65, 160], nascent transcription assays are less popular than other genomic assays, likely due to their perceived difficulty. Luckily, a variety of popular high throughput assays also have a relationship with RNA polymerase initiation and therefore could serve as proxies to nascent transcription. For example, cap associated approaches, such as CAGE, target the 5′ cap of transcripts[21, 94, 161] and are therefore a viable alternative to nascent transcription. However, CAGE provides only a subset of RNA polymerase initiation sites, biased to stable transcripts[21]. In contrast, transcription arises from only a subset of nucleosome free regions, therefore chromatin accessibility data indirectly informs on the locations of transcription initiation. Likewise, some histone marks have been associated with actively transcribed regions, such as H3K27ac and H3K4me1/2/3[162]. In principle, all of these methods provide some information on sites of RNA polymerase initiation, but with distinct detection limits, positional precision, and temporal fidelity. To leverage these datasets, a motif enrichment method is needed that seamlessly handles the uncertainty inherent in using an approximation to RNA polymerase initiation.

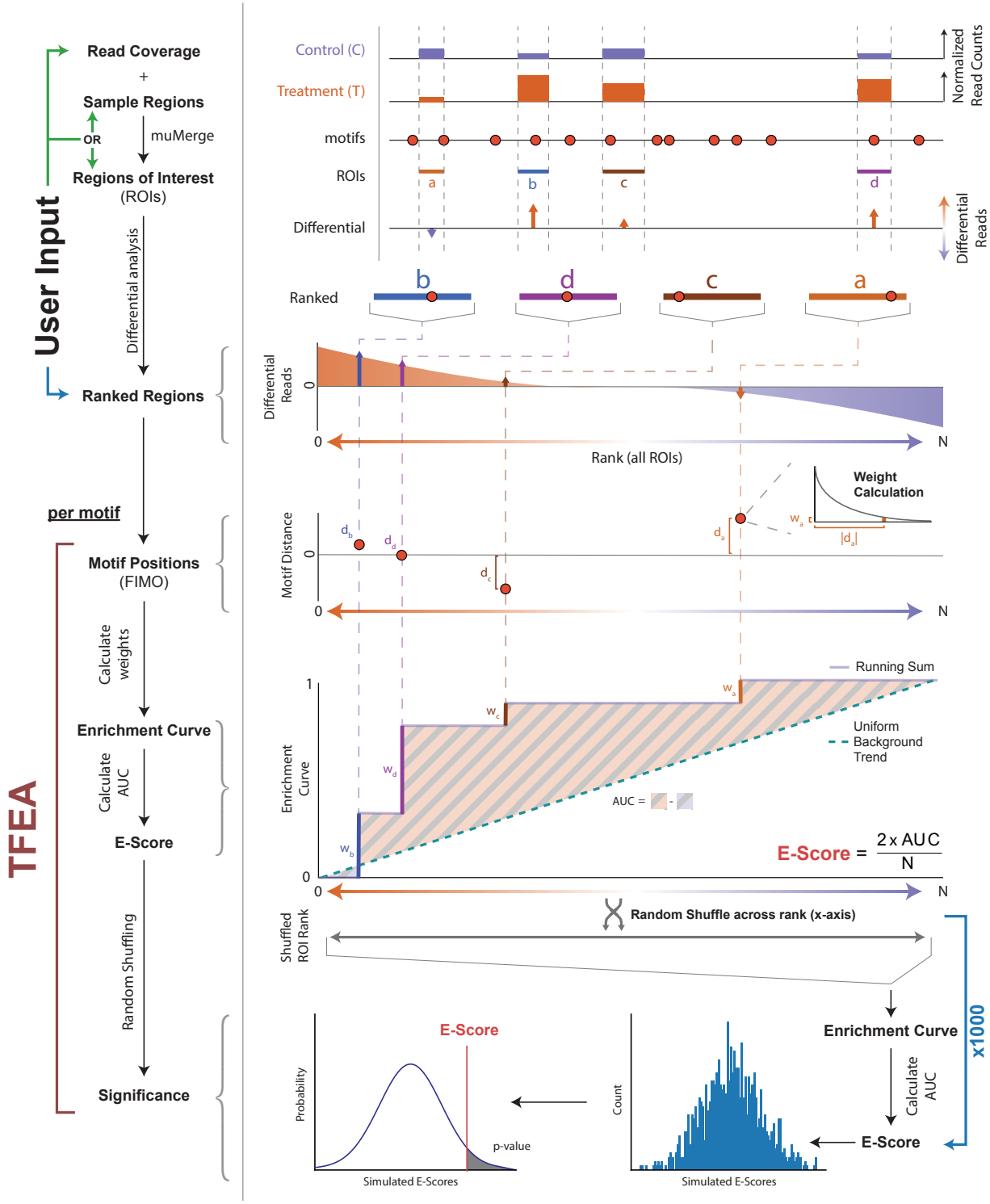
Therefore, we introduce transcription factor enrichment analysis (TFEA), a motif enrichment method specifically aimed at maximizing the informative nature of differential RNA polymerase initiation data, where positional information is critically important[8, 163]. TFEA not only accounts for the position of the motif relative to transcription initiation, but also accounts for the magnitude of transcription change (*i.e.* differential signal)[164–166]. Critically, TFEA is robust to noise in both of these sources of information (position and signal) and therefore can be

applied to a number of different regulatory datasets. Finally, TFEA is fast, computationally inexpensive, and designed with the user in mind, as we provide an easy to use command-line interface, container images (Docker and Singularity), and an importable Python 3 package. TFEA provides easy downstream analysis aimed at deciphering the temporal and mechanistic details of complex regulatory networks.

Results

Overview

Transcription factor enrichment analysis (TFEA) seeks to identify which TF(s) are causally responsible for observed changes in transcription between two data sets. An overview of this procedure is shown in Figure 4.1 (See Supplementary Figure 3.1 and 3.2 for example outputs) . Briefly, TFEA takes as input a set of RNA polymerase initiation regions and ranks them, preferably by changes in transcription levels between the two conditions. The ranked list is then used to calculate a TF motif enrichment score, which incorporates not only the differential transcription signal at initiation sites but also the distance to the nearest motif instance. The TF enrichment score is then compared to the distribution of expected scores, empirically derived, to assess statistical significance of the TF motif enrichment.



TFEA calculates motif enrichment using differential and positional information.

Figure 4.1: TFEA calculates motif enrichment using differential and positional information. The TFEA pipeline requires, minimally, a ranked list of ROIs (control in blue, treatment in orange). Optionally, a user may provide raw read coverage and regions (ROI, colored boxes labeled a-d), in which case TFEA will perform ranking using DESeq [88, 137] analysis. With a set of ranked ROIs (orange up, blue down), TFEA analyzes motif enrichment for each motif provided (red circles). For each motif, positions are determined by FIMO scans and an enrichment curve is calculated by weighting each motif instance (with weight w_i, using an exponential decay as a function of the motif distance d_i from the region center) and adding this value to a running sum. An E-Score is calculated as 2 * AUC, e.g. the area under the enrichment curve between the running sum and a uniform background (dashed line), and scaled by the number of motif instances N. For statistical significance, the ROI rank is randomly shuffled 1000 times, and E-scores are recalculated for each shuffle. The true E-Score is then compared to the distribution of E-Scores obtained from the shuffling events. For example output of TFEA see Supplementary Figure 3.1 and Supplementary Figure 3.2.

Importantly, for each cell type and condition, RNA polymerase initiates transcription from a distinct set of locations. Biologically, each RNA polymerase initiation event corresponds to an individual transcription start site (TSS). However, most sites of initiation occur in regions of bidirectional transcription with two closely, oppositely oriented TSSs[41, 167, 168]. Many assays are unable to distinguish between the two TSSs within a RNA polymerase loading zone[80] (also see Methods section "Regions of Interest"). Therefore, without loss of generality, we assume each assay provides a set of regions of interest (ROI) where each region corresponds to either a single TSS or the midpoint between bidirectional TSSs. Each ROI provides a point estimate (the midpoint of the region) and an uncertainty on that reference point (width of the region). Because initiation sites are inferred directly from data, they must first be combined across replicates and conditions in a manner that maintains high fidelity on the position of RNA polymerase initiation. Thus we first introduce and evaluate *muMerge*, a method of combining regions of interest (ROI).

***muMerge*: Combining genomic features from multiple samples into consensus regions of interest**

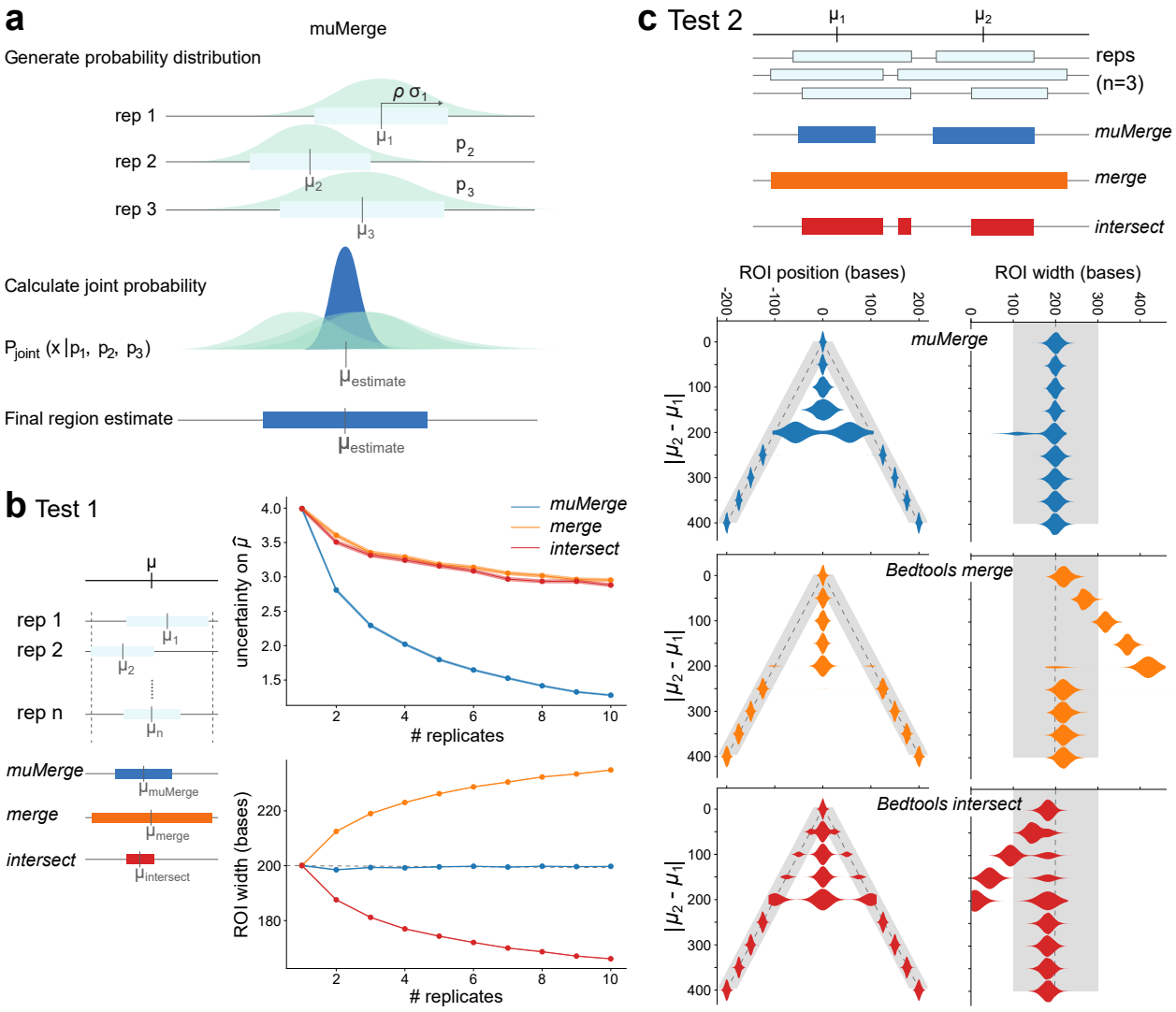
A key challenge in defining a set of consensus ROIs is retaining positional precision when combining region estimates that originate from different samples (replicates and/or conditions). To this end, we developed a statistically principled method of performing this combination called *muMerge*. In short, *muMerge* treats the ROIs from each sample as probability distributions and combines these across samples, according to whether they are replicates or different conditions, to produce a joint probability distribution that describes the highest likelihood position for polymerase initiation (See Figure 4.2a, Supplementary Figure 3.3 and Methods section "Defining ROIs with *muMerge*" for full details).

In order to demonstrate the efficacy of *muMerge*, we compare its performance to two common methods for combining regions across multiple samples—merging all samples (e.g. with *bedtools merge*) and intersecting all samples (e.g. with *bedtools interesect*). We performed two tests using simulated data (Figure 4.2b-c; Supplementary Figure 3.4). For each replicate, we

performed 10,000 simulations of sample regions for a single locus, and calculated the average performance.

Using the simulated regions, we first evaluate each methods' precision as the number of replicates increases. In Figure 4.2b, we observe that as the number of replicates increases *muMerge* converges on the correct theoretical locus position (μ) more quickly than the other two methods (*i.e.*, the vertical axis "uncertainty on $\hat{\mu}$" is the standard deviation of the distance between μ and its estimate ($\hat{\mu}$), which is computed from all 10,000 simulations), while still maintaining the correct width for the region.

The second test sought to evaluate the accuracy of these methods when inferring two closely spaced loci, with increasing distance between those loci (Figure 4.2c). While closely spaced loci are challenging to distinguish, we observe that *muMerge* smoothly transitions from calling a single inferred locus (when μ_1 and μ_2 are too close to be resolved) to two distinct loci. In contrast, the *merge* and *intersect* methods show abrupt transitions that follow increasingly poor ROI width estimates (Figure 4.2c). These tests quantitatively demonstrate the benefit of *muMerge* over the other two methods using simulated data. A comparison using experimental ChIP-seq data[69, 143], where the position of the TF motif instance is used as ground truth, further supports this conclusion (Supplementary Figure 3.5-3.6). Examples of the output from all three methods on ChIP-seq data are shown in Supplementary Figure 3.7.



muMerge precisely combines multiple samples into consensus ROIs.

Figure 4.2: *muMerge* precisely combines multiple samples into consensus ROIs. (a) A schematic for the *muMerge* method. Each sample region (light blue box) is represented by a probability distribution (green, Eq. IV.1, with centers μ_i and stdev $\rho\sigma_i$), which are combined into a joint probability distribution (dark blue peak, Eq. IV.2) from which the final ROI estimates are inferred (dark blue bar). (b) Test 1 demonstrates the position and width accuracy of a calculated ROI for a single locus, μ, as the number of sample replicates are increased (from one to ten). The three methods, *bedtools merge* (orange), *bedtools intersect* (red), and *muMerge* (dark blue), for generating ROIs from multiple samples are compared. With *muMerge* the uncertainty on $\hat{\mu}$ (*i.e.* the standard deviation of the distance between the ground truth position, μ, and its estimate, $\hat{\mu} \in \{\mu_{muMerge}, \mu_{merge}, \mu_{intersect}\}$) decreases quickly while the estimated ROI width remains essentially constant. The standard error, indicated by colored shading, is less than the line width in most cases. (c) Test 2 demonstrates the precision of the calculated ROI for two closely spaced loci, μ_1 and μ_2, as the spacing between them is increased. In this case, *muMerge* transitions from a single locus to two distinct loci more gradually (violin plots, ROI position) and the estimated ROI widths do not deviate from the expected value (violin plots, ROI width), unlike *merge* and *intersect*. In all cases, expected value and variance used for the simulations is indicated by dashed grey lines and shading, respectively. For further detail on the results of Test 1 and 2 and how the simulations were performed, see Supplementary Figure 3.4 and Methods *muMerge*: Simulating replicates for calculation of ROIs.

Transcription Factor Enrichment Analysis

Armed with the defined set of ROIs, the goal of TFEA is to determine if a given TF motif shows positional enrichment preferentially at regions with higher differential signal. Therefore an enrichment metric is necessary that accounts for not only the positional enrichment of the motif but also the underlying changes in transcription (Figure 4.1). The enrichment metric builds on previous work[8], but provides substantial improvements by eliminating arbitrary cutoffs and refines the sensitivity to motif position, which is not present in other methods[169].

In prior work, we assessed the enrichment of motifs relative to positions of RNA polymerase initiation using a co-occurrence metric hereafter referred to as a motif displacement score (MD-Score)[8]. The MD-Score is simply the ratio of TF sequence motif instances within 150 bp radius of ROI midpoints, relative to a larger local 1500 bp radius (see Supplementary Figure 3.8 for full details). Unfortunately, the MD-Score approach not only ignored alterations in transcript levels (See Supplementary Figure 3.9) but also utilized arbitrary distance thresholds to classify motif proximity in a binary fashion. To account for changes in transcription levels, we subsequently ranked ROIs by differential signal (e.g. transcription) before performing motif displacement calculations within these regions[166]. This method, referred to as differential motif displacement analysis (MDD) compared MD-Scores between the set of differentially transcribed regions to the MD-Score obtained from regions whose transcription is unchanged (see Supplementary Figure 3.10 for full details)[165, 166]. Unfortunately, the MDD-Score approach introduces an additional arbitrary threshold (e.g. to classify regions as differentially transcribed or not) and still uses the arbitrary motif distance thresholds set by the original MD-Score approach. For TFEA we sought a method that eliminates the reliance on arbitrary cutoffs.

With TFEA, we begin by leveraging the statistically robust, gold standard DESeq package[88, 137] to rank regions based not only on the differential p-value but also the direction of fold change. Each region of interest then contributes positively to the enrichment curve in a weighted fashion. These weights are determined by the distance of the motif to the reference point using an exponential function to favor closer motifs. The subsequent enrichment score (E-Score in

Figure 4.1) is proportional to the integrated difference between the observed and background enrichment curves, calculated as the area under the curve (AUC) in Figure 4.1 (see Eq. IV.8). The background (null) enrichment curve assumes uniform enrichment across all ROIs, regardless of differential signal.

By default, TFEA accounts for the known GC bias of enhancers and promoters by incorporating a correction to the enrichment score (Supplementary Figure 3.11). Once E-Scores for all TFs have been calculated, we fit a linear regression to the distribution of these scores as a function of motif GC-content. Corrected E-Scores are then calculated from the observed E-Score with the y-offset observed from the linear regression fit (see Eq. IV.11). This GC bias correction can be optionally turned off.

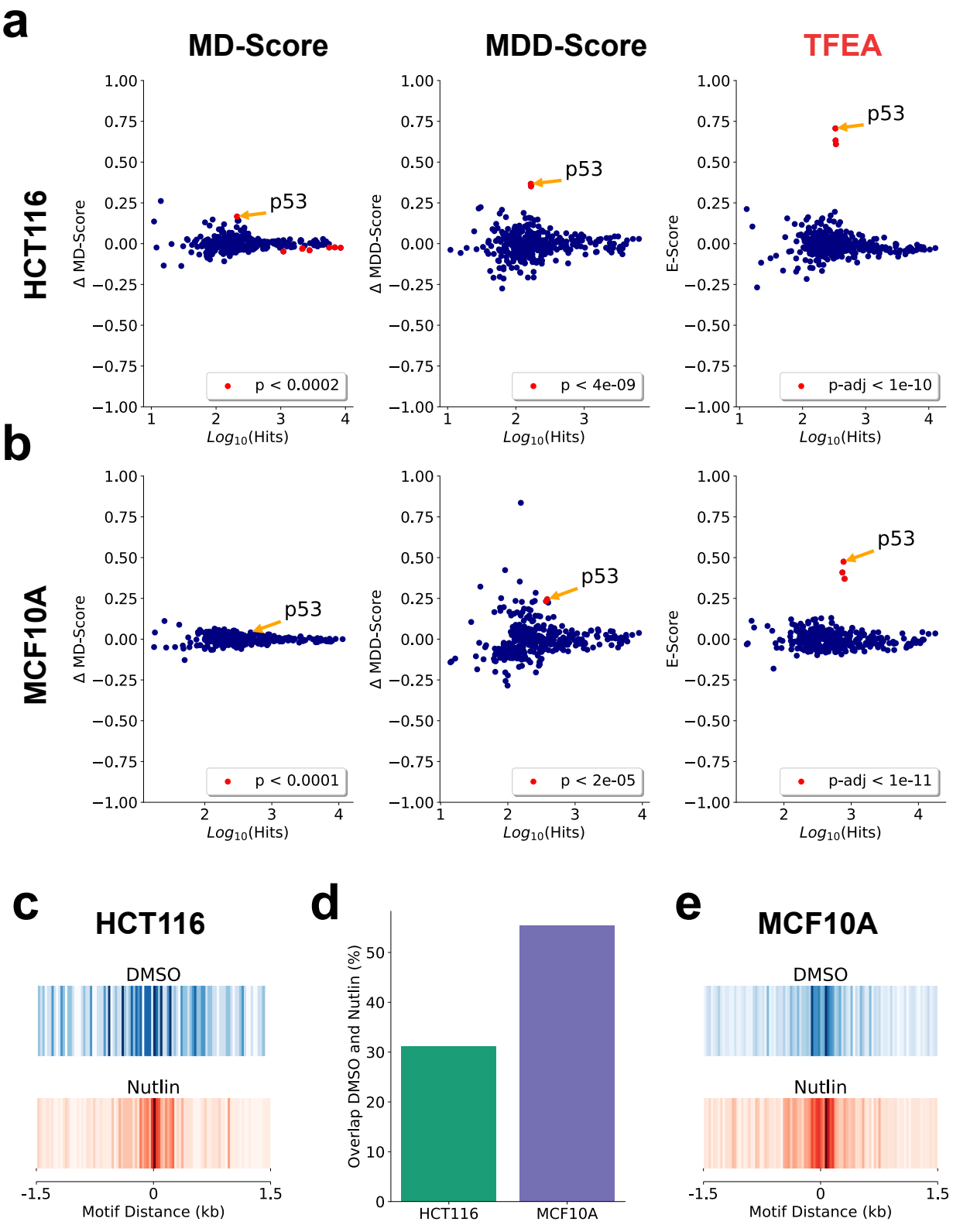
Subsequently, we assess the significance of the enrichment score by comparison to randomized ROI order, similar to GSEA[87]. To this end, we generate a null distribution of enrichment scores from random permutations, shuffling the rank order of regions and recalculating the E-Score for each shuffled permutation. The final significance of the enrichment score is then calculated from the Z-score, using the Bonferroni correction to account for multiple hypothesis testing. In this manner, TFEA provides a statistically robust and principled way of calculating motif enrichment that accounts for both differential transcription and motif position without arbitrary distance or differential transcription cutoffs.

Differential transcription signal improves motif inference over positional information alone

To assess the effectiveness of the TFEA method, we first compared its performance to both the MD-Score[8] and MDD-Score[165, 166] approaches. We examined a dataset in which a one hour Nutlin-3a treatment of HCT116 cells is used to activate *TP53*[61]. For all methods, sites of RNA polymerase loading and initiation were determined from GRO-seq data[61] using the Tfit algorithm, which leverages a mathematical model of RNA polymerase II behavior to identify RNA polymerase loading zones directly from patterns in the data[81]. These sites were then combined using *muMerge* to identify ROIs. For all methods, the significance threshold utilized was determined by comparing within treatment replicates (e.g. DMSO to DMSO) and identifying

the score at which no changes are detected (see Supplementary Figure 3.12). Using these per method thresholds, we recover *TP53* from all three approaches (Figure 4.3a). Notably, by including differential transcription information, the signal to noise ratio of *TP53* detection is drastically improved—modestly in the case of MDD and dramatically for TFEA.

We next sought to determine whether TFEA could infer the responsible TF when the underlying changes in transcription were predominantly alterations in existing transcript levels. For this test, we relied on the fact that *TP53* response in epithelial cells depends on the *TP53* family member *TP63*[170]. Because *TP53* and *TP63* have nearly identical motifs, we reasoned that the presence of a constitutively active *TP63* would result in elevated basal transcription proximal to *TP53*/*TP63* motifs. To test this hypothesis, we performed PRO-seq on MCF10A cells after one hour treatment of either DMSO (control) or Nutlin-3a, and applied all three methods to the resulting data.



TFEA improves the detection of p53 following Nutlin-3a treatment.

Figure 4.3: TFEA improves the detection of p53 following Nutlin-3a treatment. (a) Application of the MD-Score, MDD-Score, and TFEA to GRO-Seq data in HCT116 cells with 1hr Nutlin-3a or DMSO treatment [61]. MA plots contrast number of regions with motif (x-axis) to the change in each score (y-axis). Each dot is a distinct position specific scoring matrix (e.g. TF) with significant changes highlighted in red. Cutoffs determined by comparing untreated replicates (see Supplemental Figure 3.12). (b) Application of the MD-Score, MDD-Score, and TFEA to PRO-Seq data in MCF10A cells with 1hr Nutlin-3a or DMSO treatment. (c) Motif displacement distribution plot of TP53 motif instances within 1.5kb of all ROI in either DMSO (blue) or Nutlin-3a (red) (as heatmap, darker indicates more motif instances). (d) Percentage overlap of TP53 motif instances within 150bp of DMSO and Nutlin-3a ROIs. (e) Similar to (c) but in MCF10A cells. See Supplementary Data 1 for complete list of accession numbers for data utilized.

Consistent with the constitutive activity of *TP63*, we observed no change in the *TP53* motif by MD-Score analysis (Figure 4.3b, left). This is due to a larger fraction of ROIs having pre-existing transcription, prior to Nutlin-3a exposure, in MCF10A relative to HCT116 cells (Figure 4.3c-e, Supplementary Figure 3.13). While the MDD-Score method recovers *TP53* (Figure 4.3b, middle), TFEA drastically improves the signal of the *TP53* motif relative to the distribution of all other motifs (Figure 4.3b, right). For more detailed analysis of *TP53* after Nutlin-3a in HCT116 and MCF10A, see Supplementary Figs 3.14 and 3.15.

TFEA improves motif enrichment detection by incorporating positional information

We next sought to quantify the performance of TFEA with varying degrees of signal, background, and positional information. As a reference point, we leveraged the widely used MEME-Suite component AME, which quantifies motif enrichment by fitting a linear regression to ranked ROIs as a function of motif instances (Supplementary Figure 3.16) [171]. Importantly, AME does not utilize positional information.

To compare the two methods, we required biologically representative data sets with known motif enrichment, so that error rates could be readily calculated. To this end, we utilized the sites of RNA polymerase initiation detected in untreated GRO-seq datasets of HCT116 cells[61] as the base set of ROIs. As there is no second dataset for this comparison, the ROI were then arbitrarily ranked to mimic a pattern of differential transcription. Subsequently, specific instances of the HOCOMOCO[172] obtained *TP53* motif were embedded via sequence replacement into the ordered ROI list. Importantly, the position and frequency of embedded motifs (e.g. true signal) is varied to simulate distinct TF motif enrichment patterns (see Supplementary Figure 3.17 and Methods section "TFEA: Simulated motif enrichment"), allowing us to access the accuracy of both TFEA and AME.

We first measured the mean false positive rate (FPR) and mean true positive rate (TPR) across tests of varying signal and background (Figure 4.4a). We found that AME detected many false positives (defined as any motif besides *TP53*) at loose threshold cutoffs and therefore chose a strict cutoff of 1e-30 for AME. TFEA on the other hand, had a very low FPR even at loose

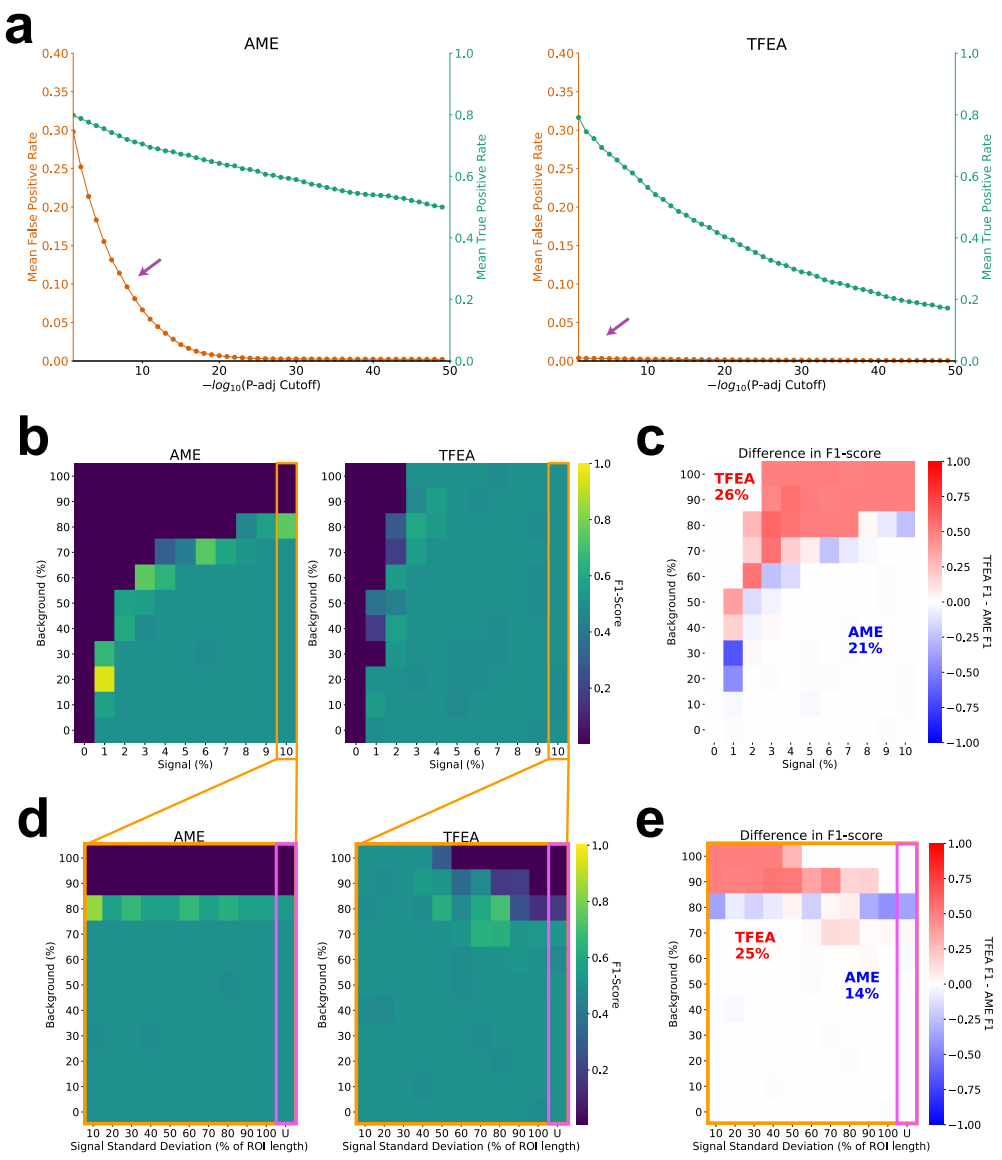


Figure 4.4: (a) Optimal cutoffs are determined using the mean true positive rate (TPR; green) and mean false positive rate (FPR; orange) across different signal and background levels as a function of varying the threshold cutoff. (b) F1 score of AME and TFEA for varied signal and background, using optimal AME cutoff 1e-30 and TFEA cutoff 0.1. (c) Difference in F1 score between TFEA and AME across all simulations (n=121; value=$F1_{TFEA} - F1_{AME}$). TFEA (red) outperforms AME (blue) in 26% of cases (value > 0) whereas AME outperforms TFEA in 21% of cases (value < 0). (d) F1 scores and (e) difference in scores for highest signal tested (10% signal), now varying the standard deviation of the signal and background. See Supplementary Figure 3.17 for more details on simulations.

thresholds with the TPR decreasing as the cutoff became stricter. We therefore chose a cutoff of 0.1 for TFEA.

We next generated two sets of simulated datasets to evaluate the performance of each method with varying signal/background (Figure 4.4a) or variance/background (Figure 4.4b). For each scenario, we generated 10 simulations and measured F1 scores for AME and TFEA. Varying signal/background (Figure 4.4b), we found that at high background levels (above 80%), AME was no longer able to detect the enrichment of *TP53*. TFEA on the other hand, was able to detect *TP53* even at high background levels by incorporating positional information. Computing the differential F1 scores between the two methods (Figure 4.4c) shows that TFEA performs well in cases where AME detects no enrichment of *TP53* (26% of cases), whereas AME outperforms TFEA in 21% of cases. Importantly, because AME does not take positional information into account, it was never able to capture cases where the level of signal and background are similar.

To further determine how TFEA handles the loss of positional information, we chose the highest signal level tested and altered the variance (standard deviation) of the signal position and the background level (Figure 4.4d). As expected, AME shows consistent behavior regardless of the positional information of the motif. In contrast, TFEA is able to distinguish signal with differing levels of positional localization. In the extreme case of no positional localization (motifs embedded with a uniform distribution), TFEA performs only slightly worse than AME (Figure 4.4e).

Finally, we sought to benchmark the runtime performance and memory usage of TFEA against AME. Here we leverage a first order Markov model (see Methods section "TFEA: Testing compute performance" to simulate increasing numbers of ROIs as input. Analyzing the core collection of HOCOMOCO TF motifs (n=401), we found that AME runtime increased non-linearly while TFEA runtime increased linearly with a single processor (Supplementary Figure 3.18a). Importantly, TFEA can utilize parallel processing, leading to notably faster runtimes. In terms of memory usage, although TFEA consumes more memory than AME, even in

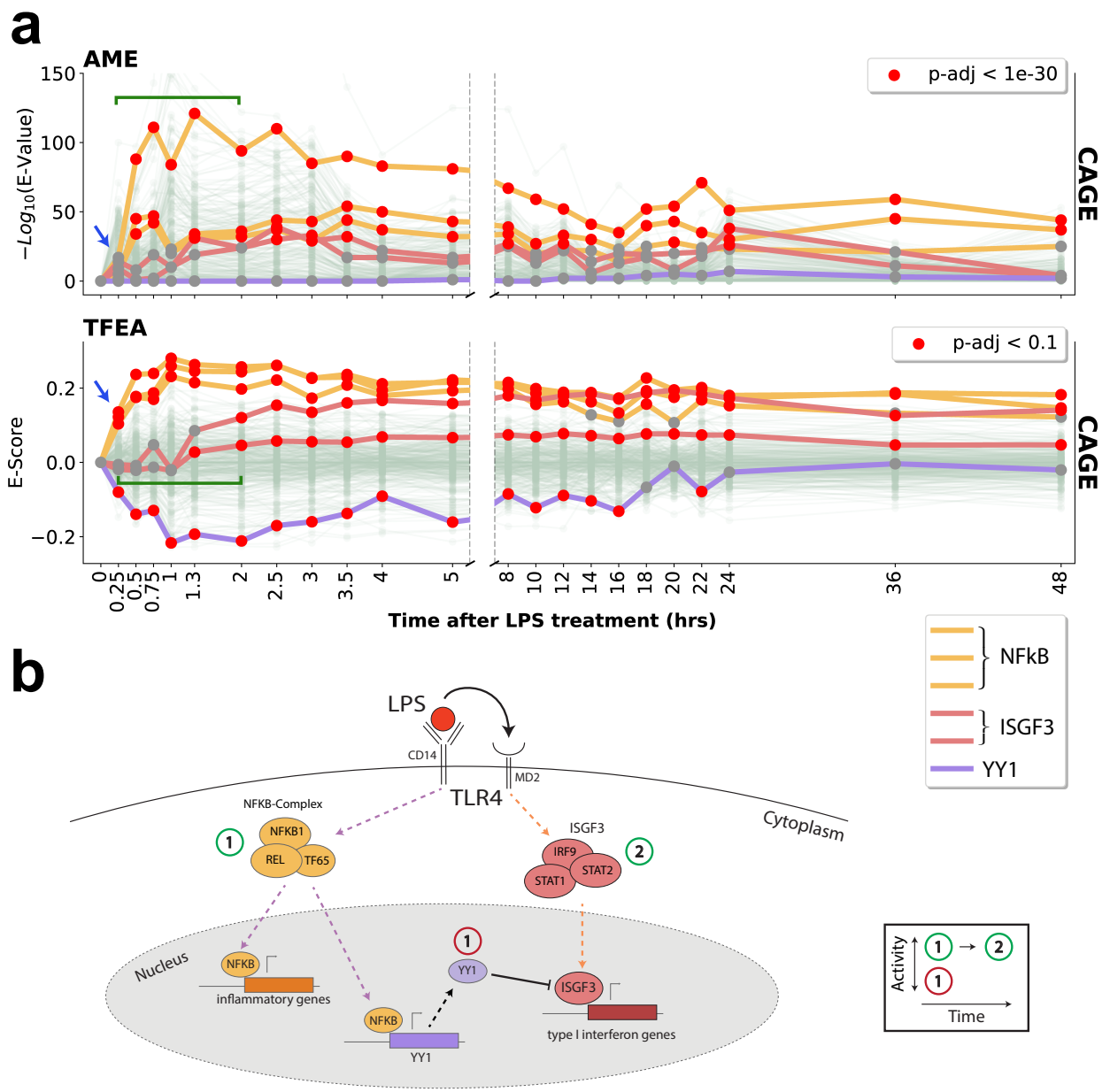


Figure 4.5: (a) Analysis of lipopolysaccharide (LPS) timeseries cap analysis gene expression (CAGE) data[173, 174] using AME and TFEA. Trajectories of activity profiles shows LPS triggers immediate activation of the NF-$\kappa\beta$ complex (TF65/RelB/NFKB1; yellow), observable at 15min (blue arrow). TFEA detects a concomitant down regulation of a set of transcription factors, exemplified here by TYY1 (purple). TFEA also resolves subsequent dynamics (green bracket) of ISGF3 activation (containing IRF9/STAT1/STAT2; red lines). (b) Schematic depicting the molecular insights gained from TFEA analysis. See Supplementary Figure 3.19 for more analysis. See Supplementary Data 1 for complete list of accession numbers for data utilized.

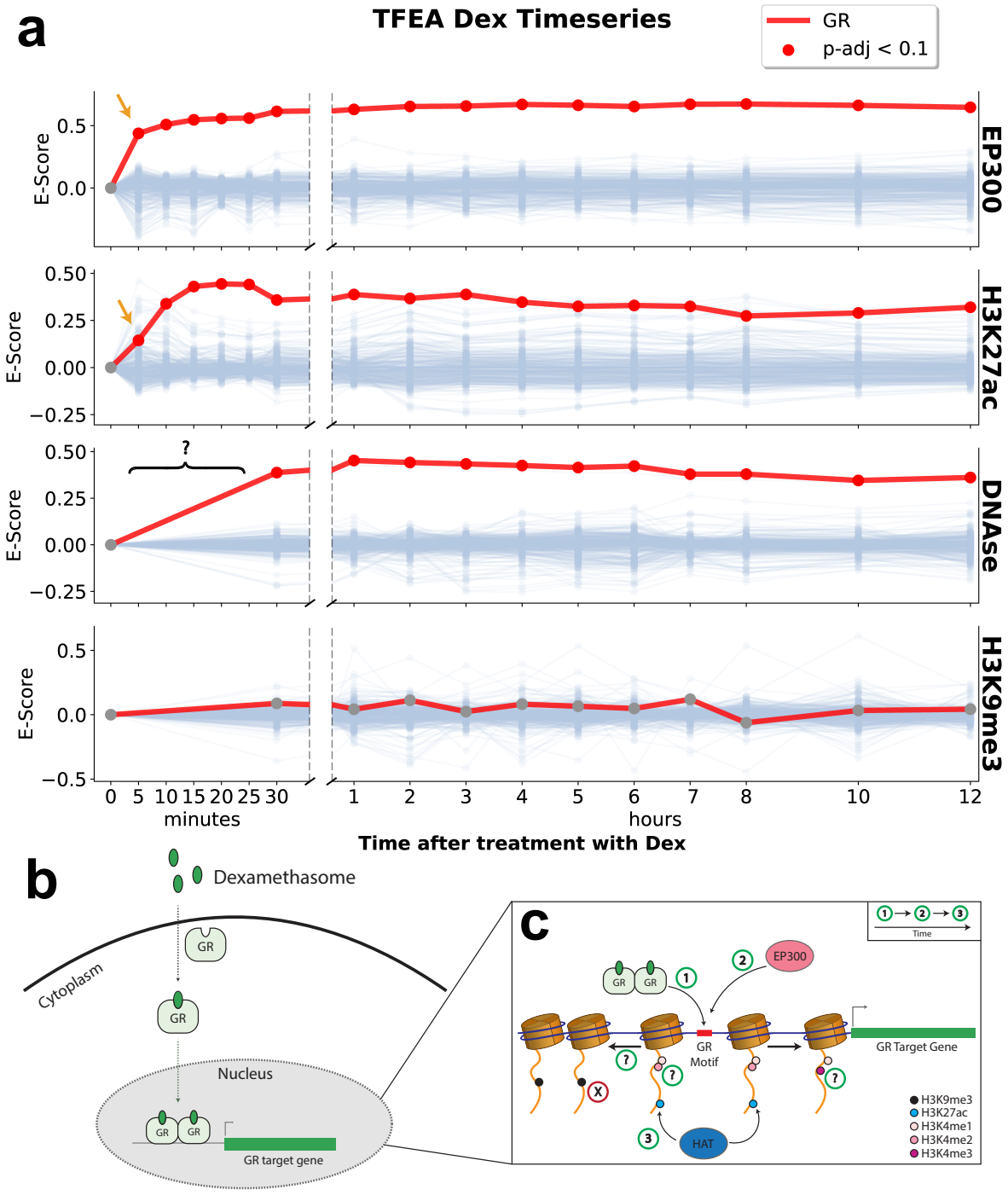
the worst case of 100,000 input regions, TFEA's memory footprint is less than 1Gb and therefore can still be run on a local desktop computer (Supplementary Figure 3.18b).

TFEA outperforms AME on experimental time series data

We next sought to examine the performance of TFEA and AME on real biological data. Here we utilized cap analysis of gene expression (CAGE), which precisely defines the transcription start site (TSS) of individual transcripts[19, 79, 173]. We analyzed a CAGE-seq timeseries dataset from the FANTOM consortium[173, 174]. In this dataset, human derived monocytes were differentiated into macrophages and treated with lipopolysaccharide (LPS), a proxy for bacterial infection. Differential expression analysis was performed on each LPS time point comparing treatment to control to obtain a list of ranked ROIs.

TFEA recovered the immediate innate immune response, exemplified by the most rapid reported (within 15 min) activation of NF-$\kappa\beta$ (*RELA*, *RELB*, and *NFKB1*; Figure 4.5a). Additionally, TFEA temporally resolved the known secondary response that arises at later time points, which includes the activation of the IFN-stimulated gene factor 3 (ISGF3)[175] complex, comprising *IRF9* and *STAT1*/*STAT2*[176]. In contrast, AME did not recover the innate immune response at the earliest time point and provided less temporal resolution when distinguishing primary and secondary responses.

Concurrent with the immediate innate immune response, TFEA identified a set of TFs that exhibit a rapid decrease in E-Scores including *ELF1*/*ELF2*[177], *YY1* [178][179], *USF1*/*USF2*[180], and *GABPA*[181]. The decreased E-Score set includes *YY1*, a transcriptional inhibitor known to be activated directly by NFκB [182]. Reduction in the E-Score of *YY1* illustrates an important limitation of TFEA—namely, that it cannot distinguish between the activation of a repressor or the loss of an activator. Ultimately, we show with this proof of principle that if the cellular response to LPS was not known *a priori*, we could temporally resolve key aspects of the regulatory network using TFEA and dense time series CAGE data (Figure 4.5b and Supplementary Figure 3.19).



TFEA captures rapid dynamics of glucocorticoid receptor (GR) following treatment with dexamethasone.

Figure 4.6: TFEA captures rapid dynamics of glucocorticoid receptor (GR) following treatment with dexamethasone. (a) TFEA correctly identifies GR (red line) from time series ChIP data on the histone acetyl-transferase p300, H3K27ac and DNase I[183]. No signal is observed in the negative control H3K9me3. TFEA shows a temporal lag in H3K27ac signal (orange arrows). (b) Known cellular dynamics of GR induced by dexamethasone (Dex). (c) Mechanistic and temporal insights gained by performing TFEA analysis, question marks indicate datasets where earlier time points were not available to resolve temporal information. See Supplementary Data 1 for complete list of accession numbers for data utilized.

TFEA works on numerous regulatory data types that inform on RNA polymerase initiation

We developed *muMerge* and TFEA for the purpose of inferring TF activity from high resolution data on transcription initiation, such as precision run-on sequencing (PRO-seq) or CAGE. However, numerous genomic datasets aimed at transcriptional regulation have a clear relationship with RNA polymerase initiation (see Methods section "Regions of Interest"). For example, RNA polymerase initiation originates in open chromatin regions. Although these data sets are less precise and are not direct readouts of polymerase initiation, the popularity of these data make them readily available. To determine whether TFEA could adequately infer TF activity from these datasets, we analyzed a timeseries dataset from ENCODE[143, 183] in which cells were treated with dexamethasone (Dex)—a known activator of the glucocorticoid receptor (GR).

TFEA correctly identifies GR as the key responding TF from the datasets that most closely capture RNA polymerase initiation (including p300, H3K27ac, and DNA accessibility), and does not identify GR for the transcriptionally repressive mark H3K9me3 (Figure 4.6a)[183, 184]. Surprisingly, the effects of p300 and H3K27ac are seen rapidly, as soon as 5min after dexamethasone treatment. Furthermore, H3K27ac deposition is temporally lagged behind its canonical acetyl-transferase p300[185–187]. Additionally, the enhancer marks H3K4me1 and H3K4me2 show strong enrichment of GR by 30min but the promoter mark H3K4me3 shows only modest enrichment, further supporting the finding that GR binds primarily at enhancers[183] (Supplementary Figure 3.20). Using the diversity of data types and dense time series, we can construct a temporally resolved mechanism of how GR effects changes in transcription (Figure 4.6b and c).

Discussion

We present here transcription factor enrichment analysis (TFEA), a computational method that seamlessly balances the information obtained from differential transcription with the position of a nearby motif, thereby allowing it to be broadly applicable to a variety of datasets that approximate RNA polymerase initiation regions. We show that TFEA outperforms existing enrichment methods when positional data is available and is comparable to these methods in the

absence of positional signal. Further, we show that TFEA, when leveraged with high-resolution time series data, can provide mechanistic insight into the order of regulatory events responding to a perturbation.

A key aspect of TFEA is the incorporation of both positional and differential information in calculating TF motif enrichment. Most motif enrichment algorithms use solely differential information, likely due to the poor positional resolution of historically popular techniques such as ChIP-Seq. Methods such as nascent transcription and CAGE provide higher resolution on the position of RNA polymerase initiation genome-wide. To leverage the improved resolution of these methods, we introduce *muMerge* - a statistically principled way of combining ROIs across replicates and conditions that better captures position and length- scale information as compared to standard merging or intersecting approaches. The presence of improved positional information greatly increases the ability to detect biologically relevant TFs.

Although TFEA makes substantial improvements in detecting which TFs are associated with changes to RNA polymerase in response to perturbations, there are several aspects of this approach that could be improved. First, TFEA inherits some limitations from its dependence on both DESeq and a collection of motifs (see Methods section "Limitations to TFEA and muMerge" for more details). More integral to the enrichment metric, TFEA motif scanning currently requires a fixed cutoff. Future iterations of the method could conceivably eliminate this cutoff, but likely this will substantially increase run times for what may only be minor gains in performance. Finally, sites of transcription initiation (both promoters and enhancers) show substantial GC bias. While we made some effort to account for this bias using linear regression, a more principled approach is desired.

Despite these caveats, TFEA recovers known TF dynamics across a broad range of data types in response to a variety of perturbations. Inevitably, the data type utilized influences the detection ability of TFEA. For example, while CAGE data provides precise resolution on the TSS, it must be deeply sequenced to detect some enhancer associated transcription events[21]. Consequently, TFs that predominantly regulate enhancers will likely be less detectable in poorly

sequenced CAGE data. On the other hand, some methods are more capable of detecting immediate changes in RNA polymerase initiation, such as precision run-on sequencing, allowing for shorter, more refined time points. As demonstrated here, TFEA is able to leverage the information from each data set by incorporating both its distinct positional and differential signal. Applying TFEA to diverse data types, using dense time series, can uncover a detailed mechanistic understanding of the key regulators that enact the cell's dynamic response to a perturbation.

Methods

TFEA

We have developed Transcription Factor Enrichment Analysis (TFEA) to identify transcription factors that demonstrate significant differential activity following a perturbation. It has been observed that, during a perturbation, the binding sites of active transcription factors co-localize with regulatory regions that exhibit strong differential RNA polymerase initiation8. TFEA leverages this observation to calculate an enrichment score that quantifies the co-localization of TF motif instances with sites of altered RNA polymerase activity.

Here we describe in detail the key steps of the TFEA pipeline (shown in Figure 4.1)—specifically, for each TF we describe how the main input (regions of interest—ROIs) are defined, how the ROIs are ranked, and how the enrichment score is subsequently calculated and GC-corrected.

Regions of Interest

One input required for TFEA is a common set of regions of interest (ROIs) on which all experimental samples are evaluated. Each region (consisting of a genomic start and stop coordinate) represents a reference point (the midpoint of the region) and an uncertainty on that reference point (the width of the region).

The biological interpretation of an ROI depends on the nature of the data type being used. However, it is assumed the data being used captures some aspect of RNA polymerase initiation (*e.g.*, CAGE-, Pol II ChIP-, p300 ChIP-, nascent-, or ATAC-seq), to varying degrees of precision, depending on the assay. Specifically, using CAGE data provides a highly precise measure of each

TSS, thus the ROI would be narrow and centered on the TSS. With nascent transcription data, such as PRO-seq or GRO-seq, the position of RNA polymerase loading and initiation (e.g. the midpoint between two bidirectional TSS)81 is often identified8, 122. RNA polymerase II ChIP also informs on the RNA polymerase loading and initiation region, but at lower resolution to nascent transcription data21. Likewise, H3K4me 1/2/3 have been shown to correlate with transcription levels21 but flank the site of initiation81. Finally, as nearly all sites of RNA polymerase loading and initiation originate within open chromatin regions, ATAC-seq data (and related accessibility metrics) are also informative81 but at lower positional precision and with more false positives (open regions without transcription)188.

Regardless of the assay, most methods identify such regions independently in each dataset (*e.g.*, a peak caller for ChIP data or Tfit for identifying sites of bidirectional transcription in nascent data). As a result, these regions will not (and should not) be exactly consistent between samples (*e.g.* some sites are condition specific and, even for shared sites, boundaries may vary). Therefore, a principled method is needed to combine the regions from all the samples into a consensus set.

Defining ROIs with muMerge

In order to combine regions from multiple samples into a consensus set of ROIs, we developed a probabilistic, principled method we call *muMerge*. Initially, *muMerge* was specifically developed for determining the set of consensus RNA polymerase loading and initiation sites observed in nascent sequencing data (by combining bidirectional calls from multiple samples) but it can be applied to peak calls generated from other regulatory data types as well (*e.g.*, ChIP, ATAC, or histone marks).

The basic assumption made by *muMerge* is that each sample is an independent observation of an underlying set of hypothetical loci—where each hypothetical locus has a precise critical point μ, of which the corresponding sample region ($[start, stop]$) is an estimate. We assume the true coordinate of the locus is more likely to be located at the center of the sample region than at

the edges, so *muMerge* represents the sample region by a standard normal probability distribution, centered on the region, whose standard deviation is related to the region width.

To calculate a best estimate (the ROI) for a given locus, *muMerge* calculates a joint probability distribution across all samples from all regions that are in the vicinity of the locus. This joint distribution is calculated by assuming:

1. replicates within a condition are independent and identically distributed (*i.i.d.*)
2. replicates *across* conditions are mutually exclusive (*i.e.*, a sample cannot represent multiple experimental conditions)

Hence *muMerge* computes the product of the normal distributions across all *replicates* within a condition and then sums these results across all *conditions*. The best estimates for the transcription loci μ (there may be multiple) are taken to be the local maxima of this joint distribution—these are the ROI positions. Finally, to determine an updated width, or confidence interval, for each ROI, *muMerge* assumes that the original sample regions whose midpoints are closest to the new position estimate are the most informative for the updated width. Thus the ROI width is calculated by a weighted sum of the widths of the original regions, weighted by the inverse of the distance to each one.

***muMerge* mathematical description:** Principally, *muMerge* makes two probabilistic assumptions about sequence samples:

- **Assumption A:** Replicate samples are independent measurements of *identical experimental conditions* and therefore any corresponding sample regions within them are independent and identically distributed (*i.i.d.*) observations of a common random variable (*i.e.*, the underlying hypothetical locus).
- **Assumption B:** Cross-condition samples are independent measurements of *mutually exclusive experimental conditions* and therefore any sample regions within them are observations of (potentially) disjoint random variables.

These two assumptions inform how *muMerge* accounts for each individual sample, when computing the most likely ROI for any given genomic location (see below for further details).

To start, the inputs to *muMerge* are a set of regions for each sample (genomic coordinates: $\{[start, stop], ...\}$) that represent the sequenced features present in the dataset, as well as an experimental conditions table that indicates the sample groupings (which samples are from which experimental condition). With these inputs, *muMerge* performs the following steps to compute a global set of ROIs:

1. Group overlapping sample regions (each group is processed one at a time)
2. Express each sample region as a positional probability distribution (Eq. IV.1)
3. Generate a joint distribution (Eq. IV.2)
4. Identify local, maximum likelihood ROI positions from the joint distribution
5. Compute ROI widths via weighted sum (Eq. IV.3)
6. Adjust the sizes of overlapping ROIs
7. Record final ROIs for the given group
8. Repeat 2–7 for all remaining groups

Now we describe these steps in detail: First, from the input samples, *muMerge* groups all sample regions that overlap in genomic coordinate (a region is grouped with all other regions it overlaps and, transitively, with any regions overlapping those). We denote a single group of overlapping regions as G_r. This grouping is done globally for all samples, resulting in a set of grouped regions $G = \{G_r\}$, such that every sample region is contained in exactly one grouping G_r (i.e., $G_r \cap G_s = \emptyset$, $\forall\, r \neq s$) (step 1). Then each group of regions, G_r, is processed individually, as the remainder of this section describes (steps 2–7). For a given group, we denote each sample region within it as the 2-tuple $(\mu_k, \sigma_k)_{ij} \in G_r$, where μ_k is the genomic coordinate (base position) of the center of the region and σ_k is the region half-width (number of bases) (shown schematically

in Supplementary Figure 3.3a "Sample Regions"). In the 2-tuple, the indices denote the k-th sample region for replicate j in condition i.

muMerge then processes the regions in G_r as follows. Each region within the group is expressed as a standard normal distribution (ϕ) as a function of base position x,

$$(\mu_k, \sigma_k)_{ij} \rightarrow p_{ij}^{(k)}(x) = \phi\left(\frac{x - \mu_k}{\rho\, \sigma_k}\right) \tag{IV.1}$$

where ρ is the "width ratio"— the ratio of the half-width sample region to the standard deviation of the normal distribution—with a default of $\rho = 1$ (user option) (shown schematically in Figure 4.2a and Supplementary Figure 3.3b "Generate probability distribution"). This distribution represents the probability of the location for the underlying hypothetical locus (μ), of which $(\mu_k, \sigma_k)_{ij}$ is an estimate. For those samples with no regions within G_r, the probability distribution is expressed as a uniform, $p_{ij}^{(k)}(x) = 1/\Delta$ where Δ is the full range encompassed by the overlapping sample regions. In other words, we assume that if the sample contains no data to inform the location of the underlying loci at that location, then all positions are equally likely for that sample. *muMerge* then calculates a joint distribution ($\mathcal{P}_{joint}(x \mid p_{ij})$) by combining all $p_{ij}^{(k)}(x)$ for the group as follows:

$$\mathcal{P}_{joint}(x \mid p_{ij}) = \sum_i \left(\prod_j \left(\sum_k p_{ij}^{(k)}(x) \right) \right) \tag{IV.2}$$

Here we are calculating the product of the replicate distributions (index j—those within a given experimental condition), consistent with our probabilistic assumption A, and the sum of the resulting distributions across experimental conditions (i index), consistent with our probabilistic assumption B (shown schematically in Figure 4.2a and Supplementary Figure 3.3c "Calculate joint probability"). Examples of how $\mathcal{P}_{joint}$ would be calculated for a given experimental set-up are given in Supplementary Figure 3.21. Though this function is not a normalized probability distribution, we are only interested in relative values of $\mathcal{P}_{joint}(x \mid p_{ij})$. Specifically, we are interested in the maxima of this function. We identify the set of maxima (which we denote $\{\widehat{\mu}_k\}$) and rank them by the function value for each position, $\mathcal{P}_{joint}(x = \widehat{\mu}_k \mid p_{ij})$. We then keep the top $M + 1$ from the ranked set, where M is the median number of regions per sample in G_r (user

option). This is our final set of estimates on the hypothetical loci positions, μ—*i.e.*, the positions of our ROIs for group G_r.

For each $\widehat{\mu}_k$, we then calculate a width for the resulting ROI. We do so for each by calculating a weighted sum over the set of all original sample regions in the group, $\{(\mu_k, \sigma_k)_{ij}\}$, weighted by the inverse of the distance from the final position estimate to each μ_k (shown in Supplementary Figure 3.3d "width estimation"). Thus the final ROI half-width, $\widehat{\sigma}_k$, is calculated as follows:

$$\widehat{\sigma}_k = \sum_i \frac{\sigma_i}{|\widehat{\mu}_k - \mu_i| + 1} \Bigg/ \sum_i \frac{1}{|\widehat{\mu}_k - \mu_i| + 1} \tag{IV.3}$$

where i indexes all sample regions in the group $G_r = \{(\mu_k, \sigma_k)_{ij}\}$. Our rationale is that the width of those sample regions that are closer to the ROI position $\widehat{\mu}_k$, are more informative for the ROI width and therefore are given a larger weight. This results in a set of ROIs $\{(\widehat{\mu}_k - \widehat{\sigma}_k,\ \widehat{\mu}_k + \widehat{\sigma}_k)\}$ (shown in Supplementary Figure 3.3e "Final region estimate").

Finally, we determine if there is overlap between any of the regions in this set of ROIs. If so, any two overlapping regions are reduced in size, symmetrically about their centers, until they no longer overlap. This is done so that any genomic position can be uniquely associated with an ROI. The final ROIs for the group are then written to an output file to be used downstream in the pipeline. This process is repeated for all groups of overlapping sample regions (i.e., $\forall\, G_r \in G$).

Ranking ROIs

With a set of ROIs identified, the next step is to rank them by differential signal. Because the goal of TFEA is to identify transcription factors that are responding to a perturbation, a ranking based on the differential transcription at the ROIs would capture the regulatory behavior of the TF. Technically, the signal in each data type actually represents different biological processes—differential transcription for nascent (PRO-seq or GRO-seq), differential accessibility (DNAse or ATAC-Seq), and differential occupancy for ChIP. Logically, we assume each is a reasonable proxy for differential transcription. There are a number of ranking metrics one could use that are based on these differential signals—for example, difference in coverage, log-fold change, or a differential significance (p-value). For TFEA, we chose to rely on a well-established

tool (DESeq) to perform our ranking, since it was designed to model the statistical variation found in sequencing data88.

For a set of ROIs, TFEA calculates read coverage for each replicate and condition using *bedtools multibamcov* (version 2.25.0)189. TFEA then inputs the generated counts table into DESeq2 (v 1.26)88 (or DESeq (v 1.38)137 if no replicates are provided) to obtain differential read coverage for all ROIs. By default, these regions are then ranked by the DESeq computed p-value, separated by positive or negative log-fold change (alternative user option to rank the ROIs purely by fold-change). In other words, the ROIs are ranked from the most significant positive fold-change to the most significant negative fold-change.

Identifying locations of motif instances

Accurately identifying the locations of motif instances relative to each ROI is a critical step in the TFEA pipeline. By default TFEA uses the motif scanning method FIMO, which is a part of the MEME suite (version 5.0.3)190. FIMO represents each TF by a base-frequency matrix and uses a zero-order background model to score each position of the input sequences. For each ROI, we scan the 3kb sequence surrounding the ROI center ($\widehat{\mu_i} \pm 1.5$kb). This 3kb window was chosen primarily to reduce computation time and is also consistent with the window used for the MD-Score method8. For each TF, we utilize a scoring threshold of 10^{-6} and keep the highest scoring position (denoted m_i), in the event more than one motif instance is identified. If no position score above the threshold, then no m_i is recorded for the ROI. Our background model is determined by calculating the average base frequency over all ROI. For this paper, we use the frequency-matrices from the HOCOMOCO database172 with a default psuedo-count of 0.1.

Enrichment Score

With the motif instances identified for each of the ranked ROIs, we now detail how TFEA calculates the enrichment score ("E-Score"—in Figure 4.1) for each transcription factor. The procedure for calculating enrichment requires two inputs:

1. N-tuple ordered list ($\widehat{\mu}_i$)—the genomic coordinates for reference points, assumed to be the centers of all ROIs (*e.g.*, consensus ROIs calculated by *muMerge*), ranked by DESeq p-value (separated by the sign of the fold-change).

2. Ordered list (m_i)—the genomic coordinates of each max-scoring motif instance (*e.g.*, motif locations generated by scanning with FIMO), for each ROI.

We first calculate the motif distance d_i for each ROI—the distance from each $\widehat{\mu}_i$ to the highest scoring motif instance m_i within 1.5kb of $\widehat{\mu}_i$. If no m_i exists within 1.5kb, then d_i is assigned a null value (Ø) (Eq. IV.4).

$$d_i = \begin{cases} |\widehat{\mu}_i - m_i|, & \text{if } m_i \text{ is present} \\ \text{Ø}, & \text{if } m_i \text{ is not present} \end{cases} \tag{IV.4}$$

We use the distribution of these distances to calculate a weighted contribution to the E-score for each motif instance. In previous work, it has been observed that the distribution of motif position relative to sites of RNA polymerase initiation decays rapidly with increased distance8. Thus we have chosen to model the motif weights with an exponential function, whose decay length is independently determined for each transcription factor, from the background motif distribution. In order to compute the weight model, we next calculate the background distribution of motif distances. We assume the majority of the ROIs experience no significant fold-change—namely, those ROIs in the middle of the ranked list. Consequently, we calculate the mean, background motif distance (Eq. IV.5) for those ROIs whose rank is between the first and third quartiles of the ordered list of ROI positions, ($\widehat{\mu}_i$), as follows

$$\bar{d} = \text{mean}\{d_i \mid \forall\, i, \text{if } Q_1 \leq i \leq Q_3 \text{ and } d_i \neq \text{Ø}\} \tag{IV.5}$$

where Q_1 and Q_3 are the first and third quartiles, respectively. Our assumption is that the inter-quartile range of the ordered list ($\widehat{\mu}_i$)—between indices Q_1 and Q_3—represents the background distribution of motif distances for the given transcription factor, and therefore defines the weighting scale for significant ROIs in our enrichment calculation. We found this to be

essential since the background distribution varies between transcription factors. This variation in the background can be attributed to the random similarity of a given motif to the base content surrounding the center of ROIs. For example, in the case of RNA polymerase loading regions identified in nascent transcription data (which demonstrate a greater GC-content proximal to μ as compared to genomic background8), GC-rich transcription factor motifs were more likely to be found proximal to each ROI by chance and thus resulted in a smaller $\bar{d}$ than would be the case for a non-GC-rich motif.

Having calculated the mean background motif distance, we proceed to calculate the enrichment contribution (*i.e.*, weight—Eq. IV.6) for each ROI in the ordered list (see "Weight Calculation" in Figure 4.1).

$$w_i = \begin{cases} e^{-d_i/\bar{d}}, & \text{if } d_i \neq \emptyset \\ 0, & \text{if } d_i = \emptyset \end{cases} \tag{IV.6}$$

In order to calculate the E-Score, we first generate the enrichment curve for the given TF (solid line in "Enrichment Curve" in Figure 4.1) and the background (uniform) enrichment curve (dashed line in "Enrichment Curve" in Figure 4.1). We define the E-Score as the integrated difference between these two (scaled by a factor of 2, for the purpose of normalization). The enrichment curve (Eq. IV.7), which is the normalized running sum of the ROI weights, and the E-Score (Eq. IV.8) are calculated as follows:

$$e(i) = \frac{\sum_{k=0}^{i} w_k}{\sum_{k=0}^{N} w_k} \tag{IV.7}$$

$$E = \frac{2}{N} \sum_{i} \left(e(i) - \frac{i}{N} \right) \tag{IV.8}$$

where i is the index for the ROI rank and i/N represents the uniform, background enrichment value for the ith of N ROIs. The background enrichment assumes every ROI contributes an equal weight w_i, regardless of its ranking position. Therefore, the enrichment curve (Eq. IV.7) will deviate significantly from background if there is correlation between the weight and ranked position of the ROIs. In this case, the E-Score will significantly deviate from zero, with $E > 0$ indicating either increased activity of an activator TF or decreased activity of a repressor TF.

Likewise, $E < 0$ indicates either a decrease in an activator TF or an increase in a repressor TF. By definition, the range of the E-Score is -1 to $+1$.

Unlike GSEA, which uses a Kolmogorov–Smirnov-like statistic to calculate its enrichment score87, the TFEA E-Score is an area-based statistic. GSEA was designed to identify if a predetermined, biologically related subset of genes is over-represented at the extremes of a ranked gene list. Therefore, the KS-like statistic is a logical choice for measuring how closely clustered are the elements of the subset, since it directly measures the point of greatest clustering and otherwise is insensitive to the ordering of the remaining elements. Conversely, because TFEA's ranked list does not contain two categories of elements (the ROIs) and all elements can contribute to the E-Score, we wanted a statistic that was sensitive to how all ROI in the list were ranked—for this reason, we chose the area-based statistic. The null hypothesis for TFEA assumes all ROI contribute equally to enrichment, regardless of their motif co-localization and rank. Hence the uniform background curve, to which the enrichment curve is compared.

In order to determine if the calculated E-Score (Eq. IV.8) for a given transcription factor is significant, we generate a E-Score null distribution from random permutations of $(\widehat{\mu}_i)$. We generate a set of 1000 null E-Scores $\{E'_i\}$, each calculated from an independent random permutation of the ranked ROIs, $(\widehat{\mu}_i)$. Our E-Score statistic is zero-centered and symmetric, therefore we assume $\{E'_i\} \sim \mathcal{N}(E_0, \sigma_E^2)$. The final E-Score for the transcription factor is compared to this null distribution to determine the significance of the enrichment.

Prior to calculating the E-Score p-value, we apply a correction to the E-Score based on the GC-content of the motif relative to that of all other motifs to be tested (user configurable). This correction was derived based on the observation that motifs at the extremes of the GC-content spectra were more likely to called as significant across a variety of perturbations. We calculate the E-Scores for the full set of transcription factors as well as the GC-content of each motif,

$\{(g_i, E_i)\}$. We then calculate a simple linear regression for the relationship between the two,

$$\widehat{b} = \bar{E} - \widehat{m}\bar{g} \tag{IV.9}$$

$$\widehat{m} = \frac{\sum_{i=1}^{n}(g_i - \bar{g})(E_i - \bar{E})}{\sum_{i=1}^{n}(g_i - \bar{g})} \tag{IV.10}$$

$$E_{GC}(g) = \widehat{b} + \widehat{m}g \tag{IV.11}$$

where $\bar{E}$ and $\bar{g}$ are the average E-Score and average GC-content. $E_{GC}(g)$ is the amount of the E-Score attributed to the GC-bias for a motif with GC-content g. Thus the final E-Score for the transcription factor is given by $E_{TF} = E - E_{GC}(g_{TF})$, the difference between Eq. IV.8 and IV.11. If GC-content correction is not performed, then Eq. IV.8 is taken to be the final E-Score. The p-value for the final TF E-Score is then calculated from the Z-score, $Z_{TF} = (E_{TF} - E_0)/\sigma_E$.

Limitations to TFEA and muMerge

Though *muMerge* and TFEA clearly demonstrate good performance, there are a number of limitations to both tools. We want to bring attention to these limitations so that users may better understand how best to apply these tools to data and interpret the results.

As implemented, *muMerge* assumes every input data set is of equal quality, by default. This means every data set is given equal weight when computing the joint probability. However, if some data sets are of low or questionable quality such that they have inaccurate bed regions, this may bias the ROI inferred by *muMerge*. We recommend removing poor quality data sets from those input to *muMerge* or weighting each data set based on its perceived quality. In short, *muMerge* cannot substitute for thoughtful quality control of each of one's samples.

Additionally, sites with regions that are very closely spaced tend to be inferred as a single ROI by *muMerge*. This can be accounted for somewhat by decreasing the value of the width ratio (ρ, default value 1), which reflects the assumed uncertainty on the location of the input sample regions. However, there is no definitive ground truth value. It should be noted, both of these limitations also apply to the *bedtools* methods of combining regions.

As implemented, TFEA depends on DESeq to order the observed transcription changes between conditions. Consequently, TFEA performs best when replicates are available. Likewise,

when DESeq assumptions are violated, this can result in unreliable region ordering. For example, when transcription factor ChIP is utilized across conditions there are often large gains in binding events. This is particularly true with environmentally stimulated TFs such as p53 or GR which can be activated by Nutlin-3a and dexamethazone, respectively. The pre-stimulated condition typically has few (if any) detectable binding sites whereas post stimulation binding is detected at hundreds to thousands of sites. In this scenario, the DESeq assumption that the bulk of sites are unchanged is violated. Even if an alternative method of ordering sites were utilized, most gained sites contain the TF motif, so no enrichment is typically observed. For this reason, we do not recommend applying TFEA to environmentally responsive transcription factor ChIP data sets or any other data type that clearly violates the statistical assumptions of DESeq.

Additionally, TFEA depends on a collection of known motifs. Unfortunately, some TFs have no known motif or one of poor quality. However, over time, the quality and numbers of TFs in the major databases have dramatically improved144. Furthermore, TFEA can only distinguish between paralogous transcription factors to the extent that they have distinct motifs. Sites of transcription initiation (both promoters and enhancers) show substantial GC bias. Consequently, short high GC content motifs, which are exceedingly common in ROIs, sometimes appear to show significant changes with a perturbation. The extent to which these signals represent a biological process or a statistical anomaly is unclear.

Finally, TFEA identifies when a TF motif associates with sites of changing RNA polymerase initiation. By ordering the differential transcription signal by the direction of change, TFEA can determine whether the identified TF is associated with a transcription gain or loss. Prior work has shown that stimulation of an activator gives rise to increased eRNA activity nearby 61, 83, 159, but loss of a repressor also leads to proximal increased eRNA activity191. Consequently, if a motif associates with transcriptional gain it may arise from either activation or repression of the TF.

Benchmarking

In order to benchmark the performance of *muMerge* and TFEA, we performed a number of simulations that isolate the different parameters of *muMerge* and TFEA, comparing the performance to that of some commonly used alternatives. We ran these alternatives: AME 5.0.5 and bedtools version 2.28.0 (*merge and intersect*) using default parameters. Here we describe how the data for each test was generated.

muMerge*: Simulating replicates for calculation of ROIs*

To test the performance of *muMerge* in a principled manner, we first generate replicate data in a way that simulates the uncertainty present in individual samples. For each replicate, we perform 10,000 simulations of sample regions for a single locus, and calculate the average performance. For each simulation we assume a precise position and width for the hypothetical locus and model the uncertainty of each sample region with a binomial and Poisson distribution, respectively. The position of each sample region, μ_i, is pulled from a symmetric binomial distribution $\mu_i \sim B(n = 100, p = 0.5)$, centered at zero. The half-width of each sample region, σ_i, is pulled from a Poisson distribution $\sigma_i \sim Pois(\lambda = 100)$. The specific distributions utilized to generate the sample regions are as follows:

$$\text{locus estimate} \equiv \begin{cases} \text{position:} & \mu_i \sim \mu + B(n = 100, p = 0.5) - np \\ \text{half-width:} & \sigma_i \sim Pois(\lambda = 100) \end{cases} \tag{IV.12}$$

Here $B(\cdot)$ is the binomial distribution centered at np with success probability 0.5 and variance $np(1 - p) = 25$. Thus, the position estimator μ_i for a single sample region is centered at μ. $Pois(\cdot)$ is the Poisson distribution, thus, the half-width for each sample region have mean and variance of $\lambda = 100$.

The first test (Supplementary Figure 3.4a) consisted of inferring a single locus (located at $\mu = 0$) from an increasing number of replicates. A sample region for each replicate was generated from Eq. IV.12. This simulation was repeated 10,000 times for each number of replicates being combined. The methods *muMerge*, *bedtools merge* and *bedtools intersect* were applied to each of

the 10,000 simulations. The average error on the midpoint (its deviation from the true locus position, $\mu = 0$) and region width were calculated for the regions generated from each method, averaged over all 10,000 simulations. The behavior of the average positional error and region width as a function of number of combined replicates is shown in Figure 4.2b (Test 1).

The second test (Supplementary Figure 3.4b) consisted of inferring two loci ($\mu_1 = -x$ and $\mu_2 = +x$) as the distance between those loci was increased (from $x = 0$ to 200). This simulation was repeated 10,000 times for each value of x (with 3 replicates). The distribution of the inferred positions and widths were plotted, using *muMerge*, *bedtools merge* and *bedtools intersect*. The distribution of positions and widths as a function of the distance between μ_1 and μ_2 are shown in Figure 4.2c (Test 2).

TFEA: Simulated motif enrichment

To generate test sequences for understanding the contribution of positional signal to motif enrichment, we randomly sampled 10,000 sequences from detected bidirectionals in untreated HCT116 cells 61. As this collection of ROI was obtained from nascent transcription data, it maintains true biological sequence signals. To simulate differential transcription, we randomly ordered the set of ROI. We then embedded instances of the TP53 motif in the highest ranked sequences with a normal distribution with $\mu = 0$ and $\sigma = 150$ (representative of signal). Importantly – p53 is known to NOT be activated in HCT116 DMSO samples61. To simulate background noise, we embedded instances of the TP53 motif with a uniform distribution to a percentage of the remaining sequences (chosen randomly). To calculate an F1 score, for each scenario of varying signal to background we generated 10 simulations. We then calculated the harmonic mean of precision and recall with the aggregate p-values of all 10 simulations measuring all 401 TF motifs within the HOCOMOCO database (total 4010 TF motifs). True positives, in this case, were the 10 instances of the TP53 motif that should be significantly enriched. Any other significantly enriched TF motifs were considered false positives. We performed two sets of tests: 1) varying the amount of motif signal relative to the amount of background and 2) varying the

standard deviation of motif position in the highest signal tested (10% signal; with the last scenario being uniform signal distribution) and the amount of background.

TFEA: Testing compute performance

The base (ATGC) content of regulatory regions was calculated from the sites of RNA polymerase initiation inferred in HCT116 DMSO (using Tfit; described in 8). One million 3kb sequences were subsequently generated based on the empirical probability of the positional base composition. We then randomly sampled (without replacement) an increasing number of sequences (up to 100,000) to be used in the computational processing tests. Run time and compute resources were measured using the Linux *time* command on a single node of a 70-node mixed-platform high-memory compute cluster running CentOS 7.4. To compute the runtime for a single processor, we added the *systime* and *usertime*. To compute memory usage for a single processor, we reran TFEA using only a single processor.

PRO-Seq in MCF10A

We generated PRO-seq libraries for MCF10A cells before (DMSO) and 1 hour after Nutlin-3a, described in detail below. Our MCF10A cells carried a WTp53 insertion at the p53 locus, as they were developed for a study of p53 isoforms104. A complete description of the cell line construction is provided here for completeness.

Cas9RNP formation: sgRNA was formed by adding tracrRNA (IDT cat# 1072533) and crRNA (TP53 exon 2, positive strand, AGG PAM site, sequence: GATCCACTCACAGTTTCCAT) in a 1:1 molecular ratio together and then heating to 95°C and then allowing to slowly cool to room temperature over 1 hour. Cas9RNP was then formed by adding purified Cas9 protein to sgRNA at a ratio of 1:1.2. 3.7μL of purified Cas9 protein at 32.4μM was added to 2.9μL of 50μM sgRNA. This was then incubated at 37°C for 15 minutes, and used at 10μM concentration within the hour.

Donor Plasmid Construction: Vector Builder was used to construct plasmid. Insert was flanked by 1.5kb homology arms, and mCherry was inserted as a selection marker.

CRISPR/Cas9 Genome Editing: MCF10A cells cultured in DMEM/F12 (Invitrogen #11330-032) media containing 5% horse serum (LifeTech #16050-122), 20ng/mL EGF ((Peprotech #AF-100-15), 0.5μg/mL Hydrocortisone (Sigma #H0888-1g), 100ng/mL Cholera toxin (Sigma #C8052-2mg), 10μg/mL insulin (Sigma #I1882-200mg), and 1x Gibco 100x Antibiotic-Antimycotic (Fisher Sci, 15240062) penicillin-streptomycin. Cells were split 24 hours prior to experiment and grown to approximately 70% confluency on a 15cm plate. Media was aspirated, and the cells were washed with PBS. 4ml of trypsin per plate were used to harvest adherent cells, after which 8mL of resuspension medium (DMEM/F12 containing 20% horse serum and 1x pen/strep) was added to each plate to neutralize the trypsin. Cells were collected in a 15ml centrifuge tube and spun down at 1,000xg for 5 minutes, then washed in PBS and spun down again at 1,000xg for 5 minutes. Cells were counted using a hemocytometer and $5x10^5$ cells were put in individual 1.5mL eppedorph tubes for transfection. Cells were re-suspended in 4.15μL Buffer R, 10μM Cas9RNP (6.6μL), 1μg WTp53 donor plasmid (1.25μL). Mixture was drawn up into a 10μL Neon pipet tip, electroporated using the Neon Transfection Kit with 10μL tips (1400V, 20ms width, 2 pulse). Transfected cells were then pipetted into 2mL of antibiotic free media. After 1 week of recovery, cells were then single cell sorted into 96 well plate based on mCherry expression. Clones were then verified with sequencing, PCR, and western blot.

Replicates A single validated clone of MCF10A WTp53 cells was selected for subsequent PRO-seq analysis. All experiments were conducted in duplicate from separate cell growths.

Nuclei Preparation: MCF10A WTp53 cells were seeded on three 25cm dishes (1x107 cells per dish) for each treatment 24 hours prior to the experiments (70% confluency at the time of the experiment). Cells were treated simultaneously with 10μM Nutlin-3a or 0.1% DMSO for 1 hour. After treatment, cells were washed 3x with ice cold PBS, and then treated with 10 ml (per 15 cm plate) ice-cold lysis buffer (10 mM Tris–HCl pH 7.4, 2 mM MgCl2, 3 mM CaCl2, 0.5% NP-40, 10% glycerol, 1 mM DTT, 1x Protease Inhibitors (1mM Benzamidine (Sigma B6506-100G), 1mM Sodium Metabisulfite (Sigma 255556-100G), 0.25mM Phenylmethylsulfonyl Fluoride

(American Bioanalytical AB01620), and 4U/mL SUPERase-In). Cells were centrifuged with a fixed-angle rotor at 1000xg for 15 min at 4°C. Supernatant was removed and pellet was resuspended in 1.5 mL lysis buffer to a homogenous mixture by pipetting 20-30X before adding another 8.5 mL lysis buffer. Suspension was centrifuged with a fixed-angle rotor at 1000xg for 15 min at 4°C. Supernatant was removed and pellet was resuspended in 1 mL of lysis buffer and transferred to a 1.7 mL pre-lubricated tube (Costar cat. No. 3207). Suspensions were then pelleted in a microcentrifuge at 1000xg for 5 min at 4°C. Next, supernatant was removed and pellets were resuspended in 500 μL of freezing buffer (50 mM Tris pH 8.3, 40% glycerol, 5 mM MgCl2, 0.1 mM EDTA, 4U/ml SUPERase-In). Nuclei were centrifuged 2000xg for 2 min at 4°C. Pellets were resuspended in 100 μL freezing buffer. To determine concentration, nuclei were counted from 1 μL of suspension and freezing buffer was added to generate 100 μL aliquots of $10x10^6$ nuclei. Aliquots were flash frozen in liquid nitrogen and stored at -80°C.

Nuclear run-on and RNA preparation: Nuclear run-on experiments were performed as described60 with the following modifications: the final concentration of non-biotinylated CTP was raised from 0.25 μM to 25 μM, a clean-up and size selection was performed using 1X AMPure XP beads (1:1 ratio) (Beckman) prior to test PCR and final PCR amplification, and the final library clean-up and size selection was accomplished using 1X AMPure XP beads (1:1 ratio) (Beckman).

Sequencing: Sequencing of PRO-Seq libraries was performed at the BioFrontiers Sequencing Facility (UC-Boulder). Single-end fragment libraries (75 bp) were sequenced on the Illumina NextSeq 500 platform (RTA version: 2.4.11, Instrument ID: NB501447), demultiplexed and converted BCL to fastq format using bcl2fastq (bcl2fastq v2.20.0.422); sequencing data quality was assessed using FASTQC (v0.11.5) and FastQ Screen (v0.11.0), both obtained from https://www.bioinformatics.babraham.ac.uk/projects/. Trimming and filtering of low-quality reads was performed using BBDUK from BBTools (v37.99) (Bushnell, n.d.) and FASTQ-MCF from EAUtils (v1.05) 192.

Data Processing

p53 ChIP data: Raw ChIP-seq data (GSE86222) from Andrysik et al. was downloaded from the SRA database 69. Data was processed with the ChIP-Flow pipeline (https://github.com/Dowell-Lab/ChIP-Flow) as follows. Reads were trimmed using BBduk from BBMap version 38.05 with the following flags ktrim=r qtrim=10 k=23 mink=11 hdist=1 maq=10 minlen=20 126. Trimmed reads were mapped to the human reference genome (GRCh38/hg38) using HISAT2 version 2.1.0 with the –very-sensitive and –no-spliced-alignment flags 193. Next, SAMtools version 1.8 194 was used to convert sam files to sorted bam files, and duplicate reads were removed with Picard Tools version 2.6.0 195. Finally, MACS2 version 2.1.1 was used to call peaks using each of the input samples for each cell line as control 196.

ENCODE data: Raw bed and bam files were downloaded directly from ENCODE (encodeproject.org). These files were input directly into the *muMerge* or TFEA pipeline for processing and analysis. AME analysis was performed on the ranked ROI list produced as an optional output from TFEA.

***muMerge* TF ChIP-seq comparison:** Peak calls for each region were scanned for an instance of the TF motif (from HOCOMOCO) using FIMO (MEME version 5.1.1), and peaks with significant hits to the TF motif (p-adj < 0.001) were retained 190. Sample regions were combined across replicates (cell types) and conditions (with or without Nutlin-3a) with *muMerge*, *bedtools merge* and *bedtools intersect* (bedtools version 2.28.0) 189. The point of interest for *muMerge* was the center of the called peak, which was expanded by 1500bp to specify the full ROI. Distance to the motif instance was calculated using the region midpoint compared to the midpoint of the best motif instance. For each method, we report the standard deviation, mean and median of distances for each region.

GRO/PRO-Seq data: All GRO-Seq and PRO-Seq data were processed using the Nextflow197 NascentFlow pipeline (v1.1 198) specifying the '–tfit' flag. Subsequent Tfit bed files from all samples were combined with *muMerge* to obtain a consensus list of ROIs.

FANTOM data: Raw expression tables for the Macrophage LPS time series were downloaded using the table extraction tool (TET) from the FANTOM Semantic catalogue of Samples, Transcription initiation, And Regulations (SSTAR; http://fantom.gsc.riken.jp/5/sstar/Macrophage_response_to_LPS). Because the annotations for regions within hg38 counts tables contained "hg19", we performed this analysis in the hg19 genome with the hg19 counts table instead of the hg38 counts table. We then performed DESeq analysis (since there were no replicates) on each time point compared to control and ranked the annotated regions within the counts table similar to Figure 4.1. We then ran TFEA and AME with default settings on each of the three donors. We displayed only data for donor 2, as this sample had the most complete time series data.

Clustering FANTOM data: We retained TFs with at least 15 significant (p-adj < 0.1) time points (representing 2/3 of all timepoints) from the TFEA output and applied K-means clustering. Clustering of the time series data was performed on the first two hours only, in order to distinguish the early responses to LPS infection. K-means clustering was conducted using the Hartigan and Wong algorithm with 25 random starts and 10 iterations for $k = 3$ 199. The optimal number of clusters was selected using the Elbow method 200.

String database analysis: Protein names from TFs that were found to be significant in at least 15 time points were taken from the HOCOMOCO database. These proteins were inputted directly into the String database (https://string-db.org). Clusters were formed by selecting the MCL clustering option with an inflation parameter of 3 (default). Network edges were selected to indicate the strength of the data support. Finally, nodes disconnected from the network were hidden.

Data Availability

We generated PRO-seq libraries for MCF10A cells with and without Nutlin-3a. MCF10A PRO-seq data generated for this study is available in GEO with accession numbers GSE142419. Additionally, a number of publicly available data sets were utilized and analyzed. These data sets are available in the Short Read Archive (SRA) or ENCODE repository with accession numbers presented in Supplemental Data 1. Additional supporting data for figures is available at the Open Science Framework201.

Code Availability

TFEA is available for download at https://github.com/Dowell-Lab/TFEA and comes with muMerge integrated. Alternatively, *muMerge* can be downloaded independently at https://github.com/Dowell-Lab/mumerge. Definition files for Singularity and Docker containers are available in the TFEA GitHub repository. Usage of these containers is recommended to simplify dependency management. Additionally, TFEA can be utilized through the web interface at https://tfea.colorado.edu/. Finally, all data analysis conducted in preparation of this manuscript is available in Juypter notebook format at:

https://github.com/Dowell-Lab/TFEA/tree/master/Jupyter_Notebooks

Acknowledgments

This work was funded in part by a National Science Foundation (NSF) ABI grant number 1759949, a National Institutes of Health (NIH) grant RO1 GM125871, and an NIH training grant T32 GM008759. We acknowledge the BioFrontiers Computing Core at the University of Colorado Boulder for providing high-performance computing resources (NIH 1S10OD012300) supported by BioFrontiers' IT. In particular we thank Matt Hynes-Grace, Jon DeMasi, and Ethan Kern for assistance in the development of the TFEA website. Finally, we also thank the BioFrontiers Institute Next-Gen Sequencing Core and the Biochemistry Shared Cell Culture Facility for their invaluable contributions to this study.

Author Contributions

J.D.R. and J.T.S. developed algorithms, analyzed data, and wrote initial manuscript; R.F.S. and Z.L.M. assisted in data analysis; C.B.L. performed experiments; J.W. assisted in designing figures; D.J.T. consulted on methodology and analysis; R.D.D. conceived the study; All authors discussed the results, their implications, and commented on the manuscript at all stages.

Competing Interests

Dr. Dowell is a founder of Arpeggio Biosciences. The other authors declare no competing interests.

CHAPTER V

REGULATORY NETWORK INFERENCE USING NASCENT RNA SEQUENCING DATA

Abstract

Gene transcription is controlled and modulated by enhancers and promoters. These regions are abundant in unstable, non-coding bidirectional RNA transcripts. Using nascent RNA transcription data across hundreds of samples, we identified ~ 500,000 regions of bidirectional transcription and linked these regions to gene targets. The identified correlated pairs, a bidirectional region and a gene, are enriched for diseases associated SNPs and often supported by independent 3D data. We present these resources as an SQL database which serves as a resource for future studies into gene regulation, enhancer associated RNAs, and transcription factors.

This work is in preparation as: Rutendo F. Sigauke, Lynn Sanford, Taylor Jones, Zachary L. Maas, Mary A. Allen and Robin D. Dowell. Atlas of nascent RNA transcripts reveals high confidence enhancer associated bidirectionals linked with genes across different tissue types. In this work, Dr. Lynn Sanford developed the SQL database backend and generated the cell type specific SPEC analysis, Taylor Jones conducted the motif scanning, Zachary L. Maas developed the methods for finding nascent data sets within the short read archive, Mary A. Allen assisted in the p53 analysis, and Dr. Robin Dowell supervised and cowrote the paper draft. Meta-data curation was assisted by the entire Dowell and Allen labs. I am responsible for the remaining analysis.

Introduction

The transcription process is conserved across organisms and it dictates which genes are expressed in a given cell state and time point [1, 2]. The well-organized regulatory process involves transcription factors (TFs) that bind to DNA at enhancer and/or promoter regions and recruit RNA-polymerase II (RNAPII) to the region nearby, thereby facilitating the transcription of the gene [1, 202]. Genome wide association studies have identified numerous variants (mostly single nucleotide polymorphisms, or SNPs) which typically reside within enhancers and promoter regions [203]. Transcription regulation is, in general, a context-specific process, requiring the

region to not only be accessible but also bound by particular transcription factors. Thus, understanding when genes are transcribed in specific cellular and temporal contexts can aid in understanding disease states. Shedding light on the potential impacts of enhancer regions on gene transcription can inform studies on transcription dysregulation in disease.

Many efforts to annotate regulatory regions have been made using ChIP-seq, pulling down on either transcription factors or related histone modifications. In particular, specific histone modifications have been associated with active or repressed [12, 204, 205]. However, these ChIP-seq experiments are limited in utility for understanding transcription regulation because they lack dynamic resolution. At short time points, histone marks rarely change even when nascent transcription assays suggest transcription changes have already happened [8]. Furthermore, the resolution in nucleotides is relatively low for ChIP – with histone peaks often measuring a kilobase or larger. An alternative to ChIP-seq data is Cap Analysis of Gene Expression (CAGE) which captures 5' caps present on RNA, including both mRNA and enhancer associated RNAs [18, 19, 79, 206, 207]. However, CAGE is biased to highly transcribed RNAs and more stable transcripts, and is therefore less robust relative to nascent transcription at detecting lowly transcribed regions [21, 208].

Another open problem in the field of transcription regulation is linking enhancers to their target genes [23]. To date, most attempts to link these entities relies on genome position (proximity), sequence signatures (such as motif patterns), and rarely 3D interaction data. Typically these approaches combine histone ChIP-seq data, TF ChIP-seq, accessibility data, 3D interactions, RNA-seq, and/or CAGE data [19, 27, 209, 210]. However, each of these assays has distinct limits on temporal dynamics or spatial resolution that muddles the identification of direct regulatory linkages.

Nascent transcription is a promising additional piece of information in the construction of gene regulatory networks. Nascent transcription response to perturbations is nearly immediate [211] and sites of RNA polymerase II initiation can be resolved to nearly base pair resolution [81]. Additionally, using CAGE data as a proxy for nascent, it was shown that eRNAs and their target

genes are correlated in transcription levels [19], which was later shown to also be true in nascent data [8]. This suggests that patterns of correlation maintained over large collections of nascent transcription data could be used to potentially identify enhancer to target gene pairings. This was essentially recently confirmed by Lidschreiber et al. who used a small collection of cancer cell lines to identify correlated pairs (enhancer to gene) [57]. In their analysis, they restricted pairs to those within 500kb of each other and compared these linkages to the simple neighboring gene approach [57].

Unaware of the Lidschreiber effort, we set out to catalog a large collection of nascent transcription data sets which could be used to both identify sites of bidirectional transcription and identify correlated enhancer to gene pairs. To this end, we present dbNascent, a catalog of manually annotated 2880 nascent RNA sequencing samples across 20 organisms. These data have been processed using standardized analysis pipelines. Annotation of bidirectional transcripts from high-quality human and mouse samples shows that they overlap previous enhancer and promoter annotations. They are also enriched for disease-associated variants. A comparison of RNA transcript classes (genes, lncRNAs, and enhancer associated RNAs) indicates that bidirectionals not previously annotated are far more tissue-specific than either genes or lncRNA transcripts. Building from previous methods, we identify bidirectional and gene transcripts that are correlated in transcription across all the samples in human data by tissues. Considering that some enhancers can be found at long distance ranges from their targets, we explore pairs that are up to 1Mb in distance [212, 213]. Using p53 as a case study, we were able to assign responsive enhancers to their target gene, thereby linking nearly all genes that respond to p53 to relevant enhancer loci. Thus we present an expansion of our knowledge of the regulatory network and a tool for hypothesis generation.

Results

A repository of nascent RNA data

We began by constructing a large repository of previously published nascent transcription data sets (Figure 5.1A). To this end, nascent RNA sequencing experiments were manually curated

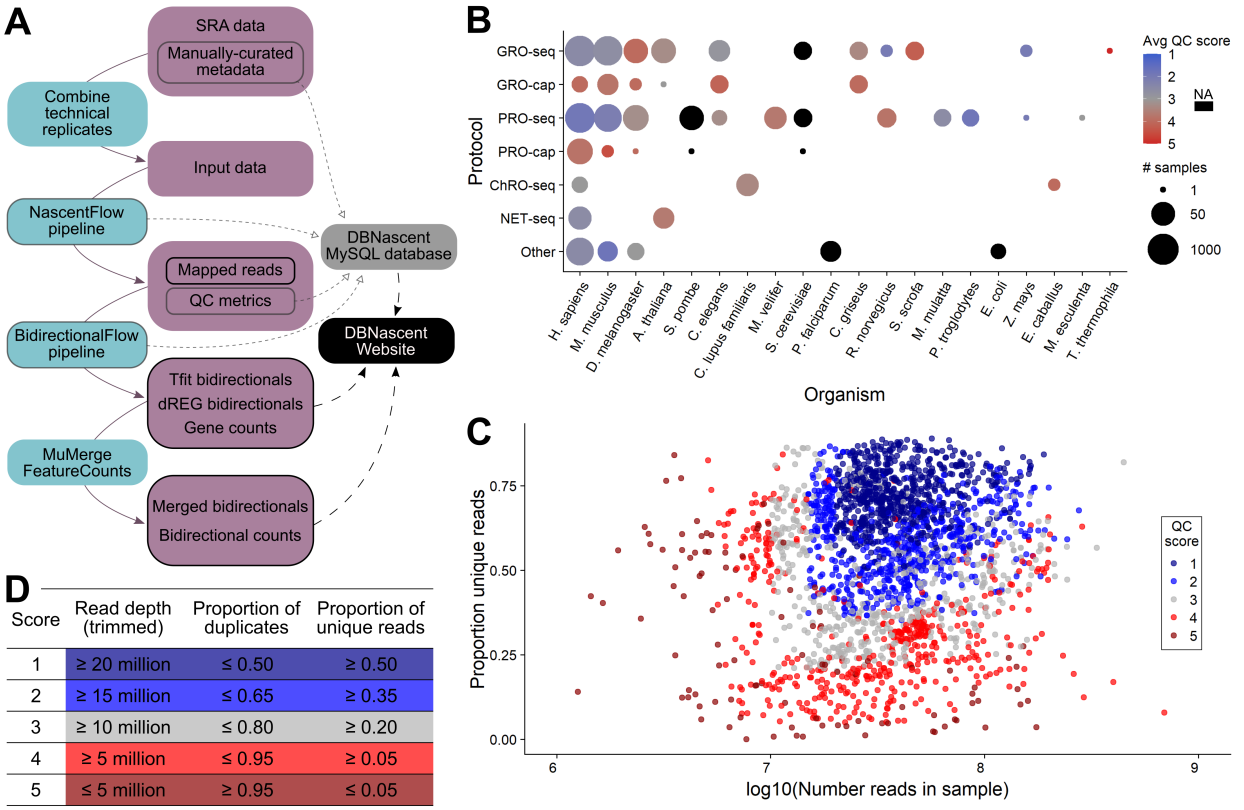
from the Gene Expression Omnibus (GEO) [214, 215] and the Sequence Read Archive (SRA) [216]. Metadata details such as organism, cell type, protocol used, library preparation, treatment type/conditions, replicate information were collected for all samples from their associated database information and/or publication (See Supplementary Table 4.1). This metadata was collected into a mySQL database (hereafter dbNascent) where all treatment condition times were annotated in reference to the time of cell harvest. Raw fastq files were processed by mapping to the relevant genome and identifying regions of bidirectional transcription using Nextflow pipelines(Figure 5.1A). Technical replicates fastq files were combined for downstream analysis.

In total, 3638 raw samples from the NIH Sequence Read Archive (SRA) were combined into 2880 biological samples across 20 organisms, collected from 287 projects, which consisted of either journal articles or Gene Expression Omnibus (GEO) datasets (Figure 5.1B). The samples were subjected to extensive quality control (QC), from which we developed a QC ranking metric based on read depth and complexity (Figure 5.1C-D). This metric was used extensively as a filtering mechanism, with most downstream analyses using high quality samples with a QC score of 1-3, unless specified otherwise. As nascent assays necessarily depend on a pull down step involving antibodies, we also sought to assess the extent of nascent RNA enrichment. To this end, an additional score was developed to attempt to identify samples that exhibited patterns of nuclear run-on (NRO) sequencing to be used as another potential filtering metric, although this was less robust (Supplementary Figure 4.1).

Of the 2880 samples in dbNascent, the majority (2387) were derived from either human or mouse cells(Figure 5.1B), and these were exclusively used for downstream analysis, i.e. identifying bidirectional transcripts. Samples were distributed across 19 and 10 tissues, for human and mouse respectively. In both cases, samples were primarily collected from cell lines or cultured primary cells (Supplementary Figure 4.2).

Bidirectional transcripts in dbNascent overlap cis-regulatory elements

Nuclear run-on assays, such as GRO-seq and PRO-seq, give readout of transcription from all cellular RNA polymerases. Consequently, they recover signal at both coding and non-coding



D

Score	Read depth (trimmed)	Proportion of duplicates	Proportion of unique reads
1	≥ 20 million	≤ 0.50	≥ 0.50
2	≥ 15 million	≤ 0.65	≥ 0.35
3	≥ 10 million	≤ 0.80	≥ 0.20
4	≥ 5 million	≤ 0.95	≥ 0.05
5	≤ 5 million	≥ 0.95	≤ 0.05

Figure 5.1: Overview of data included in dbNascent. A: Construction workflow for dbNascent. Data were derived primarily from the Sequence Read Archive (SRA) fastq files and manually curated metadata. Technical replicate fastq files were combined, then data were processed to obtain metrics on quality, bidirectional regions, and read counts. Metadata, quality control metrics, and software version information from pipelines were accumulated into a MySQL database. The dbNascent website (nascent.colorado.edu) draws from the MySQL database as well as processed analysis files for visualization. B: Samples in dbNascent were derived from twenty different organisms and multiple different protocols that are classified as nascent transcription. All species with genomes less than 25 Mb were not described well by the calculated QC score and thus are represented as NA. C: Complexity and read depth of human and mouse dbNascent samples. Two very low read depth samples have been omitted for the sake of visualization. D: Thresholds for calculation of the QC score. These thresholds may not be suitable for some species.

regions, much of which is not annotated. Therefore, to characterize regions of transcription, we opted to take a data driven approach, employing methods such as Tfit and dREG which seek to identify sites of bidirectional transcription directly from the data [81, 217, 218]. Tfit uses a mathematical model of RNA polymerase II to identify sites of polymerase loading and initiation. In contrast, dREG uses an unsupervised support vector machine approach to identify regions with

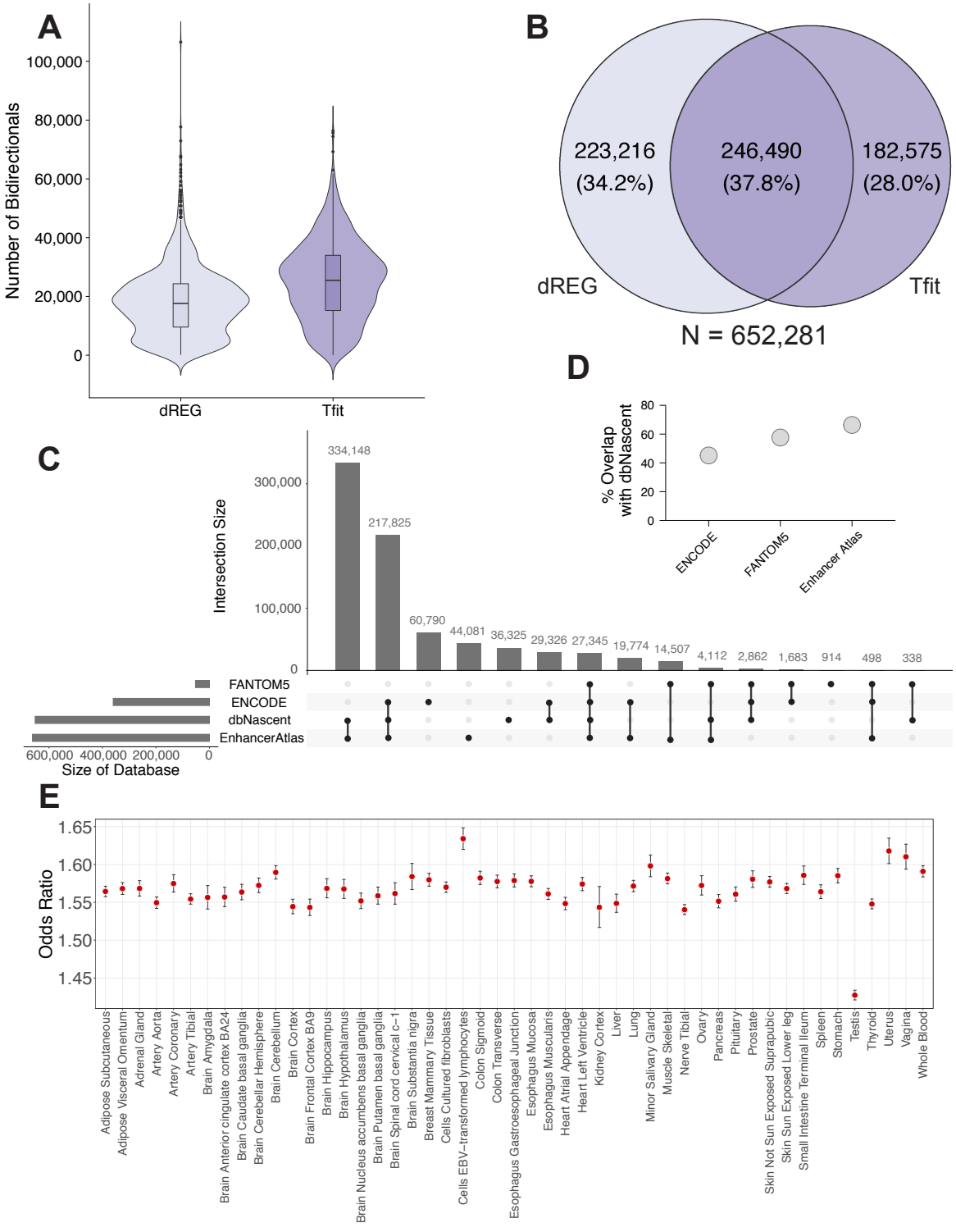
apparent bidirectional transcription. As the two methods have distinct strengths and weaknesses, we opted to use both methods and combine their results.

Across the 2866 human and 1424 mouse samples that were processed there were on average ~ 25000 bidirectional transcripts identified by Tfit and ~ 18500 by dREG. In order to get a final set of bidirectional transcripts across all human and mouse samples, bidirectional calls were combined using muMerge for each experiment for Tfit and dREG separately (see Methods Section) [59]. We used muMerge because its probabilistic framework balances the trade-offs inherent in more traditional merge and intersect approaches (see Chapter 4). The merging strategy was performed in a hierarchical manner which allowed each experiment to be merged separately, followed by merging regions by cell type. This allowed for replication of regions to be weighted higher than cross condition information. Thus, data from the same cell line is combined at a higher confidence. This step generated regions for Tfit and dREG separately(Figure 5.2 A). Since the resolution of Tfit calls at RNA polymerase initiation (typically the center region of bidirectional transcription) is better than dREG [219], Tfit calls were used when there was an overlap between the two callers. This is necessary to maintain adequate positional information for subsequent motif analysis. When comparing the two bidirectional identification methods, a majority of the calls overlapped (~40% overlap) (Figure 5.2 B).

The final set of bidirectional calls for human and mouse samples yielded 652281 and 563108 regions respectively. The bidirectional transcripts overlap introns, gene promoters and intergenic regions. We compared our bidirectional calls to annotated cis-regulatory elements from ENCODE, Enhancer Atlas and FANTOM5 [19, 204, 206, 207, 220] (Figure 5.2 C-D and Supplementary Figure 4.8). About 40% to 60% of the cis-regulatory elements are found in dbNascent bidirectional transcripts. Interestingly, 27345 human and 20506 mouse bidirectional transcripts are contained in all three databases (Figure 5.2 C and Supplementary Figure 4.7 C). Importantly, we recover a large fraction of the previously annotated cis-regulatory elements, despite having data from far fewer tissues than was used in these databases (Figure 5.2 D).

Regulatory regions have also been identified based on large scale genome-wide association studies. In particular, the GTEx consortium examined genome variation for its ability to influence expression levels. As sites of bidirectional transcription are often genetic enhancers, we reasoned that GTEx identified eQTLs should be enriched in our bidirectional regions relative to random variation. In confirmation of this, we found that bidirectional transcripts showed a higher odds for containing significant eQTL variants compared to non-significant variants (Figure 5.2 E) [203]. This further supports previous work showing an enrichment of eQTLs in enhancer regions [7].

We next turned our attention to characterizing the complete set of bidirectional transcripts identified across the database. Generally we find a wider length distribution for genes compared to bidirectional transcripts (Supplementary Figure 4.14 A) and higher median in levels of transcription in genes (Supplementary Figure 4.14 B). Furthermore, across all data sets, genes and bidirectionals have similar coefficients of variation (Supplementary Figure 4.15 and 4.16). Additionally, a principle component analysis on high quality human samples indicates that samples cluster predominantly by tissue of origin rather than quality score (Supplementary Figure 4.17).



Bidirectional transcripts in dbNascent have a high overlap with other cis-regulatory databases.

Figure 5.2: Bidirectional transcripts in dbNascent have a high overlap with other cis-regulatory databases. (A) The median number of bidirectional calls in human and mouse samples are around 20,000-25,000 calls for both dREG and Tfit. There are slightly more Tfit calls per sample compared to dREG. (B) A muMerge of all bidirectional transcripts across tissues called returned 652,281 bidirectional transcripts in human. (C-D) 40% -60% of cis-regulatory elements from other databases (ENCODE, FANTOM5 and EnhancerAtlas) overlap with dbNascent. (E) Significant GTEx eQTL variants have a higher odds to reside within called bidirectionals than non-significant variants. Equivalent information for mouse data Supplementary Figure 4.7.

Tissue specificity of transcription

Lidschreiber reported that eRNAs are more tissue specific than genes, but that conclusion was based on only 14 tissues with two replicates per tissue. Furthermore, their work was based entirely on the TT-seq protocol. Yet prior work (See Chapter 2) suggests that enhancer RNA recovery can be protocol specific. Furthermore, the work excluded lncRNAs from the analysis, leaving the question as to whether eRNAs are more or less tissue specific than lncRNAs, which have been also previously shown to tissue specific [221]. The number of cell types and tissues cataloged within dbNascent allows us to further examine this question. Of the 19 human tissues represented in the database, 12 were present in more than 5 samples. We used these 12 tissues as the basis of comparison, and further split them into noncancerous and cancerous tissues, yielding a total number of 15 tissues.

As the number of samples in each of these tissues varied widely, we chose to assess tissue specificity with the SPECS score [222], which can accommodate uneven sample size across the groups. The SPEC score ranges from 0 (indicating depletion) to 1 (indicating enrichment), with a ubiquitously transcribed gene scoring around 0.5. Considering first only genes and bidirectionals, the distribution of SPECS scores showed a larger proportion of bidirectionals having lower SPECS scores (Supplementary Figure 4.19). However, for a given tissue, both genes and bidirectionals had similar tail shapes approaching 1. Umbilical cord, noncancerous blood, and skin cancer samples showed the highest numbers of specific genes, whereas umbilical cord, skin cancer, and prostate cancer displayed the highest numbers of specific bidirectionals (Supplementary Figure 4.18).

Borrowing from the SPEC paper [222], we next assessed the change in transcription between the most specific tissue (highest SPEC score) and next highest scoring tissue. The resulting fold change will be large for each transcript which is truly present or transcribed heavily in a single tissue. Indeed, we observe a skew towards higher values for bidirectionals which is greater than lncRNAs or genes (Figure 5.3A). In line with previous work [221], lncRNAs show more tissue specificity than genes. Bidirectionals associated with genes, including both promoter

associated and intronic, were indistinguishable from genes. It is unclear the extent to which this represents a biological phenomena or a technical artifact, as it is difficult to accurately summarize transcription levels in the presence of overlapping transcripts.

The SPEC score analysis suggests that intergenic bidirectionals, most being associated with enhancers, are the most tissue specific. To further evaluate this claim we determined the number of tissues that show transcription of each region type (Figure 5.3B). By this measure, coding regions are most likely to be ubiquitously transcribed in our range of tissues, whereas intergenic bidirectionals are most likely to show tissue specific transcription. Thus enhancer associated RNAs, which arise from these bidirectional regions, are more often transcribed and potentially active in a small range of tissues as compared to genes and lncRNAs.

We next sought to determine whether the transcript levels varied distinctly across these transcript categories. To this end, we assessed transcription variation in ubiquitously transcribed and tissue-specific regions (Figure 5.3C). While ubiquitously transcribed regions showed similar variation across all region categories, tissue-specific regions varied more widely. However, some of this variation can be explained by the fact that tissue specific regions tend to be more lowly expressed.

Collectively, these comparisons indicate that genes are more ubiquitously transcribed than either lncRNAs or enhancer associated RNAs. Of the noncoding RNAs, intergenic bidirectionals (aka eRNAs) appear to be the most tissue specific and lowly transcribed. Any assessment of tissue specificity is limited by the data available. Thus, with the inclusion of more tissues and/or cell types we may find that some of these intergenic bidirectionals will be less tissue specific. However, the tissue specific nature of their transcription profile is consistent with the idea that these regions are regulatory. Distinct tissues are known to have both tissue specific transcription factors (cell type markers) and unique accessibility profiles. Enhancer associated RNAs arise from TF activity (see Chapter 4) which is largely occurring at a subset of accessible regions. Thus, the finding that intergenic bidirectionals (aka eRNAs) are the most tissue specific further emphasizes that these regulatory regions and the activities therein are distinguishing features of each cell type.

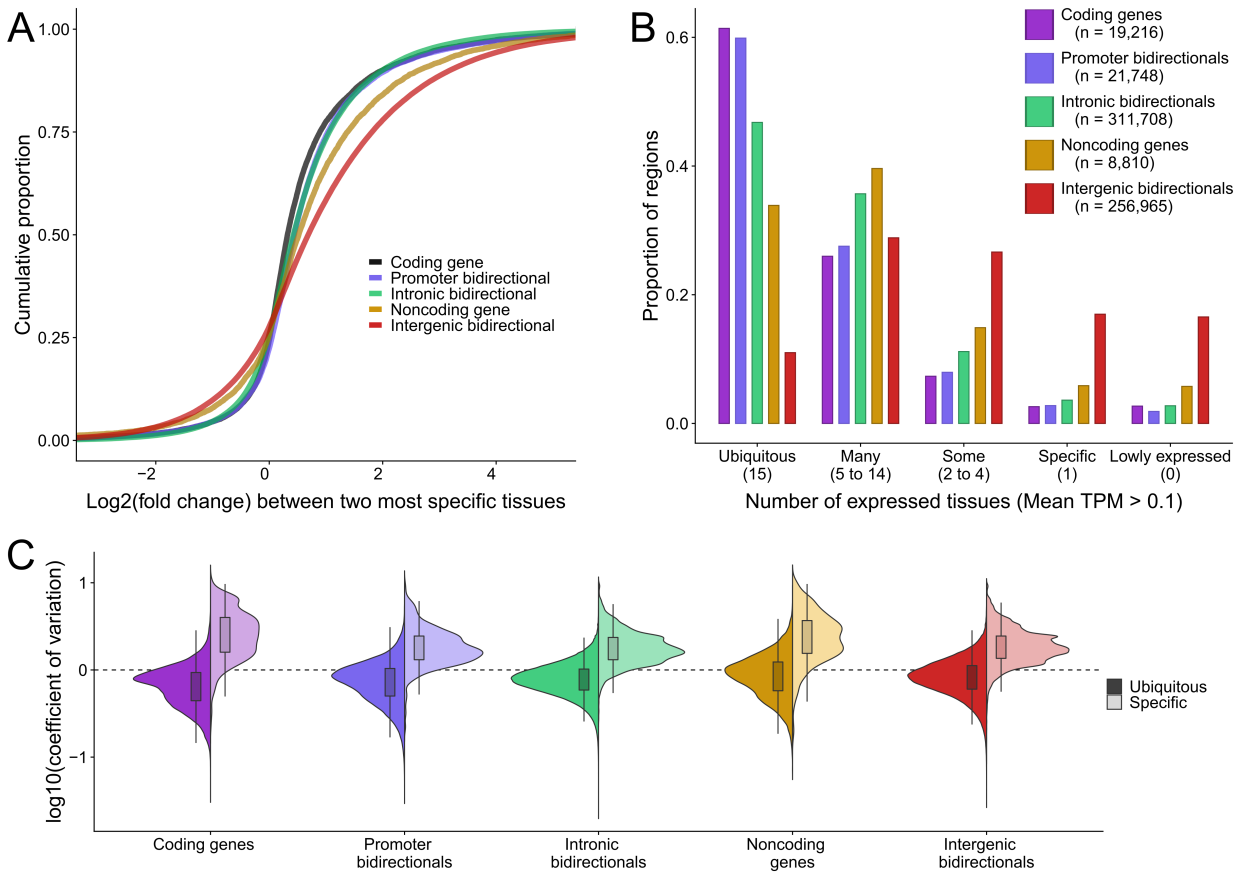


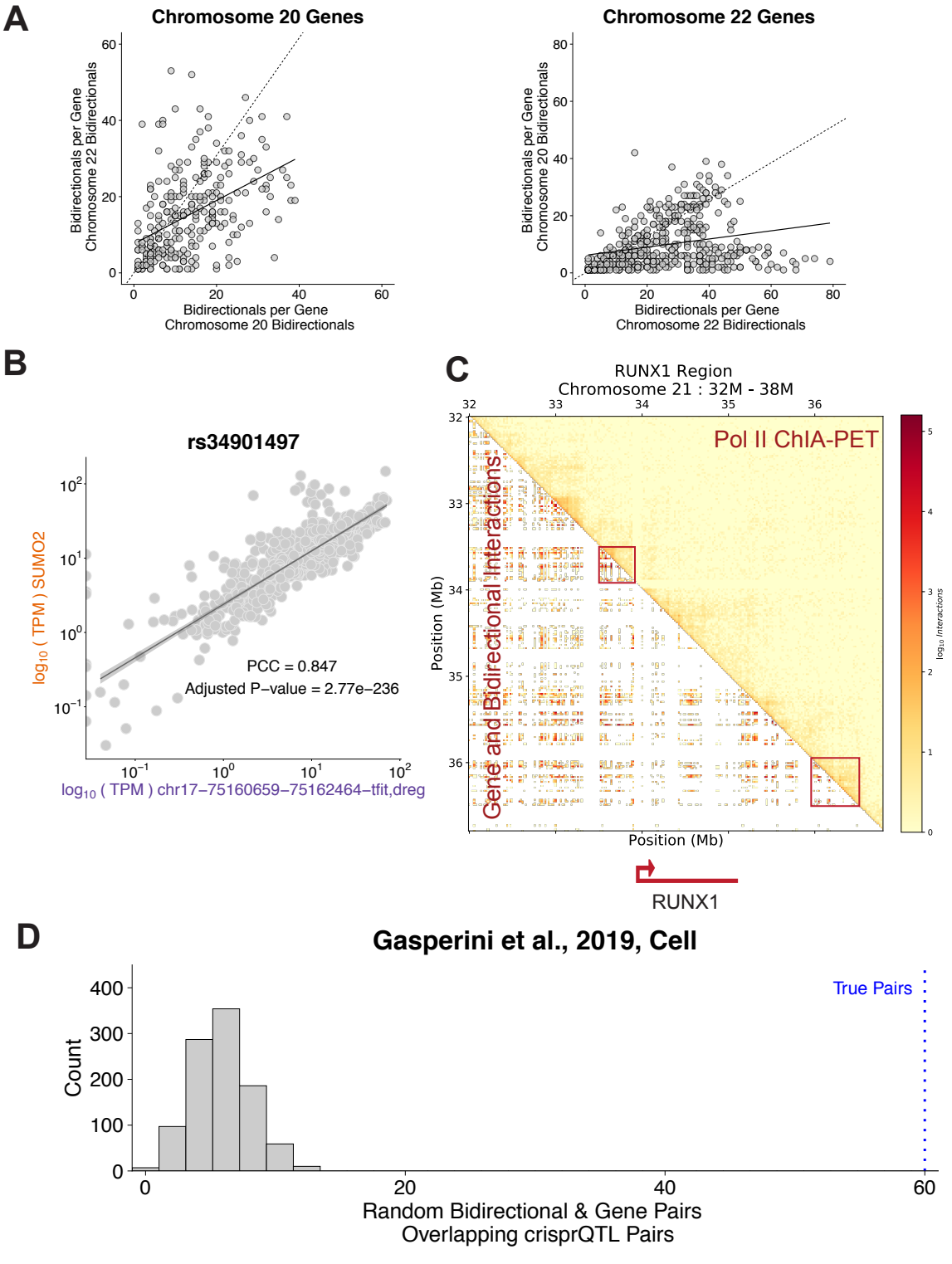
Figure 5.3: Tissue specificity of genes and bidirectionals in dbNascent. A: Cumulative distribution of fold changes between the tissue with the highest SPECS score and the tissue with the second highest SPECS score for each region. B: Each transcript is categorized based on the number of tissues in which it is transcribed. The "lowly expressed" category captures all transcripts that never meet a minimal transcription level. If a gene or bidirectional is present in fewer tissues, it would indicate a higher degree of tissue specificity. C: Variation in transcription across RNA type categories for ubiquitous and tissue-specific regions. Purple: coding genes; Blue: Promoters; Green: Intronic bidirectionals; Orange: annotated Noncoding RNAs; Red: intergenic bidirectionals.

Correlation analysis to identify putative bidirectional and gene pairs

Various methods have been developed to link enhancers to target genes with sequencing data [19, 23, 223, 224]. Initially used primarily for convenience, the closest gene approach to assigning enhancers (or ChIP sites) to target genes has turned out to be reasonably accurate. Measured 3D information is considered more accurate, but these data sets are not as readily available for most cell types. Prior work on nascent transcription showed that enhancers and their known target genes – as determined by 3D data – have correlated transcription levels[94, 121].

Therefore we sought to determine whether correlation was sufficient to identify enhancer to target gene linkages.

To this end, we calculated all pairwise gene and bidirectional correlations and identified highly correlated pairs by chromosome in a tissue specific manner (Supplementary Figure 4.21) (See Methods Section). To ensure robustness of the correlations, we filtered our tissues to those that include at least 15 samples, leaving 10 tissues to assess (Supplementary Figure 4.20 B). In this collection we found 304,250 unique pairs. Across these pairs, the median number of assigned bidirectionals to a gene is 10 and the median number of genes assigned to a bidirectional transcript is 3 (Supplementary Figure 4.23 A-B). While not a constraint of the approach, we found that most bidirectional regions within the pair were close to the gene TSS (Supplementary Figure 4.23 C). When assessing the number of tissues that support a pair, approximately 35% of pairs were supported by two or more tissues (Supplementary Figure 4.23C).



Significant gene and bidirectional transcript pairs interact in 3D space and they overlap eQTLs.

Figure 5.4: Significant gene and bidirectional transcript pairs interact in 3D space and they overlap eQTLs. (A) Assessing correlated pairs (between genes and bidirectional transcripts) from chromosomes 20 (left) and chromosome 22 (right) shows more significant interactions exist on the native chromosome rather than with bidirectionals on the swapped chromosomes. (B) Additionally, pairs found through correlated nascent RNA transcripts overlap known eQTLs from GTEx. The scatter plots show an interaction between SUMO2 and a bidirectional transcript on chromosome 17 (chr17-75160659-75162464-tfit,dreg). This interaction was also supported by 6 independent tissues (skin, colon, lung, prostate and embryo). (C) Interactions supported by 3D interactions in 25 kb bins from PolII ChIA-PET (upper right) and nascent transcription for a 25kb region around RUNX1 on chromsome 21. This comparison highlights regions with high interactions (shown in red squares). (D) Overlap of significant pairs with experimentally validated enhancer – gene pairs from Gasperini et al. 2019 [225] show a significant overlap of nascent identified pairs with the true pairs defined in Gasperini compared to random pairs.

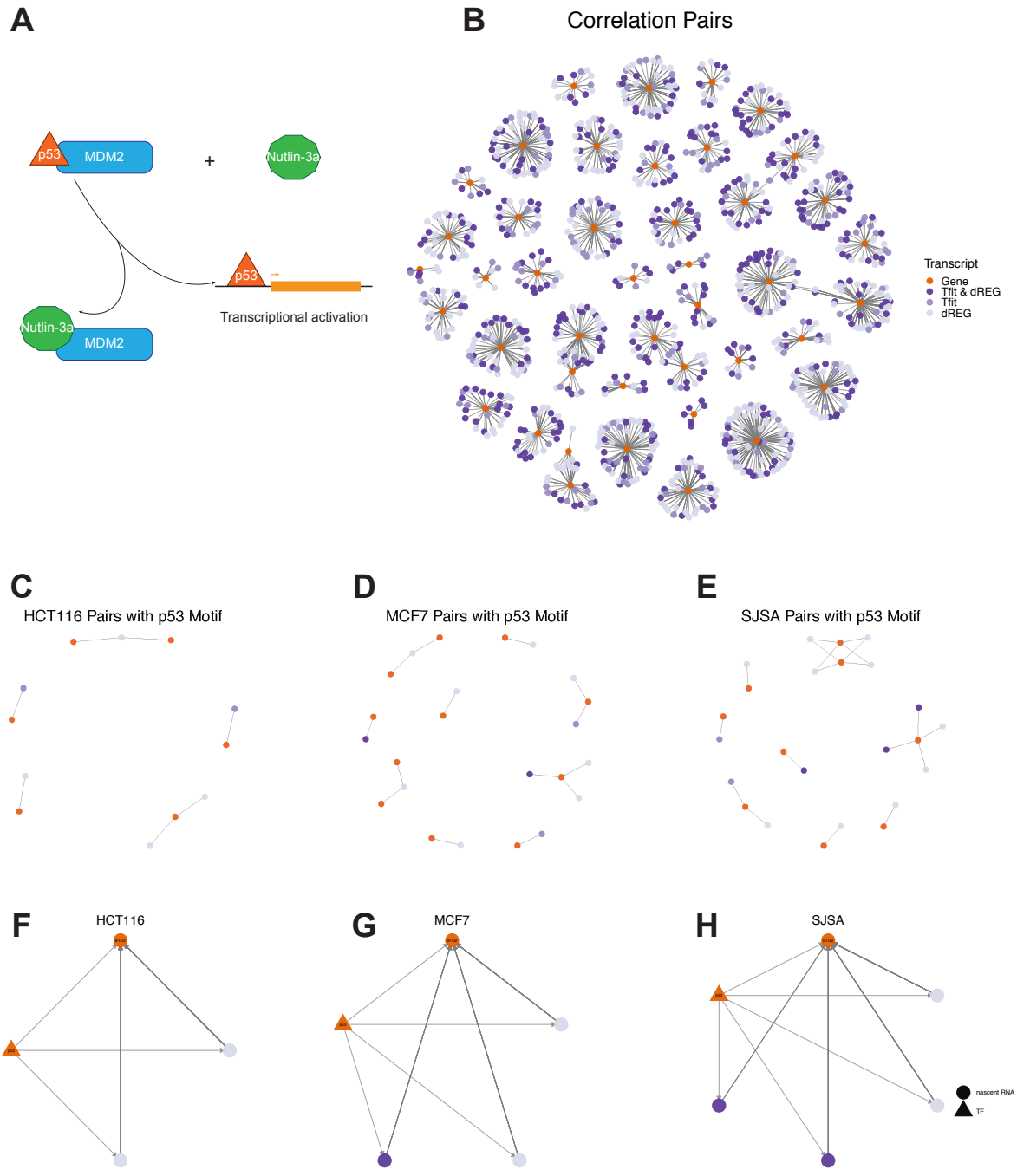
We next sought to determine whether our correlated pairs were enriched for biologically meaningful pairs. To this end, we sought to determine how many pairs would be recovered at random. We reasoned that most biologically real linkage would exist between entities on the same chromosome, thus our correlations were calculated completely within a chromosome (e.g. genes on chromosome 22 were compared to intergenic bidirectionals on chromsome 22). We next reasoned that while not a perfect negative control, correlations that arise between disparate chromosomes are more likely to arise from spurious correlation. Thus we also compared chromosome 20 and chromosome 22 transcripts by swapping the bidirectionals (e.g. chromosome 20 genes were correlated with chromosome 22 bidirectionals). Using these cross-chromosome comparisons, we found that the within chromosome comparison had more assigned pairs (Figure 5.4 A).

We next compared our recovered pairs to collections of know enhancer to gene linkages. First, we considered GTEx identified pairs and found that 10.32% of nascent derived pairs overlapped with GTEx pairs (only 0.50% of random pairs overlapped GTEx). Second, we examined 3D linkages identified within RNA polymerase II ChIA-PET in GM12878 cells. We found striking overlap between our pairs and ChIP-PET identified interactions (Figure 5.4 C). Finally, we examined the overlap of nascent derived pairs to experimentally validated enhancer and gene pairs from K562 cells, and observed a significant recovery of known interactions (Figure 5.4 D) [225]. Overall, all these comparisons suggest that a high number of our identified pairs are supported by other, orthogonal methods.

Co-transcription analysis of the p53 network.

In 2014, Dr. Mary Allen used Nutlin-3a to activate the transcription factor p53 [61]. Subsequent development of improved computational methods indicate that p53 activation results in the immediate increased transcription at 160 genes [121]. While approximately 500 intergenic bidirectionals were identified as also responding to p53 activation, it was not possible to link each responding gene to the regulatory regions driving its change. Now armed with both improved transcription factor inference methods (see Chapter 4) and putative linkages between

bidirectionals and genes (this chapter), we next sought to assess the utility of the linkages in assigning responsible regulatory regions to observed changes.



p53 activation response network.

Figure 5.5: p53 activation response network. (A) Schematic showing gene activatation by p53 upon Nutlin-3a treatment. (B) Bidirectional transcripts correlated with the 43 universally p53 responsive genes. Genes identified from three cell types[69] with 1 hour Nutlin-3a activation. Linkages to bidirectional regions from 10 tissues within the repository. Differentially transcribed bidirectional transcripts are highlighted in the cell line specific network for (C) HCT116 (D) MCF7 and (E) SJSA cells. BTG2 activation activation is distinct in (F) HCT116 cells (two bidirectional transcripts links) and (G) MCF7 cells (three bidirectional transcripts) and (H) SJSA cells (four bidirectional transcripts linked).

In this analysis, we compared the differential transcription of bidirectional and gene transcripts upon p53 activation in three different tissue types breast (MCF7), bone (SJSA), and intestine (HCT116) samples [69]. In each of these tissues, the Nutlin-3a treated (1 hr) samples were compared to their DMSO controls, and differentially transcribed genes and bidirectional regions were selected. We identified 43 previously known p53 response genes that were differentially transcribed across all tissues. All 43 genes were correlated with at least one bidirectional transcript (Figure 5.5 B). When we required that a p53 motif be present in the bidirectional transcript or gene promoter, only 8 genes were dropped (Supplementary Figure 4.25). Additionally, we identified cell type response networks for each tissue (Figure 5.5 C-E). These cell type specific responses are exemplified by BTG2 gene (Figure 5.5 F,G,H and Supplementary Figures 4.26, 4.27, 4.28).

The framework shows that we can assign bidirectional transcripts, and by extension enhancer regions, to the genes for which they are likely responsible in regulating the p53 induced transcription increase. Importantly these assignments are made in a context-specific manner with high confidence. Armed with experimental conditions, the bidirectional transcript and gene transcript pairs mentioned in Section can be narrowed down to give pairs where both halves of the pair are responsive to a perturbation. These are then the highest confidence linkages that suggest directly testable hypothesis on causal physical relationships between the transcription factor (in this case p53), an associated enhancer (the bidirectional region) and its target gene (the target gene in the correlated pair).

Discussion

Here we present dbNascent, an atlas that catalogs published nascent RNA sequencing data, the largest database of nascent RNA data. These data were manually curated for experiment relevant metadata and analyzed using standardized pipelines. The result is a collection of data sets with comparable information and quality control metrics in a MySQL database.

We further identified regions of bidirectional transcription across all high-quality human and mouse samples. These regions were merged in a hierarchical fashion to generate experiment

specific, cell type specific, tissue specific and consensus bidirectional transcript annotations. The consensus regions overlap with previously annotated cis-regulatory elements from ENCODE and FANTOM5. Additionally, we assessed the tissue specificity of all the transcribed regions by comparing genes, lncRNAs, and bidirectional transcripts, showing that bidirectional transcripts are more tissue-specific than lncRNAs, which are in turn more specific than genes. These observations are supported by previous findings [57, 221]. Our annotations expand the landscape of transcribed cis-regulatory elements.

Finally, we assign cis-regulatory regions to likely target genes using a correlation based framework. We identified correlated bidirectional and gene transcripts across human tissues. The correlated pairs we identified overlap experimentally validated enhancer—gene pairs as well as eQTLs from GTEx, supporting the use of these data to investigate enhancer assignment. Since dbNascent contains the largest collection of nascent RNA samples compared to other databases such EnhancerAtlas [220], we expanded the interactions observed to more tissues. We then used the p53 response network to demonstrate the utility of our correlated pairs, and in so doing identify enhancer to gene associations for nearly all of the p53 response genes.

Given that the method we use to identify pairs relies on correlations of transcription levels, spurious correlations are a real concern. To curb the false positive rate, we added constraints for assigning bidirectionals to a gene. Namely, allowed correlations that were supported by the majority of samples, pairs that were within a 1 Mb window, and had a false discovery rate of less than 0.001 on the correlation p-values. These filter steps likely enrich for true correlations but may do so at some cost with respect to less frequent but real interactions. Additionally, we estimated the relative enrichment of true correlations relative to spurious ones by using a cross chromosome comparison. However, it is well worth noting that there may be real correlations in the cross chromosome data. Nevertheless, the higher level of within chromosome correlations suggest that our data collection contains biological signal. Finally, as a transcription factor regulates many locations across the genome, there may be correlations within our dataset that represent the activity of the TF more generally moreso than the direct enhancer to gene linkage being tested.

Importantly, the p53 activation analysis shows that given a perturbation experiment, we can assign bidirectionals to genes with high confidence. Between patterns of differential transcription in response to TF activation and the patterns of bidirectional to gene correlation, it is possible to assign apparent responsibility to many enhancer to gene linkages. These are now readily testable hypothesis upon which perturbation experiments could be used to confirm causality.

It is also worth noting that the correlation linkages identified here could be used more generally to infer gene regulatory networks (GRNs). Correlation based network inference methods [226] for GRNs are, in theory, an excellent starting point for these analysis. However, our experience indicates that the increase in data set size that arises from including enhancers (many more enhancers than genes) makes the practical utilization of these tools challenging. Further work on building networks from these data would offer a great condition-specific resource for further experimental validation.

Methods And Materials

All the code and methods used in the meta-analysis of nascent RNA sequencing experiments can be found on this github page https://github.com/Dowell-Lab/DBNascent_Analysis. The samples were processed on a compute cluster running CentOS Linux v7.

Mouse samples were mapped to the mm10 reference genome and human samples to the hg38 genome. Databases used for comparisons that were mapped to older reference genomes were lifted to the specified genomes above using liftOver 227.

Nascent RNA sequencing experiments metadata collection

Nascent RNA sequencing experiments were manually curated from the Gene Expression Omnibus (GEO) 214, 215 and the Sequence Read Archive (SRA) 216. All treatment condition times were annotated in reference to the time of cell harvest. In addition to standard metadata, we also curated protocol specific information including the ratio of biotin labeled NTPs was collected for PRO-seq experiments, and antibodies used for the NET-seq protocols. Papers that had other high throughput experiments (including RNA-seq, ATAC-seq, ChIP-seq and 3D chromatin assays

such as HiC) that were performed along with nascent RNA sequencing were noted. Two rounds of data curation were implemented where the first round was meant for data entry, and the second round for entry verification. In total, 3638 samples were manually curated from 320 SRA projects (SRPs) and 287 papers.

Preprocessing nascent RNA sequencing experiments

All SRR accessions were downloaded from the SRA and extracted with SRA Toolkit (v2.8.0, v2.9.2). Replicate information was used, where available, to combine technical replicates by concatenating fastq files. New samples resulting from technical replicate combination within a given experiment were given SRZ designations with a number equivalent to the first numerical SRR contained within. In one case, technical replicates were combined using data from multiple papers as a result of further resequencing of previous samples. Combined samples in this case were given SRM designations, with numeric conventions the same as the SRZs. In total, 2880 samples were generated from the original 3638 samples after technical replicate concatenation. This collection of 2880 samples was then the source of all downstream analysis. All samples in the database were trimmed, mapped to reference genomes, and assessed for sample quality using an in-house NextFlow pipeline (https://github.com/Dowell-Lab/Nascent-Flow), run with NextFlow v20.07.1.

Mapping reads to reference genomes

Fastq files were trimmed for adapter sequences and low quality bases using BBMap (v38.05), then aligned to reference genomes with HISAT2 (v2.1.0). Downstream mapped read files (CRAM files and IGV-compatible TDF files) were generated with Samtools (v1.8, v1.10), Bedtools (v2.25.0, v2.28.0), and IGVtools (v2.3.75), and in some cases were done so through an additional NextFlow pipeline (https://github.com/Dowell-Lab/Downfile_pipeline), run with Nextflow v20.07.1. For practical reasons, software versions were occasionally changed to process some samples. All versions used to process a specific sample are linked to that sample in the database.

Quality control and quality tiers

Samples were assessed for quality using metrics from the following software packages: FastQC (v0.11.8), HISAT2 (v2.1.0), Preseq (v2.0.3), RSeQC (v3.0.0), Picardtools (v2.6.0), and BBMap (v38.05). Three specific metrics were used to classify samples into quality 'tiers' for filtering purposes: read depth after trimming, proportion of duplicates (as assessed using Picardtools), and complexity (as assessed using the modeled value for unique reads in 10 million output by Preseq). Thresholds were determined to classify samples into one of five tiers (see Figure 5.1).

Identifying bidirectional transcripts

Regions of nascent transcription were identified using Tfit 81 and dREG 217, 228. Identification of regions of transcription with dREG followed the recommended pipeline where mapped reads were filtered based on a minimum map quality score (MAPQ) greater than 1. BigWig input files for dREG were generated by converting filtered BAM files using `bamToBed`, then BED files were converted to bedGraph format using `bedtools genomecov` (bedtools version 2.28.0), and finally the BigWig files were generated using `bedGraphToBigWig` (from https://www.encodeproject.org/software/bedgraphtobigwig/) 229. Identification of bidirectional transcription with Tfit followed a pipeline where multimapped reads and reads with low map quality score were filtered as shown: `samtools view -@ 16 -h -q 1 ${SRR}.bam | grep -P '(NH:i:1|^@)' | samtools view -h -b > ${SRR}.filtered.bam`. Input bedGraph files were generated using `genomeCoverageBed`. Tfit was run in a two step processes, first with the template matching module to identify sides of bidirectional transcription, then these regions were used as input to fit the precise RNA polymerase behavior (Supplemental Figure 4.3). Since both dREG and Tfit are compute intensive, only high quality data sets (QC < 4) were processed using dREG. The nextflow pipeline used for characterizing bidirectional transcription with both Tfit and dREG can be found on GitHub (https://github.com/Dowell-Lab/Bidirectional-Flow).

Merging regions of bidirectional transcription

Regions (from replicates, conditions, and bidirectional calling methods) were merged using muMerge version 1.1.0 (https://github.com/Dowell-Lab/mumerge) 9. Since muMerge combines regions in a probabilistic way, replicate information and sample conditions were taken into account for the merging processes. Tfit and dREG bidirectional transcript calls were first muMerged separately by paper based on the experimental setup (that is by cell/tissue type, experimental condition and replicate information) creating

`Paper_genome_celltype_tfit/dreg.bed`. The criteria used was the same as shown in the original muMerge paper 9; where joint distribution for each region was calculated such that product of replicate distributions was taken, then sum of the resulting distributions was taken across the different experimental conditions.

The muMerge bed files by experiment were combined based on the cell/tissue types for Tfit and dREG, where the same cell/tissue types were treated as “replicates" and the different cell/tissue types were the “conditions". Selection of samples to merge was based on the percent of calls overlapping TSS, the GC content of the 300bp regions around the center of the calls and the paper QC (Supplement Figure 4.5 A and B). Bidirectional calls from papers with GC content greater than 0.5, percent TSS regions less 0.5 and paper QC less than 4 (Supplement Figure 4.5 C). Samples from 61 mouse papers and 101 human papers were used for the final mumerge step. Lastly, the dREG and Tfit muMerge files were combined such that Tfit calls were used for overlapping regions (Shown schematically in Supplementary Figure 4.4).

Some general observations on the two methods were made. First, we found that dREG broke bidirectional regions identified by Tfit into smaller chunks. Despite this, Tfit calls slightly more regions per sample compared to dREG (Figure 5.2). Second, both methods struggled to call bidirectionals within introns when the gene was transcribed robustly, though generally dREG was better in these regions. Third, dREG called a larger number of low level bidirectionals than Tfit (Supplemental Figure 4.15). Finally, when samples have poor quality, both dREG and Tfit tend to call more gene TSS regions as these are more highly transcribed and therefore have the most

robust signal (Supplemental Figures 4.6 and 4.5). More generally, we see a higher level of TSS regions called by dREG versus Tfit. As no gold standard exists for these regions, it is unclear which of the two methods is more accurate. There are regions that are recovered by both methods, and regions unique to each bidirectional transcript caller. So, in this paper we combined calls from both methods, keeping track of their origin, i.e. bidirectional caller.

Overlapping bidirectional transcripts with cis-regulatory elements

Overlaps of the merged regions were assessed to enhancer annotations from FANTOM5, ENCODE and Enhancer Atlas using bedtools intersect version 2.28.0 20, 204, 206, 220. Additionally, these regions were also overlapped with disease associated variants from GTEx and odds ratio calculated 203. The odds ratio was calculated by counting the number of GTEx eQTL variants that overlapped bidirectional transcripts for both significant and non-significant variants and getting the fraction of the variant overlapping bidirectional transcripts versus not overlapping the transcripts (Supplement Figure 4.11).

$$\text{Odds Ratio}_\text{t} = \frac{sb_t/sn_t}{nb_t/nn_t} \tag{V.1}$$

Where t is a given GTEx tissue, sb_t are GTEx eQTL variants that fall in bidirectional transcripts, sn_t are significant variants that fall outside of bidirectional transcripts, nb_t are non-significant variants that fall within bidirectionals and nn_t are non-significant GTEx variants that do not fall in bidirectional regions.

Calculating base content

Base composition for bidirectional transcript calls from dREG and Tfit was defined as the ration of GC content in the center 300bp (typically contains the RNA loading position) relative to larger bounding 3kb region. Notably, this is the same window used by TFEA (Chapter 4).

Counting reads

Reads were counted using featureCounts from RSubread 230. For gene transcripts, sense strand reads over gene bodies (750bp from the TSS) were counted as these show more consistency across nuclear run-on protocols 231. Reads on both stands were counted for bidirectionals

transcripts. In both cases, multimapping reads were ignored. However, unlike for bidirectional transcripts, multi-feature overlap in genes was allowed.

Normalizing read counts

Normalization of counts was done using transcripts per million (TPM) normalization as shown below 232, 233:

$$\mathrm{TPM_i} = \frac{r_i/l_i}{\sum_1^j r_j/l_j} \times 10^6 \tag{V.2}$$

where r_i are the mapped reads for transcript i (for all genes and bidirectional transcripts), l_i is the transcript length and $\sum_j r_j/l_i$ sums all j length normalized transcripts. The ratio is multiplied by a scaling factor of 10^6. The counts for genes was normalized over full length of the longest transcript and the 5′ end of that transcripts were truncated. To avoid double counting reads, bidirectional transcripts that overlap genes were removed from the normalization step. The total number of transcripts included 5′ end truncated genes, and intergenic bidirectional transcripts that did not fall with 500bp of the TSS and 100bp downstream of the TES.

Calculating summary statistics

The summary statistics described below were calculated using R version 3.6.0 111. For all samples, the average and median transcription values were calculated based on the normalized counts. Across sample coefficient of variation (CV) for human samples were calculated as follows:

$$\mathrm{CV_i} = \frac{\sigma_i}{\mu_i} \tag{V.3}$$

where for transcript i the standard deviation (σ_i) for normalized counts is divided by the average normalized counts (μ_i).

Motif scanning

Whole genome motif scanning was performed by FIMO in from the MEME suite (version 5.0.3)234. The motif databases for both human and mouse were pulled from HOCOMOCO (v11)145. The distance was calculated from the center of the bidirectional transcripts to the center of all high quality motif hits within 1500bp of that transcript.

Correlation and Co-transcription Analysis

Building the transcriptional regulatory network from nascent RNA data was split into three steps (1) finding pairs of highly correlated genes and bidirectional transcripts, (2) filtering high confidence pairs (3) pulling out high confidence TF motif hits in bidirectional transcripts (See Supplementary Figures 4.21 and 4.24).

Step 1: Pairwise correlation of gene and bidirectional transcripts Pearson's correlation coefficients between genes and bidirectionals were calculated using WGCNA (version 1.70-3) 39, 235. Comparison were among transcribed regions within a chromsome (i.e. no chross chromosome comparisons). The input to WGCNA was normalized counts for genes where the 5′ end was truncated and bidirectionals transcripts. These counts were log transformed such that transcripts with zero counts were excluded from the pairwise calculations as shown below:

$$\text{Transformed TPM}_\text{i} = \log_{10}(\text{TPM}_\text{i} + 1) \tag{V.4}$$

Given the samples with transcription, the pearson's correlation coefficient (PCC) was calculated as follows:

$$\text{r}_{\text{x,y}} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \tag{V.5}$$

where x are the genes, y bidirectional transcripts and n are the number of samples transcribed for both gene and bidirectional transcripts (i.e. excluding samples with zero counts). The p-value was calculated from the Student's t-distribution and the t statistics was calculated as:

$$\text{t} = \sqrt{n-2} \times r/\sqrt{1-r^2} \tag{V.6}$$

where n are the number of samples and r is the PCC. The output from the correlation calculations included the transcript identifiers, distance between the pairs, the PCC, the p-value and adjusted p-values 39, 236.

The code used to calculate pairwise correlations can be found on this GitHub repository https://github.com/Dowell-Lab/bidir_gene_pairs.

Step 2: Filtering for high confidence pairs By default, the analysis was performed for tissues with over 15 samples, and pairs that were supported by over 5% of the samples were considered reported. Additionally, pairs with pearsons correlation coefficient (PCC) greater than or less than 0.6, with an adjusted p-value less than 0.01 and with 1 Mb of each other were considered significant pairs.

In reporting high confidence pairs, more stringent parameters were applied. Since the sample sizes from the tissue correlations were smaller, an even more stringent filter was applied for the correlated pairs. In order for a pair to be considered significant it had to be supported by over 75% of the samples, have an absolute PCC of greater than 0.8 and an adjusted p-value less than 0.001. For pairs that were derived from using all 880 of the human samples, the minimum number of samples required to support a pair was 440 (5% of the samples) and the absolute PCC had to be greater than 0.6. The 299,120 significant pairs from the 10 tissue correlations and 151,538 pairs for correlations derived from all samples were combined yielded 373,247 unique pairs.

Step 3: Assigning TFs to networks Assignment of TFs to the networks was based on the motif presence in the bidirectional transcript region. In order for a TF to be assigned to a bidirectional transcript the motif had to be within 1500bp of the center of the transcript (See p53 motif example below).

p53 response network

Published GRO-seq samples from HCT116 (colorectal carcinoma), MCF7 (breast carcinoma) and SJSA (osteosarcoma) cell lines were analyzed for differential transcription across genes and bidirectionals 69. Differential transcription analysis between the control (DMSO) and treated (nutlin-3a, a MDM2 inhibitor) samples was done using DESeq2 (version 1.26.0) 88. A total of 43 differentially transcribed genes were shared across the three cell lines. We identified the bidirectional–gene pairs and found that all 43 genes had a bidirectional region correlated to

them via the pairwise correlation method mentioned above (Subsection). After filtering for bidirectional transcripts with a p53 motif, we identified 32 genes with at least one bidirectional region assigned. Lastly, we further filtered for bidirectional transcripts that were differentially transcribed in each of the cell lines.

Overlap of pairs with eQTLs and crisprQTLs

Gene and bidirectional pairs derived from the co-expression method were overlapped with pairs from crisprQTLs validated enhancer – gene pairs from Gasperini and company 225. The gene and bidirectional pairs were randomly shuffled 1000 times, and the overlaps with the crisprQTLs assessed. The random pairs and true pair overlaps were compared and plotted as a histogram (See Figure 5.4D). For eQTLs from GTEx, the randomization was only done once 203.

Evaluation of relative false positive rate

To evaluate how often we get false positive pairs, genes and bidirectional transcripts from chromosomes 22 and 20 were swapped such that genes from chromosome 20 were correlated with bidirectional transcripts from chromosome 22 and vice versa. The number of assigned bidirectional regions to each gene were compared with the true pairs from the original chromosome using the same methodologies as the within chromosome comparisons described above.

CHAPTER VI

CONCLUSIONS AND FUTURE DIRECTIONS

Summary Of Contributions

In this thesis, I integrate several computational concepts and sequencing data with the ultimate goal of understanding how gene expression is regulated. The process of gene expression regulation involves various components including enhancers, promoters, and TFs. These interactions have been characterized and represented by gene regulatory networks (GRNs). I introduce a method that integrates a more direct measure of these elements for building GRNs with nascent RNA sequencing data, which measures RNAs that are actively transcribed by RNA polymerase. Nascent RNA data yields gene transcript and bidirectional transcripts at enhancer regions (giving rise to enhancer RNAs/eRNAs). I show that, as with other computational methods, the quality of the input data is paramount to the extraction of informative signals. Via implementation of a co-transcription analysis on nascent data, my method recovered known interactions associated with diseases. Lastly, I refined the context-specific GRNs by introducing experimental data that activated p53, a TF that is a tumor suppressor with many implications in cancer. With this framework, I show that the nascent RNA derived interactions combined with perturbation experiments offer a method that enables us to understand the tissue-specific GRNs in response to stimuli. Importantly, I am presenting an additional toolkit for exploratory research in basic and translation research.

Understanding the limitations of input data is key to determining the bounds of our inferences. In the past, GRNs were inferred using gene expression data such RNA-seq or microarray data [39]. These data captured the gene expression and the derived networks identified genes with correlated expression profiles across samples. In an attempt to incorporate TFs into these GRNs, TF motif instances upstream of genes were used as a proxy for TF presence. Alternatively, chromatin accessibility data such as ATAC-seq has been used as a substitute for active enhancers and genes, and co-accessible regions have been used as a potential linker for $accessible\ enhancer \rightarrow expressed\ gene$ [237]. Other attempts at building a more complete

GRN combined RNA-seq and ATAC-seq data [55, 238]. In this thesis I show that, unlike the above-mentioned inputs for GRNs, nascent RNA sequencing data gives a readout of transcription at enhancers and genes, offering a more complete representation of the gene regulation process in a single experiment.

Over the last decade, there has been an expansion of published nascent RNA experiments, however, there has not been a systematic assessment of these methods. In this thesis, I present a much-needed comparison of the widely used nascent RNA protocols (GRO-seq and PRO-seq) and the main library preparation methods (RNA ligation, template switching, circularization, and random priming) from the same cell line and treatment conditions. In summary, the main differences between the protocols are observed at the 5' end of transcripts. In contrast, for the library preparation methods, the quality control metrics varied. Interestingly, I showed that we can distinguish nascent protocols with a combination of discrete wavelet transform (DWT) analysis and support vector machine (SVM) on the detail coefficients that contain the noise signal. Despite these differences, Chapter 2 reveals that the subsequent biological conclusions were consistent regardless of the protocol.

Nascent transcription clearly shows transcription, which is predominantly bidirectitonal, arises from nearly all regions of RNA polymerase II initiation and corresponds tightly with enhancer marked regions (by ENCODE data). Thus, methods of identifying bidirectional regions [81, 121, 228, 239] are constantly being developed. Some are supervised such as FStitch a hidden Markov model and Tfit an exponentially-modified Gaussian mixture model of RNA polymerase activity [81, 121] whereas others are unspervised pattern detection approaches such as dREG and SVR [228, 239]. To date, only a singular benchmark has compared these approaches [219]. From this benchmark, I designed and developed a standardized pipeline for processing nascent transcription data into bidirectionally transcribed regions (Chapter 3). In chapter 3 I highlight the proper processing of nascent RNA data taking into account the quality of the samples. Put together, chapters 2 and 3 lay the foundations for downstream analyses.

In Chapter 5 I present dbNascent, a database of about three thousand publicly available nascent RNA data. This database offers a unique resource for understanding transcription and transcription regulation. Along with dbNascent, I also present a summary of all transcribed regions in human and mouse data sets housed in the database and show that these regions overlap annotated cis-regulatory elements. I also show that the regions of bidirectional transcription are enriched for disease-associated SNPs from GTEx, consistent with other studies. I then employ correlation-based methods to identify enhancer and gene pairs that are enriched for previously validated pairs. As a starting point, I ask for bidirectional and gene transcripts that are highly correlated within a one-megabase distance. Informed by findings from Chapter 4, which introduces transcription factor enrichment analysis (TFEA) – a method to infer TF activity from nascent RNA data, I also use TF motif data to link p53 responsive enhancers to the genes they likely regulate.

Future Work

The current work presented in this thesis offers a starting point for further research. Protocol comparisons presented in Chapter 2 could be expanded to include other nascent RNA sequencing protocols such as TT-seq and NET-seq. While these protocols are not as widely used, understanding their biases and how those biases might influence the recovery of biologically meaningful signals would allow for the proper integration of these data sets to subsequent meta-analysis studies. Furthermore, the DWT+SVM method for labeling nascent RNA sequencing data could be expanded to automatically label nascent RNA experiments on the Sequencing Read Archive (SRA) and the Gene Expression Omnibus (GEO). Taking it a step further, the method could be extended to labeling other next-generation sequencing protocols as a way to validate human-inputted metadata as human error is inevitable. Lastly, one can imagine using the detail coefficients from the DWT to measure the overall quality of a given sample.

Chapter 2 gives guidelines for processing nascent RNA data in an annotation-agnostic manner with established tools used in the field (dREG, FStitch and Tfit). Since the submission of

this thesis, other tools have been developed (e.g. PINTS [240]), and a broader comparison would be informative to both subsequent analysis and further methods development.

TFEA (Chapter 3) was a great improvement from the previous TF activity inference since it accounted for the differential change in bidirectional transcription levels [8, 9]. The current experimental setup for TFEA assumes a pairwise comparison, so extending the experimental setup to account for varying experimental designs, such as time series, would yield more meaningful results. Currently TFEA identifies transcription factors with enriched positional and transcriptional response signal, but does not identify which regions can be *assigned* to the TF of interest. Assignment is an inherently more difficult problem, as it is related to identifying causality. Two things could be included into future TFEA versions that assist in this endeavor. First, the leading edge of the enirchment score would effectively identify the set of bidirecitonals where the TF is most enriched (and are therefore likely direct targets). Second, the correlation linkages identified in Chapter V could be used to improve the enrichment score – as a TF need not be associated with a gene's promoter if it binds and activates at one of the gene's linked enhancers.

The other tool developed in the TFEA work is muMerge, which offers a merging strategy that was informed by the sample and replicate information. Adding sample quality information to down weight poor samples in the muMerging step of the TFEA process may lead to more informative regions – particularly in large scale merging strategies as those used in Chapter V. As transcription factor motif information is positionally oriented relative to the site of RNA polymerase initiation, it is critically important that the loading position be identified precisely. Merging strategies inherently introduce uncertainty in this position, which is a limitation to any meta-analysis on TF activity. It is also worth noting that muMerge reduces the width of regions as more data is included – a side effect of the muMerge strategy of assuming width is related to the variance on the loading position. Unfortunately, the smaller regions rapidly no longer reflect the extent of transcription. Balancing these conflicting encodings of region width (uncertainty and extent of transcription) is an area of future work.

One of the main contributions of this thesis is the resource of nascent RNA data sets annotated systematically and processed with standardized methods. This database offers a unique resource for understanding transcription and transcription regulation. I used these data to start building GRNs by first pairing bidirectional transcripts to genes and then linking TF motif data to the transcribed regions. However, due to the complexity of assigning reads to intronic bidirectionals – where the bidirectional and gene are overlapping – intronic bidirectionals were excluded from the GRN inference. Future work to improve upon this limitation could be informed by reads from the opposite stands of a gene to disambiguate which reads in the overlap region arise from the gene versus the overlapping bidirectional transcript.

Importantly, dbNascent offers a resource of nascent RNA sequencing data and other experiments that are associated with a specific study. The manually curated metadata are an invaluable resource for any downstream analysis. There are matched experiments for some of the data sets such as RNA-seq, ChIP-seq, ATAC-seq, and HiC experiments that can be implemented to better understand condition-specific GRNs. These data can also be used to study how well nascent GRNs can be recapitulated with other sequencing data sets. In addition to my work, other laboratory projects are leveraging the nascent repository – including ongoing work on nascent transcription spike-ins.

Since dbNascnet contains various perturbations and time series data, it is possible to take advantage of the fact that enhancer transcription precedes (temporally) the transcription of the gene it targets [38]. One can apply the Granger Causal (GC) framework to link enhancers to their target genes (or gene promoters). The GC framework by definition implies that the future of variable Y (gene transcription) can be predicted from the past of variable X (enhancer transcription) allowing the linking on $enhancer\ x \rightarrow gene\ y$ [40, 241]. These condition-specific interactions can act as validations for GRNs presented in Chapter V.

Previous work in the lab showed that we can infer TF activity by comparing TF motif colocalization with bidirectional transcripts in a nascent RNA experiment [8]. This requires a computationally expensive process of generating background sequences that represent the base

compositions of the bidirectional transcript sequences. Comparing the TF motif colocalization between the simulated and the true set, we can infer active TFs in a single experiment. Adding TF activity inference to this framework of GRN inference would provide TF anchors to the networks.

Finally, dbNascent offers a valuable resource for exploring the transcriptional process. The $enhancer \rightarrow gene$ interactions combined with experimental perturbations present researchers with a manageable set of candidate regulators for downstream experimental validation.

REFERENCES

1. Buffry A, Mendes C, and McGregor A. The functionality and evolution of eukaryotic transcriptional enhancers. In: *Advances in genetics*. Vol. 96. Elsevier, 2016:143–206.

2. Trefflich S, Dalmolin RJ, Ortega JM, and Castro MA. Which came first, the transcriptional regulator or its target genes? An evolutionary perspective into the construction of eukaryotic regulons. Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms 2019:194472.

3. Spitz F and Furlong EE. Transcription factors: from enhancer binding to developmental control. Nature reviews genetics 2012;13:613–26.

4. Calo E and Wysocka J. Modification of enhancer chromatin: what, how, and why? Molecular cell 2013;49:825–37.

5. Halfon MS. Studying transcriptional enhancers: the founder fallacy, validation creep, and other biases. Trends in Genetics 2019;35:93–103.

6. Kim Tk, Hemberg M, Gray JM, et al. Widespread transcription at neuronal activity-regulated enhancers. Nature 2010;465:182–7.

7. Danko CG, Hyland SL, Core LJ, et al. Identification of active transcriptional regulatory elements from GRO-seq data. Nat Meth 2015;12:433–8.

8. Azofeifa JG, Allen MA, Hendrix JR, Read T, Rubin JD, and Dowell RD. Enhancer RNA profiling predicts transcription factor activity. Genome research 2018.

9. Rubin JD, Stanley JT, Sigauke RF, et al. Transcription factor enrichment analysis (TFEA) quantifies the activity of multiple transcription factors from a single experiment. Communications biology 2021;4:1–15.

10. Farh KKH, Marson A, Zhu J, et al. Genetic and epigenetic fine mapping of causal autoimmune disease variants. Nature 2015;518:337–43.

11. Preissl S, Gaulton KJ, and Ren B. Characterizing cis-regulatory elements using single-cell epigenomics. Nature Reviews Genetics 2022:1–23.

12. Affymetrix ENCODE Transcriptome Project and Cold Spring Harbor Laboratory ENCODE Transcriptome Project. Post-transcriptional processing generates a diversity of 5'-modified long and short RNAs. Nature 2009;457:1028–32.

13. The ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. Nature 2012;489:57–74.

14. Heintzman ND, Stuart RK, Hon G, et al. Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome. Nature genetics 2007;39:311–8.

15. Heintzman ND, Hon GC, Hawkins RD, et al. Histone modifications at human enhancers reflect global cell-type-specific gene expression. Nature 2009;459:108–12.

16. Rada-Iglesias A, Bajpai R, Swigut T, Brugmann SA, Flynn RA, and Wysocka J. A unique chromatin signature uncovers early developmental enhancers in humans. Nature 2011;470:279–83.

17. Luo Y, Hitz BC, Gabdank I, et al. New developments on the Encyclopedia of DNA Elements (ENCODE) data portal. Nucleic acids research 2020;48:D882–D889.

18. Carninci P, Kvam C, Kitamura A, et al. High-efficiency full-length cDNA cloning by biotinylated CAP trapper. Genomics 1996;37:327–36.

19. Andersson R, Gebhard C, Miguel-Escalada I, et al. An atlas of active enhancers across human cell types and tissues. Nature 2014;507:455–61.

20. Consortium TF, RIKEN PMI the, (DGT) C, et al. A promoter-level mammalian expression atlas. Nature 2014;507:462–70.

21. Core LJ, Martins AL, Danko CG, Waters CT, Siepel A, and Lis JT. Analysis of nascent RNA identifies a unified architecture of initiation regions at mammalian promoters and enhancers. Nat Genet 2014;46:1311–20.

22. Qin T, Lee C, Li S, et al. Comprehensive enhancer-target gene assignments improve gene set level interpretation of genome-wide regulatory data. Genome Biology 2022;23:1–30.

23. Hariprakash JM and Ferrari F. Computational biology solutions to identify enhancers-target gene pairs. Computational and structural biotechnology journal 2019;17:821–31.

24. Vučićević D, Corradin O, Ntini E, Scacheri PC, and Ørom UA. Long ncRNA expression associates with tissue-specific enhancers. Cell cycle 2015;14:253–60.

25. Whalen S, Truty RM, and Pollard KS. Enhancer–promoter interactions are encoded by complex genomic signatures on looping chromatin. Nature genetics 2016;48:488–96.

26. Cao Q, Anyansi C, Hu X, et al. Reconstruction of enhancer–target networks in 935 samples of human primary cells, tissues and cell lines. Nature genetics 2017;49:1428–36.

27. Fishilevich S, Nudel R, Rappaport N, et al. GeneHancer: genome-wide integration of enhancers and target genes in GeneCards. Database 2017;2017.

28. Greene CS, Krishnan A, Wong AK, et al. Understanding multicellular function and disease with human tissue-specific networks. Nature genetics 2015;47:569–76.

29. Castro DM, De Veaux NR, Miraldi ER, and Bonneau R. Multi-study inference of regulatory networks for more accurate models of gene regulation. PLoS computational biology 2019;15:e1006591.

30. Huynh-Thu VA, Irrthum A, Wehenkel L, and Geurts P. Inferring regulatory networks from expression data using tree-based methods. PloS one 2010;5:e12776.

31. Aibar S, González-Blas CB, Moerman T, et al. SCENIC: single-cell regulatory network inference and clustering. Nature methods 2017;14:1083–6.

32. Pliner HA, Packer JS, McFaline-Figueroa JL, et al. Cicero predicts cis-regulatory DNA interactions from single-cell chromatin accessibility data. Molecular cell 2018;71:858–71.

33. Sakaue S, Weinand K, Isaac S, et al. Tissue-specific enhancer-gene maps from multimodal single-cell data identify causal disease alleles. medRxiv 2022:2022–10.

34. Swanson E, Lord C, Reading J, et al. Simultaneous trimodal single-cell measurement of transcripts, epitopes, and chromatin accessibility using TEA-seq. Elife 2021;10:e63632.

35. Mimitou EP, Lareau CA, Chen KY, et al. Scalable, multimodal profiling of chromatin accessibility, gene expression and protein levels in single cells. Nature biotechnology 2021;39:1246–58.

36. Chepelev I, Wei G, Wangsa D, Tang Q, and Zhao K. Characterization of genome-wide enhancer-promoter interactions reveals co-expression of interacting genes and modes of higher order chromatin organization. Cell research 2012;22:490–503.

37. Li G, Ruan X, Auerbach RK, et al. Extensive promoter-centered chromatin interactions provide a topological basis for transcription regulation. Cell 2012;148:84–98.

38. Arner E, Daub CO, Vitting-Seerup K, et al. Transcribed enhancers lead waves of coordinated transcription in transitioning mammalian cells. Science 2015;347:1010–4.

39. Langfelder P and Horvath S. WGCNA: an R package for weighted correlation network analysis. BMC bioinformatics 2008;9:1–13.

40. Granger CW. Investigating causal relations by econometric models and cross-spectral methods. Econometrica: Journal of the Econometric Society 1969:424–38.

41. Core L and Lis J. Transcription Regulation Through Promoter-Proximal Pausing of RNA Polymerase II. Science 2008;319:1791.

42. Kwak H, Fuda NJ, Core LJ, and Lis JT. Precise Maps of RNA Polymerase Reveal How Promoters Direct Initiation and Pausing. Science 2013;339:950–3.

43. Schwalb B, Michel M, Zacher B, et al. TT-seq maps the human transient transcriptome. Science 2016;352:1225–8.

44. Schofield JA, Duffy EE, Kiefer L, Sullivan MC, and Simon MD. TimeLapse-seq: adding a temporal dimension to RNA sequencing through nucleoside recoding. Nature methods 2018;15:221–5.

45. Wissink EM, Vihervaara A, Tippens ND, and Lis JT. Nascent RNA analyses: tracking transcription and its regulation. Nature Reviews Genetics 2019;20:705–23.

46. Marbach D, Costello JC, Küffner R, et al. Wisdom of crowds for robust gene network inference. Nature methods 2012;9:796–804.

47. Haury AC, Mordelet F, Vera-Licona P, and Vert JP. TIGRESS: trustful inference of gene regulation using stability selection. BMC systems biology 2012;6:1–17.

48. Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological) 1996;58:267–88.

49. Faith JJ, Hayete B, Thaden JT, et al. Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. PLoS biology 2007;5:e8.

50. Chen X and Xuan J. Bayesian Inference of Gene Regulatory Network. In: *Bayesian Inference on Complicated Data*. IntechOpen, 2019.

51. Huynh-Thu VA and Sanguinetti G. Gene regulatory network inference: an introductory survey. Gene Regulatory Networks: Methods and Protocols 2019:1–23.

52. Spearman C. The proof and measurement of association between two things. The American journal of psychology 1987;100:441–71.

53. Breiman L. Random forests. Machine learning 2001;45:5–32.

54. Huynh-Thu VA, Irrthum A, Wehenkel L, and Geurts P. Inferring regulatory networks from expression data using tree-based methods. PloS one 2010;5:e12776.

55. Aibar S, González-Blas CB, Moerman T, et al. SCENIC: single-cell regulatory network inference and clustering. Nature methods 2017;14:1083–6.

56. Azofeifa J, Allen MA, Lladser M, and Dowell R. FStitch: A Fast and Simple Algorithm for Detecting Nascent RNA Transcripts. In: *Proceedings of the 5th ACM Conference on*

Bioinformatics, Computational Biology, and Health Informatics. BCB '14. Newport Beach, California: ACM, 2014:174–83.

57. Lidschreiber K, Jung LA, Emde H von der, et al. Transcriptionally active enhancers in human cancer cells. Molecular systems biology 2021;17:e9873.

58. Hunter S, Sigauke R, Stanley J, Allen M, and Dowell R. Protocol Variations in Run-On Transcription Dataset Preparation Produce Detectable Signatures in Sequencing Libraries. Research Square Preprint Server 2021.

59. Rubin JD, Stanley JT, Sigauke RF, et al. Transcription factor enrichment analysis (TFEA): Quantifying the activity of hundreds of transcription factors from a single experiment. Nature Communications Biology 2021.

60. Mahat DB, Kwak H, Booth GT, et al. Base-pair-resolution genome-wide mapping of active RNA polymerases using precision nuclear run-on (PRO-seq). Nature Protocols 2016;11:1455–76.

61. Allen MA, Mellert H, Dengler V, et al. Global analysis of p53-regulated transcription identifies its direct targets and unexpected regulatory mechanisms. eLife 2014;3:e02200.

62. Rothschild G and Basu U. Lingering Questions about Enhancer RNA and Enhancer Transcription-Coupled Genomic Instability. Trends in Genetics 2017;33:143–54.

63. Wang D, Garcia-Bassets I, Benner C, et al. Reprogramming transcription by distinct classes of enhancers functionally defined by eRNA. Nature 2011;474:390–4.

64. Kim SSY, Dziubek A, Alisa Lee S, and Kwak H. Nascent RNA sequencing of peripheral blood leukocytes reveal gene expression diversity. bioRxiv 2019.

65. Barbieri E, Hill C, Quesnel-Vallieres M, Barash Y, and Gardini A. Rapid and scalable profiling of nascent RNA with fastGRO. bioRxiv 2020.

66. Shivram H and Iyer VR. Identification and removal of sequencing artifacts produced by mispriming during reverse transcription in multiple RNA-seq technologies. RNA 2018;24:1266–74.

67. Sarantopoulou D, Tang SY, Ricciotti E, et al. Comparative evaluation of RNA-Seq library preparation methods for strand-specificity and low input. Scientific Reports 2019;9:13477.

68. Wang L, Felts SJ, Van Keulen VP, Pease LR, and Zhang Y. Exploring the effect of library preparation on RNA sequencing experiments. Genomics 2019;111:1752–9.

69. Andrysik Z, Galbraith MD, Guarnieri AL, et al. Identification of a core TP53 transcriptional program with highly distributed tumor suppressive activity. Genome research 2017;27:1645–57.

70. Adelman K and Lis JT. Promoter-proximal pausing of RNA polymerase II: emerging roles in metazoans. Nat Rev Genet 2012;13:720–31.

71. Roberts TC, Hart JR, Kaikkonen MU, Weinberg MS, Vogt PK, and Morris KV. Quantification of nascent transcription by bromouridine immunocapture nuclear run-on RT-qPCR. Nature protocols 2015;10:1198.

72. Orioli A, Praz V, Lhôte P, and Hernandez N. Human MAF1 targets and represses active RNA polymerase III genes by preventing recruitment rather than inducing long-term transcriptional arrest. Genome Research 2016;26:624–35.

73. Smith JP, Dutta AB, Sathyan KM, Guertin MJ, and Sheffield NC. PEPPRO: quality control and processing of nascent RNA profiling data. Genome Biology 2021;22:155.

74. Daubechies I. Ten lectures on wavelets. SIAM, 1992.

75. Lee GR, Gommers R, Waselewski F, Wohlfahrt K, and O'Leary A. PyWavelets: A Python package for wavelet analysis. Journal of Open Source Software 2019;4:1237.

76. Jonkers I, Kwak H, and Lis JT. Genome-wide dynamics of Pol II elongation and its interplay with promoter proximal pausing, chromatin, and exons. Elife 2014;3:e02407.

77. Day DS, Zhang B, Stevens SM, et al. Comprehensive analysis of promoter-proximal RNA polymerase II pausing across mammalian cell types. Genome Biology 2016;17:120.

78. Gao T and Qian J. EnhancerAtlas 2.0: an updated resource with enhancer annotation in 586 tissue/cell types across nine species. Nucleic Acids Research 2019;48:D58–D64.

79. Shiraki T, Kondo S, Katayama S, et al. Cap analysis gene expression for high-throughput analysis of transcriptional starting point and identification of promoter usage. Proceedings of the National Academy of Sciences 2003;100:15776–81.

80. Cardiello JF, Sanchez GJ, Allen MA, and Dowell RD. Lessons from eRNAs: understanding transcriptional regulation through the lens of nascent RNAs. Transcription 2020;11:3–18.

81. Azofeifa JG and Dowell RD. A generative model for the behavior of RNA polymerase. Bioinformatics 2016;33:227–34.

82. Hah N, Danko C, Core L, et al. A Rapid, Extensive, and Transient Transcriptional Response to Estrogen Signaling in Breast Cancer Cells. Cell 2011;145:622–34.

83. Hah N, Murakami S, Nagari A, Danko CG, and Kraus WL. Enhancer transcripts mark active estrogen receptor binding sites. Genome Research 2013;23:1210–23.

84. Shen H and Maki CG. Pharmacologic activation of p53 by small-molecule MDM2 antagonists. Current pharmaceutical design 2011;17:560–8.

85. Su Z, Łabaj PP, Li S, et al. A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium. Nature Biotechnology 2014;32:903–14.

86. Mootha VK, Lindgren CM, Eriksson KF, et al. PGC-1α-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. Nature Genetics 2003;34:267–73.

87. Subramanian A, Tamayo P, Mootha VK, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. Proceedings of the National Academy of Sciences of the United States of America 2005;102:15545–50.

88. Love MI, Huber W, and Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome biology 2014;15:550.

89. Min IM, Waterfall JJ, Core LJ, Munroe RJ, Schimenti J, and Lis JT. Regulating RNA polymerase pausing and transcription elongation in embryonic stem cells. Genes & Development 2011;25:742–54.

90. Mahat DB, Salamanca HH, Duarte FM, Danko CG, and Lis JT. Mammalian Heat Shock Response and Mechanisms Underlying Its Genome-wide Transcriptional Regulation. Molecular Cell 2016;62:63–78.

91. Dukler N, Booth GT, Huang YF, et al. Nascent RNA sequencing reveals a dynamic global transcriptional response at genes and enhancers to the natural medicinal compound celastrol. Genome Research 2017;27:1816–29.

92. Booth GT, Parua PK, Sansó M, Fisher RP, and Lis JT. Cdk9 regulates a promoter-proximal checkpoint to modulate RNA polymerase II elongation rate in fission yeast. Nature Communications 2018;9:543.

93. Aoi Y, Smith ER, Shah AP, et al. NELF Regulates a Promoter-Proximal Step Distinct from RNA Pol II Pause-Release. Molecular Cell 2020;78:261–274.e5.

94. Andersson R, Refsing Andersen P, Valen E, et al. Nuclear stability and transcriptional directionality separate functionally distinct RNA species. Nat Commun 2014;5.

95. Wang J, Zhao Y, Zhou X, Hiebert SW, Liu Q, and Shyr Y. Nascent RNA sequencing analysis provides insights into enhancer-mediated gene regulation. BMC Genomics 2018;19:633.

96. Wissink EM, Vihervaara A, Tippens ND, and Lis JT. Nascent RNA analyses: tracking transcription and its regulation. Nat Rev Genet 2019;20:705–23.

97. Marioni J, Mason C, Mane S, Stephens M, and Gilad Y. RNA-seq: An assessment of technical reproducibility and comparison with gene expression arrays. Genome Research 2008;18:1509.

98. Steinparzer I, Sedlyarov V, Rubin JD, et al. Transcriptional Responses to IFN-γ Require Mediator Kinase-Dependent Pause Release and Mechanistically Distinct CDK8 and CDK19 Functions. Molecular Cell 2019;76:485–499.e8.

99. Leek JT, Scharpf RB, Bravo HC, et al. Tackling the widespread and critical impact of batch effects in high-throughput data. Nature Reviews Genetics 2010;11:733–9.

100. Goh WWB, Wang W, and Wong L. Why Batch Effects Matter in Omics Data, and How to Avoid Them. Trends in Biotechnology 2017;35:498–507.

101. Somekh J, Shen-Orr SS, and Kohane IS. Batch correction evaluation framework using a-priori gene-gene associations: applied to the GTEx dataset. BMC Bioinformatics 2019;20:268.

102. Zhang Y, Parmigiani G, and Johnson WE. ComBat-seq: batch effect adjustment for RNA-seq count data. NAR Genomics and Bioinformatics 2020;2.

103. Sanitá Lima M and Smith DR. Don't just dump your data and run. EMBO reports 2017;18:2087–9.

104. Levandowski CB, Jones T, Gruca M, Ramamoorthy S, Dowell RD, and Taatjes DJ. The Δ40p53 isoform inhibits p53-dependent eRNA transcription and enables regulation by signal-specific transcription factors during p53 activation. PLoS Biology 2021;19:e3001364.

105. Van Rossum G and Drake FL. Python 3 Reference Manual. Scotts Valley, CA: CreateSpace, 2009.

106. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011;12:2825–30.

107. Hunter JD. Matplotlib: A 2D graphics environment. Computing in Science & Engineering 2007;9:90–5.

108. Wickham H. ggplot2: Elegant Graphics for Data Analysis. Springer, 2016.

109. Wilke CO. cowplot: Streamlined Plot Theme and Plot Annotations for 'ggplot2'. 2020.

110. Meyer D, Dimitriadou E, Hornik K, Weingessel A, and Leisch F. e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. 2021.

111. R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria, 2019.

112. Kuhn M. Building Predictive Models in R Using the caret Package. Journal of Statistical Software, Articles 2008;28:1–26.

113. Core LJ, Waterfall JJ, and Lis JT. Nascent RNA Sequencing Reveals Widespread Pausing and Divergent Initiation at Human Promoters. Science 2008;322:1845–8.

114. Nojima T, Gomes T, Grosso ARF, et al. Mammalian NET-Seq Reveals Genome-wide Nascent Transcription Coupled to RNA Processing. Cell 2015;161:526–40.

115. Chu T, Rice EJ, Booth GT, et al. Chromatin run-on reveals nascent RNAs that differentiate normal and malignant brain tissue. bioRxiv 2017.

116. Li W, Notani D, Ma Q, et al. Functional roles of enhancer RNAs for oestrogen-dependent transcriptional activation. Nature 2013;498:516–20.

117. Sartorelli V and Lauberth SM. Enhancer RNAs are an important regulatory layer of the epigenome. Nature Structural & Molecular Biology 2020;27:521–8.

118. Allison KA, Kaikkonen MU, Gaasterland T, and Glass CK. Vespucci: a system for building annotated databases of nascent transcripts. Nucleic Acids Research 2013.

119. Chae M, Danko CG, and Kraus WL. groHMM: a computational tool for identifying unannotated and cell type-specific transcription units from global run-on sequencing data. BMC Bioinformatics 2015;16:222.

120. Fuda NJ, Ardehali MB, and Lis JT. Defining mechanisms that regulate RNA polymerase II transcription in vivo. Nature 2009;461:186–92.

121. Azofeifa JG, Allen MA, Lladser ME, and Dowell RD. An Annotation Agnostic Algorithm for Detecting Nascent RNA Transcripts in GRO-Seq. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2017;14:1070–81.

122. Wang Z, Chu T, Choate LA, and Danko CG. Identification of regulatory elements from nascent transcription using dREG. Genome Research 2019;29:293–303.

123. Zhao Y, Dukler N, Barshad G, Toneyan S, Danko CG, and Siepel A. Deconvolution of Expression for Nascent RNA sequencing data (DENR) highlights pre-RNA isoform diversity in human cells. Bioinformatics 2021.

124. Sanchez GJ, Richmond PA, Bunker EN, et al. Genome-wide dose-dependent inhibition of histone deacetylases studies reveal their roles in enhancer remodeling and suppression of oncogenic super-enhancers. Nucleic Acids Research 2017;46:1756–76.

125. Sprang M, Krüger M, Andrade-Navarro MA, and Fontaine JF. Statistical guidelines for quality control of next-generation sequencing techniques. Life Science Alliance 2021;4.

126. Bushnell B. BBMap: a fast, accurate, splice-aware aligner. 2014.

127. Daley T and Smith AD. Predicting the molecular complexity of sequencing libraries. Nature Methods 2013;10:325.

128. 1000 Genome Project Data Processing Subgroup, Wysoker A, Handsaker B, et al. The Sequence Alignment/Map format and SAMtools. Bioinformatics 2009;25:2078–9.

129. Ramírez F, Diehl S, Manke T, Dündar F, and Grüning BA. deepTools: a flexible platform for exploring deep-sequencing data. Nucleic Acids Research 2014;42:W187–W191.

130. Quinlan AR and Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics 2010;26:841–2.

131. Robinson JT, Thorvaldsdóttir H, Winckler W, et al. Integrative genomics viewer. Nature Biotechnology 2011;29:24.

132. Proudfoot NJ. Transcriptional termination in mammals: Stopping the RNA polymerase II juggernaut. Science (New York, N.Y.) 2016;352:aad9926–aad9926.

133. Ramsköld D, Wang ET, Burge CB, and Sandberg R. An Abundance of Ubiquitously Expressed Genes Revealed by Tissue Transcriptome Sequence Data. PLOS Computational Biology 2009;5:1–11.

134. Shandilya J and Roberts SGE. The transcription cycle in eukaryotes: From productive initiation to RNA polymerase II recycling. Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms 2012;1819:391–400.

135. Costa-Silva J, Domingues D, and Lopes FM. RNA-Seq differential expression analysis: An extended review and a software tool. PLOS ONE 2017;12:1–18.

136. Soneson C and Delorenzi M. A comparison of methods for differential expression analysis of RNA-seq data. BMC Bioinformatics 2013;14:91.

137. Anders S and Huber W. Differential expression analysis for sequence count data. Genome Biology 2010;11:R106.

138. Robinson MD, McCarthy DJ, and Smyth GK. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics (Oxford, England) 2010;26:139–40.

139. Kulakovskiy IV, Medvedeva YA, Schaefer U, et al. HOCOMOCO: a comprehensive collection of human transcription factor binding sites models. Nucleic Acids Research 2013;41:D195–D202.

140. Puc J, Kozbial P, Li W, et al. Ligand-dependent enhancer activation regulated by topoisomerase-I activity. Cell 2015;160:367–80.

141. Tripodi IJ, Allen MA, and Dowell RD. Detecting Differential Transcription Factor Activity from ATAC-Seq Data. Molecules (Basel, Switzerland) 2018;23.

142. Sasse SK, Gruca M, Allen MA, et al. Nascent transcript analysis of glucocorticoid crosstalk with TNF defines primary and cooperative inflammatory repression. Genome research 2019;29:1753–65.

143. Davis CA, Hitz BC, Sloan CA, et al. The Encyclopedia of DNA elements (ENCODE): data portal update. Nucleic Acids Research 2018;46:D794–D801.

144. Lambert SA, Jolma A, Campitelli LF, et al. The Human Transcription Factors. Cell 2018;172:650–65.

145. Kulakovskiy IV, Vorontsov IE, Yevshin IS, et al. HOCOMOCO: expansion and enhancement of the collection of transcription factor binding sites models. Nucleic acids research 2016;44:D116–D125.

146. Fornes O, Castro-Mondragon JA, Khan A, et al. JASPAR 2020: update of the open-access database of transcription factor binding profiles. Nucleic Acids Research 2019;48:D87–D92.

147. Whitfield TW, Wang J, Collins PJ, et al. Functional analysis of transcription factor binding sites in human promoters. Genome Biology 2012;13:R50.

148. Spivakov M. Spurious transcription factor binding: Non-functional or genetically redundant? BioEssays 2014;36:798–806.

149. Cusanovich DA, Pavlovic B, Pritchard JK, and Gilad Y. The Functional Consequences of Variation in Transcription Factor Binding. PLoS Genet 2014;10:e1004226.

150. MacQuarrie KL, Fong AP, Morse RH, and Tapscott SJ. Genome-wide transcription factor binding: beyond direct target regulation. Trends Genet 2011;27:141–8.

151. Balwierz PJ, Pachkov M, Arnold P, Gruber AJ, Zavolan M, and Nimwegen E van. ISMARA: automated modeling of genomic signals as a democracy of regulatory motifs. Genome Research 2014;24:869–84.

152. Jiang S and Mortazavi A. Integrating ChIP-seq with other functional genomics data. Briefings in Functional Genomics 2018;17:104–15.

153. Hao S and Baltimore D. The stability of mRNA influences the temporal order of the induction of genes encoding inflammatory molecules. Nature Immunology 2009;10:281–8.

154. Tani H, Mizutani R, Salam KA, et al. Genome-wide determination of RNA stability reveals hundreds of short-lived noncoding transcripts in mammals. Genome research 2012;22:947–56.

155. Gallego Romero I, Pai AA, Tung J, and Gilad Y. RNA-seq: impact of RNA degradation on transcript quantification. BMC Biology 2014;12:42.

156. Paulsen MT, Veloso A, Prasad J, et al. Use of Bru-Seq and BruChase-Seq for genome-wide assessment of the synthesis and stability of RNA. Methods 2014;67:45–54.

157. Muhar M, Ebert A, Neumann T, et al. SLAM-seq defines direct gene-regulatory functions of the BRD4-MYC axis. Science 2018;360:800–5.

158. Yao L, Berman BP, and Farnham PJ. Demystifying the secret mission of enhancers: linking distal regulatory elements to target genes. Crit Rev Biochem Mol Biol 2015;50:550–73.

159. Luo X, Chae M, Krishnakumar R, Danko CG, and Kraus WL. Dynamic reorganization of the AC16 cardiomyocyte transcriptome in response to TNFα signaling revealed by integrated genomic analyses. BMC Genomics 2014;15:155–5.

160. Judd J, Wojenski LA, Wainman LM, et al. A rapid, sensitive, scalable method for Precision Run-On sequencing (PRO-seq). bioRxiv 2020.

161. Tome JM, Tippens ND, and Lis JT. Single-molecule nascent RNA sequencing identifies regulatory domain architecture at promoters and enhancers. Nature Genetics 2018;50:1533–41.

162. Bannister AJ and Kouzarides T. Regulation of chromatin by histone modifications. Cell Research 2011;21:381–95.

163. Bailey TL and Machanick P. Inferring direct DNA binding from ChIP-seq. Nucleic Acids Research 2012;40:e128–e128.

164. Lesluyes T, Johnson J, Machanick P, and Bailey TL. Differential motif enrichment analysis of paired ChIP-seq experiments. BMC Genomics 2014;15.

165. Sasse SK, Gruca M, Allen MA, et al. Nascent transcript analysis of glucocorticoid crosstalk with TNF defines primary and cooperative inflammatory repression. Genome Research 2019;29:1753–65.

166. Gruca MA, Gohde MA, and Dowell RD. Annotation Agnostic Approaches to Nascent Transcription Analysis: Fast Read Stitcher and Transcription Fit. Methods in Molecular Biology 2020;to appear.

167. Seila AC, Calabrese JM, Levine SS, et al. Divergent transcription from active promoters. Science 2008;322:1849–51.

168. Scruggs BS, Gilchrist DA, Nechaev S, et al. Bidirectional Transcription Arises from Two Distinct Hubs of Transcription Factor Binding and Active Chromatin. Molecular Cell 2015;58:1101–12.

169. Nielsen MM, Tataru P, Madsen T, Hobolth A, and Pedersen JS. Regmex: a statistical tool for exploring motifs in ranked sequence lists from genomics experiments. Algorithms for Molecular Biology 2018;13:17.

170. Karsli Uzunbas G, Ahmed F, and Sammons MA. Control of p53-dependent transcription and enhancer activity by the p53 family member p63. Journal of Biological Chemistry 2019;294:10720–36.

171. McLeay RC and Bailey TL. Motif Enrichment Analysis: a unified framework and an evaluation on ChIP data. BMC Bioinformatics 2010;11:165.

172. Kulakovskiy IV, Vorontsov IE, Yevshin IS, 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 2018;46:D252–D259.

173. Forrest ARR, Kawaji H, Rehli M, et al. A promoter-level mammalian expression atlas. Nature 2014;507:462–70.

174. Baillie JK, Arner E, Daub C, et al. Analysis of the human monocyte-derived macrophage transcriptome and response to lipopolysaccharide provides new insights into genetic aetiology of inflammatory bowel disease. PLoS Genetics 2017;13.

175. Molle C, Goldman M, and Goriely S. Critical role of the IFN-stimulated gene factor 3 complex in TLR-mediated IL-27p28 gene expression revealing a two-step activation process. The Journal of Immunology 2010;184:1784–92.

176. Nan J, Wang Y, Yang J, and Stark GR. IRF9 and unphosphorylated STAT2 cooperate with NF-κB to drive IL6 expression. Proceedings of the National Academy of Sciences 2018;115:3906–11.

177. Curina A, Termanini A, Barozzi I, et al. High constitutive activity of a broad panel of housekeeping and tissue-specific cis-regulatory elements depends on a subset of ETS proteins. Genes & Development 2017;31:399–412.

178. Joo M, Wright JG, Hu NN, et al. Yin Yang 1 enhances cyclooxygenase-2 gene expression in macrophages. American Journal of Physiology. Lung Cellular and Molecular Physiology 2007;292:L1219–1226.

179. Zhang XC, Liang HF, Luo XD, et al. YY1 promotes IL-6 expression in LPS-stimulated BV2 microglial cells by interacting with p65 to promote transcriptional activation of IL-6. Biochemical and biophysical research communications 2018;502:269–75.

180. Jüttner S, Cramer T, Wessler S, et al. Helicobacter pylori stimulates host cyclooxygenase-2 gene transcription: critical importance of MEK/ERK-dependent activation of USF1/-2 and CREB transcription factors. Cellular Microbiology 2003;5:821–34.

181. Xue HH, Bollenbacher-Reilley J, Wu Z, et al. The Transcription Factor GABP Is a Critical Regulator of B Lymphocyte Development. Immunity 2007;26:421–31.

182. Siednienko J, Maratha A, Yang S, Mitkiewicz M, Miggin SM, and Moynagh PN. Nuclear factor κB subunits RelB and cRel negatively regulate Toll-like receptor 3-mediated β-interferon production via induction of transcriptional repressor protein YY1. Journal of Biological Chemistry 2011;286:44750–63.

183. McDowell IC, Barrera A, D'Ippolito AM, et al. Glucocorticoid receptor recruits to enhancers and drives activation by motif-directed binding. Genome Research 2018;28:1272–84.

184. Li Q, Su A, Chen J, Lefebvre YA, and Hache RJG. Attenuation of Glucocorticoid Signaling through Targeted Degradation of p300 via the 26S Proteasome Pathway. Molecular Endocrinology 2002;16:2819–27.

185. Jin Q, Yu LR, Wang L, et al. Distinct roles of GCN5/PCAF-mediated H3K9ac and CBP/p300-mediated H3K18/27ac in nuclear receptor transactivation. The EMBO Journal 2011;30:249–62.

186. Weinert BT, Narita T, Satpathy S, et al. Time-Resolved Analysis Reveals Rapid Dynamics and Broad Scope of the CBP/p300 Acetylome. Cell 2018;174:231–244.e12.

187. Raisner R, Kharbanda S, Jin L, et al. Enhancer activity requires CBP/P300 bromodomain-dependent histone H3K27 acetylation. Cell Reports 2018;24:1722–9.

188. Tripodi IJ, Chowdhury M, Gruca M, and Dowell RD. Combining signal and sequence to detect RNA polymerase initiation in ATAC-seq data. PLOS ONE 2020;15:1–18.

189. Quinlan AR and Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics 2010;26:841–2.

190. Grant CE, Bailey TL, and Noble WS. FIMO: scanning for occurrences of a given motif. Bioinformatics 2011;27:1017–8.

191. Lam MTY, Cho H, Lesch HP, et al. Rev-Erbs repress macrophage gene expression by inhibiting enhancer-directed transcription. Nature 2013;498:511–5.

192. Aronesty E. Comparison of Sequencing Utility Programs. The Open Bioinformatics Journal 2013;7:1–8.

193. Kim D, Langmead B, and Salzberg SL. HISAT: a fast spliced aligner with low memory requirements. Nature methods 2015;12:357–60.

194. Li H, Handsaker B, Wysoker A, et al. The sequence alignment/map format and SAMtools. Bioinformatics 2009;25:2078–9.

195. Institute B. Picard toolkit. `http://broadinstitute.github.io/picard/`. 2019.

196. Zhang Y, Liu T, Meyer CA, et al. Model-based analysis of ChIP-Seq (MACS). Genome biology 2008;9:1–9.

197. Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, and Notredame C. Nextflow enables reproducible computational workflows. Nature Biotechnology 2017;35:316–9.

198. Tripodi IJ and Gruca MA. Nascent-Flow. 2018.

199. Hartigan JA and Wong MA. Algorithm AS 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics) 1979;28:100–8.

200. Bholowalia P and Kumar A. EBK-means: A clustering technique based on elbow method and k-means in WSN. International Journal of Computer Applications 2014;105:975–8887.

201. Dowell R. TFEA Figure Data. OSF. April 1. osf.io/wprmd. 2021.

202. Spitz F and Furlong EEM. Transcription factors: from enhancer binding to developmental control. Nat Rev Genet 2012;13:613–26.

203. Consortium G. The GTEx Consortium atlas of genetic regulatory effects across human tissues. Science 2020;369:1318–30.

204. Consortium EP et al. An integrated encyclopedia of DNA elements in the human genome. Nature 2012;489:57.

205. Davis CA, Hitz BC, Sloan CA, et al. The Encyclopedia of DNA elements (ENCODE): data portal update. Nucleic acids research 2018;46:D794–D801.

206. Abugessaisa I, Ramilowski JA, Lizio M, et al. FANTOM enters 20th year: expansion of transcriptomic atlases and functional annotation of non-coding RNAs. Nucleic Acids Research 2021;49:D892–D898.

207. Lizio M, Harshbarger J, Shimoji H, et al. Gateways to the FANTOM5 promoter level mammalian expression atlas. Genome biology 2015;16:1–14.

208. Hirabayashi S, Bhagat S, Matsuki Y, et al. Dynamics and Topology of Human Transcribed Cis-regulatory Elements. bioRxiv 2019:689968.

209. Whalen S, Truty RM, and Pollard KS. Enhancer–promoter interactions are encoded by complex genomic signatures on looping chromatin. Nature genetics 2016;48:488–96.

210. Carullo NV, Phillips III RA, Simon RC, et al. Enhancer RNAs predict enhancer–gene regulatory links and are critical for enhancer function in neuronal systems. Nucleic acids research 2020;48:9550–70.

211. Mahat DB, Salamanca HH, Duarte FM, Danko CG, and Lis JT. Mammalian Heat Shock Response and Mechanisms Underlying Its Genome-wide Transcriptional Regulation. Molecular Cell 2016;62:63–78.

212. Lettice LA, Heaney SJ, Purdie LA, et al. A long-range Shh enhancer regulates expression in the developing limb and fin and is associated with preaxial polydactyly. Human molecular genetics 2003;12:1725–35.

213. Visel A, Rubin EM, and Pennacchio LA. Genomic views of distant-acting enhancers. Nature 2009;461:199–205.

214. Edgar R, Domrachev M, and Lash AE. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic acids research 2002;30:207–10.

215. Barrett T, Wilhite SE, Ledoux P, et al. NCBI GEO: archive for functional genomics data sets—update. Nucleic acids research 2012;41:D991–D995.

216. Leinonen R, Sugawara H, Shumway M, and Collaboration INSD. The sequence read archive. Nucleic acids research 2010;39:D19–D21.

217. Danko CG, Hyland SL, Core LJ, et al. Identification of active transcriptional regulatory elements from GRO-seq data. Nat Meth 2015;12:433–8.

218. Danko CG, Choate LA, Marks BA, et al. Dynamic evolution of regulatory element ensembles in primate CD4+ T cells. Nature Ecology & Evolution 2018;2:537–48.

219. Yao L, Liang J, Ozer A, Leung AKY, Lis JT, and Yu H. A comparison of experimental assays and analytical methods for genome-wide identification of active enhancers. Nature Biotechnology 2022:1–10.

220. Gao T and Qian J. EnhancerAtlas 2.0: an updated resource with enhancer annotation in 586 tissue/cell types across nine species. Nucleic acids research 2020;48:D58–D64.

221. Cabili MN, Trapnell C, Goff L, et al. Integrative annotation of human large intergenic noncoding RNAs reveals global properties and specific subclasses. Genes & Development 2011;25:1915–27.

222. Everaert C, Volders PJ, Morlion A, Thas O, and Mestdagh P. SPECS: a non-parametric method to identify tissue-specific molecular features for unbalanced sample groups. BMC Bioinformatics 2020;21:58.

223. Xu H, Zhang S, Yi X, Plewczynski D, and Li MJ. Exploring 3D chromatin contacts in gene regulation: the evolution of approaches for the identification of functional enhancer-promoter interaction. Computational and structural biotechnology journal 2020;18:558–70.

224. Wang J, Zhao Y, Zhou X, Hiebert SW, Liu Q, and Shyr Y. Nascent RNA sequencing analysis provides insights into enhancer-mediated gene regulation. BMC genomics 2018;19:1–18.

225. Gasperini M, Hill AJ, McFaline-Figueroa JL, et al. A genome-wide framework for mapping gene regulation via cellular genetic screens. Cell 2019;176:377–90.

226. Langfelder P and Horvath S. WGCNA: an R package for weighted correlation network analysis. BMC Bioinformatics 2008;9:559.

227. Hinrichs AS, Karolchik D, Baertsch R, et al. The UCSC genome browser database: update 2006. Nucleic acids research 2006;34:D590–D598.

228. Wang Z, Chu T, Choate LA, and Danko CG. Identification of regulatory elements from nascent transcription using dREG. Genome Research 2019;29:293–303.

229. Kent WJ, Zweig AS, Barber G, Hinrichs AS, and Karolchik D. BigWig and BigBed: enabling browsing of large distributed datasets. Bioinformatics 2010;26:2204–7.

230. Liao Y, Smyth GK, and Shi W. featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. Bioinformatics 2014;30:923–30.

231. Hunter S, Sigauke RF, Stanley JT, Allen MA, and Dowell RD. Protocol variations in run-on transcription dataset preparation produce detectable signatures in sequencing libraries. BMC genomics 2022;23:1–18.

232. Li B and Dewey CN. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. BMC bioinformatics 2011;12:1–16.

233. Wagner GP, Kin K, and Lynch VJ. Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples. Theory in biosciences 2012;131:281–5.

234. Bailey TL, Johnson J, Grant CE, and Noble WS. The MEME Suite. Nucleic Acids Research 2015;43:W39–W49.

235. Zhang B and Horvath S. A general framework for weighted gene co-expression network analysis. Statistical applications in genetics and molecular biology 2005;4.

236. Benjamini Y and Hochberg Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal statistical society: series B (Methodological) 1995;57:289–300.

237. Bravo González-Blas C, Minnoye L, Papasokrati D, et al. cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data. Nature methods 2019;16:397–400.

238. Bravo González-Blas C, De Winter S, Hulselmans G, et al. SCENIC+: single-cell multiomic inference of enhancers and gene regulatory networks. bioRxiv 2022:2022–8.

239. Chu T, Wang Z, Chou SP, and Danko CG. Discovering Transcriptional Regulatory Elements From Run-On and Sequencing Data Using the Web-Based dREG Gateway. Current protocols in bioinformatics 2019;66:e70.

240. Yao L, Liang J, Ozer A, Leung AKY, Lis JT, and Yu H. A comparison of experimental assays and analytical methods for genome-wide identification of active enhancers. Nature biotechnology 2022;40:1056–65.

241. Arnold A, Liu Y, and Abe N. Temporal Causal Modeling with Graphical Granger Methods. In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2007:66–75.

242. Sasse SK, Gruca M, Allen MA, et al. Nascent transcript analysis of glucocorticoid crosstalk with TNF defines primary and cooperative inflammatory repression. Genome Research 2019.

243. Allen MA, Mellert H, Dengler V, et al. Global analysis of p53-regulated transcription identifies its direct targets and unexpected regulatory mechanisms. eLife 2014;3:e02200.

244. Niskanen EA, Malinen M, Sutinen P, et al. Global SUMOylation on active chromatin is an acute heat stress response restricting transcription. Genome Biology 2015;16:153.

245. Dukler N, Booth GT, Huang YF, et al. Nascent RNA sequencing reveals a dynamic global transcriptional response at genes and enhancers to the natural medicinal compound celastrol. Genome Research 2017;27:1816–29.

APPENDIX A

ABBREVIATIONS

NRO	Nuclear Run-on.
PRO-seq	Precision Run-on Sequencing.
GRO-seq	Global Run-on Sequencing.
TT-seq	Transient transcriptome sequencing.
RO-seq	Run-on Sequencing.
RNAPII	RNA polymerase II.
eRNA	Enhancer RNA.
DWT	Discrete Wavelet Transform.
GRN	Gene Regulatory Network.
TRN	Transcriptional Regulatory Network.
TSS	Transcription Start Site.

APPENDIX B

SUPPLEMENT TO CHAPTER 2

The reader is encouraged to refer to the primary publication for the Supplementary Tables, described here for completeness.

Supplemental Table 1 — Sample Information

Sample information for all RO-seq libraries used in analyses. Information is as follows: cell type, treatment, time point, enrichment protocol, library preparation method, replicate number, depth, complexity metrics, and SRA identifiers

Additional File 2

Supplemental Figure 2.1 Complexity curves and read distributions of public and in-house GRO-RPR datasets, indicating trends of lower quality for our libraries with this preparation.

Supplementary Figures

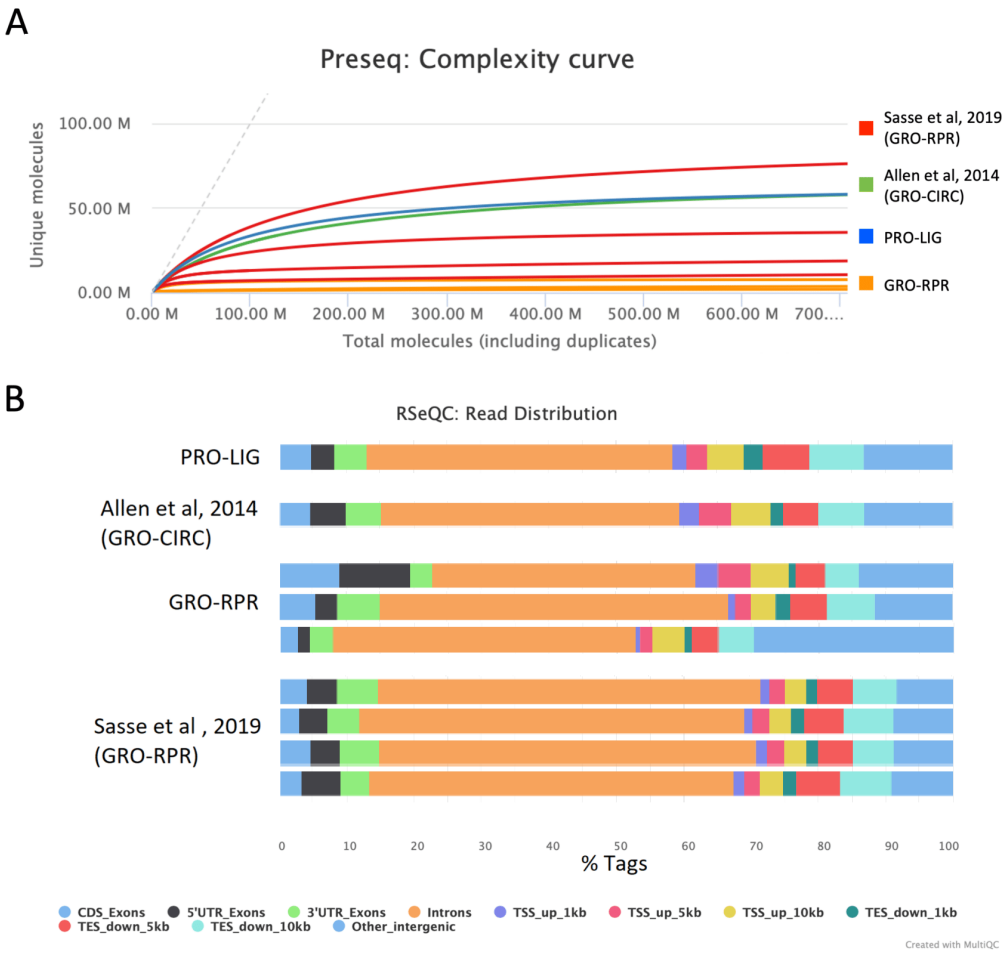


Figure 2.1: (A) Complexity curves of 4 publicly available GRO-RPR datasets (242: SRR8429046, SRR8429047, SRR8429054, SRR8429055), our in-house generated GRO-RPR datasets (see Supplemental Table 1, Materials and Methods. SRR14355654, SRR14355657, and SRR14355653), one PRO-LIG dataset (SRR14355672), and one publicly available GRO-CIRC dataset (243: SRR1105737). While the most complex library was from a GRO-RPR preparation, we found that the majority of these RPR datasets tended to be of lower complexity. Despite this trend, we contend that there is insufficient data to determine whether this is a fault of our handling or a feature of RPR library preparations with RO-seq datasets. (B) Read distribution plots of the datasets described in (A). While many regions were consistent regardless of protocol, was considerable variation in read distributions within the GRO-RPR datasets, especially comparing the proportion of reads found in 5' UTR regions and intergenic regions. As such, we chose to summarize additional quality metrics and library characteristics for our GRO-RPR datasets (Fig. 2.2D,2.3D,2.4B, see also Supplementary Table 1), with the understanding that their poor quality influence these metrics. GRO-RPR datasets were otherwise not used for further comparative analysis. From top to bottom, the samples are as follows: SRR14355672 (PRO-LIG); SRR1105737 (GRO-CIRC); SRR14355654, SRR14355657, SRR14355653 (GRO-RPR); SRR8429046, SRR8429047, SRR8429054, SRR8429055242.

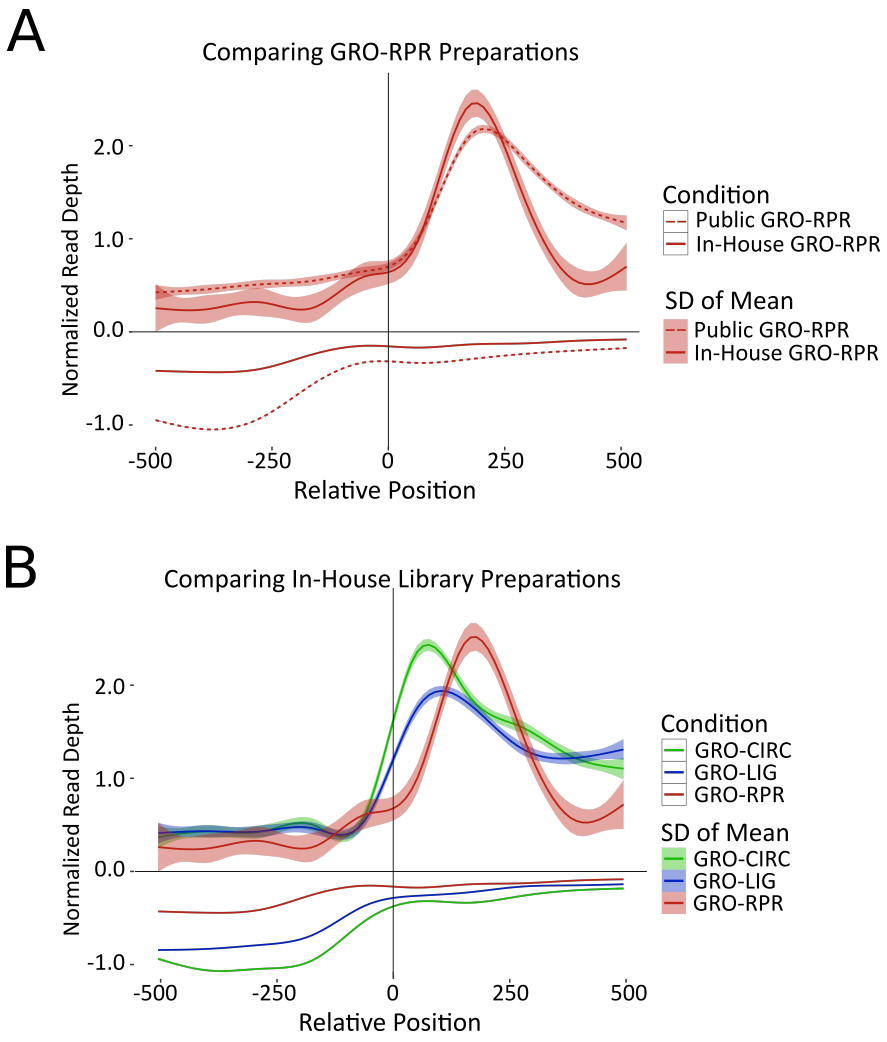


Figure 2.2: (A) Metagenes of public GRO-RPR and in-house GRO-RPR libraries. All GRO-RPR datasets display a similar gap in coverage near the annotated TSS. Note that each public GRO-RPR library was subsampled to 20 million reads such that the comparison was performed at the same depth. (Public GRO-RPR data: SRR8429046, SRR8429047, SRR8429054, SRR8429055) (B) Metagenes of in-house libraries, including GRO-RPR libraries. Each library was subsampled to 20 million reads to match the lower depth of the GRO-RPR libraries. Additionally, we note that our GRO-RPR libraries are lower complexity. For both metaplots, genes shorter than 2000 bp, genes with significant signal 1 kb upstream (>1% of upstream bases covered), and genes with low coverage (TPM < .01) were removed. (n=1428) (GRO-CIRC: SRR1105736, SRR1105737. GRO-LIG: SRR14355673, SRR14355674. GRO-RPR: SRR14355653, SRR14355654, SRR14355657)

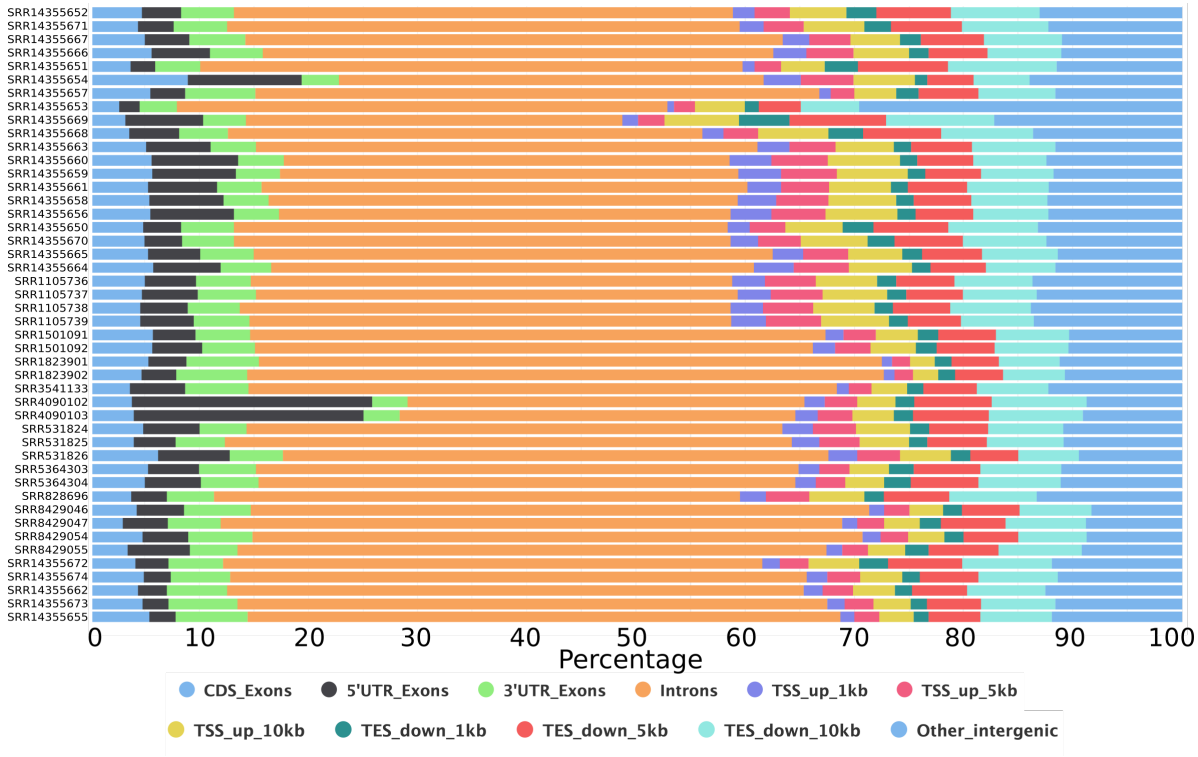


Figure 2.3: Read distributions were generated from RSeQC, see Materials and Methods, Supplemental Table 1.

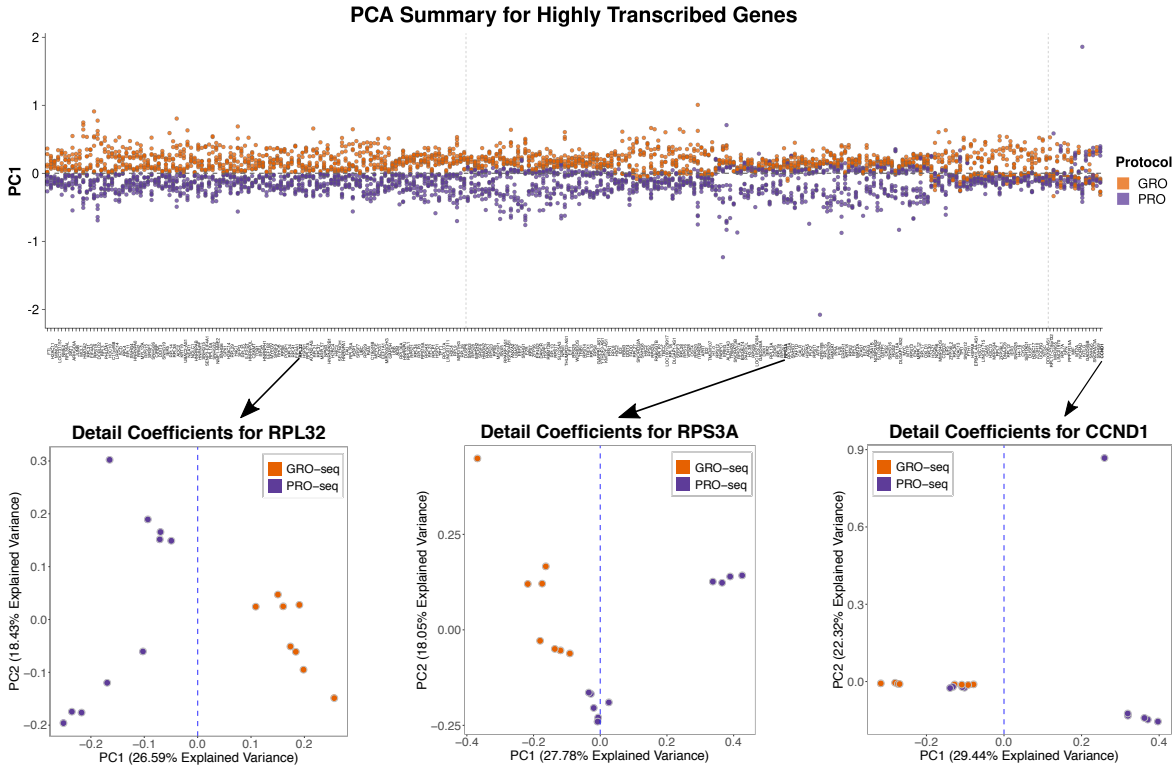


Figure 2.4: (Top) PC1 effectively separates GRO and PRO libraries for 39.8% (117 genes) of the set of 294 highly transcribed genes while 55.1% (162 genes) of the genes separates the libraries on PC1 and PC2. (Bottom) PC1 and PC2 results for each library are shown for three example genes: RPL32 (separates on PC1), RPS3A (separates on a plane in the PC1/PC2 space), and CCND1 (not separable with these PC).

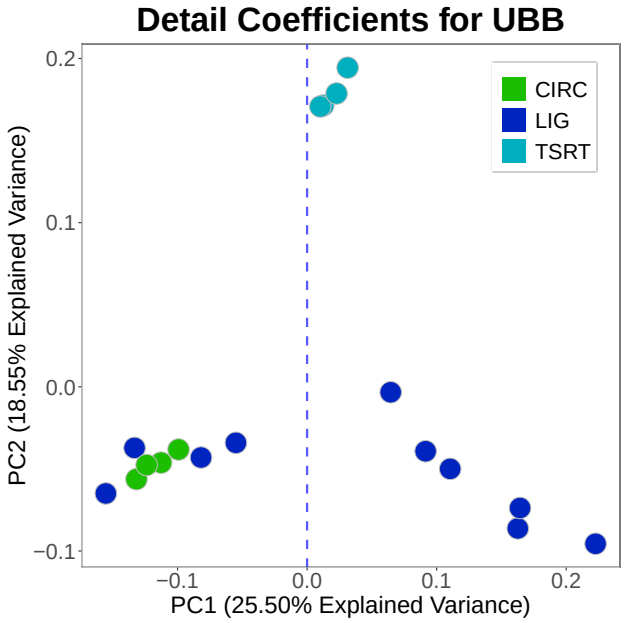


Figure 2.5: PCA results for UBB locus, as in Figure 2.2F. Results are colored by library preparation method. At this locus, the results cluster less distinctly by library preparation method, compared to the enrichment protocol.

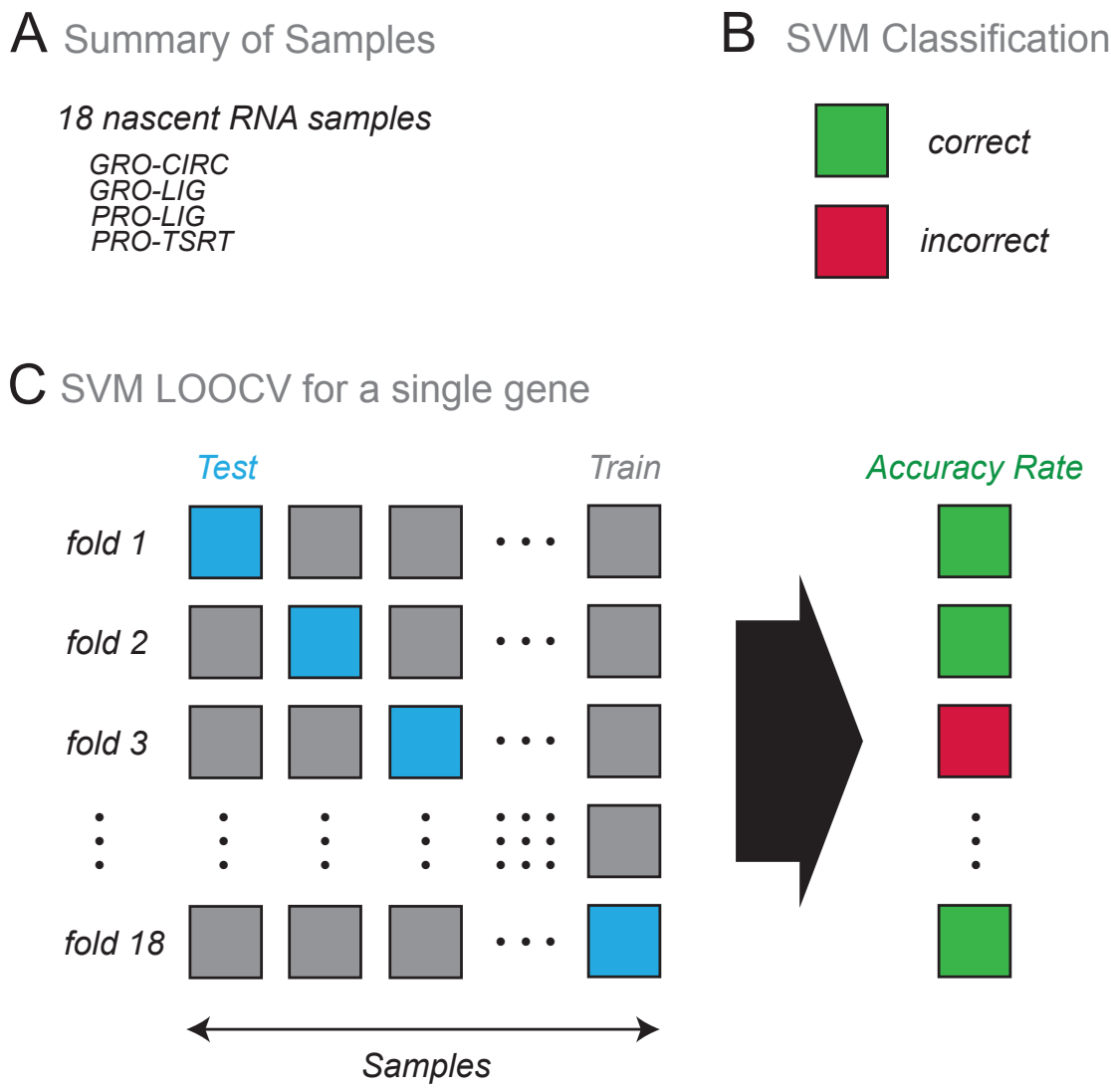


Figure 2.6: (A) Eighteen nascent RNA sequencing samples were used as input, from GRO-CIRC, GRO-LIG, PRO-LIG and PRO-TSRT libraries. (B) SVM classification was considered correct if the protocol was inferred from the data. (C) Given a gene, eighteen consecutive leave one out tests were performed. In each, one sample was selected as a test sample while the other samples were used as the training set. The SVM classification was subsequently evaluated for accuracy. Based on the SVM LOOCV method, a majority of the genes (>75%) accurately classified the protocol for the 18 samples.

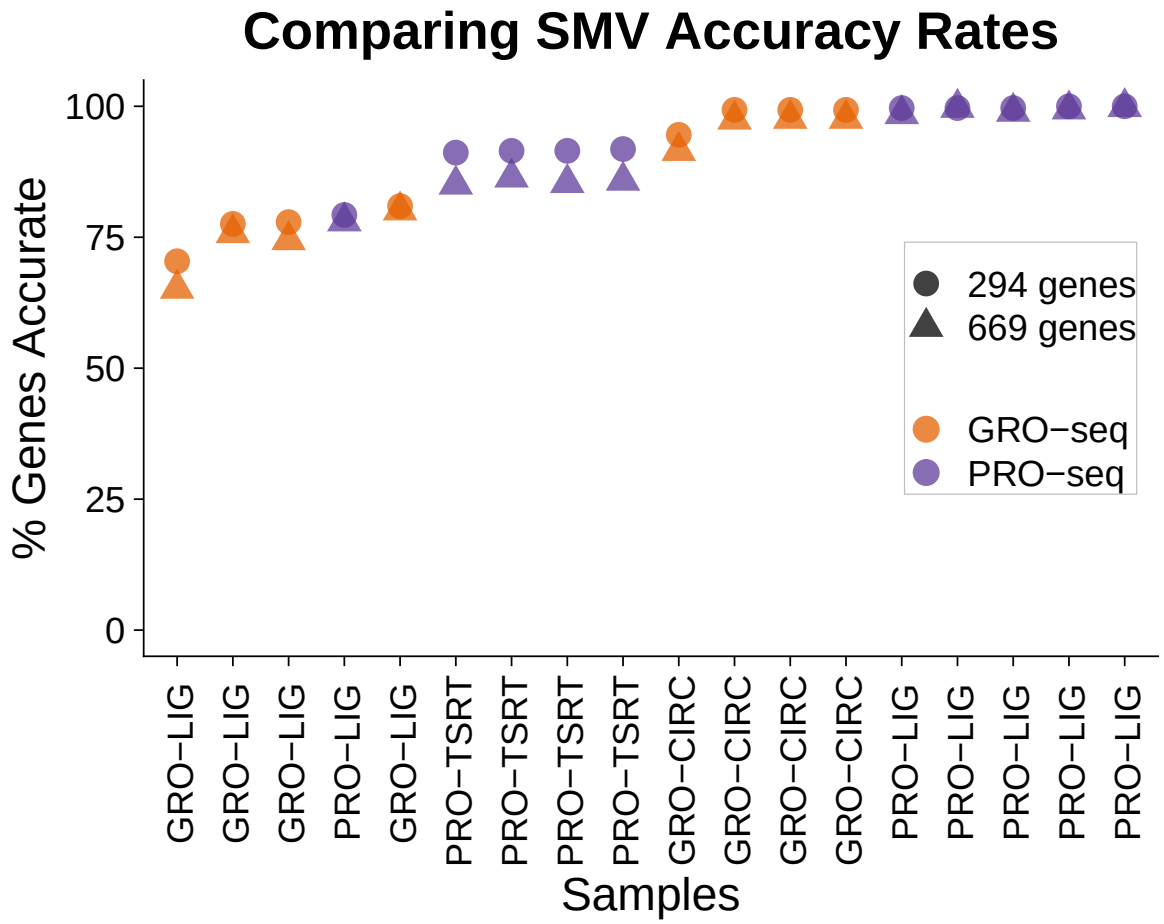


Figure 2.7: The accuracy rate for the classifier remained mostly unchanged for both the top 294 and top 669 genes with high coefficient of variation (CV less than 0.85 and average TPM greater than 100).

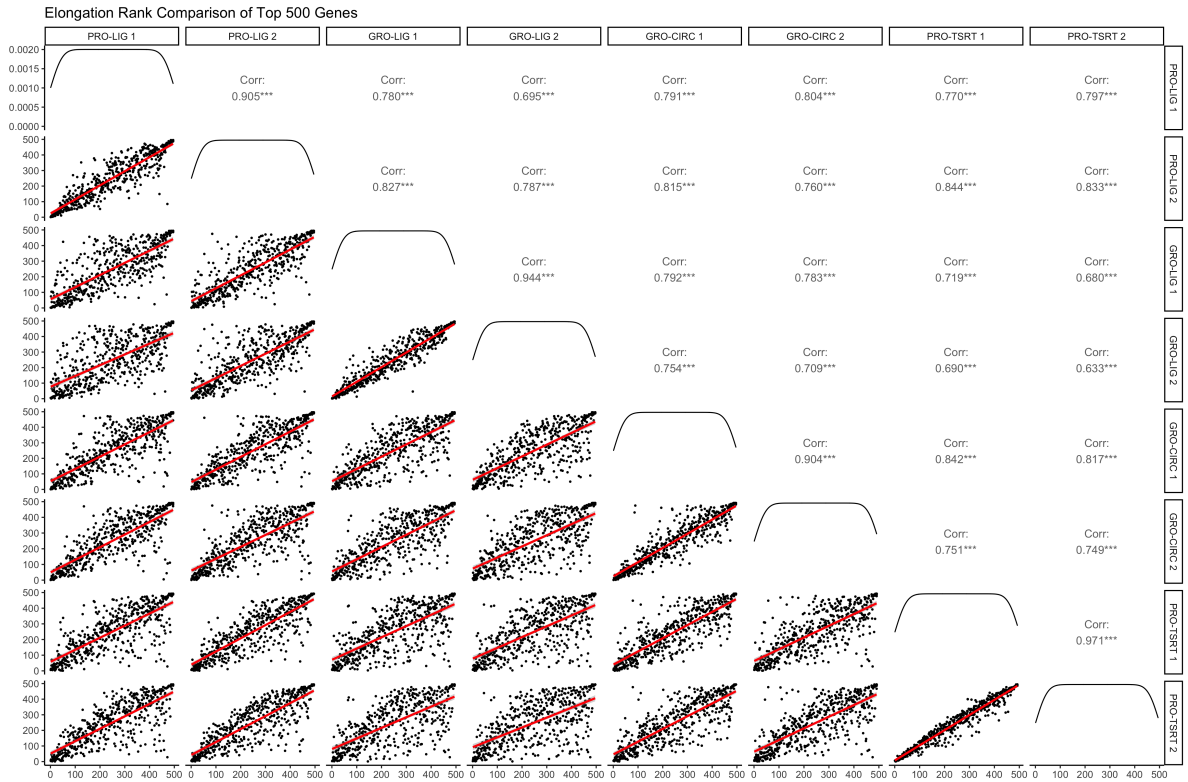


Figure 2.8: Only the top 500 genes (by TPM) were considered. There is considerably more correlation in elongation regions versus pause regions at these genes, suggesting more variability occurs near the TSS across protocols. Each replicate dataset is a biological replicate (see Supplemental Table 1).

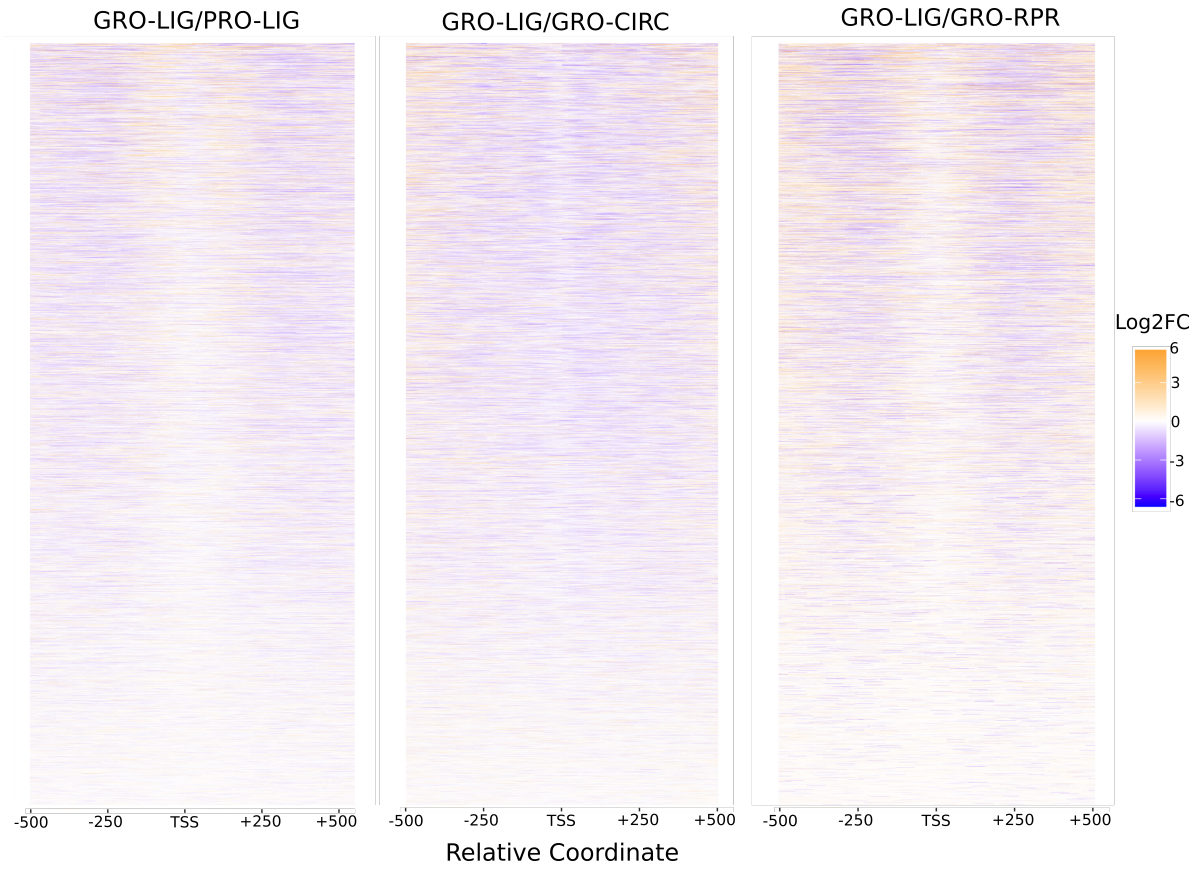


Figure 2.9: (TSS +/- 500 bp, 10 bp per window; RefSeq hg38 gene annotations were used.) Genes shorter than 2000 bp were not included. A pseudocount of 1 was added to all libraries to avoid undefined values. There is comparatively lower coverage near the TSS in many genes, representing the center of bidirectional transcription. This is especially prevalent in GRO-RPR and PRO-LIG libraries.

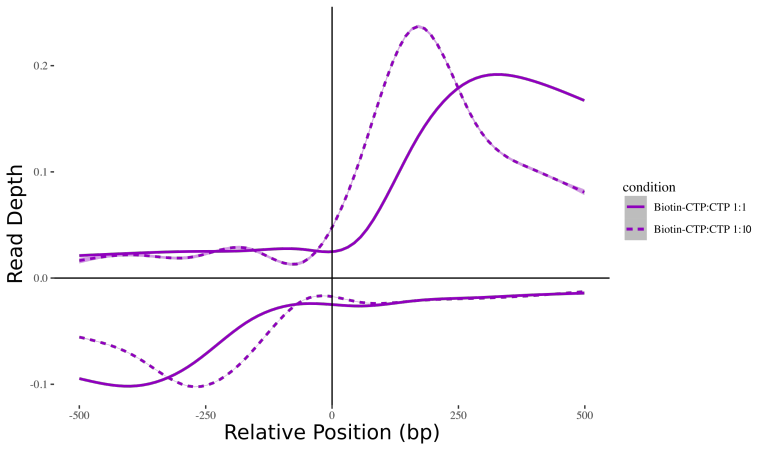


Figure 2.10: Libraries generated from HCT116 cell treated with DMSO, using the PRO-LIG protocol and library preparation strategies. Libraries differed only in the relative amounts of unlabeled CTP added (See Materials and Methods).

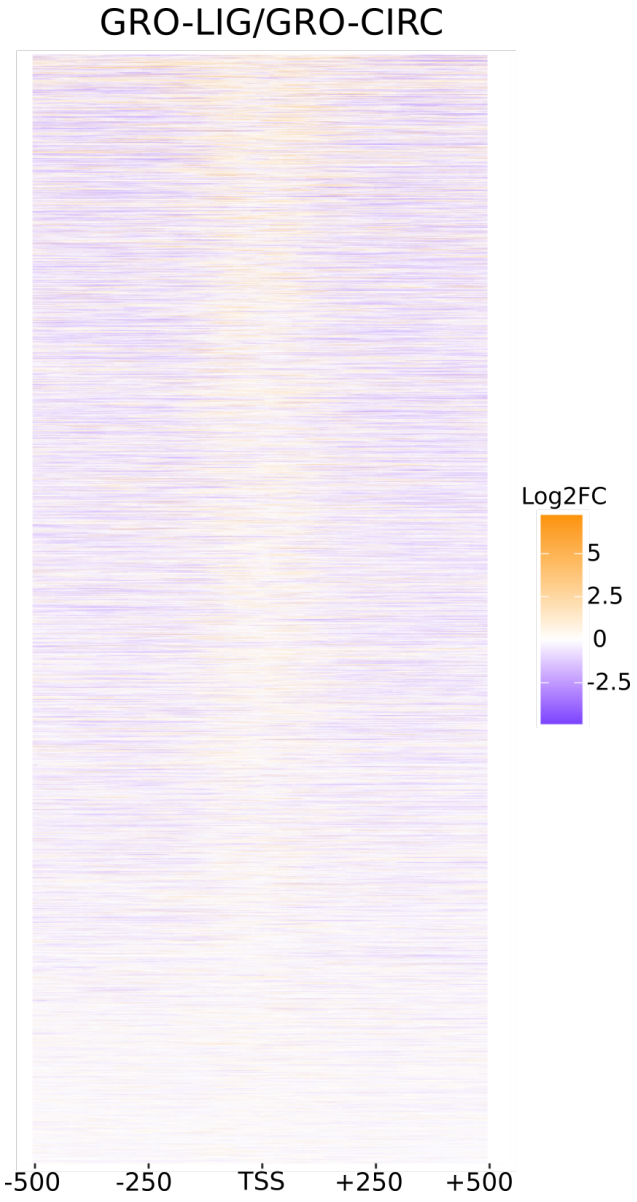


Figure 2.11: (TSS +/- 500 bp, 10 bp per window; RefSeq hg38 gene annotations were used.) Genes shorter than 2000 bp were not included. A pseudocount of 1 was added to all libraries to avoid undefined values. There is considerably more signal in the analyzed GRO-LIG library near the TSS, suggesting additional factors such as size selection contribute to disparities near these regions. (Public GRO-LIG: SRR1501091, SRR1501092; Public GRO-CIRC: SRR4090102, SRR4090103)

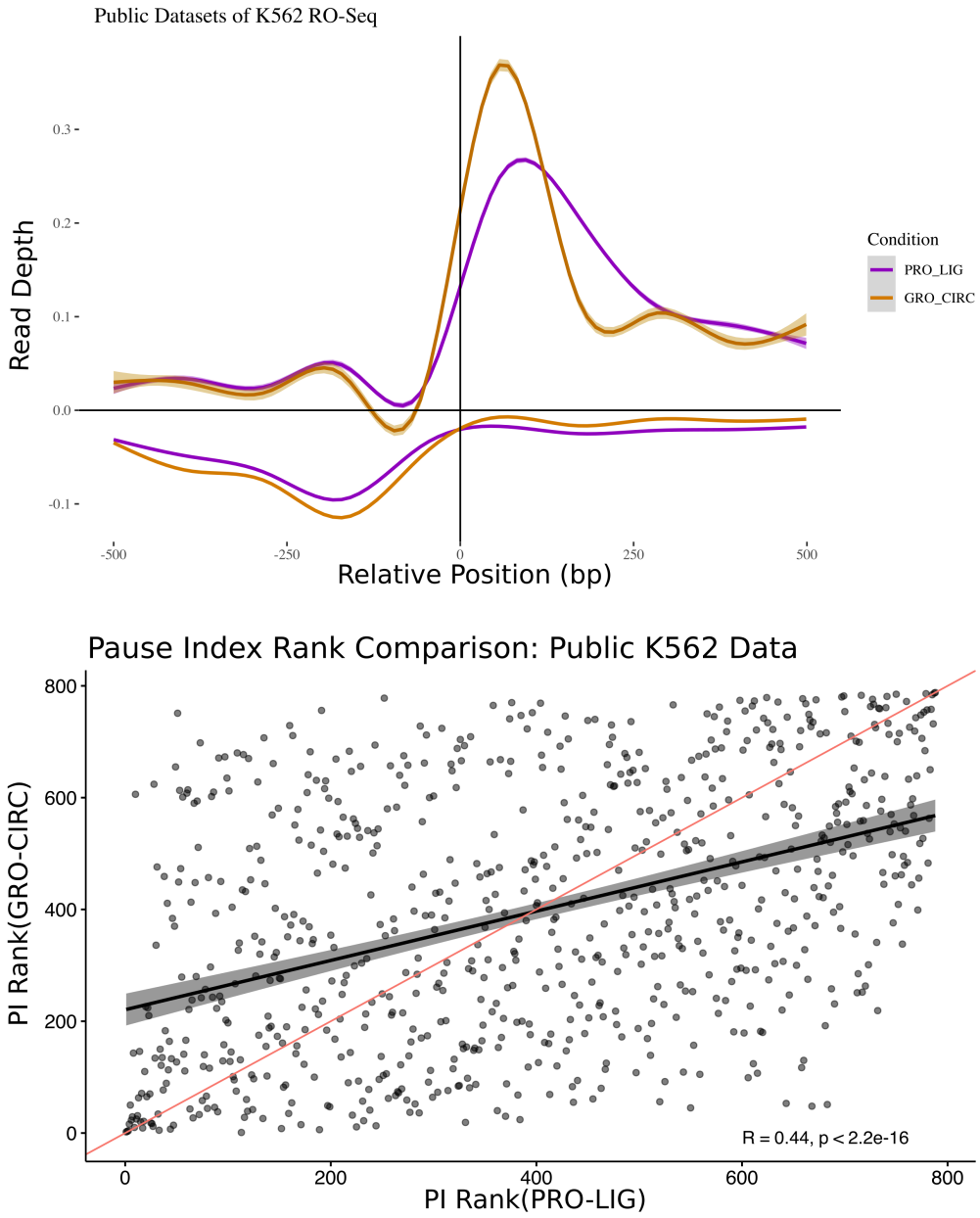


Figure 2.12: (Top) Metagenes of public datasets244, 245. Libraries were generated from K562 Cells treated with DMSO and prepped with either PRO-LIG or GRO-CIRC methods. PRO-LIG libraries were prepared with all 4 NTPs labeled with biotin during the run-on reaction. While the peak of these distributions occur at different relative locations than our datasets, we note that the PRO library still shows a peak that is further downstream than the comparative GRO library. (Bottom) Public data244, 245 were subjected to analysis as in Fig. 2.3C, left (see Supplemental Table 1). PI regions were defined as in Fig. 2.3. Notably, the rank correlation remains low (R=0.44) consistent with PI differences being driven by protocol. Public GRO-CIRC: SRR1823901 and SRR1823902. Public PRO-LIG: SRR5364303 and SRR5364304, see Supplemental Table 1.

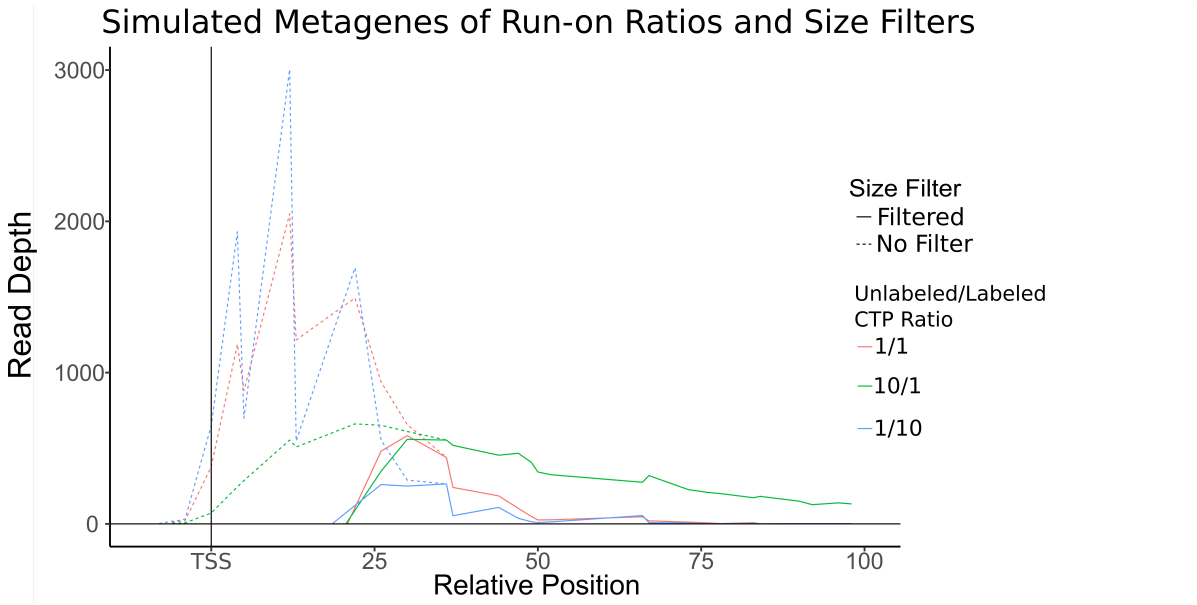


Figure 2.13: Reads were generated *in silico* from a simulated gene template (see Materials and Methods), using run-on ratios to inform read positions and length. Small reads (approx. <25bp, see Materials and Methods) were either filtered out (solid lines), or kept in (dotted lines). As expected, with increasing NTP concentration the peak moves downstream (dashed lines). However, the size selection subsequently alters the location of the visible peak (solid lines) based on the proportion of the data that passes beyond the filter. In this way, the two protocol steps interact to influence the location observed for the 5′ peak. Here, for example, both the filtered 10/1 (green) and 1/1 (red) tracks report a 5′ peak near 28 bp, whereas the filtered 1/10 (blue) track reports a 5′ near 38 bp. Additionally, the read distribution in shifted towards the TSS in the filtered 1/1 track relative to the 10/1 track.

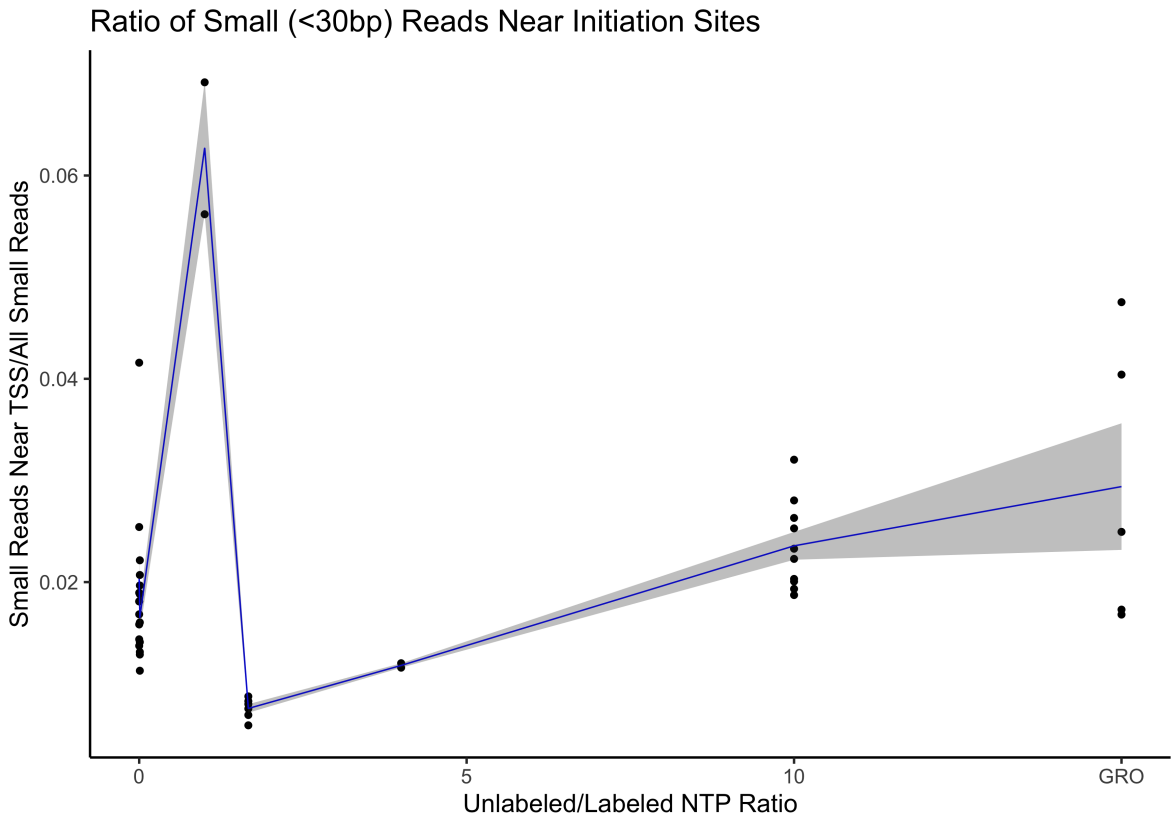


Figure 2.14: We reasoned that this ratio would be informative of the mixture of labeled and unlabeled NTPs in the run-on reaction. Based on publicly available data and our own in-house data (see Materials and Methods for full list of samples analyzed), there appears to be a trend in this ratio, although not a monotonically increasing function. The scarcity of different run-on ratios in public data do not warrant an estimate on an "ideal" ratio from these data; however, we note that these data are consistent with our in silico simulations.

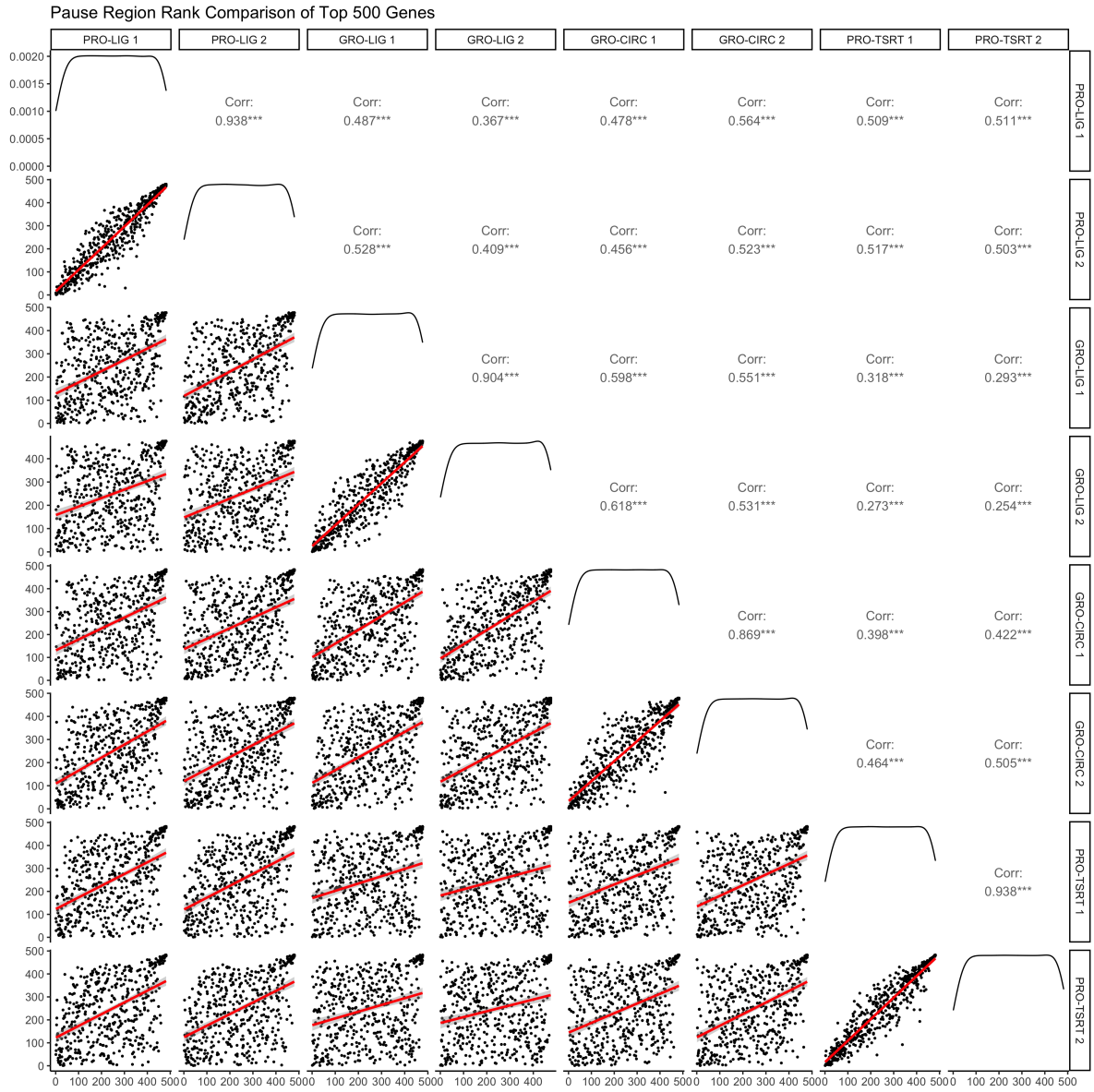


Figure 2.15: (pause region:-50 to +250 from RefSeq hg38 TSS annotation). There is considerable variation between protocols at these regions. Replicates shown are biological replicates (see Supplemental Table 1).

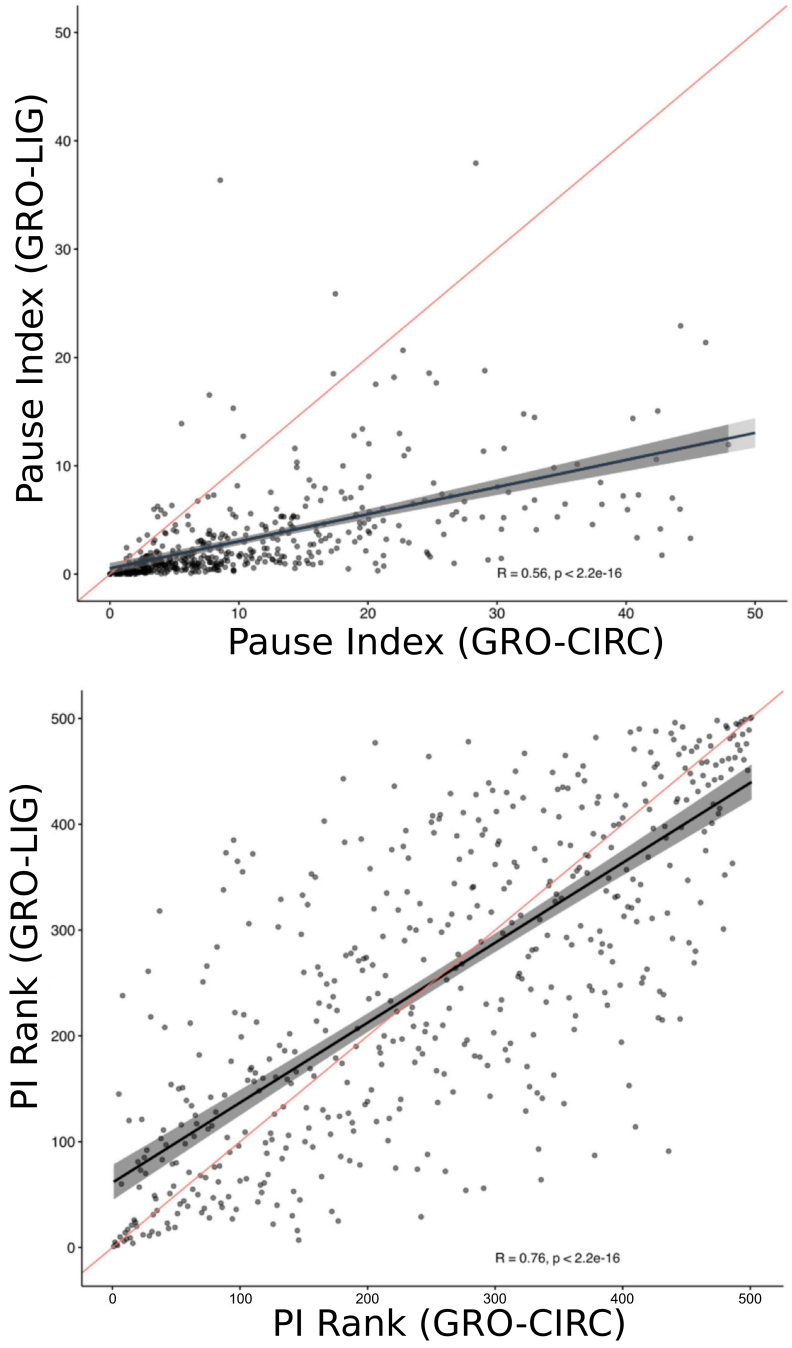


Figure 2.16: Pause indices generated using a different pause region definition than Fig. 2.3E. Namely here the pause ratio is TSS to +80, elongation region +81:TES-1000 (genes shorter than 2000 bp were not included) and features were counted with featureCounts. In spite of using both a distinct interval and counting scheme, the pausing ratio remains poorly correlated (here Pearson R=0.56, Spearman R=0.76).

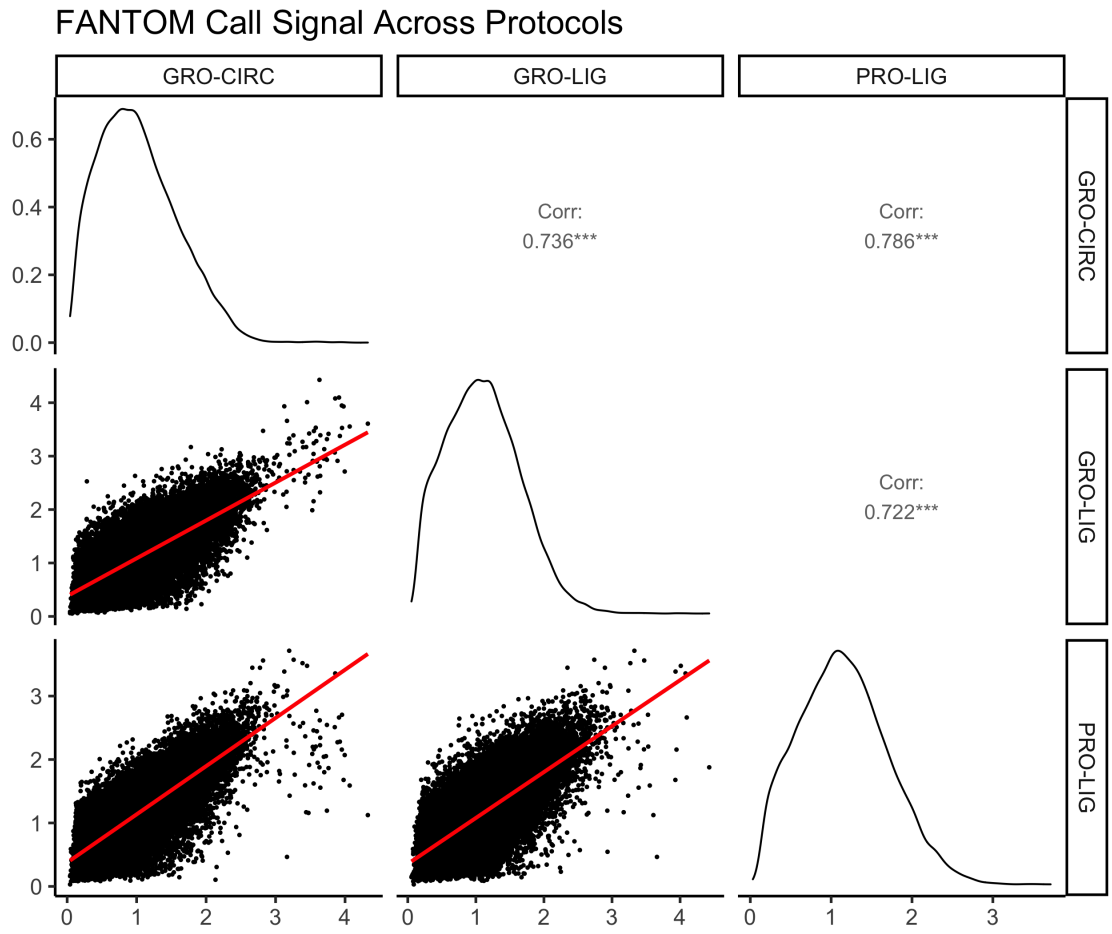


Figure 2.17: FANTOM annotations 20 are generated from CAGE data, thus we reasoned that FANTOM annotated regions would be highly transcribed enhancers. Correlation levels are high between all protocols at these regions, albeit with considerable variation near select sites.

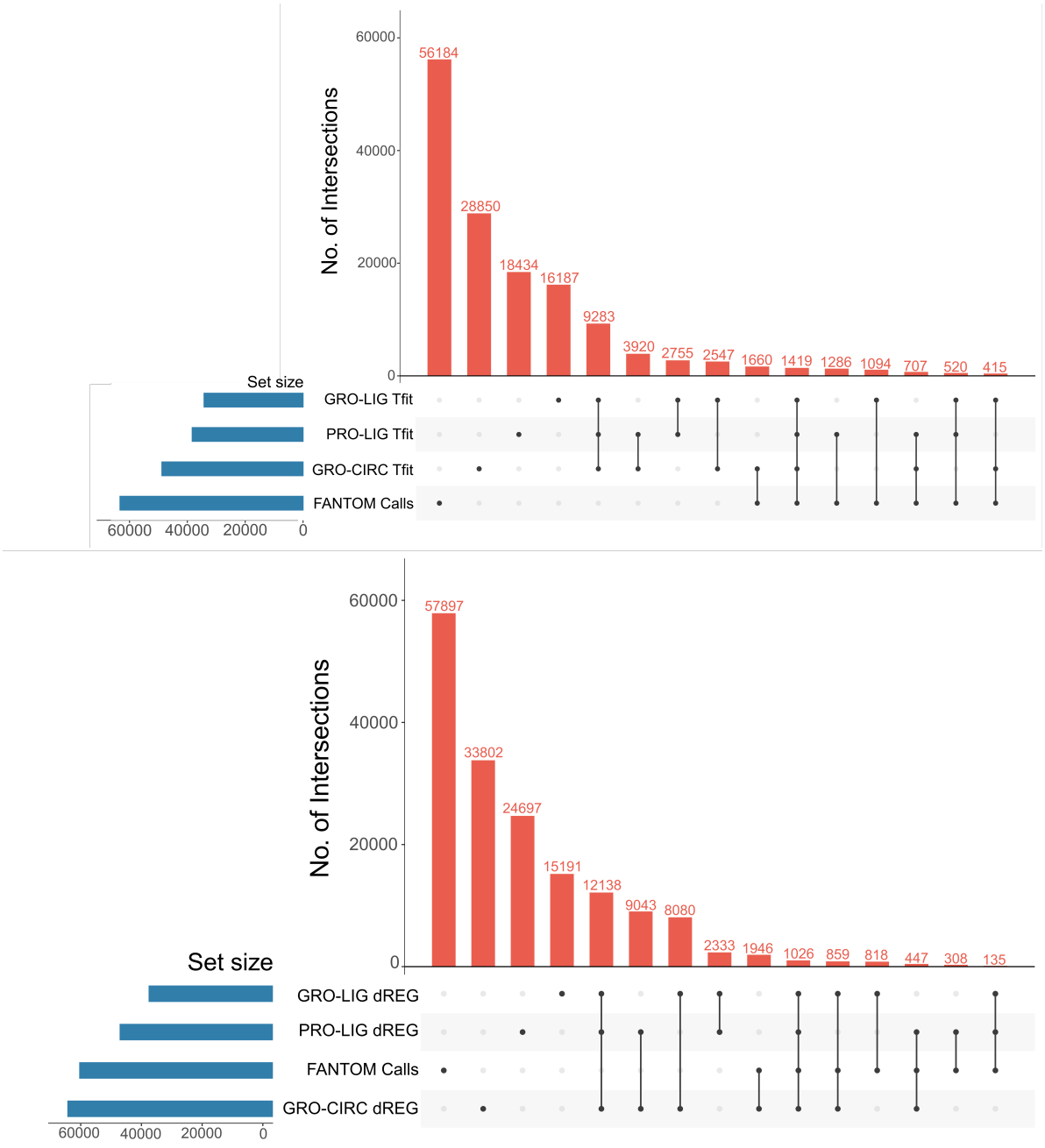


Figure 2.18: Bidirectional calls for equal numbers of DMSO-treated biological replicates were combined to form each set (PRO-LIG: n=2, combined depth 83.3 million reads (SRR14355652, SRR14355672); GRO-LIG: n=2, combined depth 108 million reads (SRR14355673, SRR14355674); GRO-CIRC: n=2, combined depth 212 million reads (SRR1105736, SRR1105737) (see Supplemental Table 1)). We observe frequent instances where each method does not call a region, despite the presence of bidirectional transcription, as shown in Fig. 2.4D,E. While this effect is depth dependent, there are notable regions where the strength of signal is strongly protocol dependent even after correcting for disparities in depth (Fig. 2.4G,H).

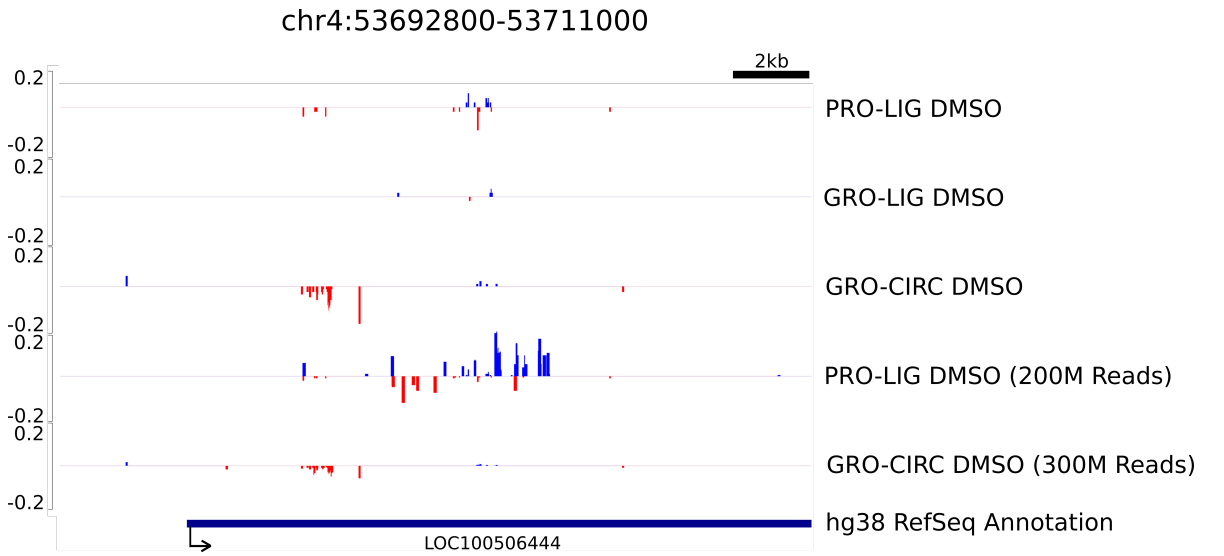


Figure 2.19: Read depths were normalized by CPM. Biological and technical replicates were combined to increase effective depth, as indicated in the bottom two read tracks (PRO-LIG: SRR14355650, SRR14355651 SRR14355652, SRR14355672; GRO-CIRC: SRR1105736, SRR1105737, SRR828696, see Supplemental Table 1).

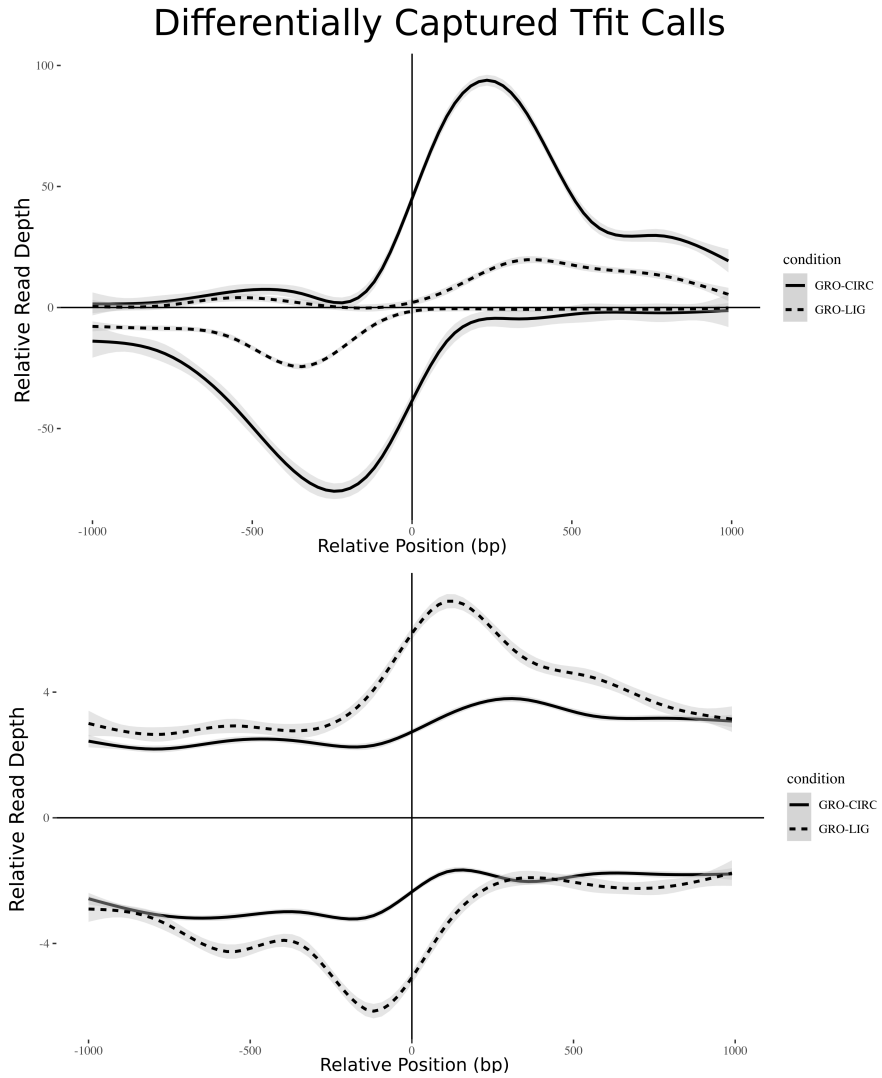


Figure 2.20: Tfit calls across all replicates and treatments were combined together using *muMerge* for both GRO-LIG and GRO-CIRC libraries. Combined enhancers for GRO-LIG were then merged with combined enhancers for GRO-CIRC using bedtools merge (v2.28.0). Counts over these regions were used as input for DESeq1 (See also Materials and Methods). Differentially transcribed enhancers (Fig 2.4F, Materials and Methods) were used as inputs for metagene construction of GRO-CIRC (Top) and GRO-LIG (Bottom) preferentially obtained regions. Reads counts were normalized by CPM.

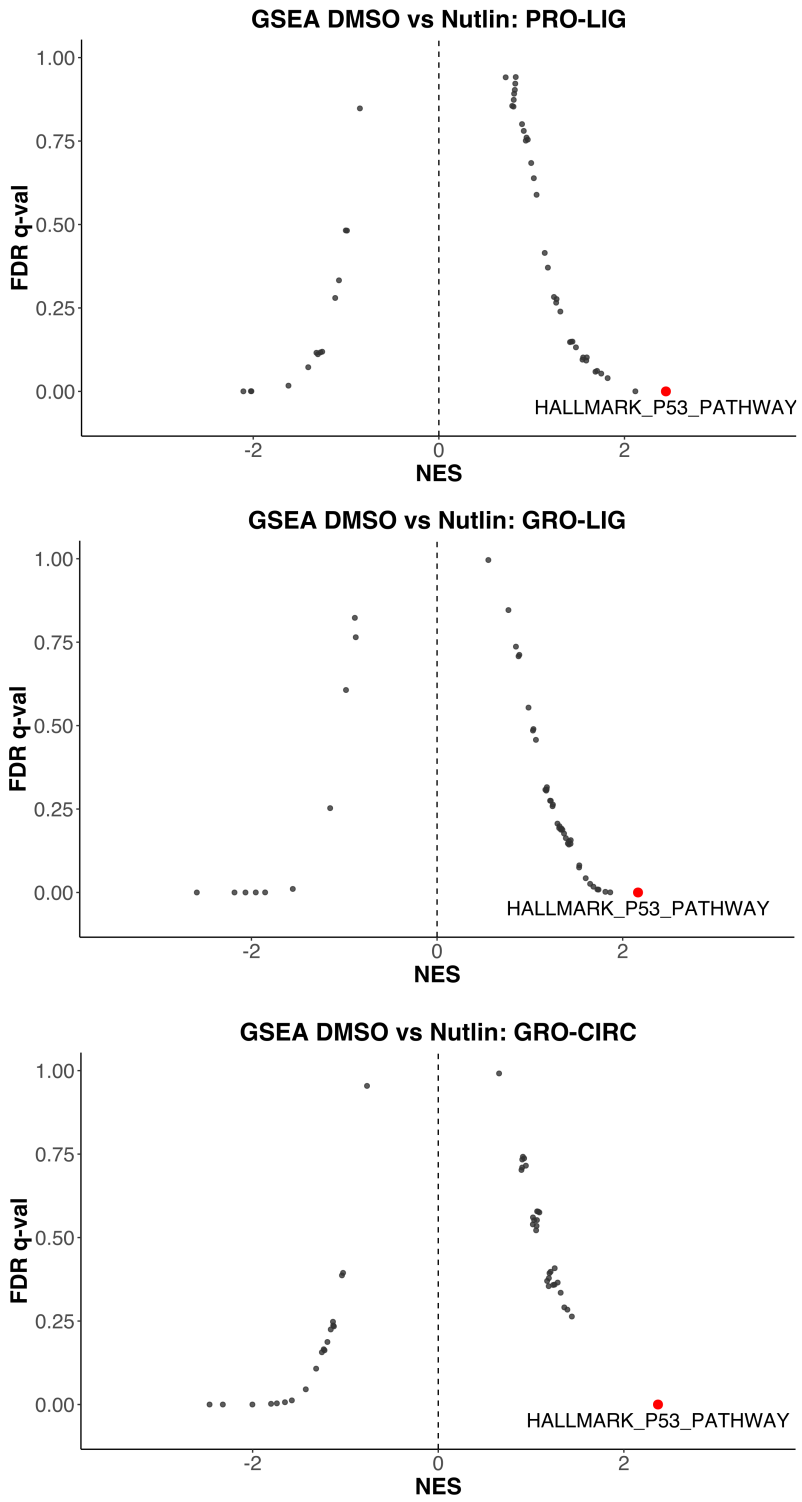


Figure 2.21: Gene region definitions were adjusted to exclude the 5′ pause peak, as per Fig 2.5A. In spite of library variations, the HALLMARK_P53_PATHWAY (red) is the strongest hit in all comparisons.

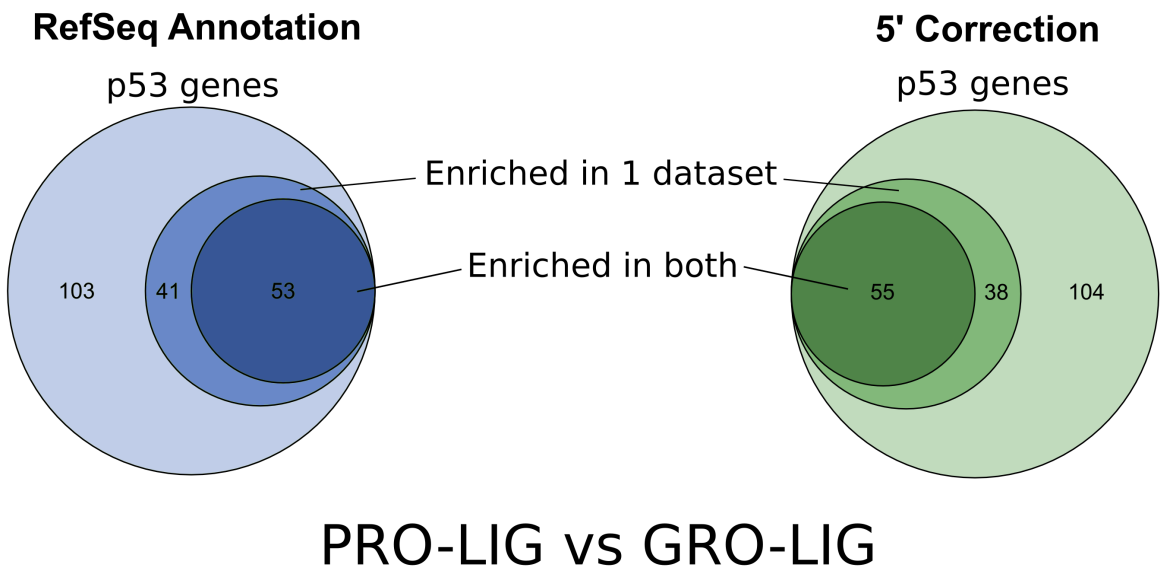


Figure 2.22: Analysis was performed using counts over gene bodies (Left, hypergeometric test p-value=5.54e-15), and using a 5′ correction (Right, hypergeometric test p-value= 8.87e-17), as in Fig. 2.5A (see also Materials and Methods).

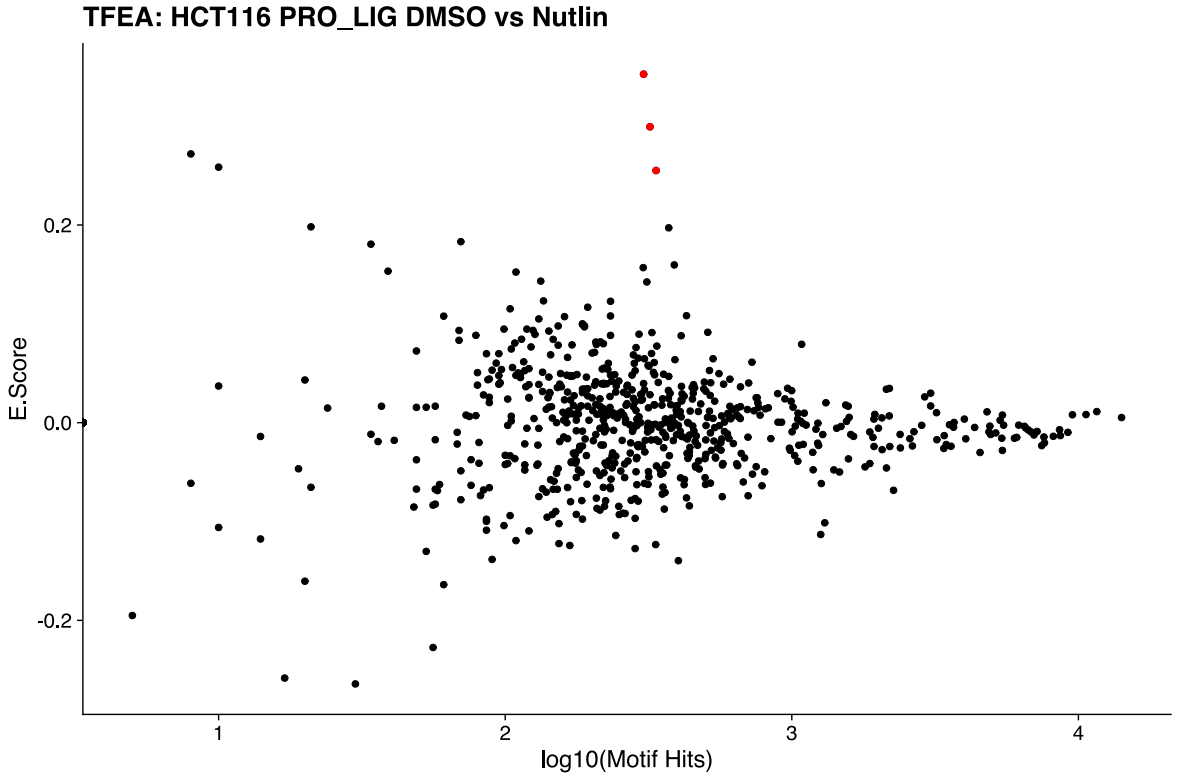


Figure 2.23: Regions were combined using *muMerge*, as in Fig. 2.5E,F. Red dots indicate transcription factors belonging to the p53 family (TP53, TP63, TP73).

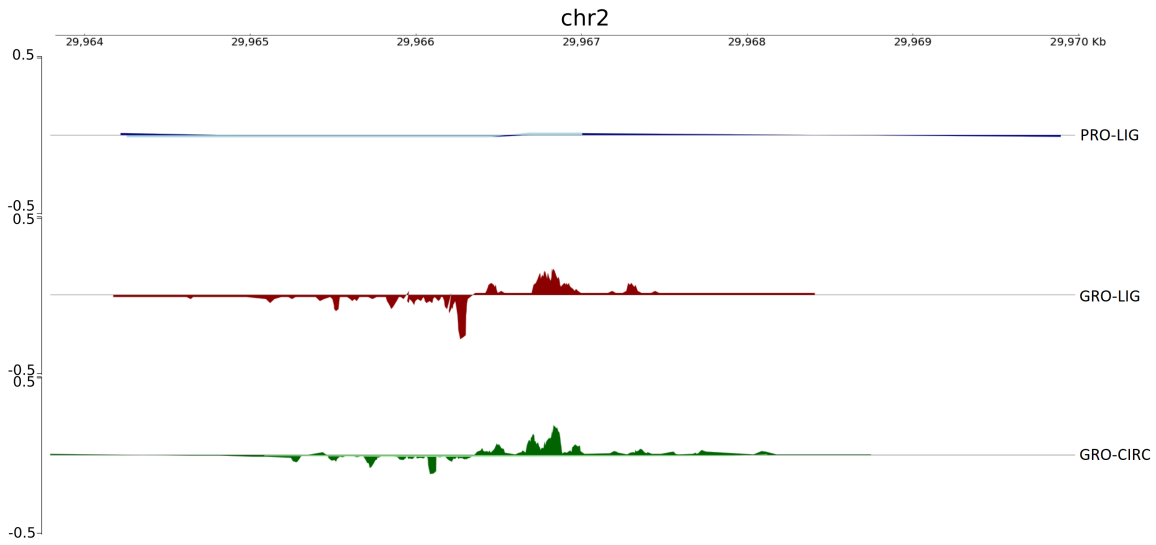


Figure 2.24: Darker colors represent transcription level in Nutlin-3a treated libraries, while lighter colors represent levels found in DMSO-treated libraries. (Notably DMSO levels are nearly zero.) Read counts are normalized by CPM.

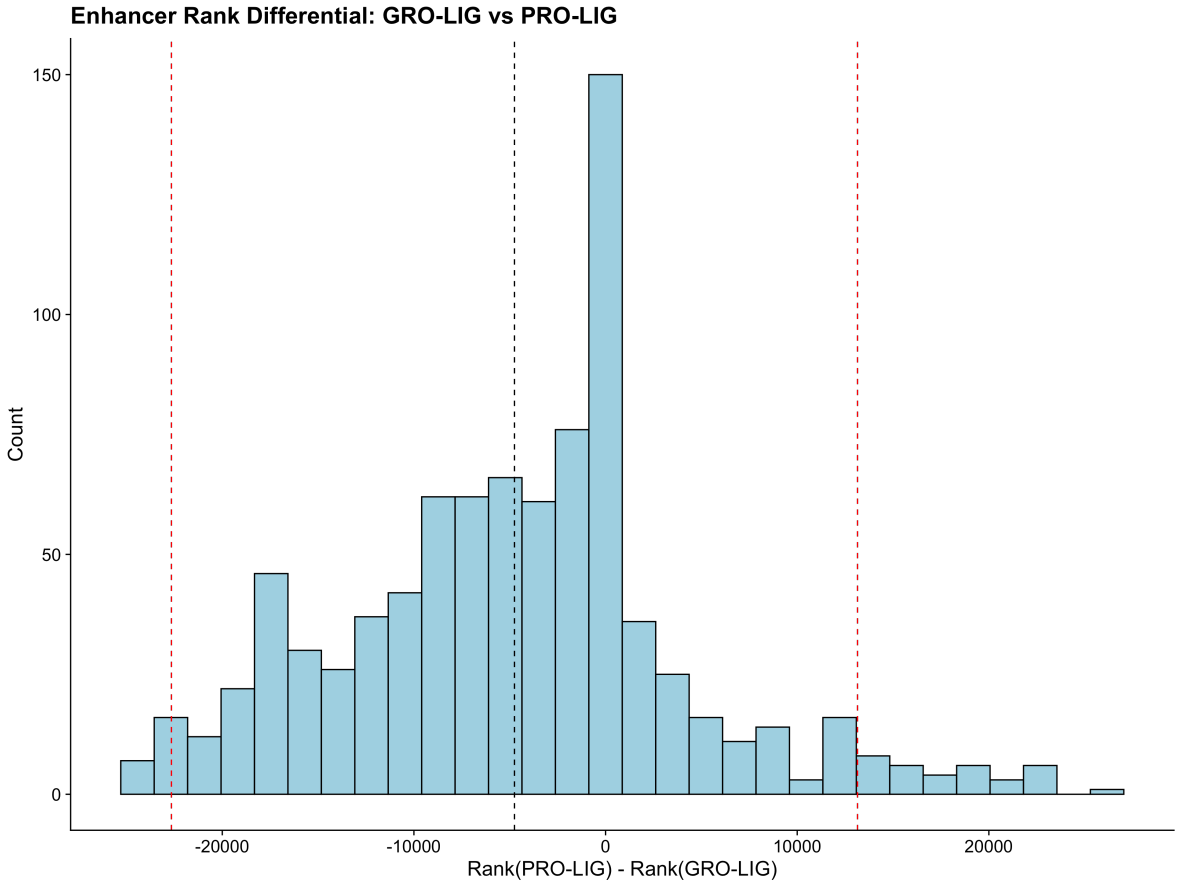


Figure 2.25: Ranks were determined within TFEA through DESeq2. p53 enhancers which were more than 2 standard deviations (red dotted lines) from the mean (black dotted line) were considered to be differentially captured in GRO-seq or PRO-seq.

APPENDIX C

SUPPLEMENT TO CHAPTER 4

The reader is referred to the publication for the supporting tables describing the accession numbers of all data utilized within this chapter.

File name: Supplementary Data 1 Description: Accession numbers for data utilized to generate Figures in the paper (Figure 3, Figure 5, Figure 6, and Supplementary Figure 6), one tab per figure.

Table for Figure 3: Accession Table with all samples summary for HCT116 GRO-seq samples treated with either Nutlin or DMSO. Data collected from Allen 2014.

Table for Figure 5: FANTOM Project Numbers Table for data from Forrest 2014, Baillie 2017 with LPS: lipopolysaccharide time series CAGE data.

Table for Figure 6: Accession numbers used in Figure 6, data from Davis 2018, Mcdowell 2018. Samples treated with Dex: dexamethasone.

Table for Supplemental Figure 6: Accession numbers used in Supplemental Figure 6, data from Davis 2018, Andrysik 2017.

Supplemental Figures

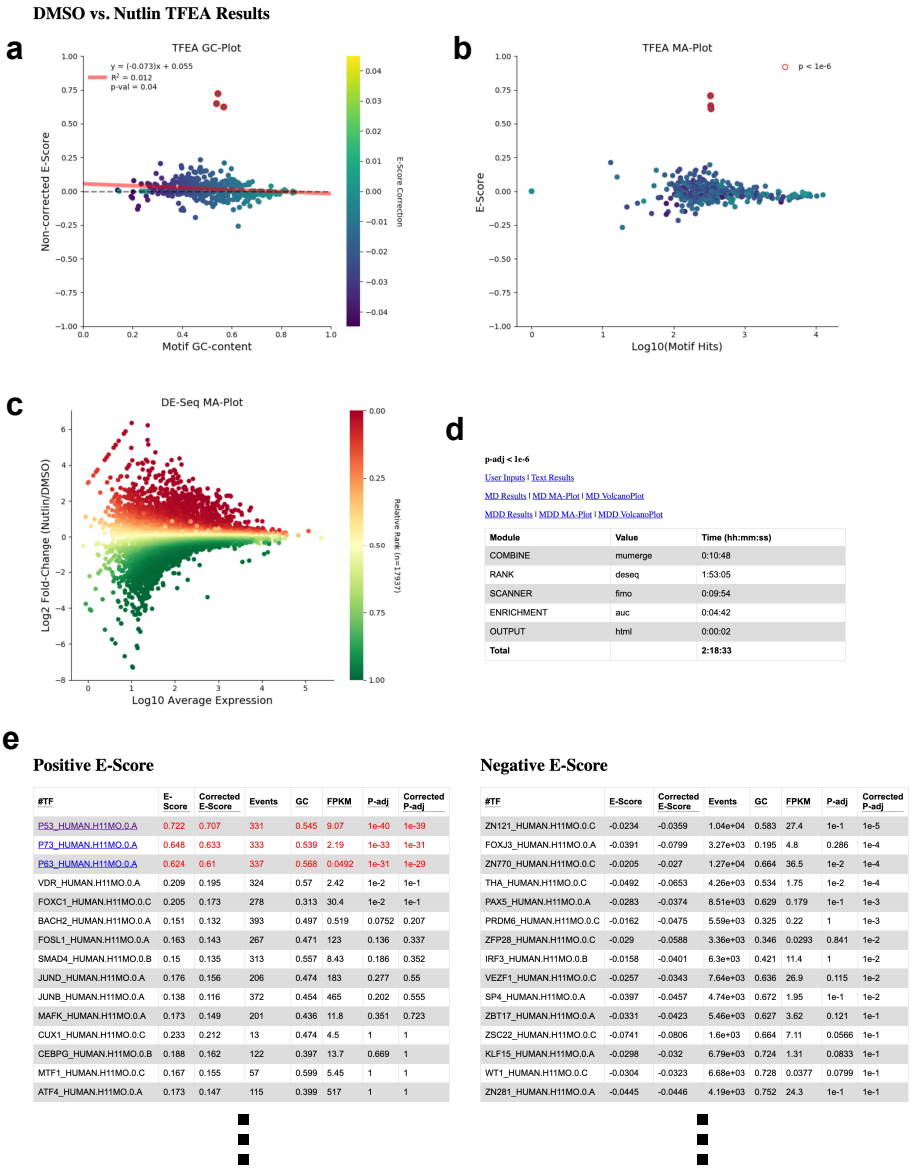


Figure 3.1: An example of TFEA main HTML results page.(a) Pre-GC correction showing the E-Score of each motif (y-axis) as a function of GC-content (x-axis). Red line: linear regression fit; dots colored by the amount to correct. (b) A scatter plot (colored as in a), similar to an MA-plot, showing the GC-corrected E-Scores (y-axis) vs the number of motif hits within regions (<1.5kb; x-axis) for each motif analyzed. (c) An MA-plot of the ROIs generated from DESeq2. (d) A table listing the inputs, text results, MD-Score (motif displacement score) and MDD-Score (differential motif displacement) results (as clickable links), as well as the time taken to complete each step of the TFEA process. (e) A list of motifs that exhibit positive (left) or negative (right) enrichment ordered by adjusted p-value. Significant motifs appear as red and have clickable links (blue) to individual results pages with more detailed information (see Supplementary Figure 3.2). List are truncated for readability. Data is HCT116 dataset, as used in Figure 4.3a61.

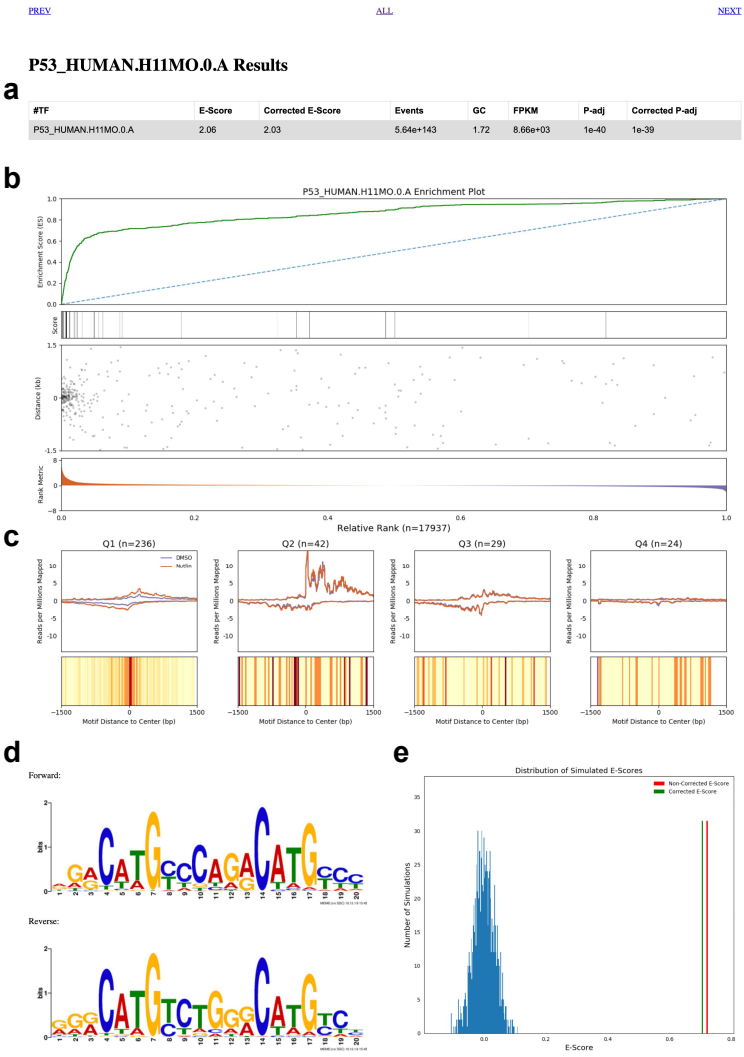


Figure 3.2: This page is reached by clicking on the corresponding motif in Supplementary Figure 3.1e. (a) Summary statistics for the motif of interest, in this case p53 from HOCOMOCO v11172. E-Score: Enrichment Score, Corrected E-Score: Enrichment score after GC correction using linear regression, Events: Number of motif hits within all ROIs, GC: GC content of motif, FPKM: Fragments per kilobase per million (with respect to the gene associated with the TF), P-adj: adjusted p-value of the E-Score, Corrected P-adj: adjusted p-value of corrected E-score. (b) Enrichment plots showing (from top to bottom) the running sum statistic (green line), the individual scores of each ROI (as a heatmap, darkness is greater score), scatter plot of motif hits within ROIs relative to the reference point (labeled 0), and the ranking of ROIs based on differential transcription (red: positive; blue: negative). (c) For each quartile, summarize motif containing ROI within the quartile via Top: Meta plot of read coverage over ROIs. Bottom: Motif displacement distribution (as heatmap: red is max; yellow is min) summarizing the motif positions relative to the reference point. (d) Logos of forward and reverse complement position specific scoring matrix of the motif analyzed. (e) Histogram of E-Scores from randomly shuffling the rank order of ROIs (blue) with true non-corrected E-score (red) and GC-corrected E-score (green).

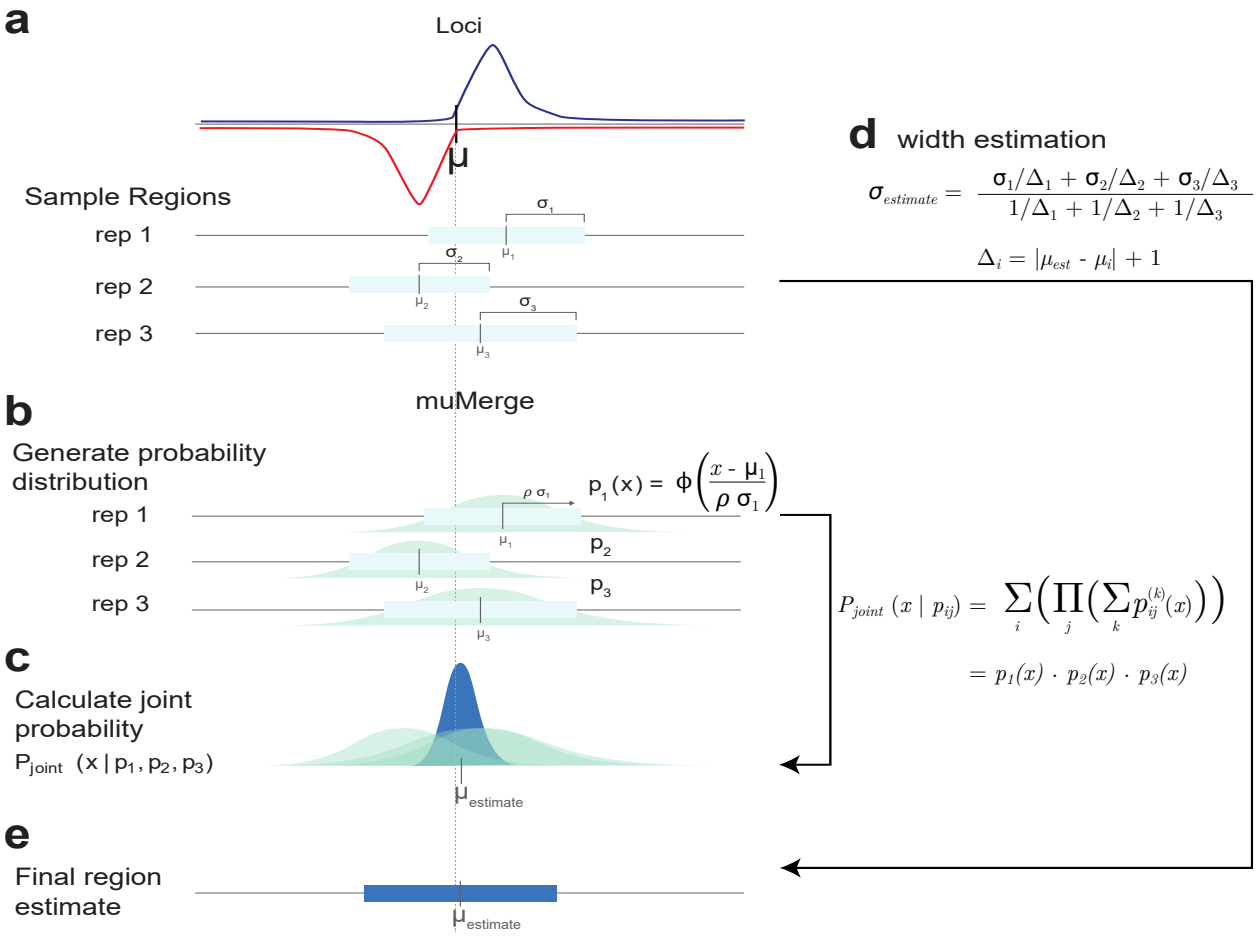


Figure 3.3: (a) The goal of *muMerge* is to combine multiple sample regions (light blue boxes)—centers μ_i and half-widths σ_i—which originate from different replicates and/or conditions (rep 1,2,3), but are measurements of the same underlying loci μ, into a consensus set of regions of interest (ROIs). Red and blue lines represent a hypothetical bidirectional signal centered at μ. (b) *muMerge* assumes that each sample region is an estimate on the location of a genomic locus of interest and models this probability (p_i) as a normal distribution (ϕ, light green distributions) centered on the middle of each sample region μ_i, and standard deviation related to the region's half-width σ_i (scaled by ratio parameter ρ—default = 1). (c) Subsequently, a joint probability (p_{joint}, dark blue distribution—Eq. IV.2 in main text), is calculated from the sample distributions (k-index sum over within-sample peaks, j-index product over within-condition replicates, and i-index sum over conditions), and the estimate for the consensus position ($\mu_{estimate}$) is the local maxima of this joint distribution. (d) Next, to calculate the best estimate for the width of the ROI, a weighted average of the original sample region widths is calculated ($\sigma_{estimate}$)—Eq. IV.3 in main text. It is assumed that the sample regions closest to the consensus position are the most accurate representation of the underlying locus, so the weighted average of the widths is calculated such that more weight ($1/\Delta_i$) is given to the sample regions closer to $\mu_{estimate}$. (e) Thus, the final *muMerge* region estimate (dark blue box) is given by ($\mu_{est} - \sigma_{est}$, $\mu_{est} + \sigma_{est}$). The method is described in detail in Methods section "Defining ROIs with *muMerge*".

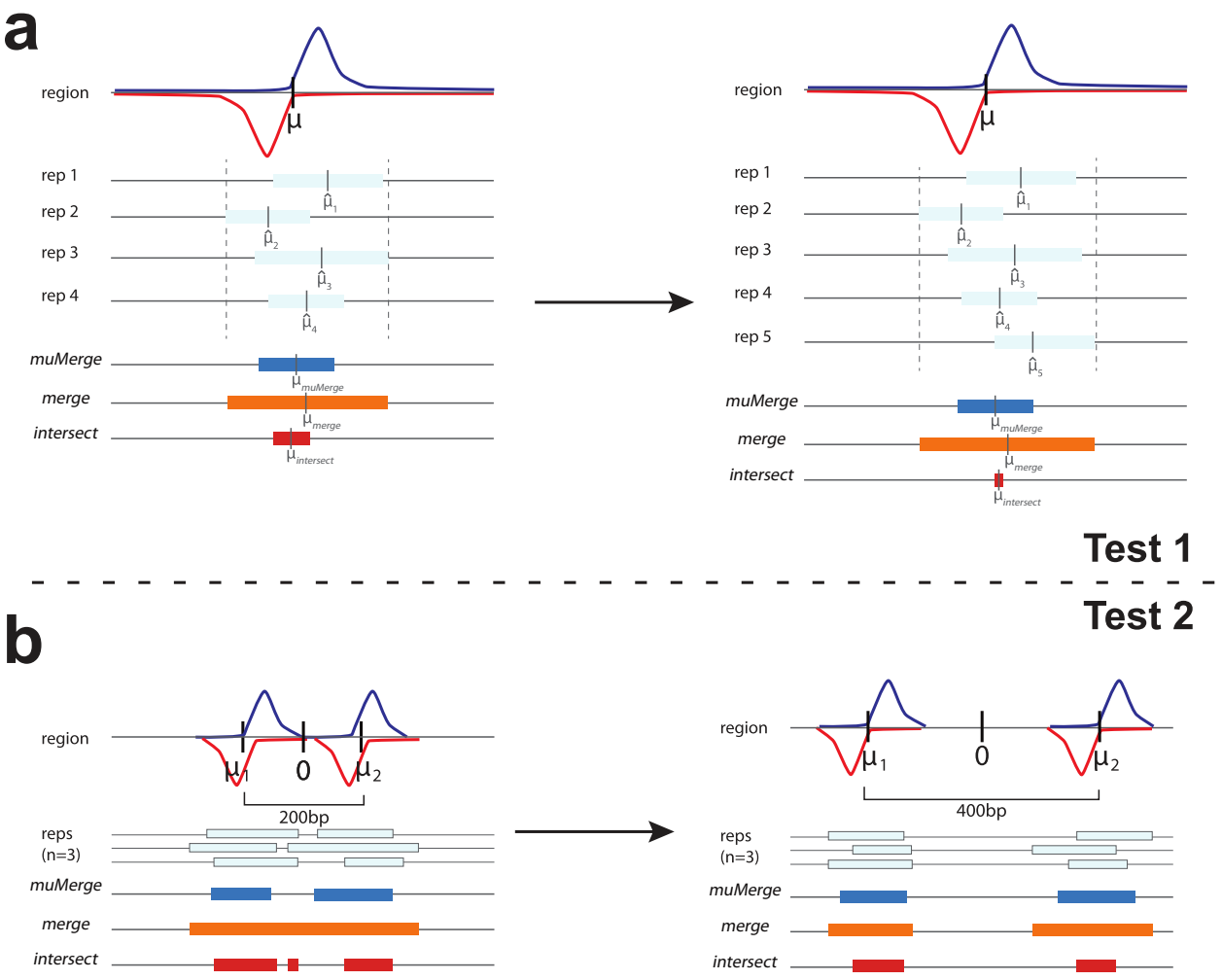


Figure 3.4: The results of these two tests are shown in Figure 4.2. (a) The first test involves sampling regions from a single theoretical locus with increasing number of replicates (light blue boxes). *muMerge* (dark blue) retains correct length and μ position, while *bedtools merge* (orange) tends to increase ROI length and *bedtools intersect* (red) tends to decrease ROI length with increasing number of replicates. The quantitative results of this test are shown in Figure 4.2b of the main text. (b) The second test to determine performance involves sampling from two theoretical loci as a function of inter-locus spacing ($|\mu_2 - \mu_1|$) (light blue). For closely spaced loci, *muMerge* (dark blue) correctly separates the two loci whereas *bedtools merge* (orange) is more likely to generate a single ROI, and *bedtools intersect* (red) is more likely to generate multiple separate ROI (in this example, three). The quantitative results of this test are shown in Figure 4.2c of the main text. For both tests, the top cartoon depicts bidirectional signal on two strands (blue: positive strand; red: negative strand). Regions inferred from individual replicates in light blue. ROI ascertained by *muMerge* (dark blue), *bedtools merge* (orange) and *bedtools intersect* (red) shown for comparison.

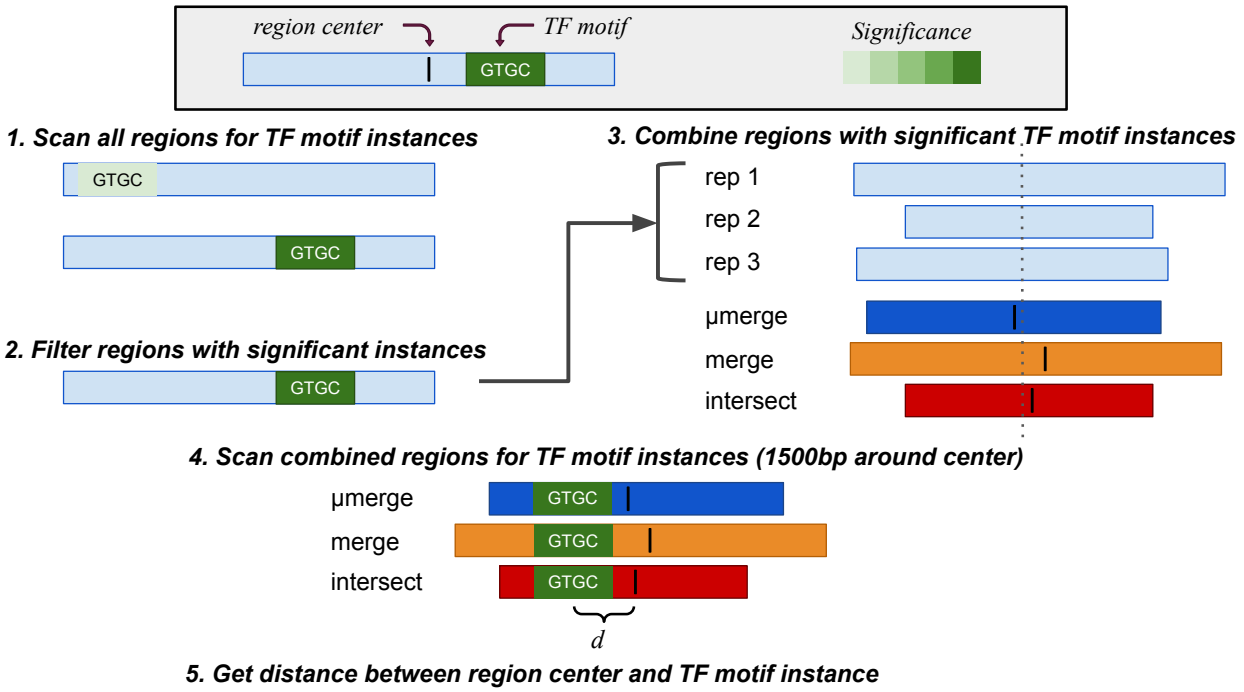


Figure 3.5: (1) Given ChIP-seq peak call regions (light blue boxes), the first step is to scan for TF motif instances (green) for all samples with FIMO190. (2) Peak regions with significant TF motif hits (dark green) (p-adj < 0.001) are retained, and (3) significant replicate regions are combined with either *muMerge* (dark blue), *bedtools merge* (orange) and *bedtools intersect* (red)189. (4) Combined regions are expanded ±1500 bp around the center (black vertical line) of the region and TF motif instances are determined. (5) Finally, the distance between the region center and the center of the best motif instance is calculated.

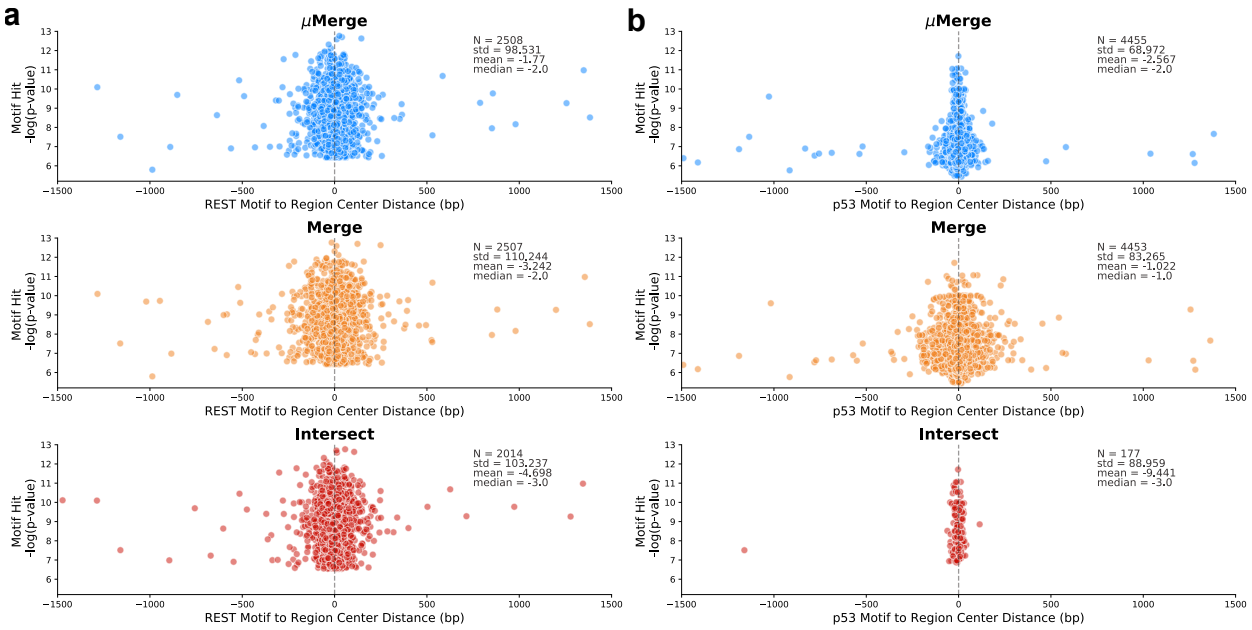


Figure 3.6: Using the procedure of Supplementary Figure 3.5, scatter plots show the distance of the centers of ROIs (inferred, using the three methods, from identified ChIP peaks) to the corresponding TF binding motif. (a) Two REST143 ChIP-seq replicates were combined using *muMerge* (dark blue), *bedtools merge* (orange) or *bedtools intersect* (red). The distance between the midpoint of the resulting region and the midpoint of the best motif instance is plotted (x-axis) relative to the motif score (y-axis). We note the mean is closest to zero for *muMerge* (dark blue) which also has the smallest standard deviation (std). This scenario has only two replicates which produces the smallest difference between the methods (consistent with Fig. 4.2b), since two samples is the least amount of replicate statistical power. However, *muMerge* still outperforms the other two methods—smallest standard deviation and mean closest to zero. (b) Similar comparison for p5369 where cell types (HCT116, MCF7, and SJSA) are used as replicates (one sample per cell type) and combined across two conditions (DMSO and Nutlin-3a)—six samples in total. In this case, with multiple conditions, greater number of replicates, and "noisier" data (i.e. multiple cell lines), *muMerge* (dark blue) significantly outperforms the other two methods—*bedtools merge* (orange) produces large deviation from the motif while *intersect* (red) is only able to infer non-zero ROI for ChIP peaks in both conditions, which happen to correlate with highly significant motif instances (p-value $< 10^7$), missing $> 97\%$ of the ROI identified by the others ($N = 177$ vs. $N \sim 4450$). Conversely, *muMerge* infers ROIs for a broad range of motif instance significance, and demonstrates the lowest deviation from the motif location (standard deviation $\sim$ 69 bases). Example regions for the p53 comparison are shown in Supplementary Figure 3.7. See Supplementary Data 1 for complete list of accession numbers for data utilized.

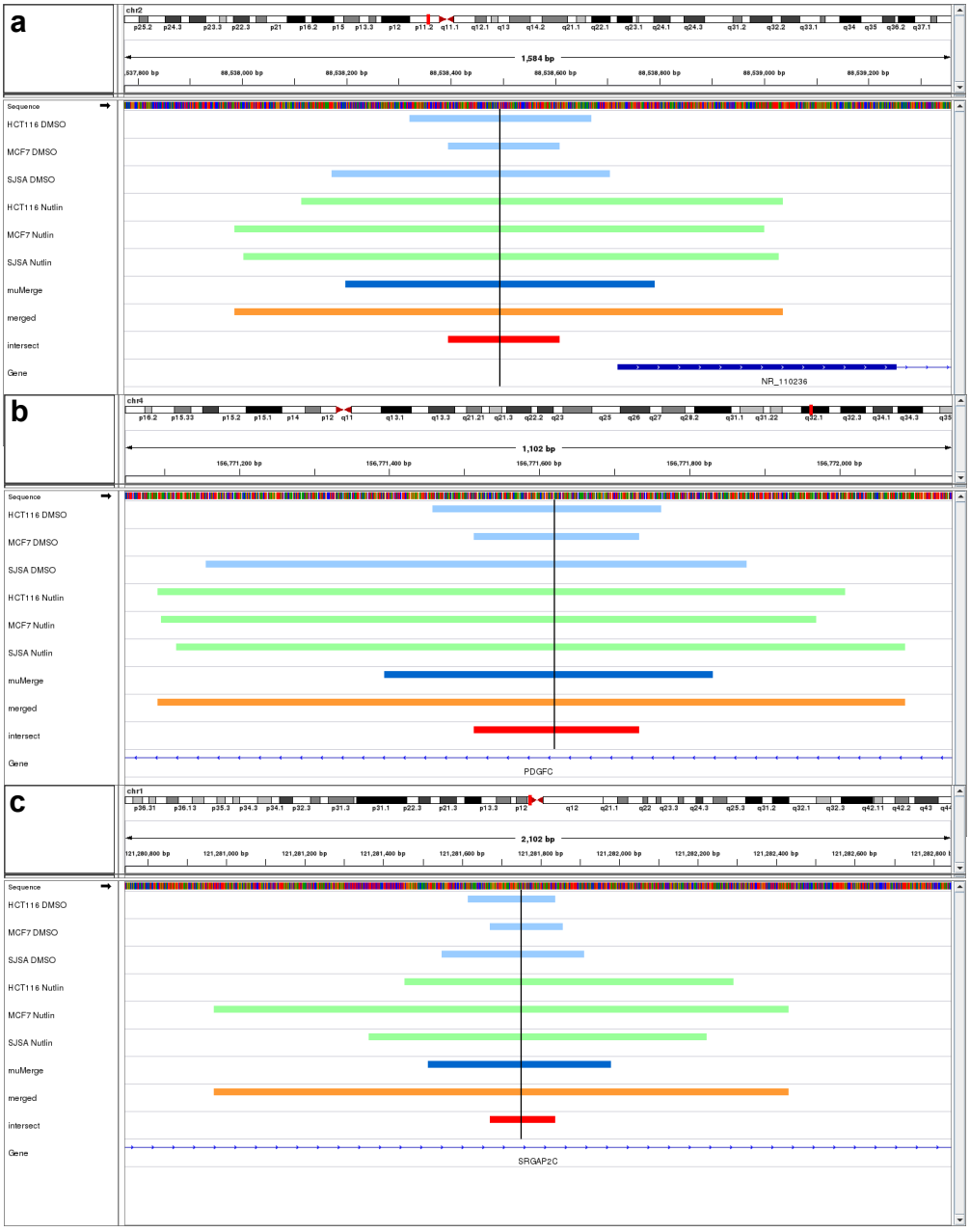


Figure 3.7: Three regions (a-c) from the p53 comparison of Supplementary Figure 3.6 are shown for the three cell types (treated as replicates) and two conditions (DMSO: light blue and Nutlin-3a: green). In all three cases, the motif location (vertical black line) is accurately inferred by all three methods. Consistent with the results in Figure 4.2b/c (and depicted in Supplementary Figure 3.4), *merge* represents the upper limit on the ROI size and *intersect* represents the lower limit. Conversely, *muMerge* strikes a balance between these two extremes.

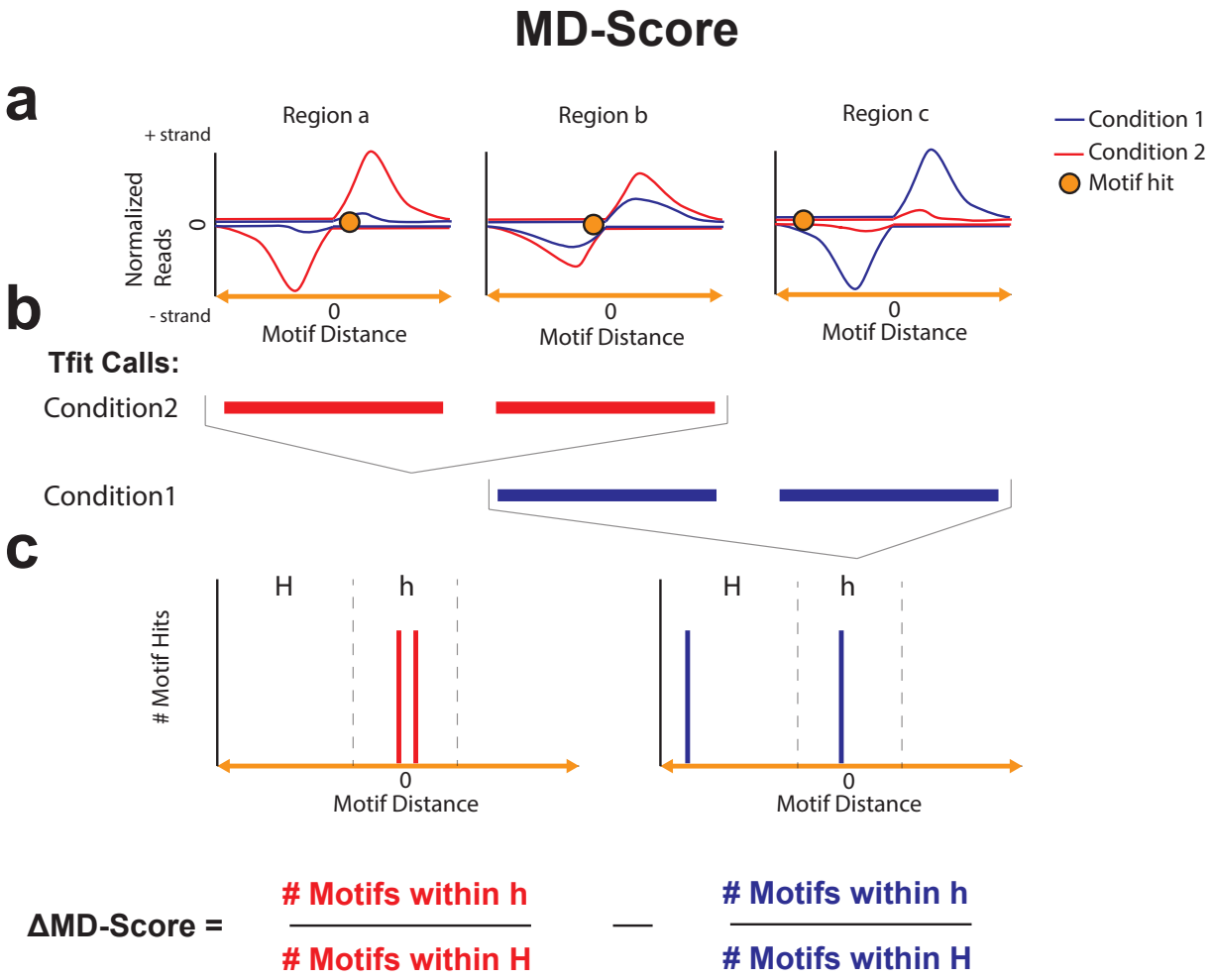


Figure 3.8: (a) Cartoon depicting typical histograms of nascent transcription data (condition 1: blue, condition 2: red) for three example regions (regions a, b and c). Orange dot represents a TF motif instance. (b) Regions of RNA polymerase initiation identified in each dataset (red, blue boxes), for example as called by Tfit81. These regions are the inputs to the MD-score (motif displacement score) approach8. (c) Motif displacement distribution histograms plot position of motif (vertical bars) relative to reference point (labeled 0) for both conditions (red and blue). The MD-Score is the fraction of motif instances within the inner window (h=150 bp) divided by the total motif hits in the larger window (H=1500 bp; note H encompasses h). MD-Scores are calculated independently in each of the two conditions to obtain the difference.

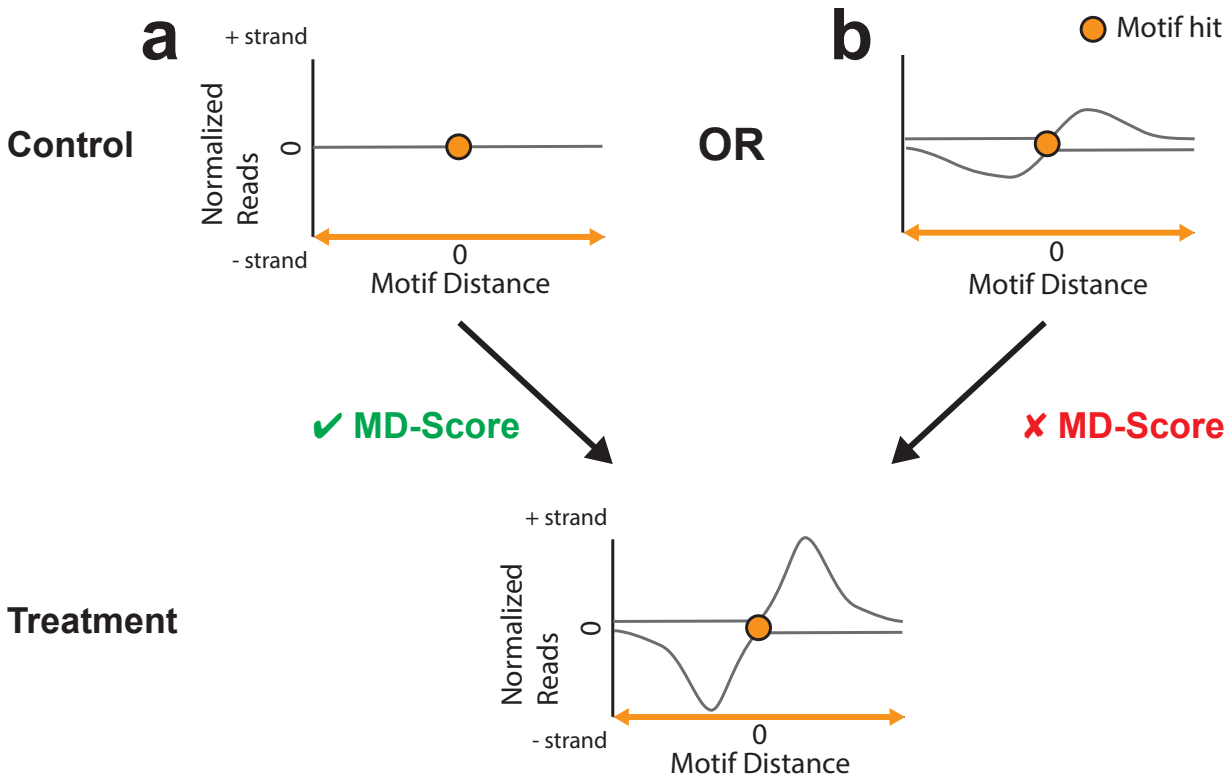


Figure 3.9: A given locus in the treatment can arise from either (a) a region of no signal in the control; or (b) increase in signal at a pre-existing region within the control sample. Importantly, the first case increases the Δ MD-Score whereas the second does not.

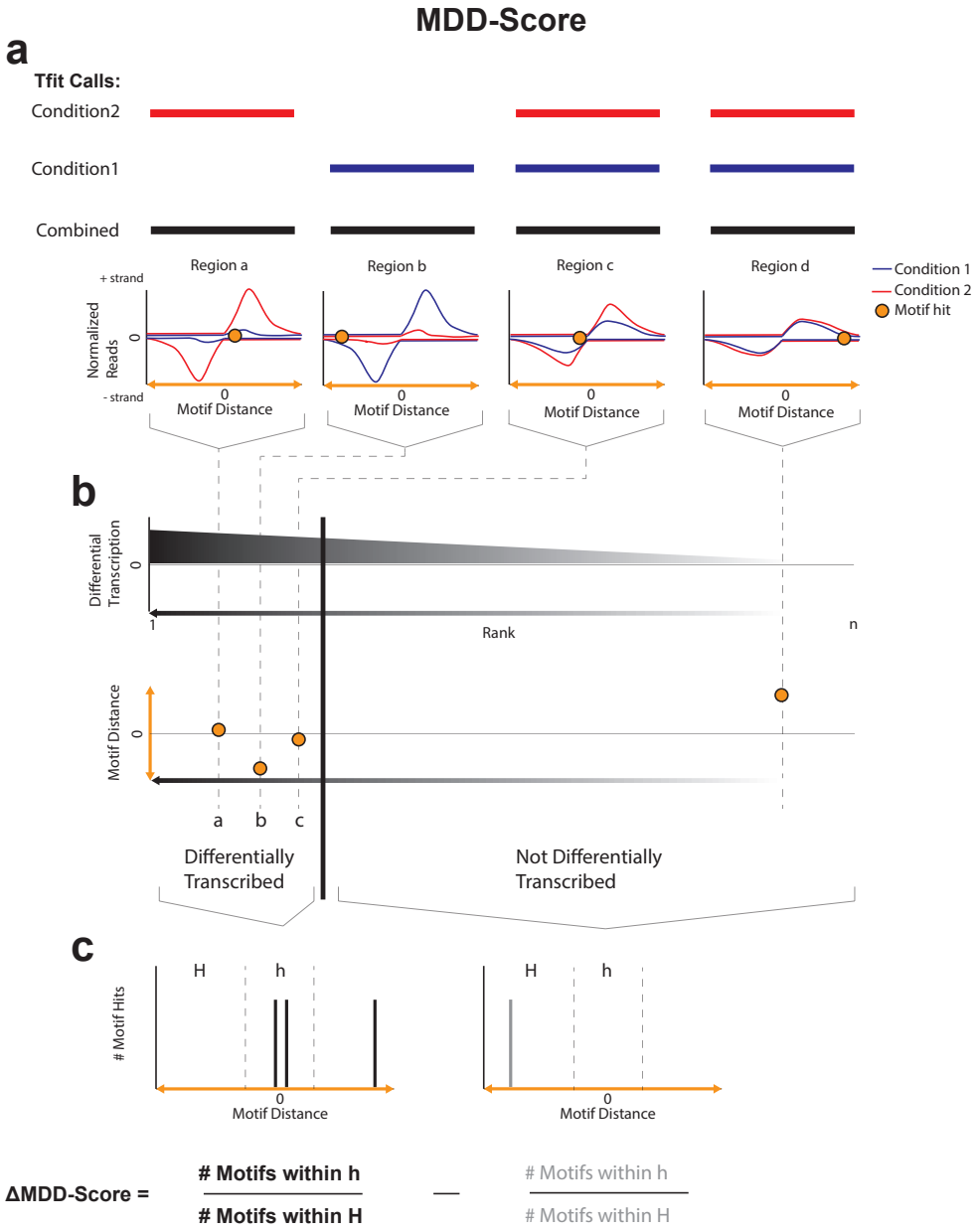


Figure 3.10: The differential MD-Score method (referred to as MDD-Score)165, 166 begins with (a) a collection of regions called in one or more conditions (red and blue boxes). Combined regions (black) are then (b) ranked by DESeq or DESeq2 p-value (depending on replicate number) and a cutoff segregates the differentially transcribed subset. (c) The MDD-Score is then the difference of MD-Score between the differentially transcribed set (black histogram) and the not differentially transcribed (grey histogram).

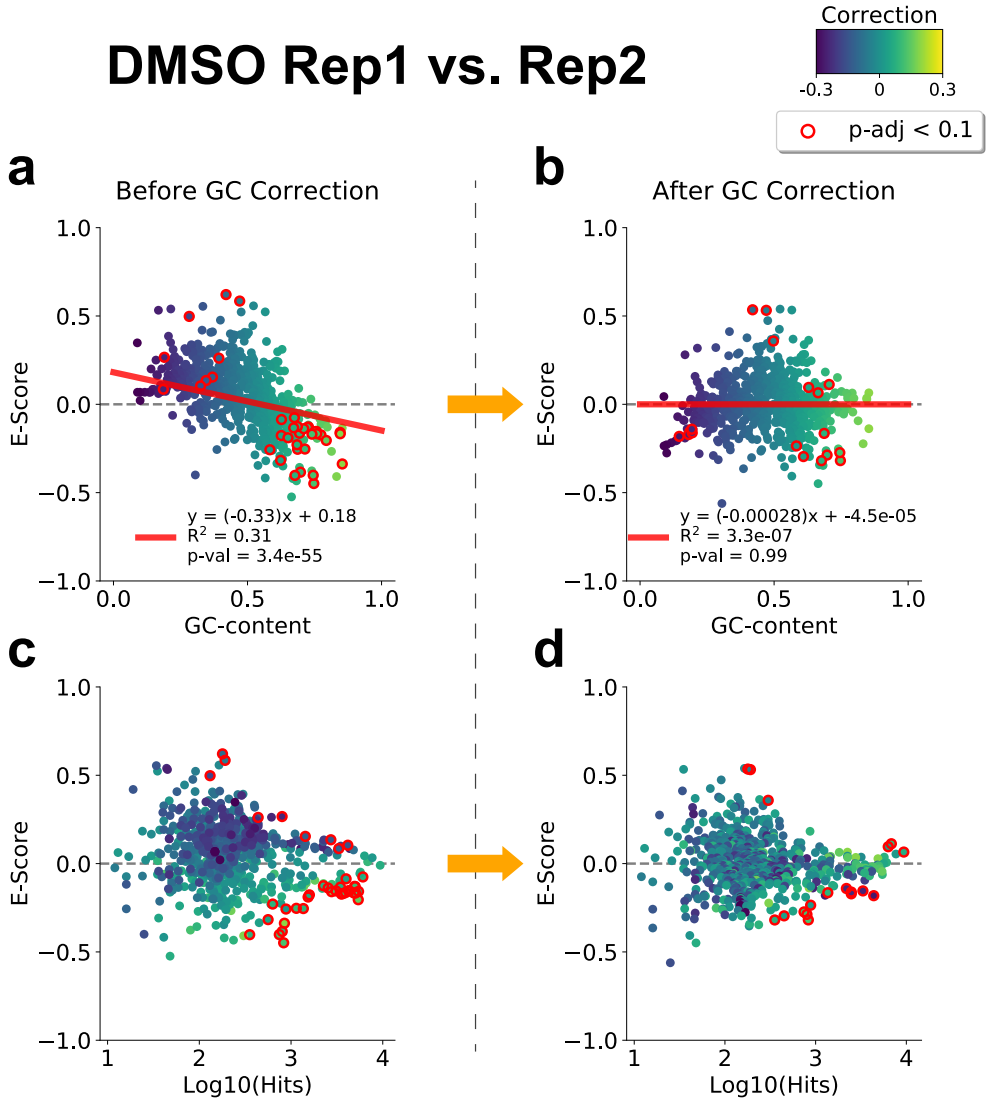


Figure 3.11: We observed that motif E-Scores often correlated with their GC-content. (a) Scatter plot of E-Score (y-axis) vs. GC-content (x-axis) of motifs, comparing replicate 1 vs. replicate 2 (DMSO condition) before GC-correction (red line: linear regression fit). (b) Scatter plot of E-Score (y-axis) vs. GC-content (x-axis) of motifs after GC correction (red line: linear regression fit). (c) MA plot of E-Score (y-axis) vs. Log10 number of motif hits within regions of interest (x-axis) before GC correction. (d) MA plot of E-Score (y-axis) vs. Log10 number of motif hits within regions of interest (x-axis) after GC-correction. These MA plots show that the underlying distribution of E-Scores relative to number of motif hits does not significantly change after GC-correction. All panels are data in HCT116 DMSO condition (SRR1105736, SRR1105737 61), dots are colored by the amount to be corrected due to GC-bias, red outline dots are p-adj < 0.1.

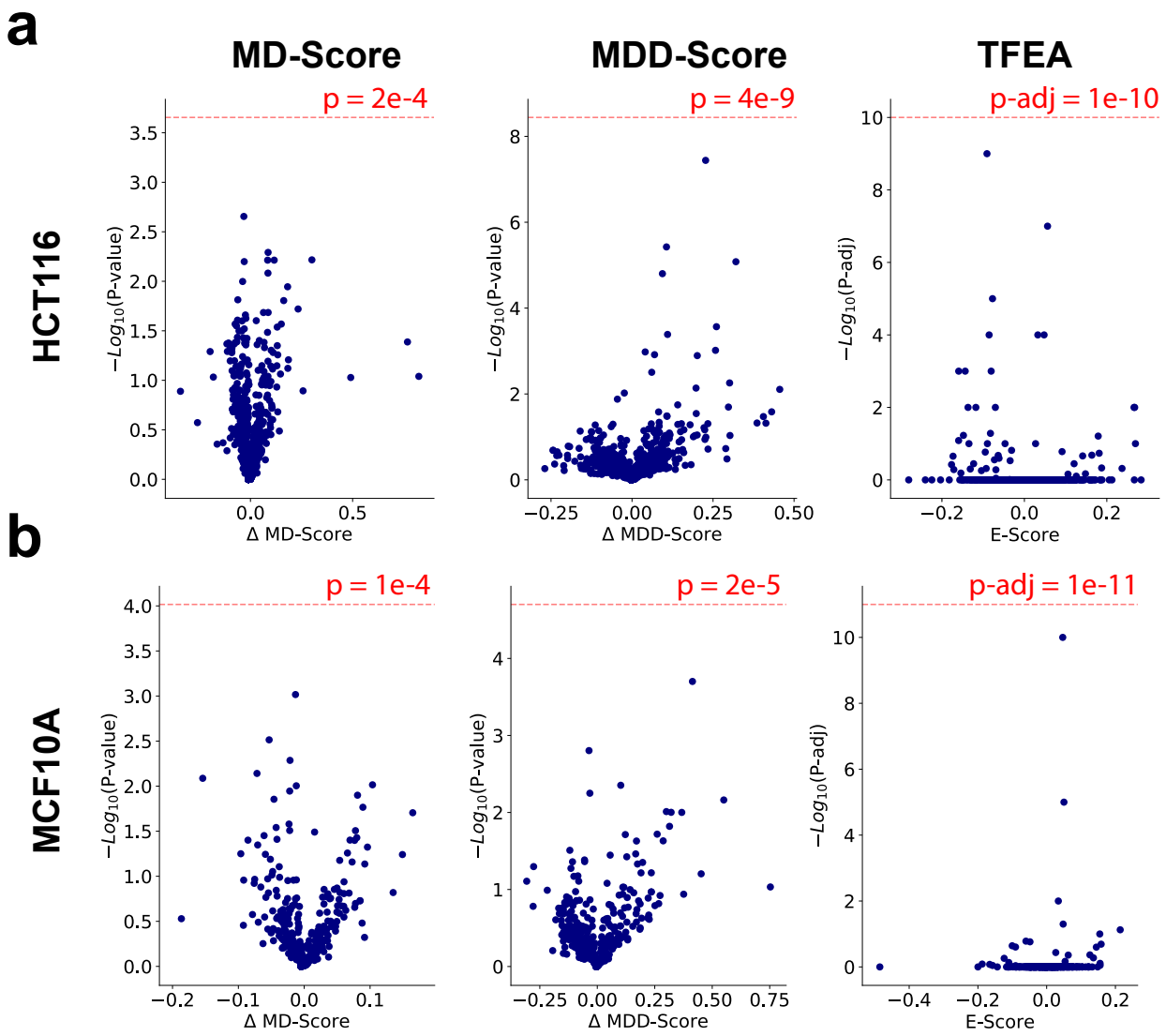


Figure 3.12: To choose a threshold cutoff for each of the three methods, DMSO replicates were compared and the threshold at which no false positives are obtained was determined. To be conservative, an additional order of magnitude is added for stringency. We performed this for each method in both (a) HCT116 and (b) MCF10A cells.

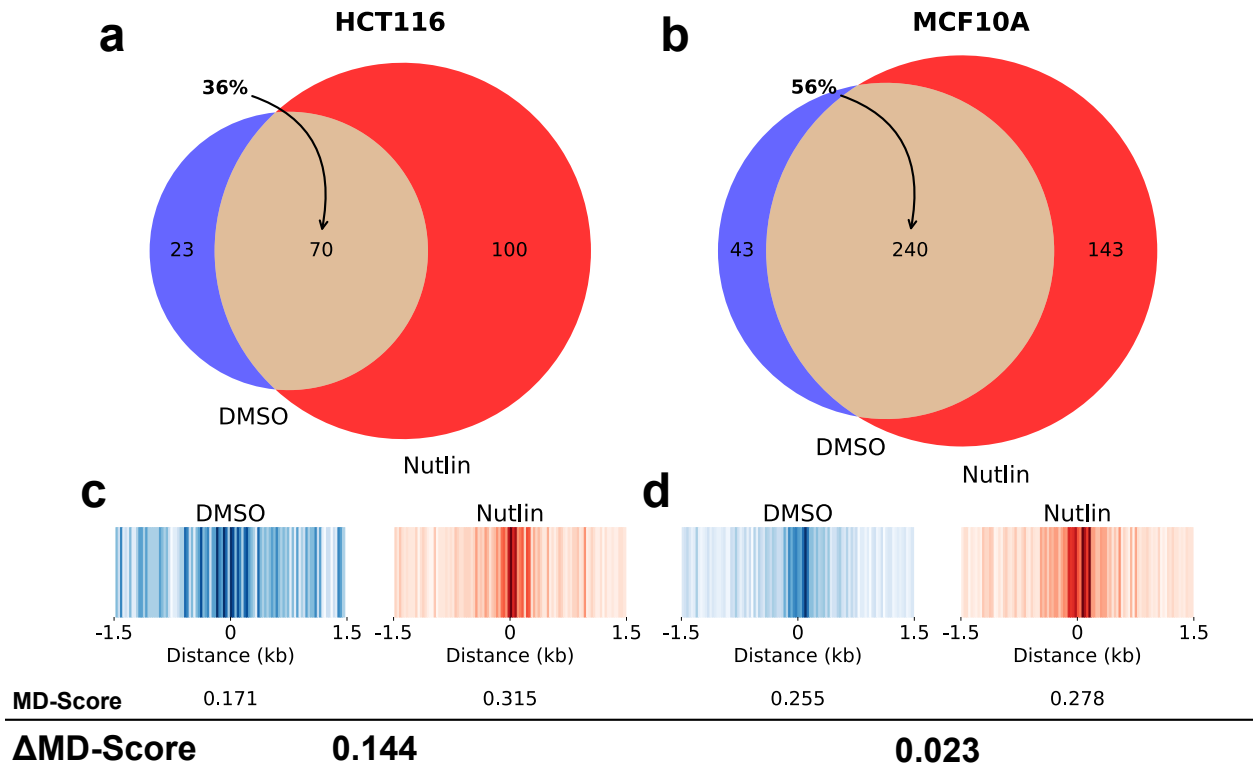


Figure 3.13: The response to Nutlin-3a visualized as Venn diagrams of (a) HCT116 and (b) MCF10a cells show a distinct p53 response, with a larger proportion (in MCF10A cells) of existing sites of RNA polymerase initiation (DMSO, blue) that respond to Nutlin-3a (red; overlap shown in tan). In both cases, only regions with p53 motif within 150 bps of the point of interest (midpoint of ROI) are shown. Motif displacement distributions of TP53 motif within 1.5 kb of ROI midpoints for (c) HCT116 or (d) MCF10A cells in DMSO (blue) and Nutlin-3a (red) conditions shows a higher co-localization of p53 in DMSO treated MCF10A cells. Bottom: MD-Score quantification for each condition followed by the observed ΔMD-Score for the Nutlin-3a response in each cell type.

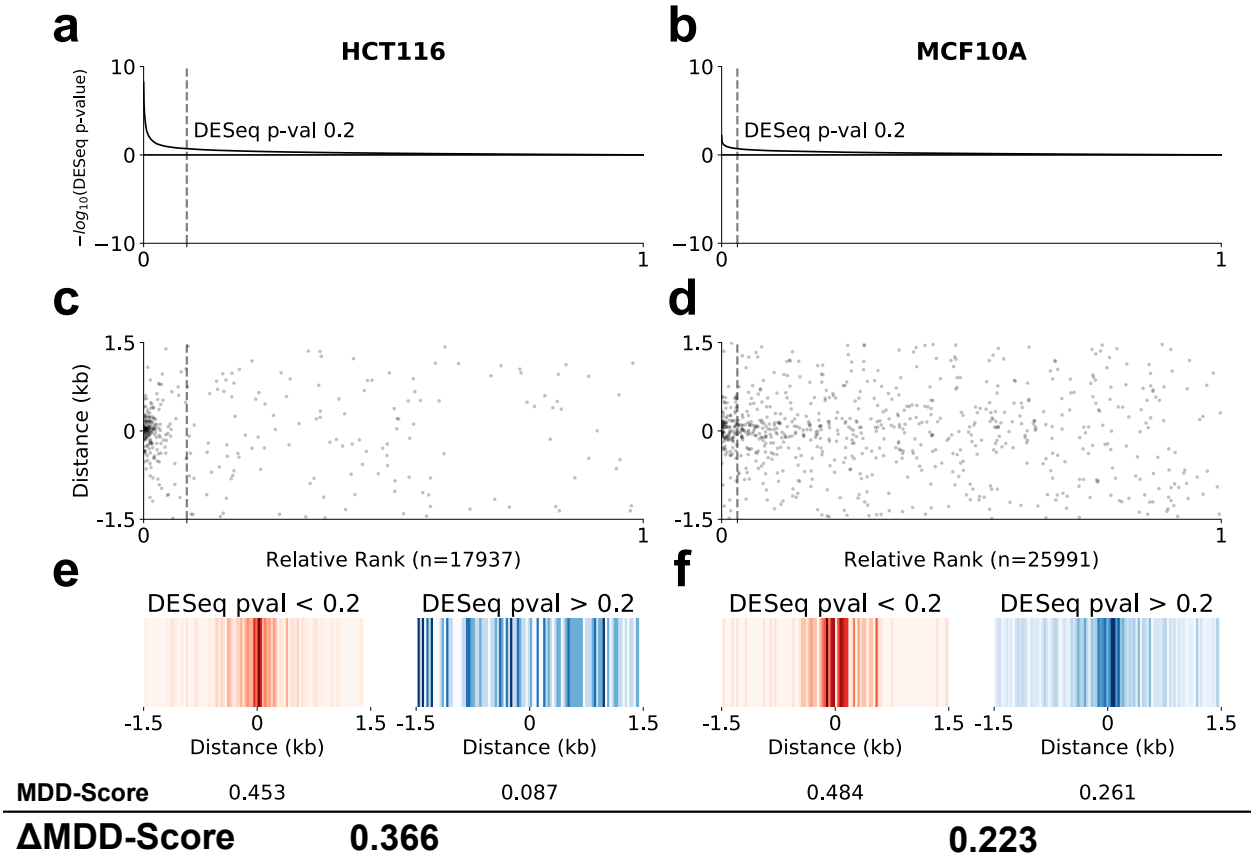


Figure 3.14: The MDD-Score approach detects p53 response in both (a) HCT116 and (b) MCF10a cells. By default, a loose DESeq2 p-value of 0.2 is chosen to identify the set of differentially transcribed ROI (similar to 165). Scatterplots show instances of TP53 motif across ranked ROI for (c) HCT116 and (d) MCF10A cells. The presence of constitutive TP63 activity leads MCF10a cells to have a higher background signal around TP53 motifs. Motif displacement distribution heatmaps for (e) HCT116 and (f) MCF10A cells, further emphasize the increased background presence of the TP53 motif in MCF10A cells. Red is control (DMSO), blue is Nutlin-3a treated. HCT116 data from SRR1105736, SRR1105737, SRR1105738, SRR1105739.

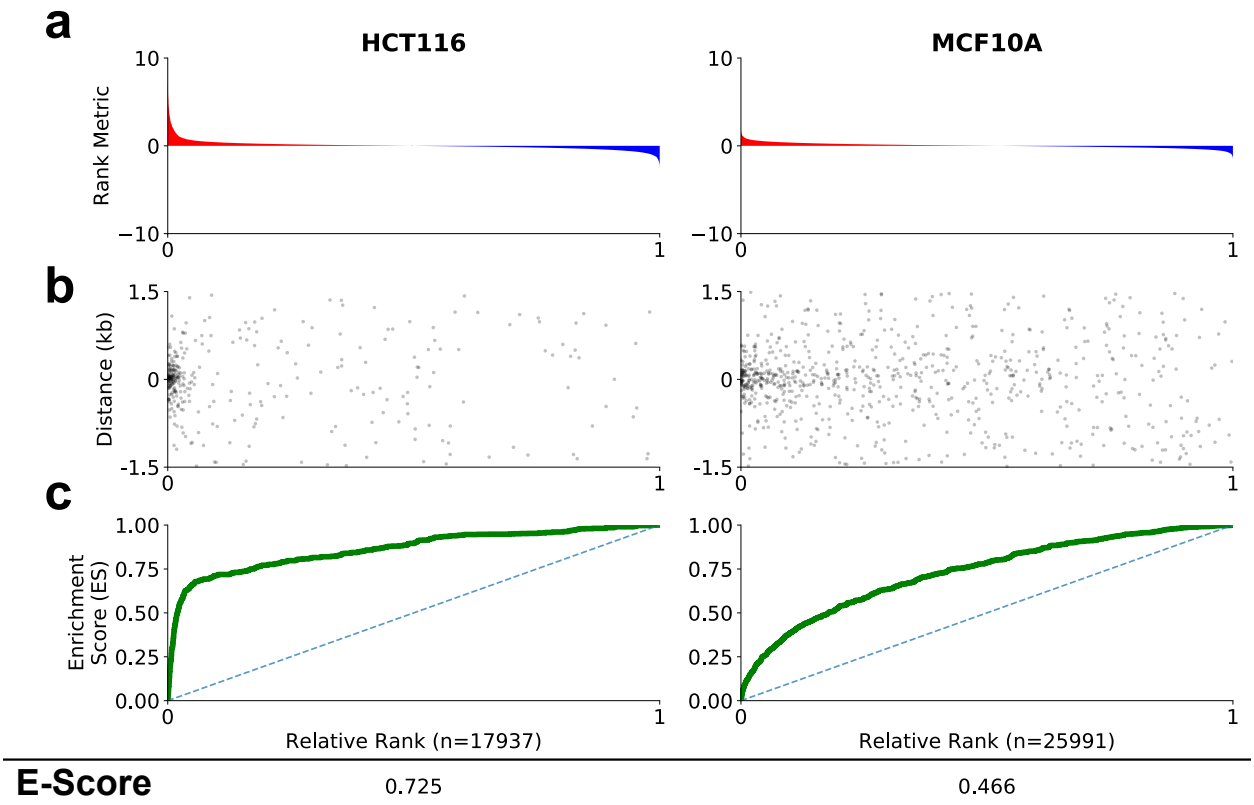


Figure 3.15: (a) ROI are ranked by differential transcription. Red: increased transcription, blue: decreased. (b) Instances of the TP53 motif are detected within ranked ROIs. (c) TFEA measures motif enrichment as the E-Score, calculated as 2* AUC (ie. area under the curve) between the running sum of ROI scores (green line) and the uniform distribution (dashed blue line). HCT116 data from SRR1105736, SRR1105737, SRR1105738, SRR1105739.

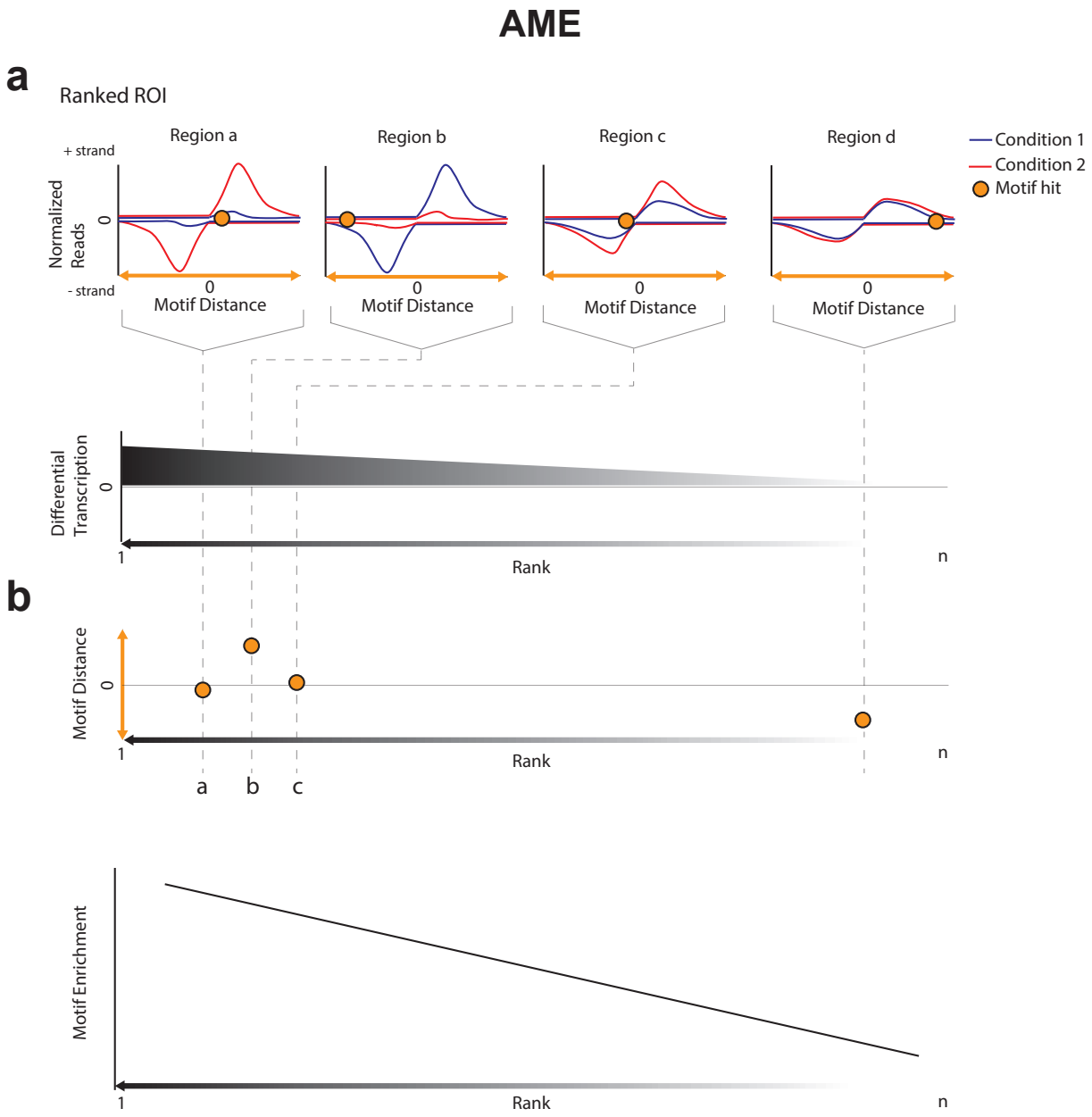


Figure 3.16: Analysis of Motif Enrichment (AME) is part of the MEME suite and requires (a) a ranked list of regions of interest (ROIs, labeled a-d) as input. AME then performs (b) linear regression on the motifs as a function of rank, ignoring the distance to motif (orange circles) information.

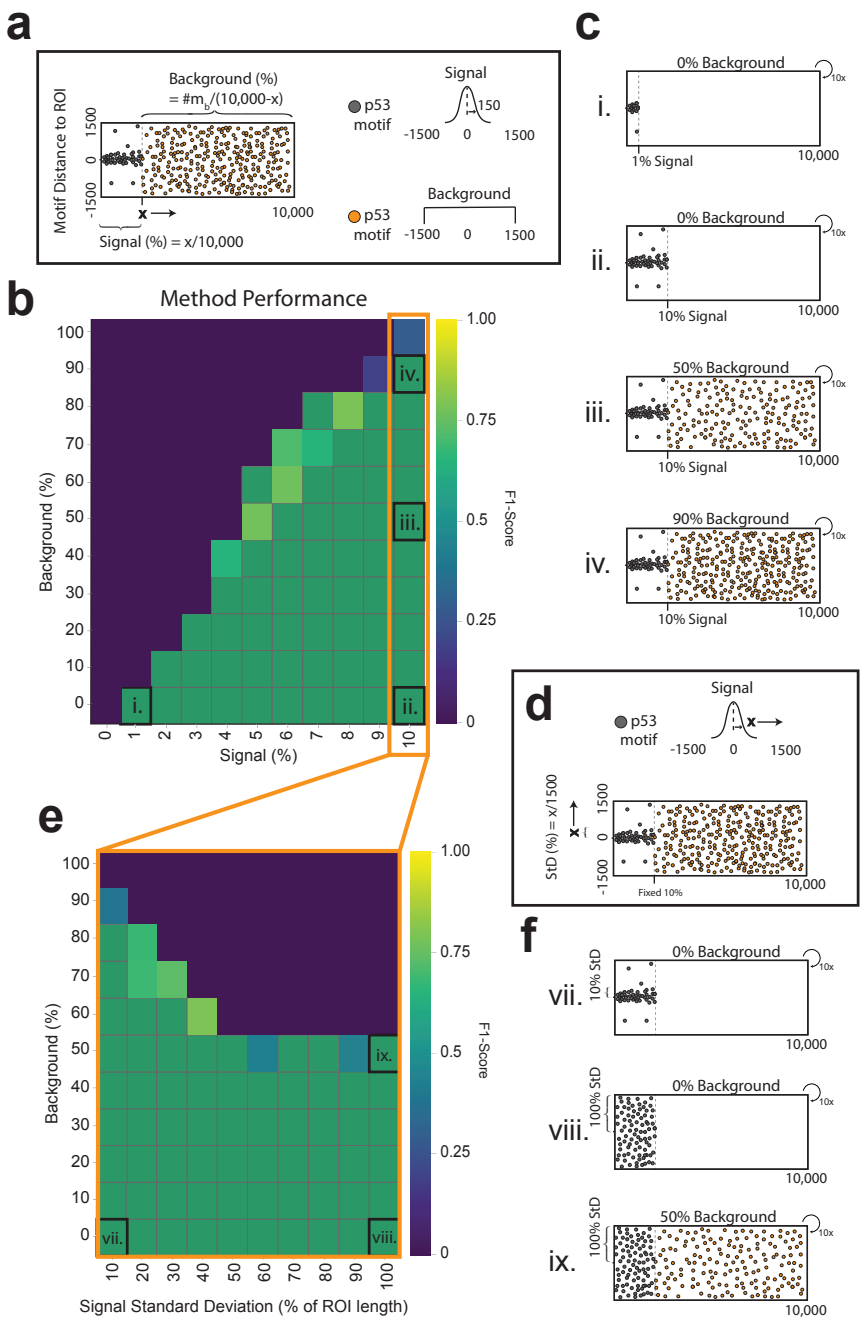


Figure 3.17: (a) A description of key concepts of motif embedding strategy for both signal (grey) and background (orange). (b) F1-Score (as heatmap) for varying fraction of ROI with signal (x-axis) and background (y-axis). Representative tests cases are labeled (i-iv) and their (c) respective embedding strategies are shown. (d) A description of additional criteria utilized for altering the variability of signal embedding. (e) For 10% signal, we additionally alter the signal standard deviation (x-axis) vs background (y-axis). Representative cases (vii-ix) are labeled and their (f) respective embedding strategies are shown.

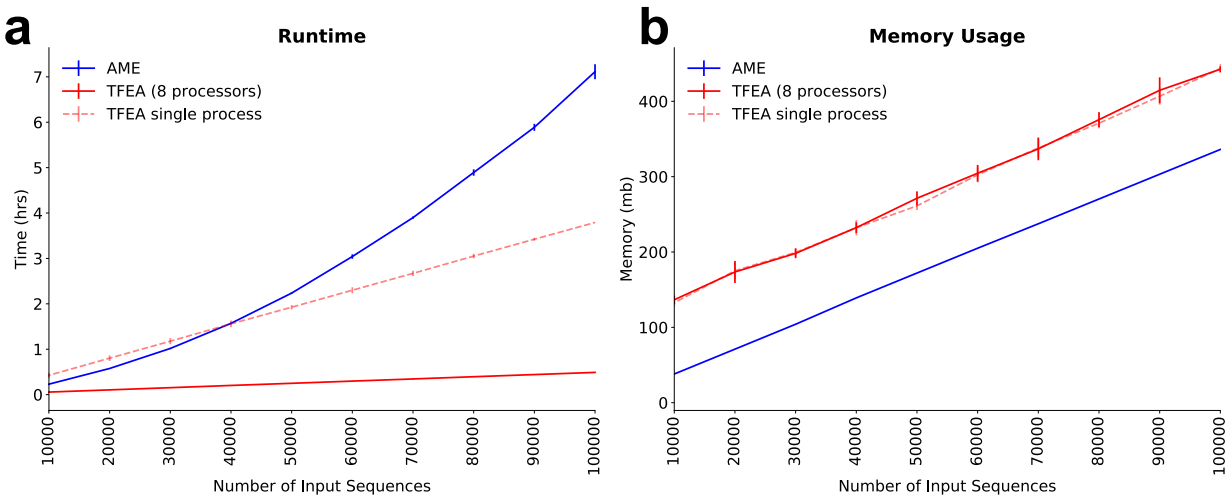


Figure 3.18: (a) Runtime statistics for AME (solid blue; parallel processing not supported) and TFEA (8 processors: solid red; 1 processor: dashed red) with varying numbers of input ROI (bars = standard deviation of 10 runs). (b) Memory usage statistics comparing AME to TFEA.

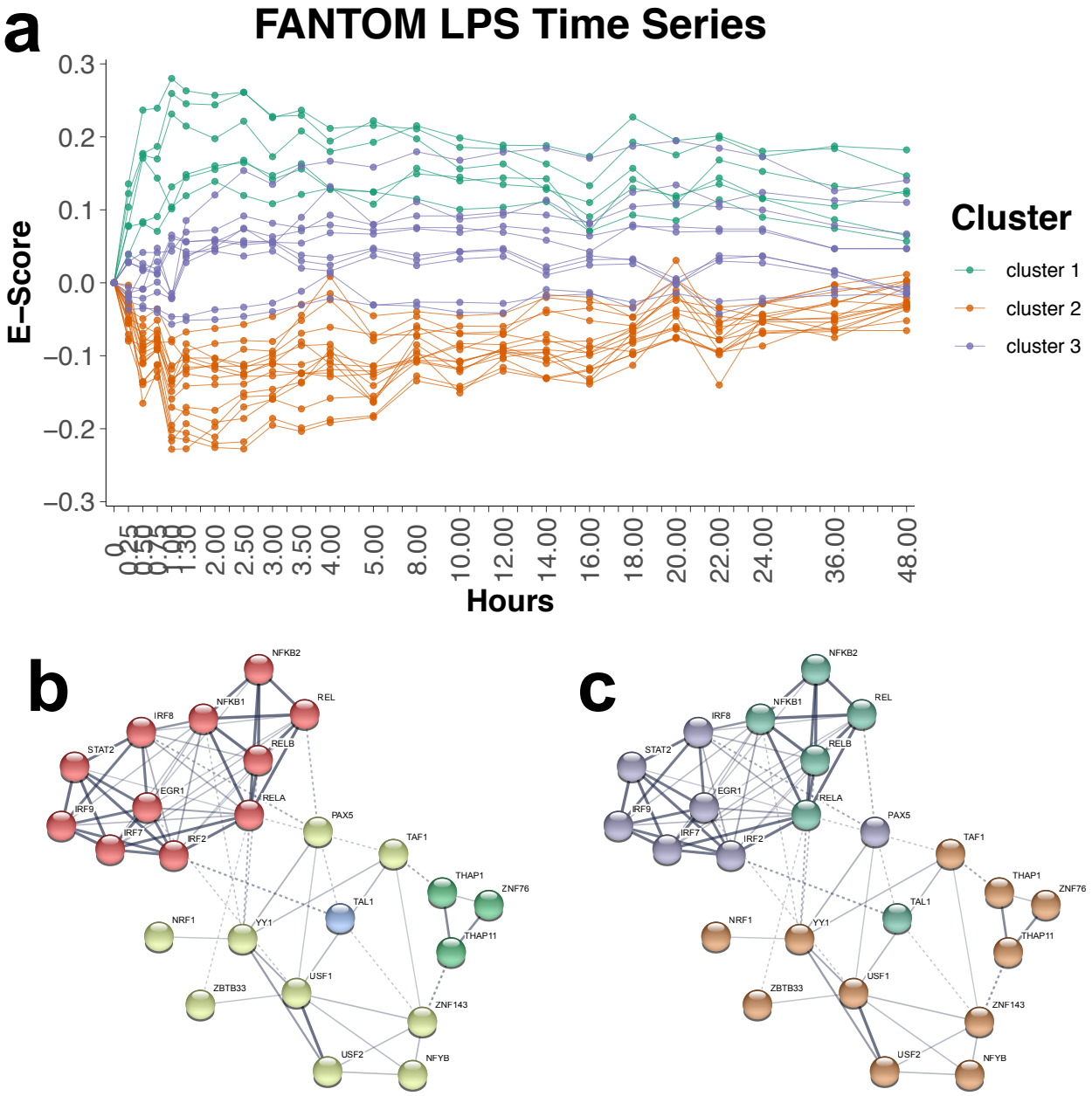


Figure 3.19: We applied k-means clustering to the subset of TFs that were significant (by TFEA) in at least 15 time points (~2/3 of all timepoints; n=32 TFs). (a) Time series traces of significant TFs colored by resulting cluster. The three main clusters correspond to the immediate increased response (cluster 1, green), the immediate decreased response (cluster 2, orange) and the later responding TFs (cluster 3, purple). (b) Alternatively the TFs can be analyzed using the String database using the Markov cluster algorithm. (c) Superposition of the coloring scheme in (a) onto the network cluster of (b).

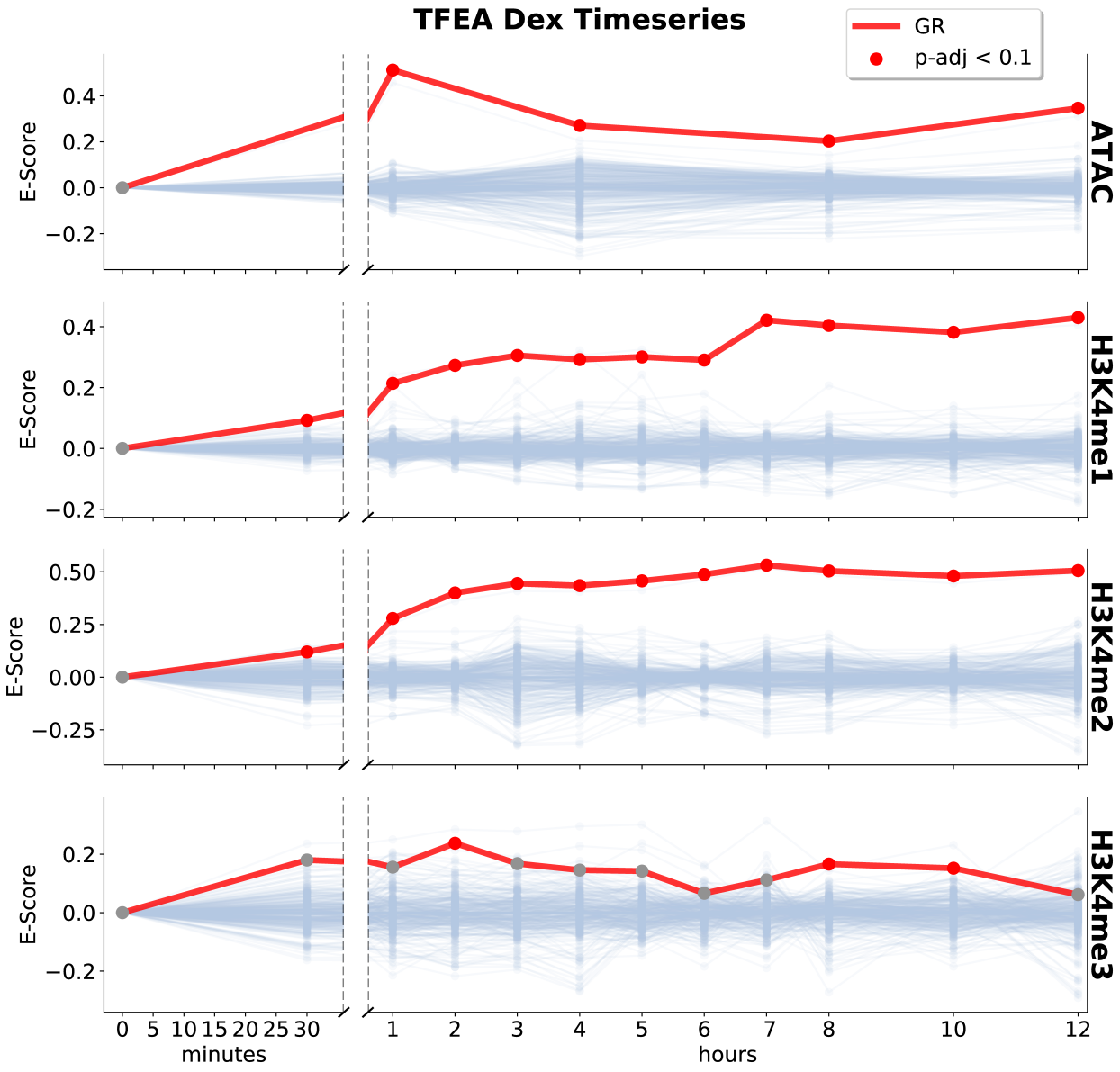
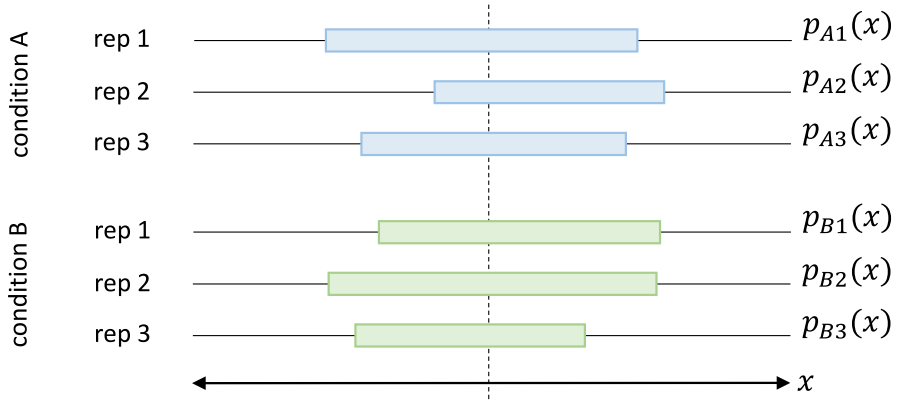


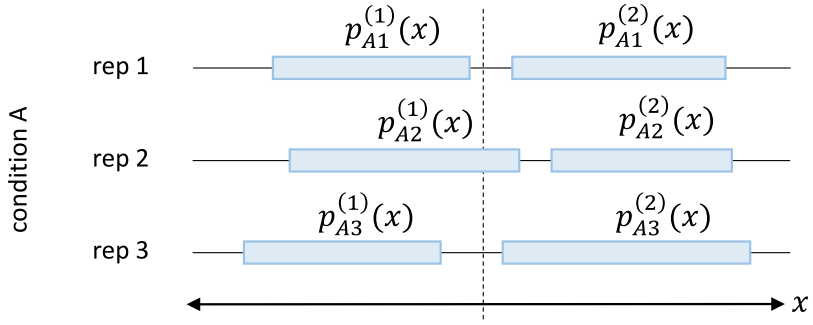
Figure 3.20: TFEA is able to recover GR (red line) in many distinct data sets including ATAC, H3K4me1, and H3K4me2. Interestingly, TFEA only detects moderate enrichment of GR in H3K4me3, in agreement which previous results indicating that GR primarily binds to enhancers (which do not have the H3K4me3 mark)183. Grey lines are trajectories of other TFs.

a 1 peak, 3 replicates, 2 conditions



$$P_{joint}(x) \propto p_{A1}(x) \cdot p_{A2}(x) \cdot p_{A3}(x) + p_{B1}(x) \cdot p_{B2}(x) \cdot p_{B3}(x)$$

b 2 peaks, 3 replicates, 1 condition



$$P_{joint}(x) \propto [p_{A1}^{(1)}(x) + p_{A1}^{(2)}(x)] \cdot [p_{A2}^{(1)}(x) + p_{A2}^{(2)}(x)] \cdot [p_{A3}^{(1)}(x) + p_{A3}^{(2)}(x)]$$

Figure 3.21: .
Here we have two hypothetical examples of overlapping sample regions (rep 1,2,3) from two different experiments. The corresponding calculation of $\mathcal{P}_{joint}$ (from individual sample probability distributions $p_{ij}^{k}(x)$) is shown for both examples, as a function of genomic coordinate x. (a) A genomic location for an example experiment consisting of two conditions (condition A: blue, condition B: green), each of which has three replicates (rep). Each $p_{ij}^{(k)}(x)$ is the normal distribution representing the corresponding sample region—Eq. IV.1 in the Methods. (b) A genomic location for an example experiment consisting of a single condition, which has three replicates. This region contains two distinct (but closely spaced) loci in each replicate. NOTE: due to the positioning of individual sample regions, *bedtools merge* would produce a single, large ROI and *bedtools intersect* would produce three separate ROIs with the middle one being very narrow. Conversely, *muMerge* would produce two distinct ROIs—the ideal outcome for this particular example.

APPENDIX D

SUPPLEMENT TO CHAPTER 5

This section contains supplement figures and table to chapter 5. The figures are referenced in the main text and expand on the main text figures.

Supplemental Figures

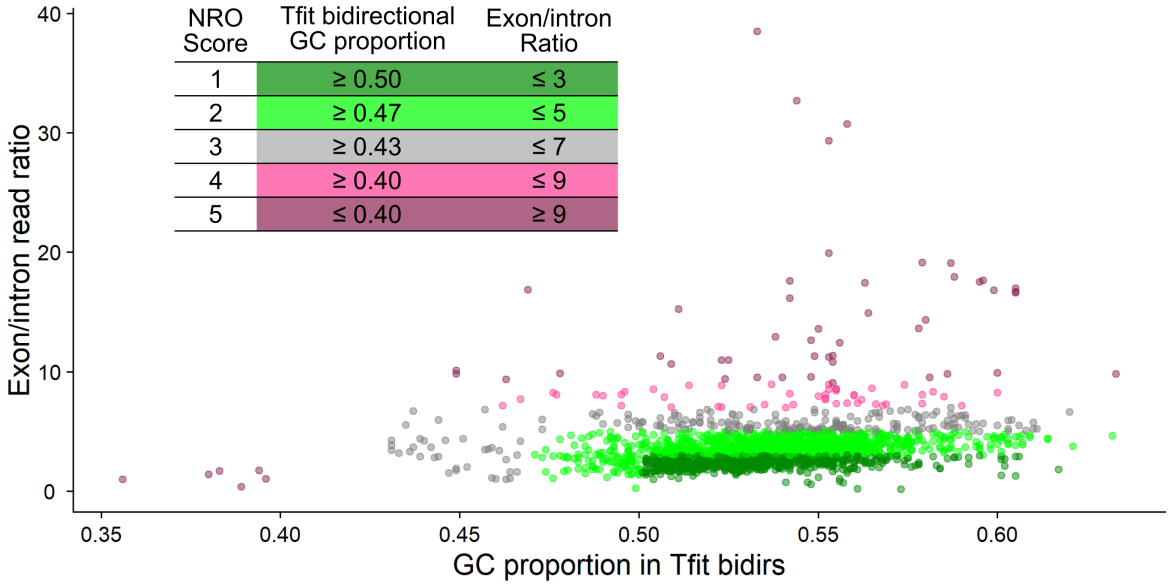


Figure 4.1: Distribution of NRO scores for human and mouse samples in dbNascent. If Tfit was successfully run on a sample, it was incorporated into the NRO score, otherwise the score was solely based on exon/intron ratio.

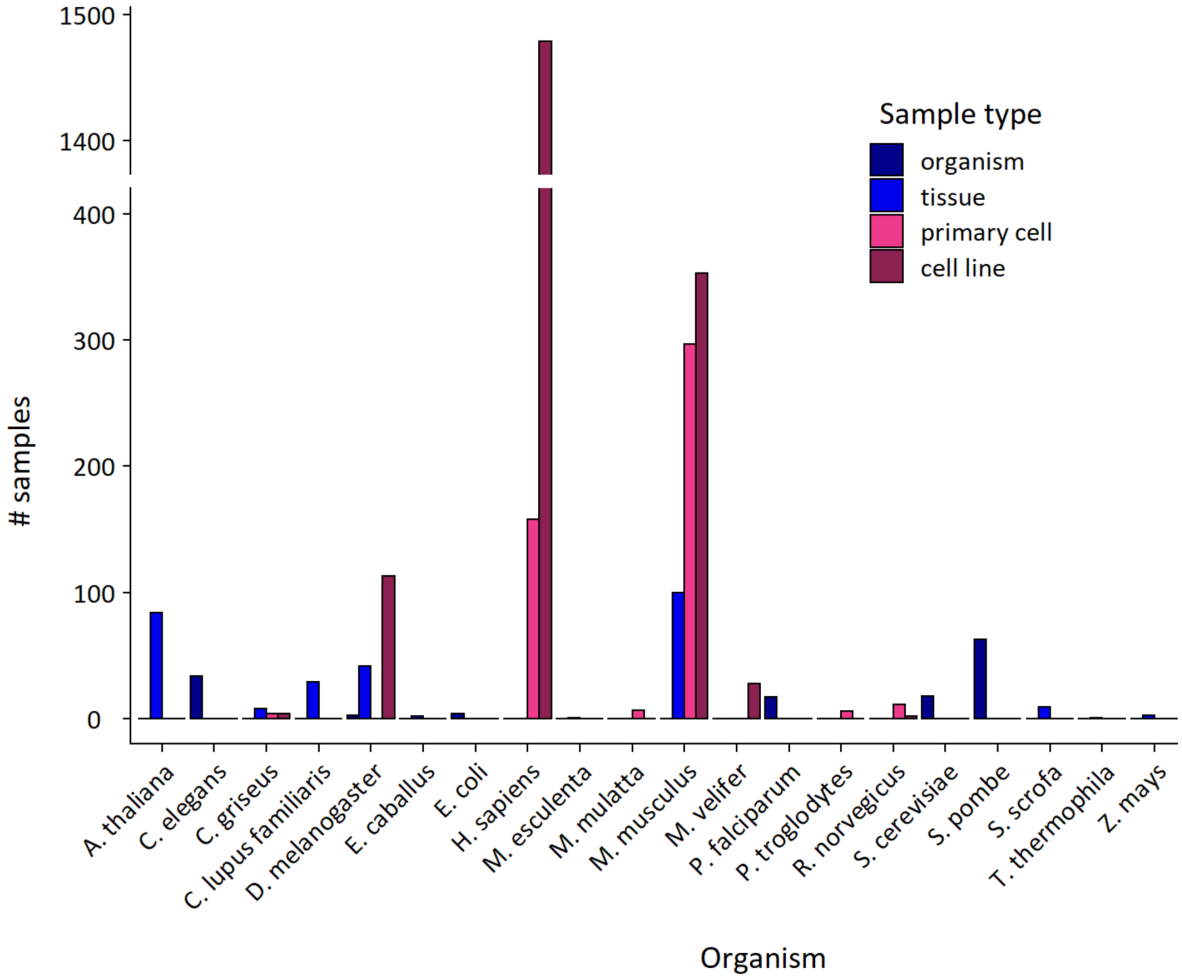


Figure 4.2: Sample types represented in dbNascent. The most represented cell type across the database are cell types, followed by primary cells. Most of these sample types are found in human and mouse samples.

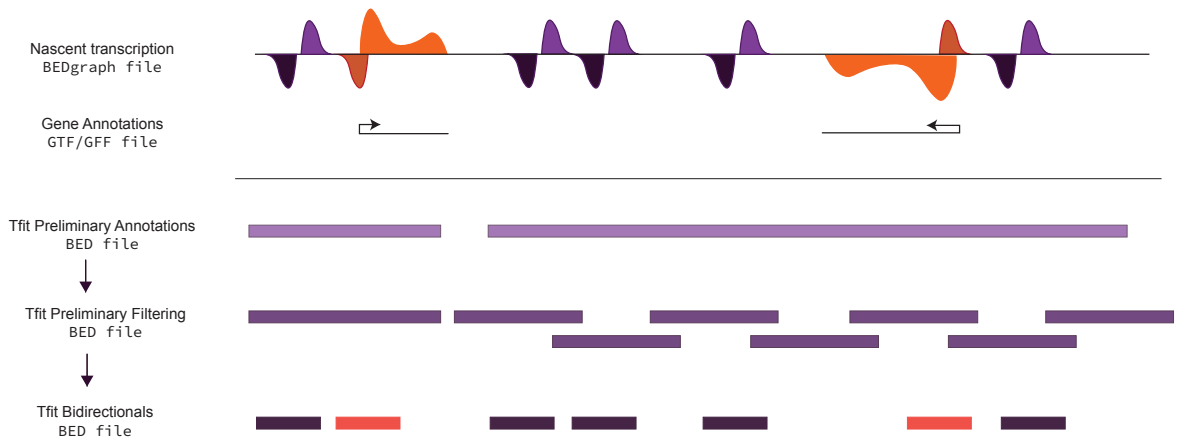


Figure 4.3: Updated Tfit preliminary filter. This schematics shows the optimized Tfit preprocessing step. Tfit takes as input coverage files in the format of BEDgraphs. In this preprocessing step we take into account gene regions (GTF file format) and use these regions to inform Tfit. This updated method returns gene annotations in regions where bidirectional transcripts overlap genes.

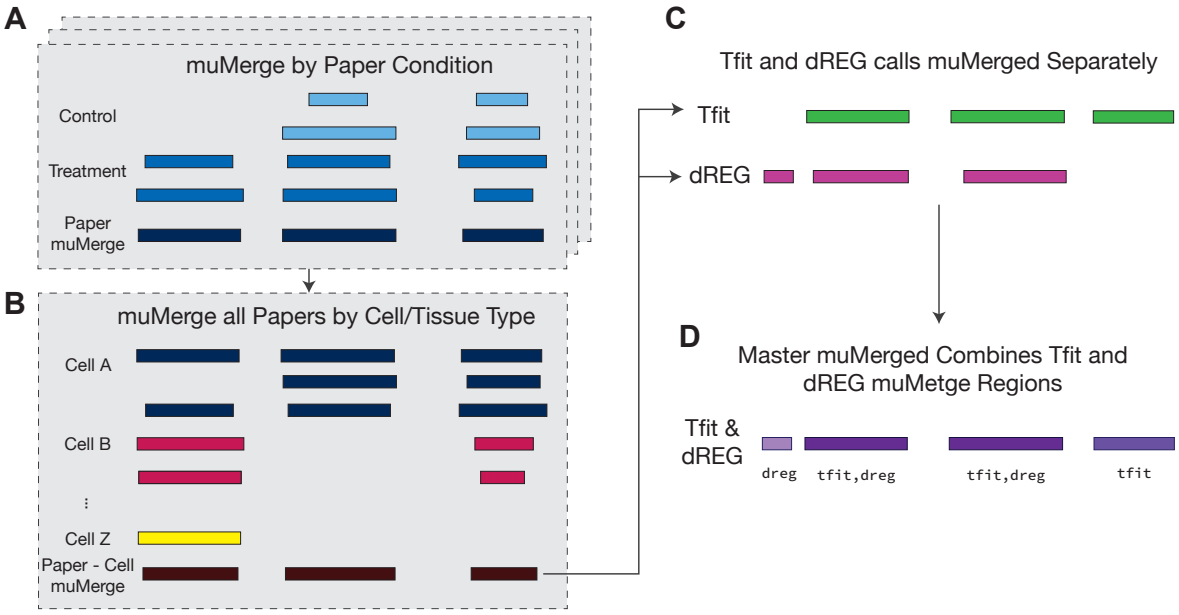


Figure 4.4: Defining regions of nascent RNA transcription across multiple experiments. In each experimental setup, regions were muMerged using the conditions specific to the paper experimental design. Once all the papers were merged, the paper muMerged regions were comined based on the cell/tissue type. The muMerge was performed for Tfit and dREG separately, and to combine the regions, Tfit regions were used when Tfit and dREG overlapped and respective unique regions from Tfit and dREG were also used.

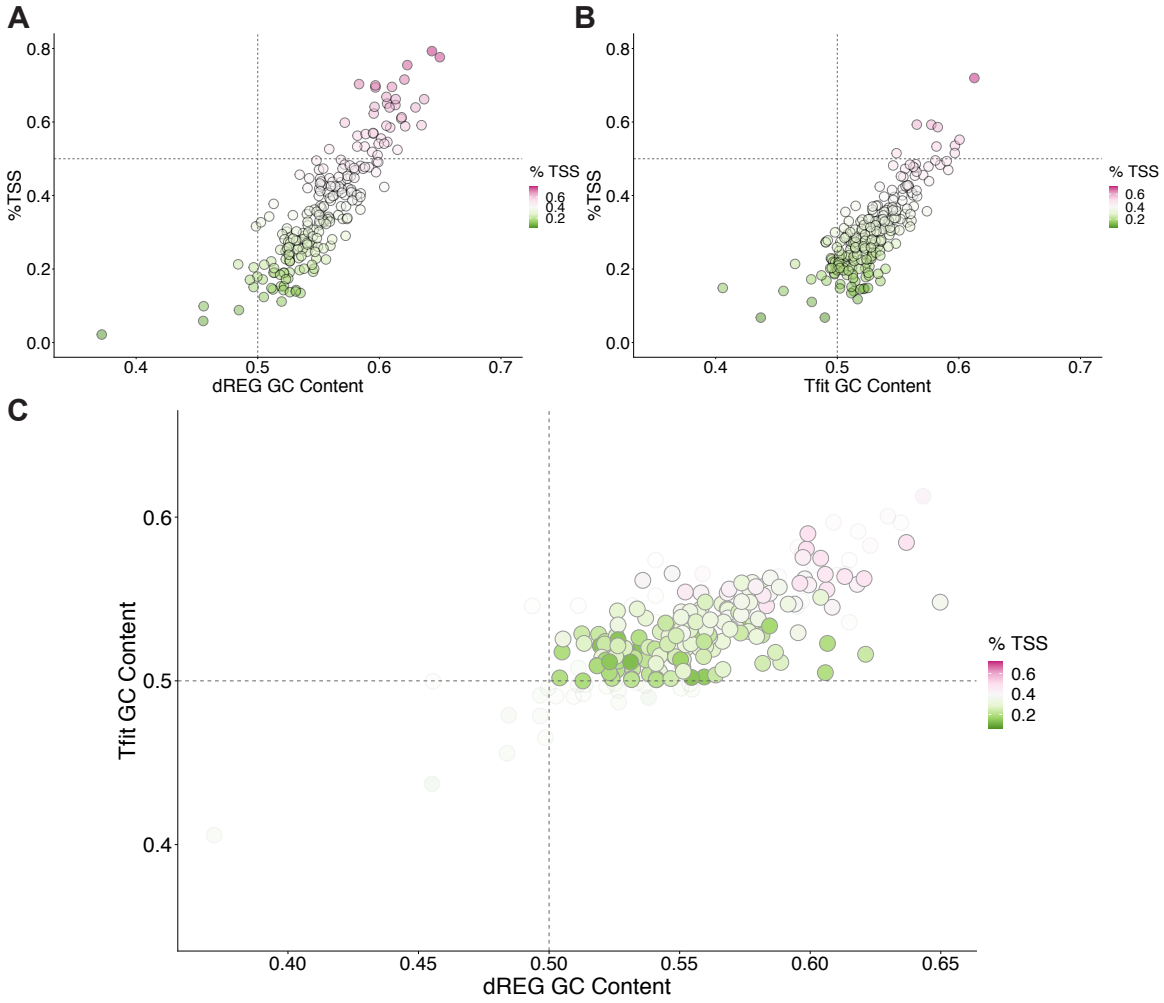


Figure 4.5: Base composition and fraction of TSS overlapping bidirectionals. (A) dREG and (B) Tfit GC content and percent of TSS overlapping bidirectionals. (C) dREG versus Tfit GC content and the points are colored by %TSS for Tfit regions. The shaded points represent papers filtered out from downstream analyses.

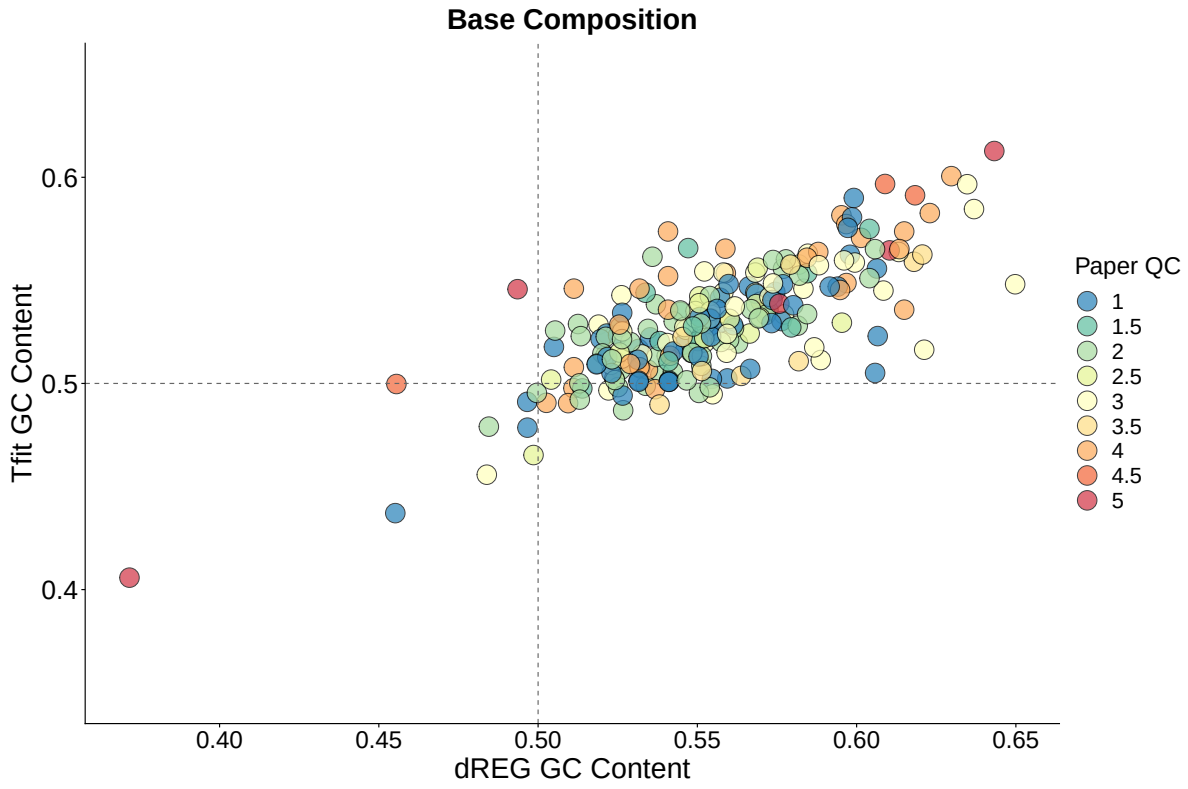


Figure 4.6: Base composition and average paper quality of called bidirectional transcripts. Each point represents a paper, the x-axis represents the GC content of regions called by dREG and the y-axis are regions called by Tfit. The points are colored by the average paper quality. In general, we see that samples poor QC scores (4-5) tend to have a higher GC composition (moreso in dREG calls).

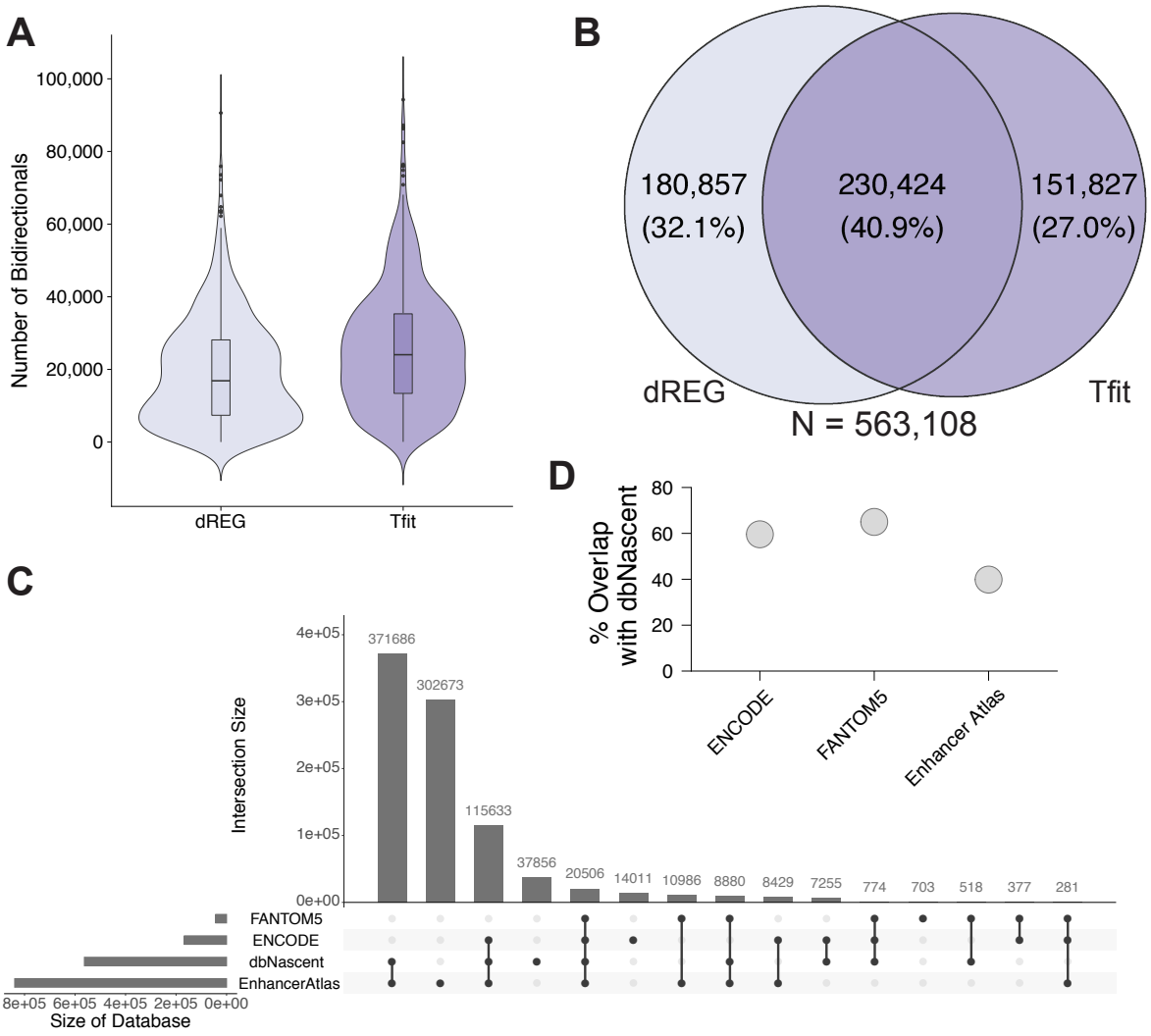


Figure 4.7: Summary of mouse muMerged regions. Parallel to Figure 5.2 but for mouse data. (A) Distribution of bidirectional calls for both dREG and Tfit show median calls around 20000 with Tfit calling slighly more regions. (B) Overlap of muMerge regions from Tfit and dREG show about 40% overlap between the regions. (C) Comparing the final muMerge regions with other databases shows greatest overlap with Enhancer Atlas. A total of 20506 regions are found in all databases. (D) Over 40% of ENCODE, FANTOM5 and Enhancer Atlas overlap with dbNascent.

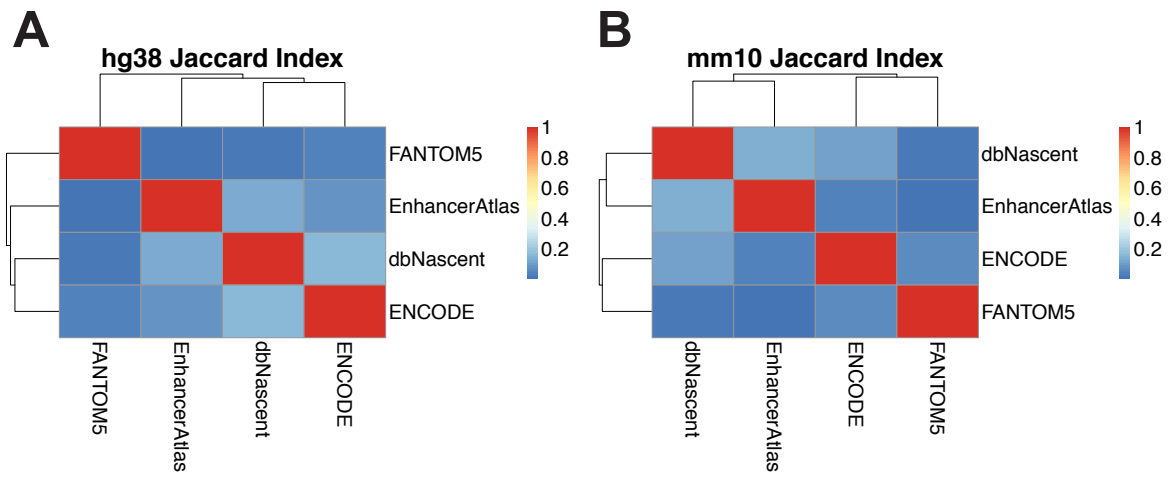


Figure 4.8: Jaccard indices between enhancer databases. Jaccard indices between cis-regulatory regions from dbNascent ENCODE, EnhancerAtlas, and FANTOM5 for (A) human and (B) mouse annotations. dbNascent calls are more similar to all the other databases in human samples. In mouse samples, dbNascent and enhancer atlas are more similar.

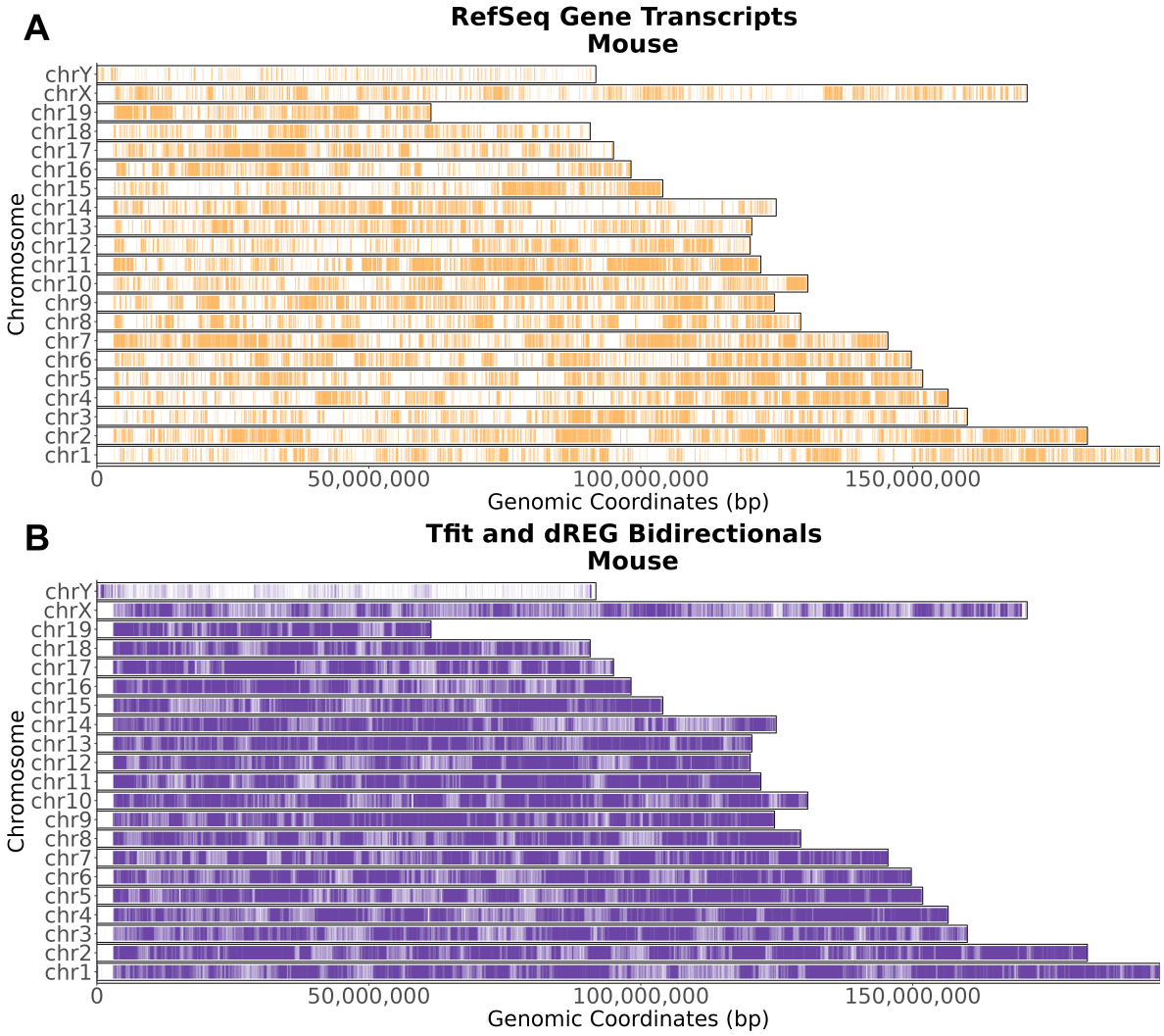


Figure 4.9: Mouse transcribed regions. (A) Genes and (B) bidirectionals transcribed marked as a point on each chromosome. The called bidirectionals and genes span across the same mouse genome locations.

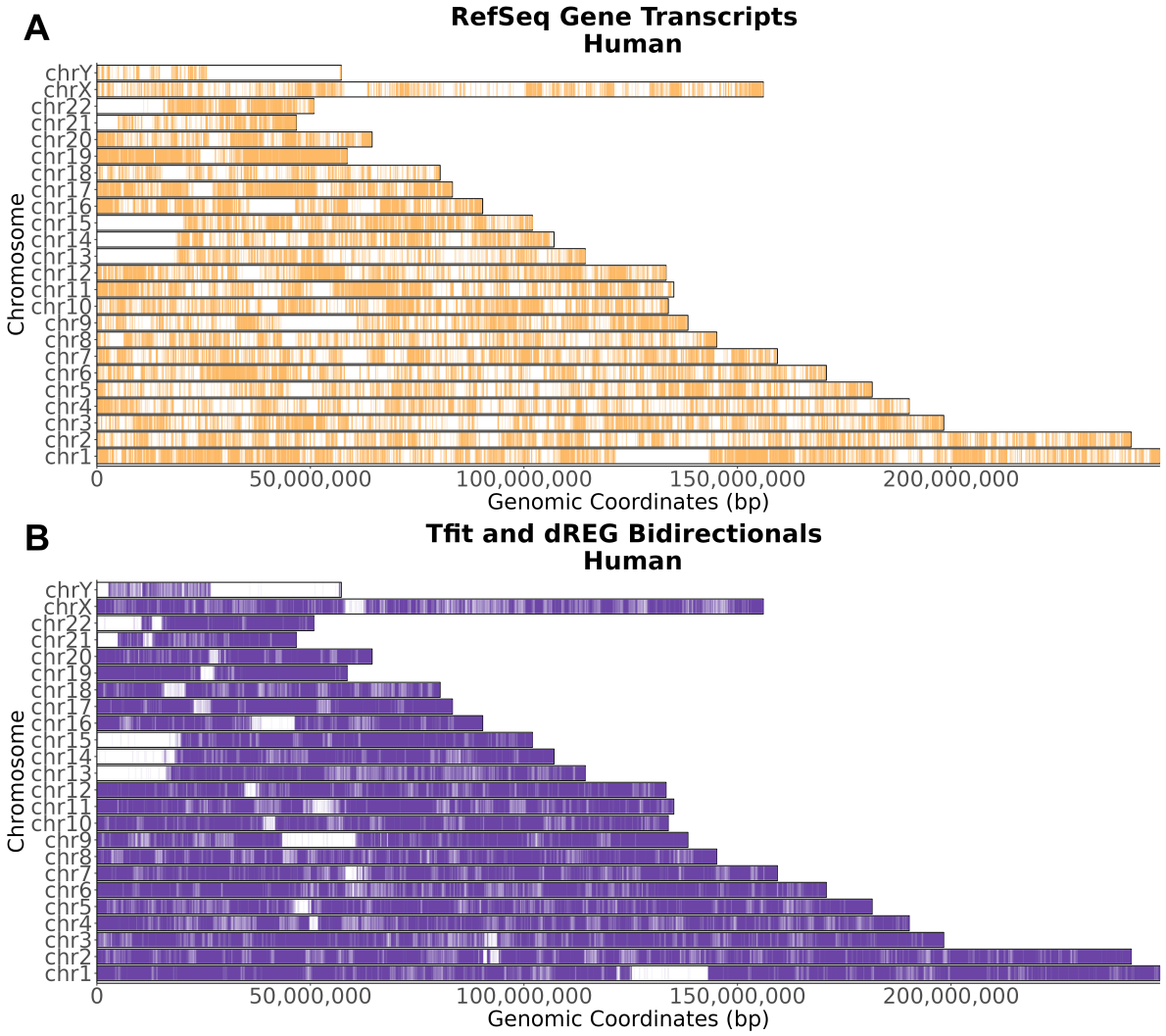


Figure 4.10: Human transcribed regions. A) Genes and (B) bidirectionals transcribed marked as a point on each chromosome. The called bidirectionals and genes span across the same human genome locations.

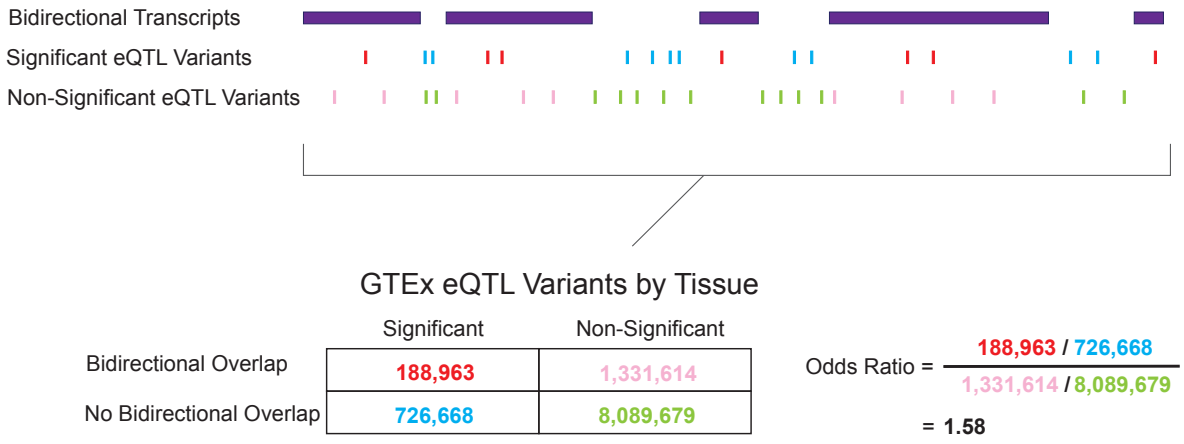


Figure 4.11: Schematic for odds ratio for GTEx eQTLs and bidirectional transcript overlap with GTEx Breast Mammary Tissue eQTKL variant counts. To calculate odds ratio, eQTL variants that overlapped bidirectional transcripts and those that did not overlap bidirectional transcripts were counted for both significant and non-significant GTEx eQTLs variants. The Odds Ratios were calculated for all GTEx tissues.

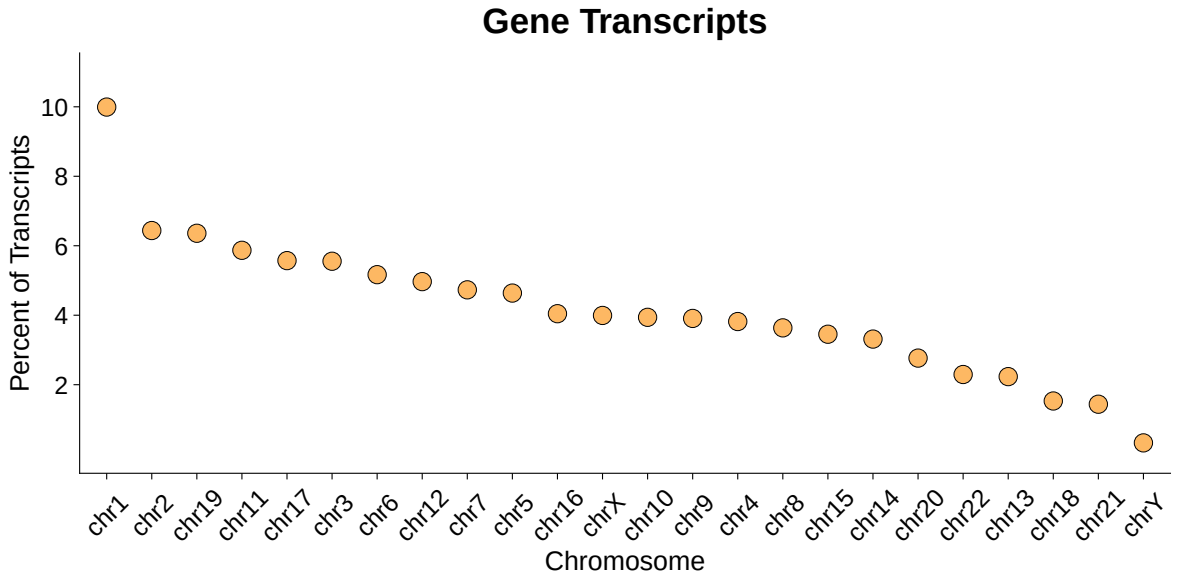


Figure 4.12: Percentage of gene transcripts per chromosome in human samples. Larger chromosome have more gene transcription.

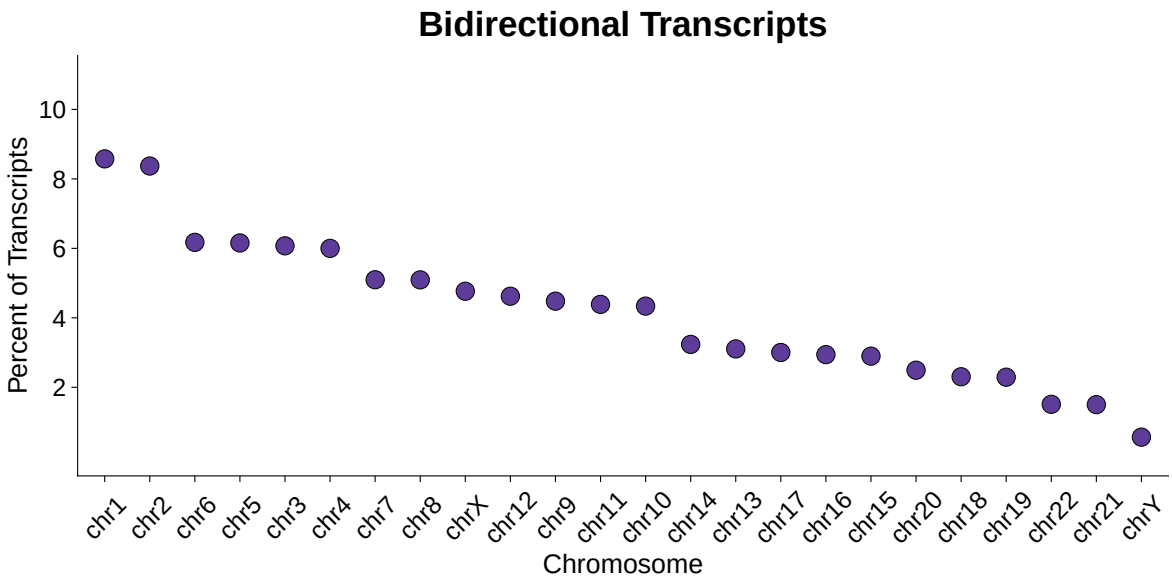


Figure 4.13: Percentage of bidirectional transcripts per chromosome in human samples. Larger chromosome have more bidirectional transcription.

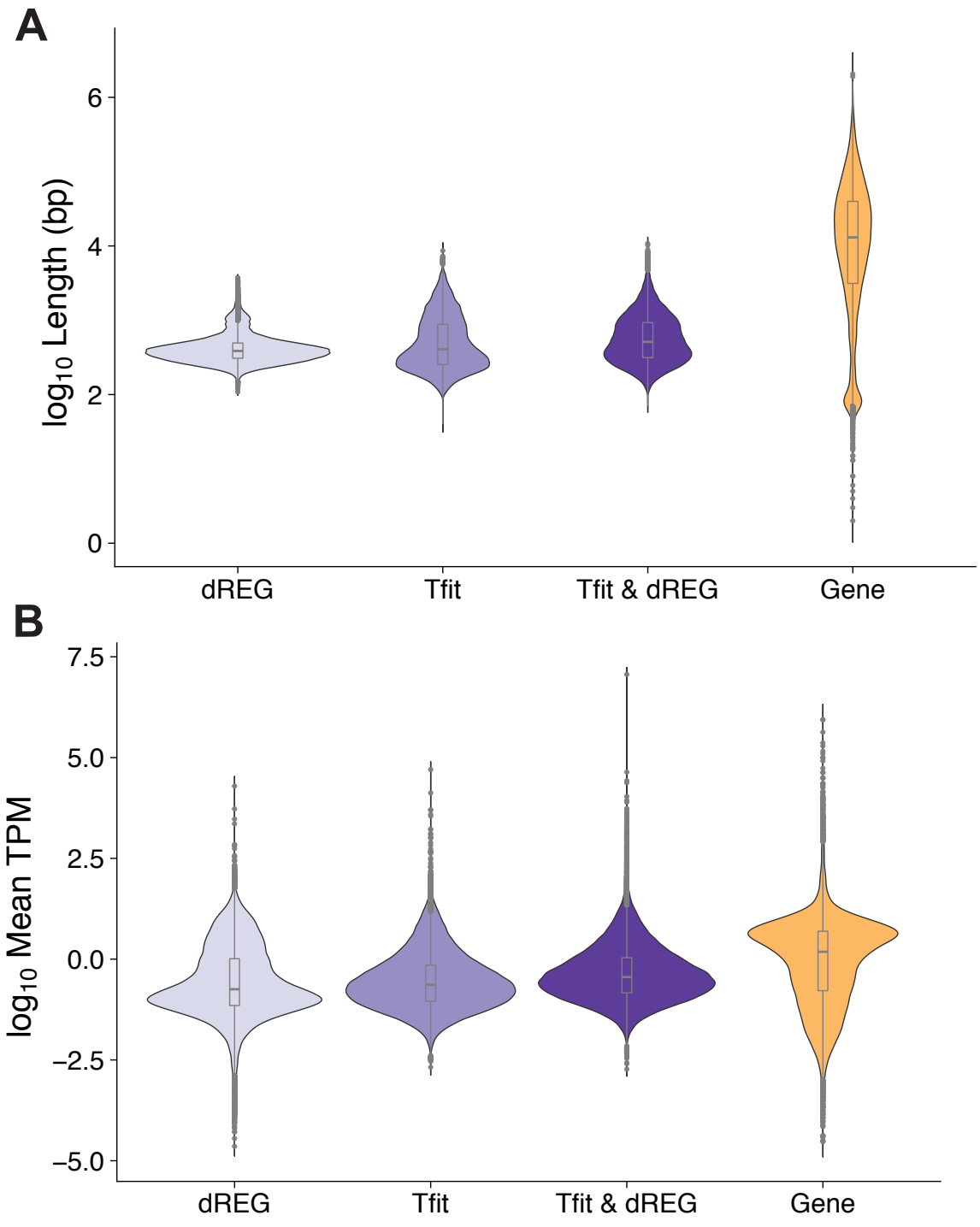


Figure 4.14: Transcript features for human transcripts. (A) The length distribution of transcripts shows regions called by dREG have the tightest distribution followed by Tfit and genes have a wider range compared to bidirectional transcripts. (B) Normalized gene counts show that genes have a higher transcripts per million (TPMs) compared to bidirectional transcripts.

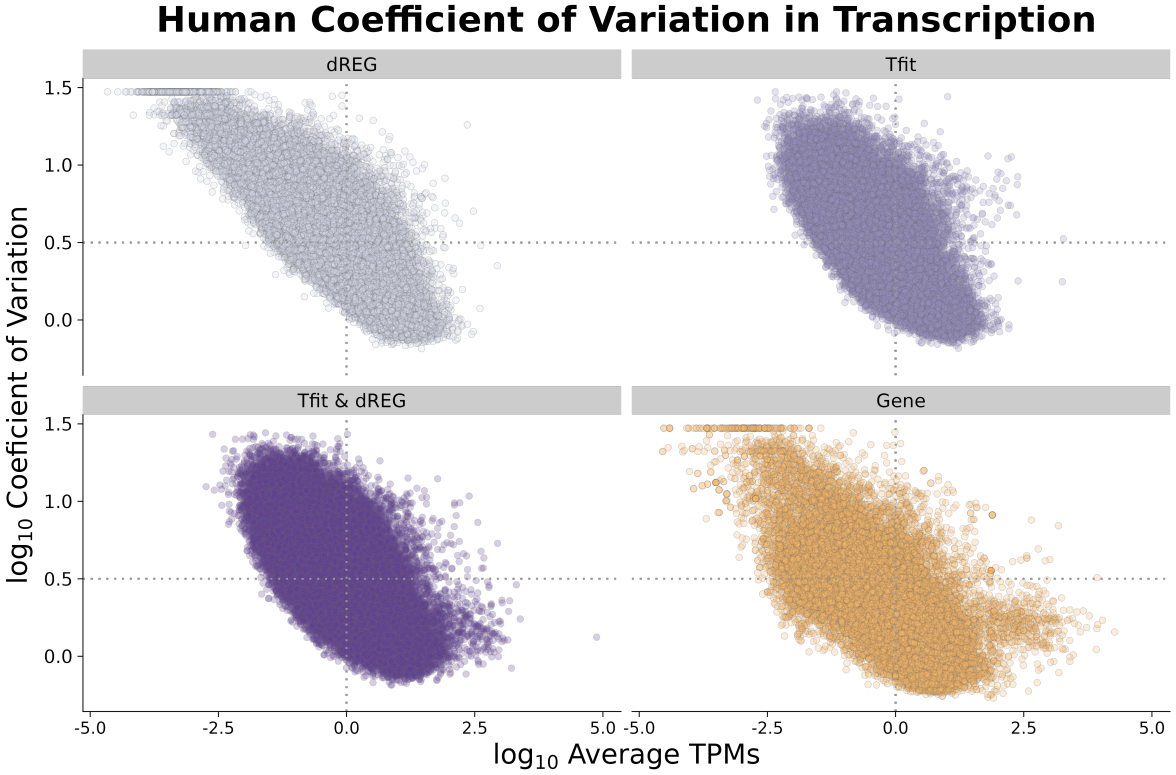


Figure 4.15: Coefficient of variation across human transcripts. All transcripts types (genes, dREG bidirectionals and Tfit bidirectionals) show a similar average normalized counts versus coefficient of variation profile. Transcripts with higher average normalized counts have lower coefficient of variation.

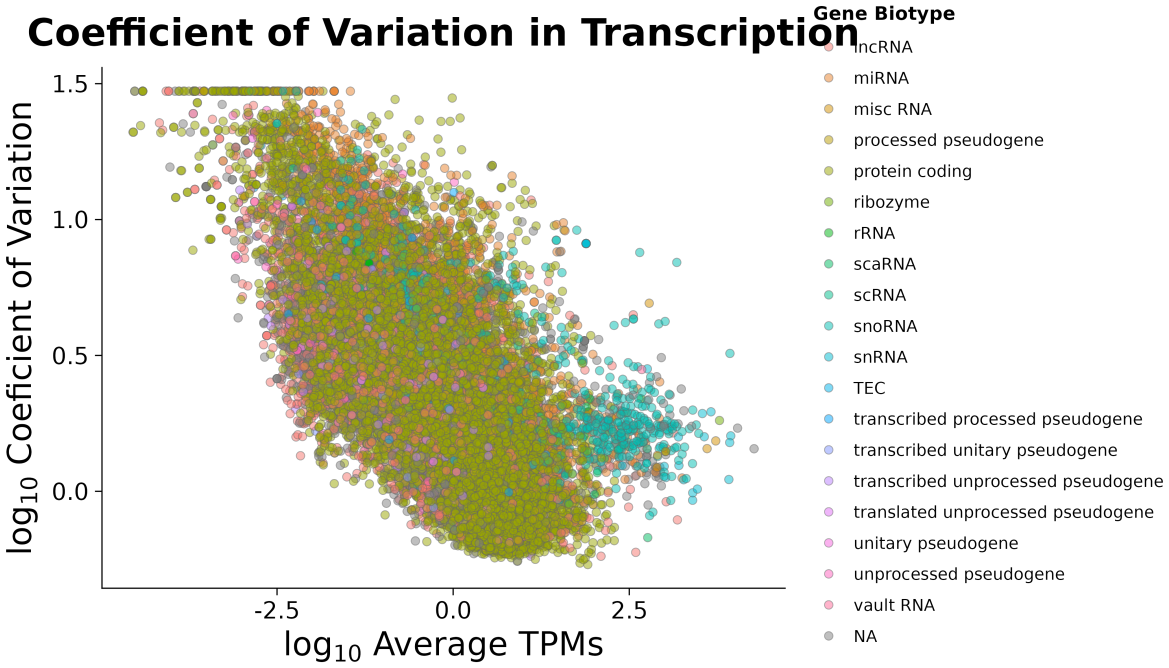


Figure 4.16: Coefficient of variation across human gene transcripts colored by biotype. All biotypes types show a similar average normalized counts versus coefficient of variation profile.

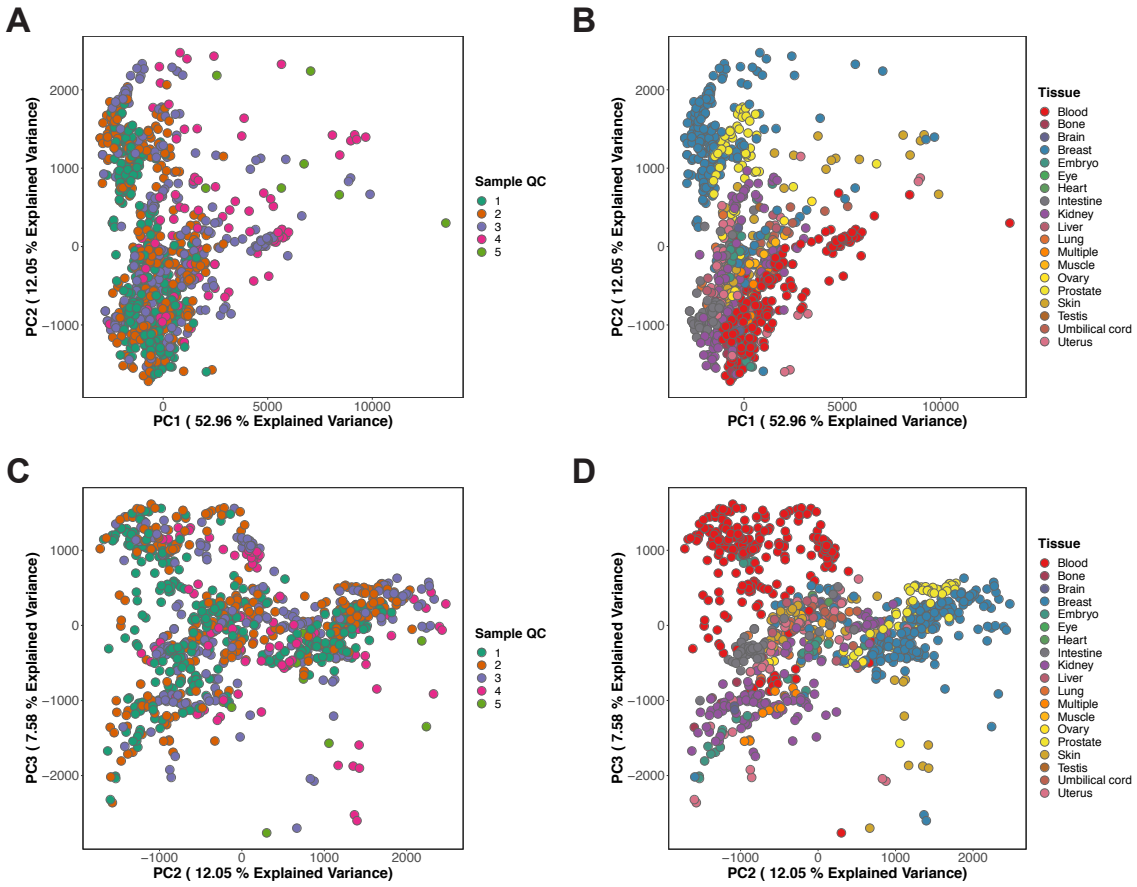


Figure 4.17: Principle component analysis across all human genes and bidirectionals show tissue type clusters. (A-B) All 880 sample point shown in PC 1 and PC 2 space colored by sample quality score (A) or tissue type (B). (C-D) All 880 sample point shown in PC 2 and PC 3 space colored by sample quality score (C) or tissue type (D).

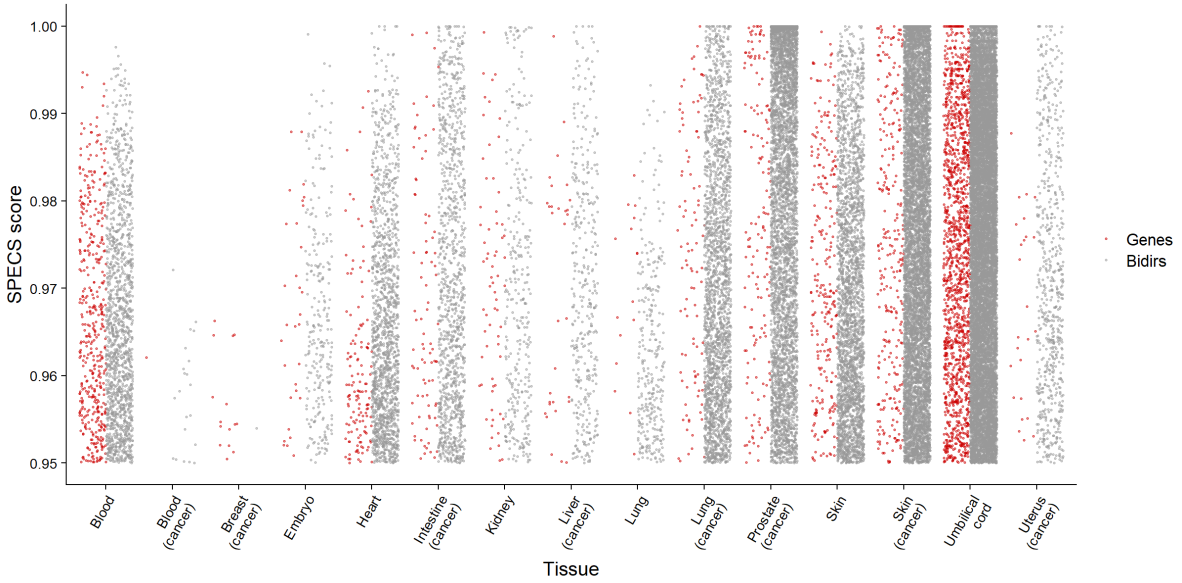


Figure 4.18: Highly specific genes and bidirectionals based on SPECS score. A SPECS score of greater than 0.95 is considered highly specific for a tissue. In general, the relative proportions for tissue-specific genes and bidirectionals correlate in a particular tissue.

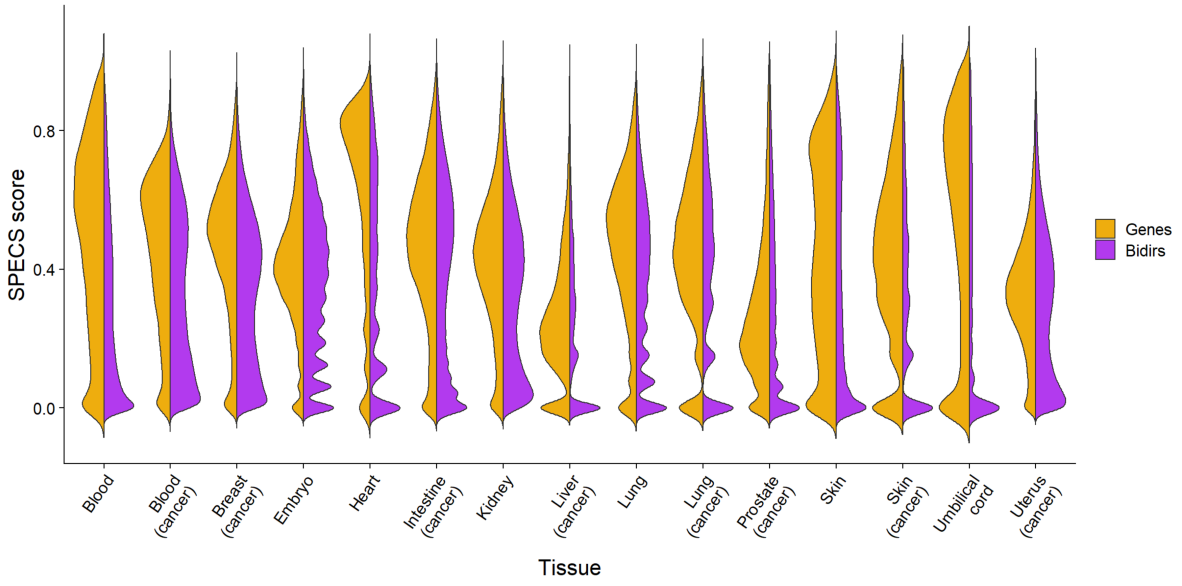


Figure 4.19: Distribution of SPECS scores across genes and bidirectionals for each tissue present in dbNascent. Tissues were filtered to have > 5 total samples of QC score < 4.

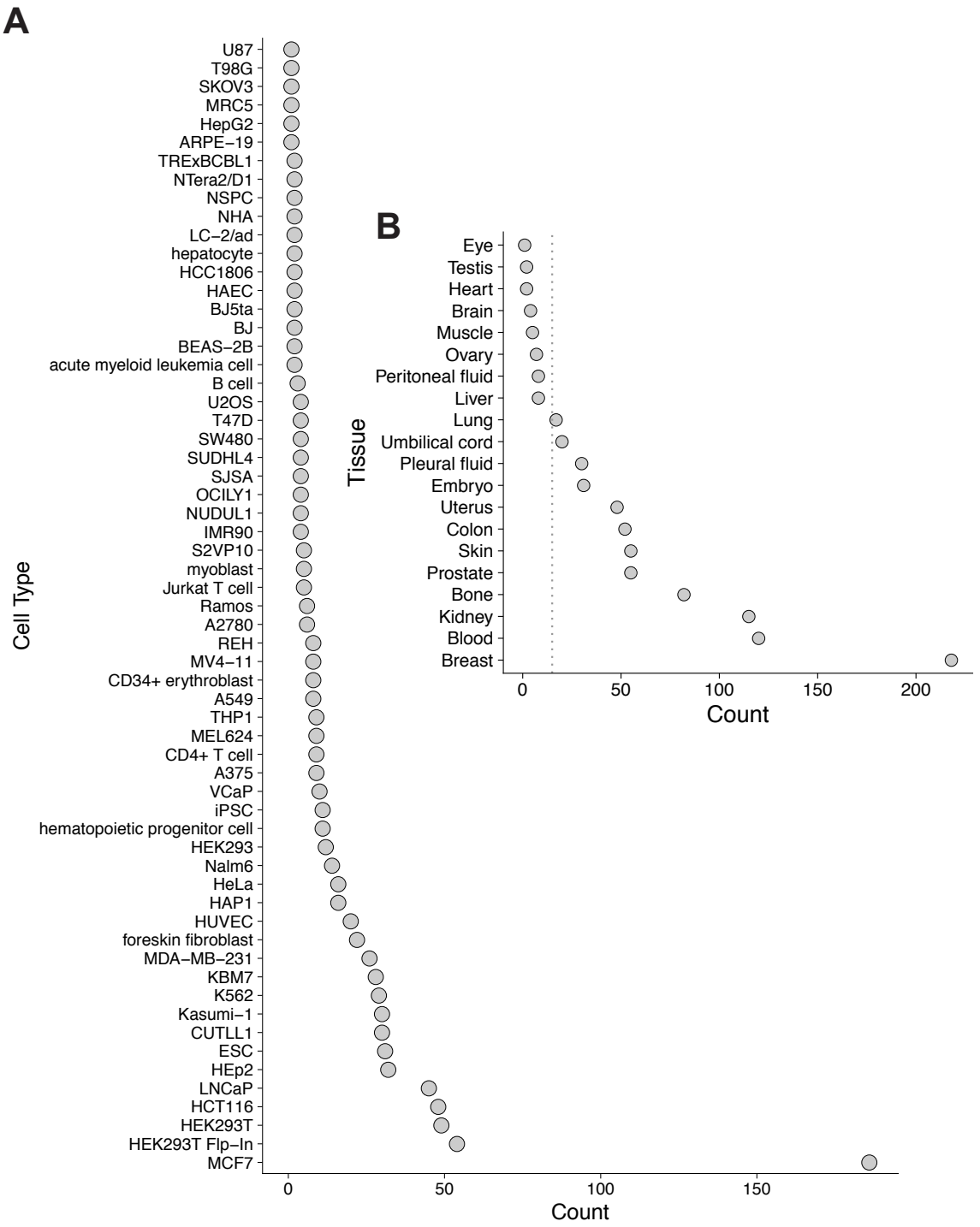


Figure 4.20: Number of samples for human by cell type and tissue types. (A) Counts of samples by cell type as labeled in GEO. Most samples in the analysis pipeline for high quality samples are MCF7. (B) Summary of of the sample samples labeled by tissue type agree with the cell type level summary and show that most samples are breast tissues. Further more, 10 tissues (Blood, Breast, Embryo, Intestine, Kidney, Lung, Prostate, Skin, Umbilical cord, Uterus) have greater than 15 samples (right of the dotted line).

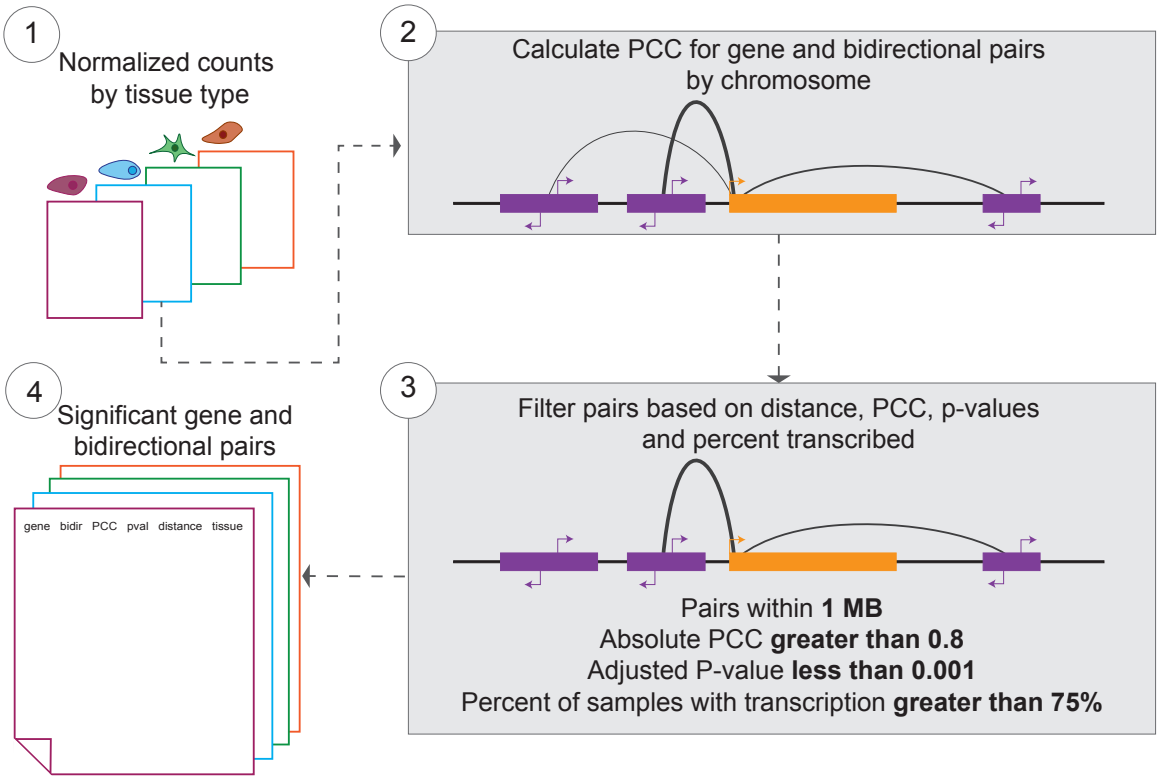


Figure 4.21: Tissue specific correlations. (1) Counts across transcripts are normalized by library size and length. The samples are separated by tissue type (Blood, Breast, Embryo, Intestine, Kidney, Lung, Prostate, Skin, Umbilical cord, Uterus). (2) Within each tissue, correlations between genes and bidirectional transcripts are computed. (3) The pairs are filtered based on distance (less than 1Mb), pearsons correlation coefficent (PCC) greater than 0.8 or less than -0.8, an adjusted p-value less than 0.001 and the pairs should be supported by majority of samples (greater than 75%).

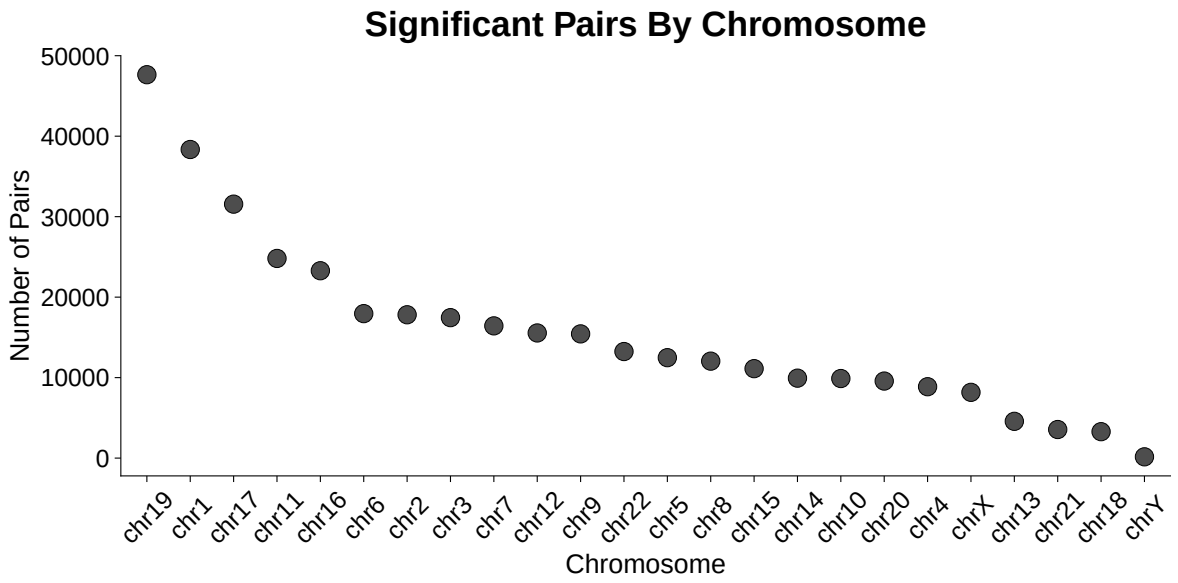


Figure 4.22: Number of gene and bidirectional pairs per chromosome in human samples. Chromosomes with more transcription have more pairs identified.

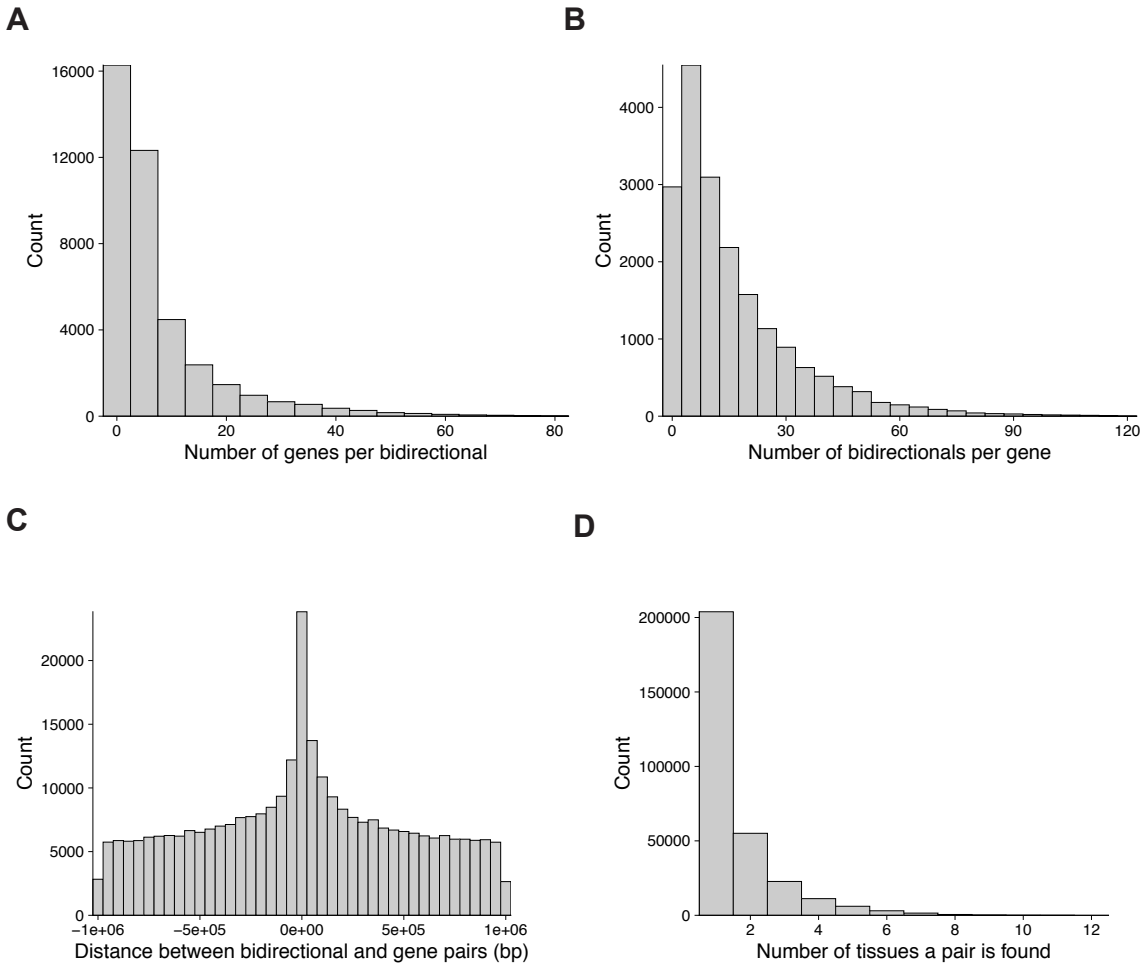


Figure 4.23: Summary of gene and bidirectional pairs identified by the tissue specific interactions. Distributions of the (A) number of genes a bidirectional transcript, (B) number of bidrectionals assigned to a genes, (C) distance between pairs and (D) the number of tissues a $gene \rightarrow bidirectiona$ pair is found.

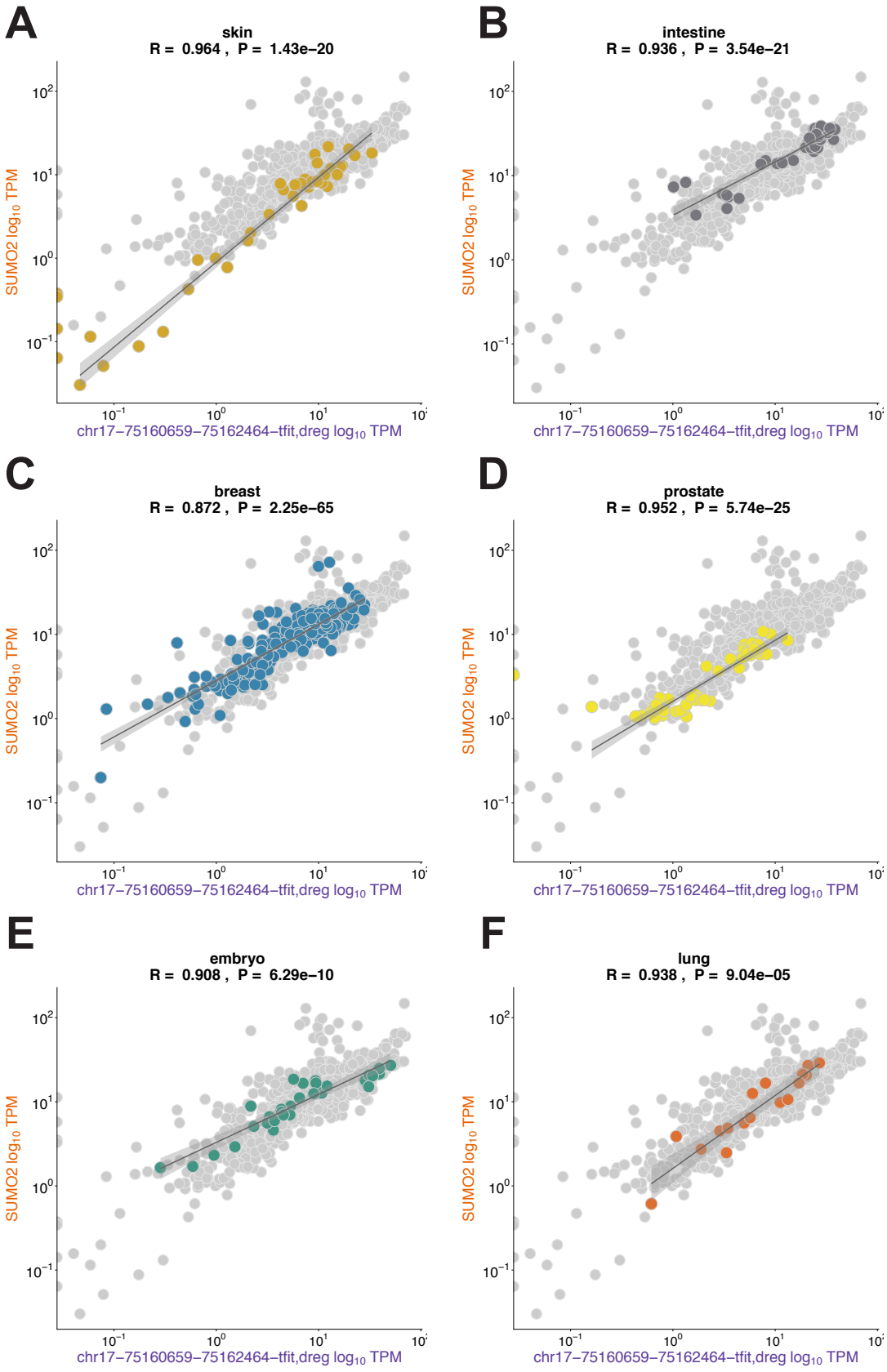

A
skin
R = 0.964 , P = 1.43e−20
B
intestine
R = 0.936 , P = 3.54e−21
C
breast
R = 0.872 , P = 2.25e−65
D
prostate
R = 0.952 , P = 5.74e−25
E
embryo
R = 0.908 , P = 6.29e−10
F
lung
R = 0.938 , P = 9.04e−05
SUMO2 log10 TPM
chr17−75160659−75162464−tfit,dreg log10 TPM

Figure 4.23: SUMO and bidirectional paired with analysis pipeline. This pair is also supported by eQTL found in skin tissue (A). Additionally, this pair was also found to be significant in five other tissues namely (B) intestine, (C) breast, (D) prostate, (E) embryo and (F) lung tissues.

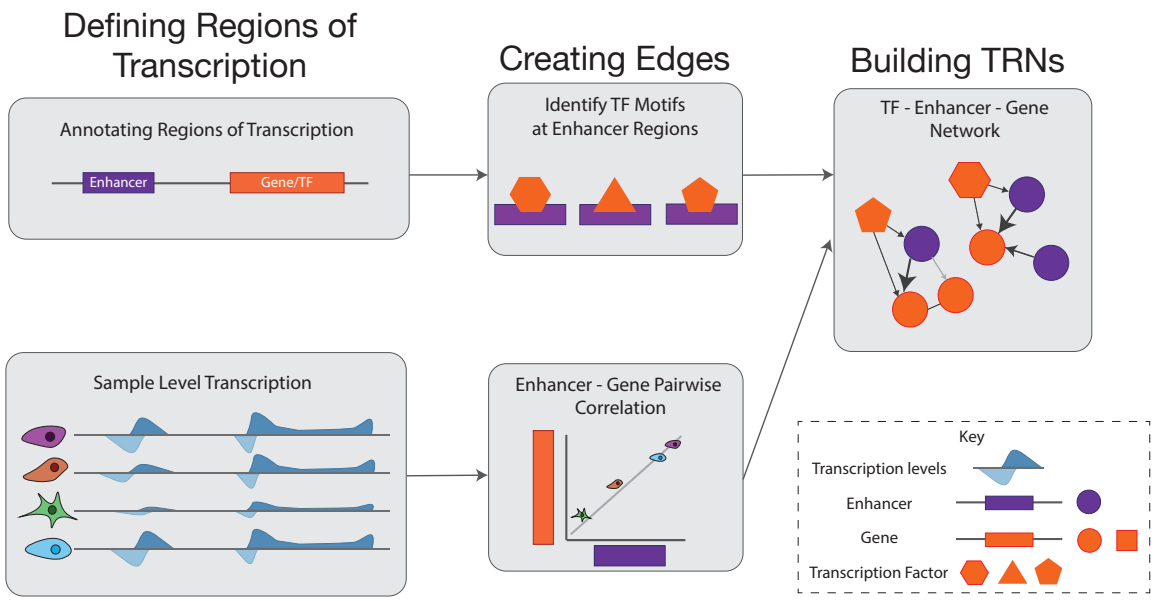


Figure 4.24: Building enhanced gene regulatory networks (GRNs) from nascent RNA data. The inputs include annotated regions of transcription (genes and bidirectionals). Bidirectional transcripts are assigned to genes based on correlation of their transcription. The transcription factor (TF) assignments are derived from TF motif instances in the regions on bidirectional transcription.

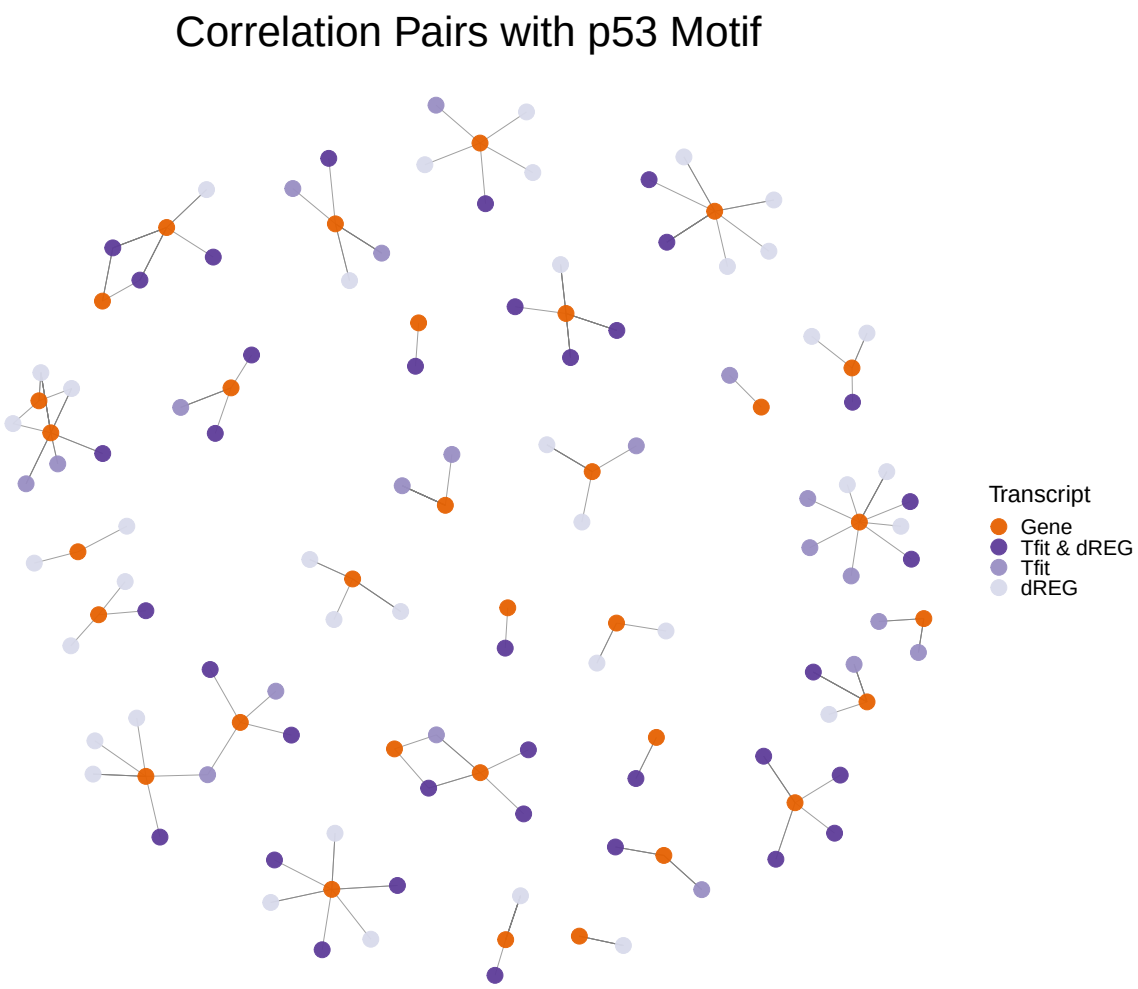


Figure 4.25: p53 responsive genes linked to bidirectional transcripts with a p53 motif. All 43 genes differentially transcribed across the tested cell linked (HCT116, MCF7 and SJSA) are paired to bidirectionals that have a p53 motif instance.

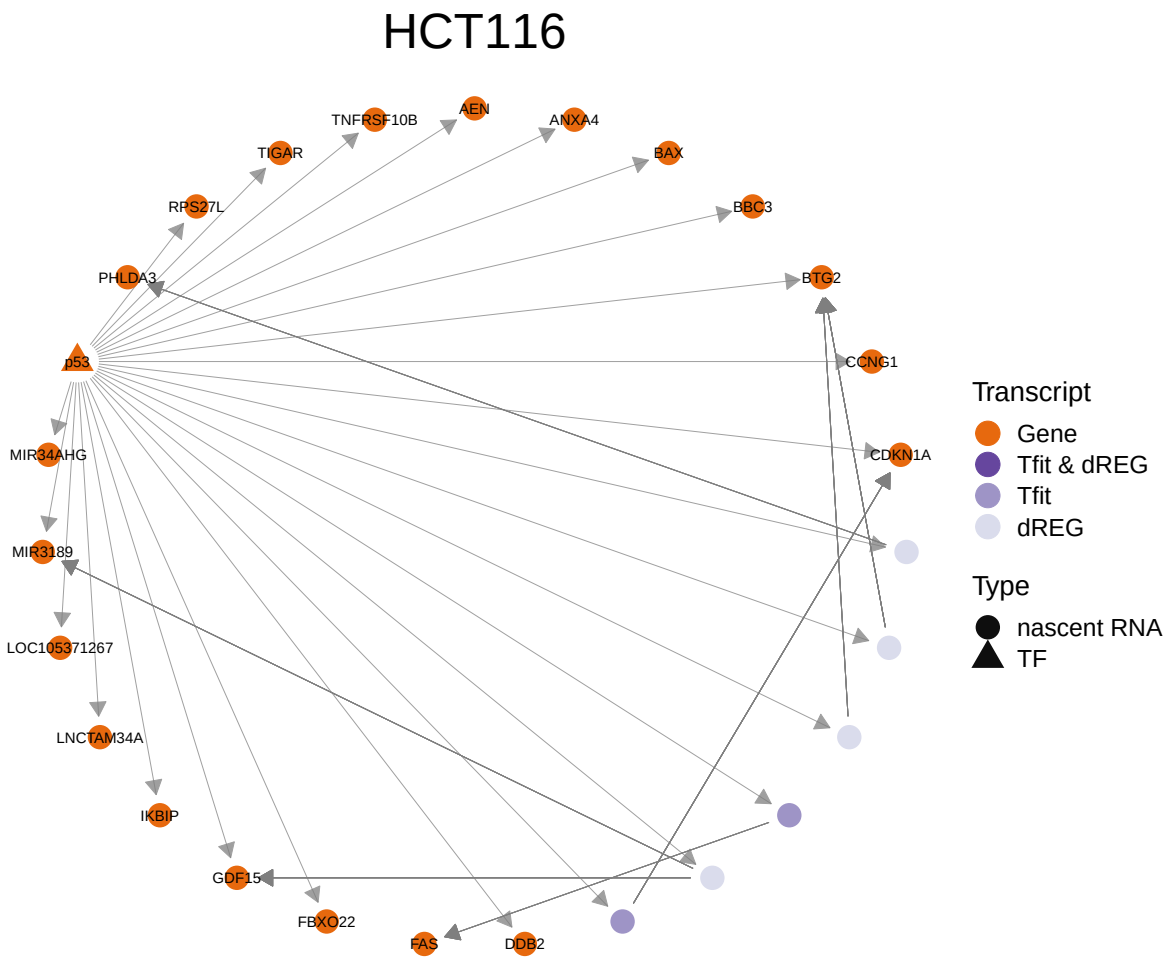


Figure 4.26: HCT116 p53 responsive network. p53 network in HCT116 cell lines. The triangle represents p53 motif and the arrow means the motif was found in the region (gene promoter or bidirectional transcript).

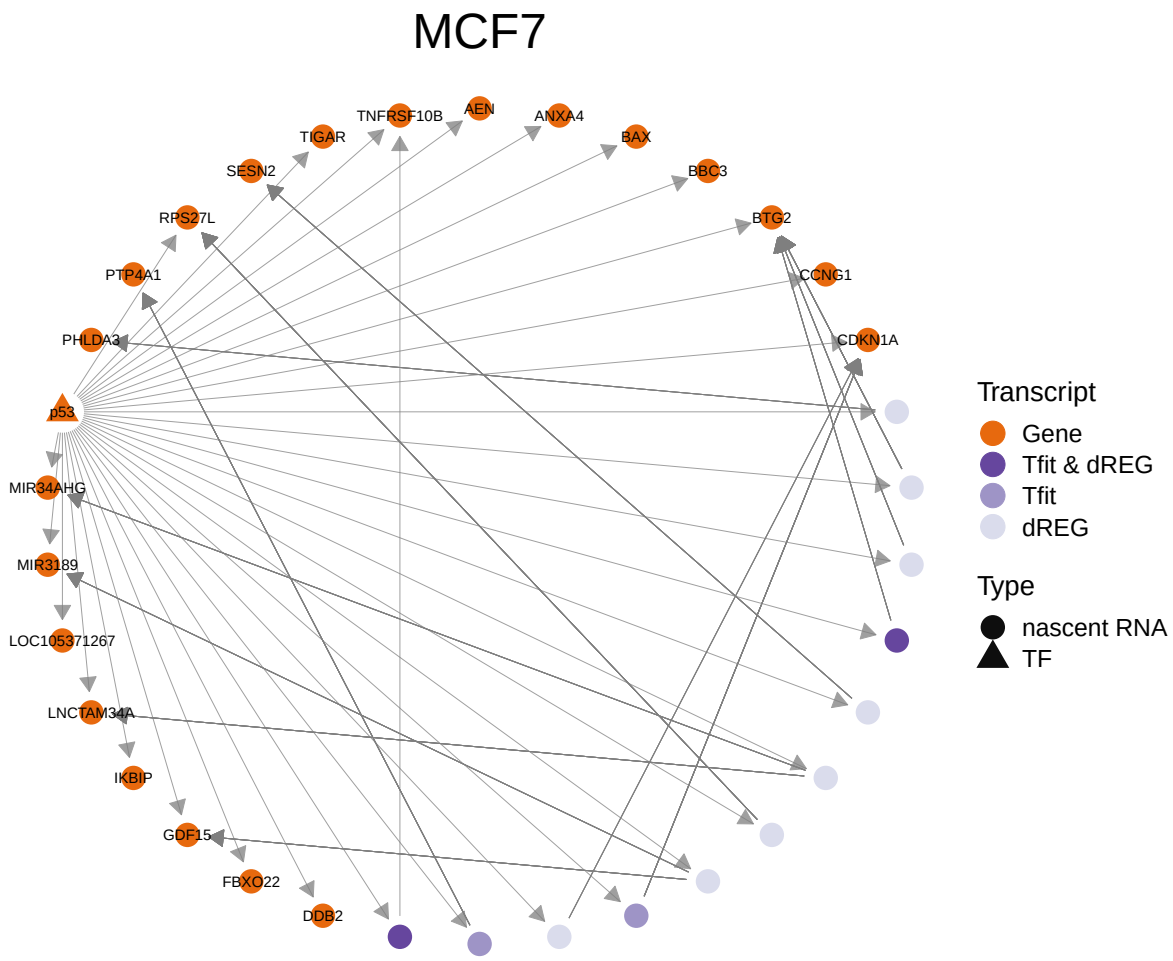


Figure 4.27: MCF7 p53 responsive network. p53 network in MCF7 cell lines. The triangle represents p53 motif and the arrow means the motif was found in the region (gene promoter or bidirectional transcript).

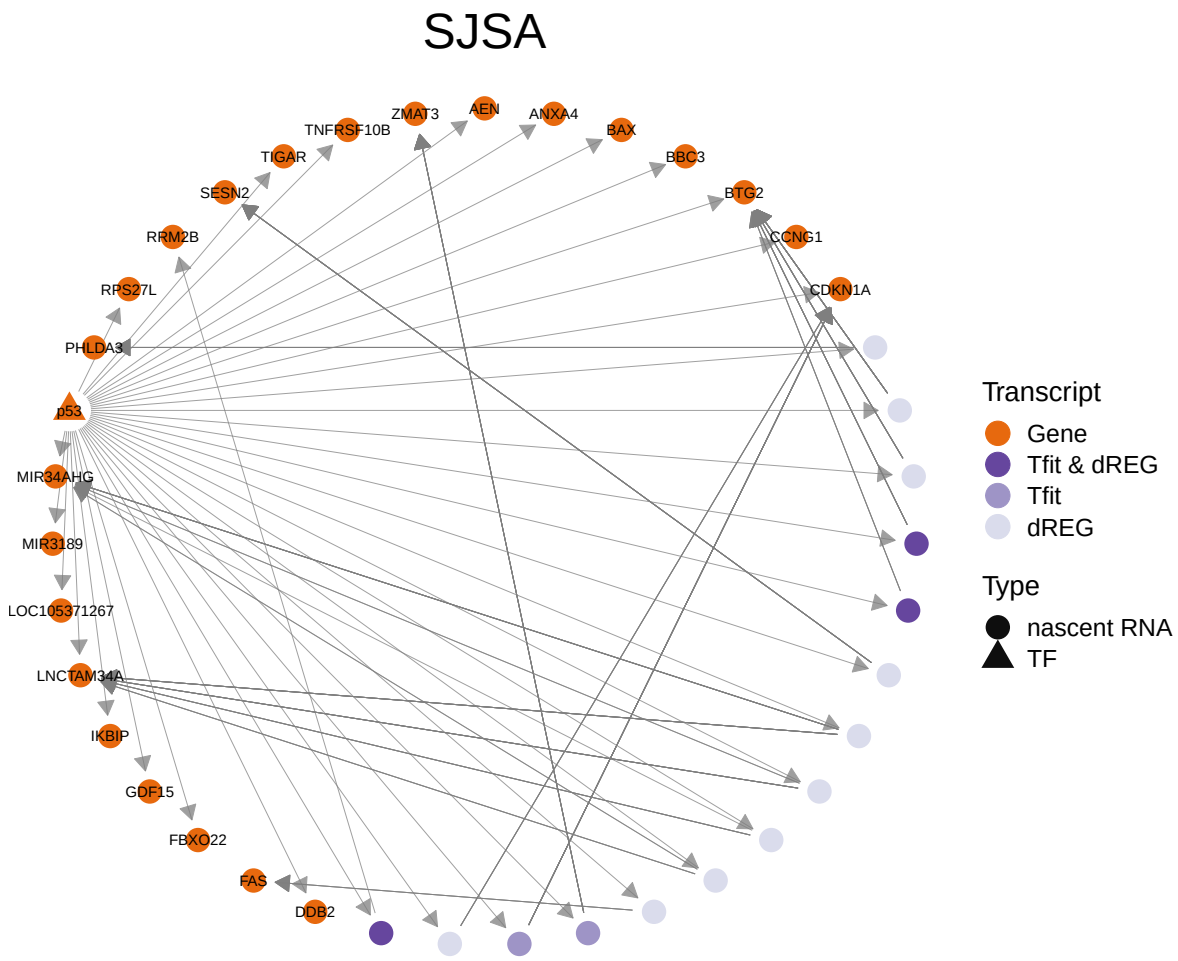


Figure 4.28: SJSA p53 responsive network. p53 network in SJSA cell lines. The triangle represents p53 motif and the arrow means the motif was found in the region (gene promoter or bidirectional transcript).

Summary of metadata curation

Metadata Collected	Description
Paper identifier	The paper where the samples and results were published presented in the format *Author Year Descriptive Term* (e.g. Allen2014global)
Sample identifier	The project id (SRP) and the sample id (SRR)
Replicate	Specify whether the sample was technical or biological replicate
Organisms	The scientific name of organism (formatted as H. Sapiens or D. Melonogaster)
Genetic background	The cell type or tissue used
Modification	Genetic modifications such as RNAi, shRNA
Treatment	The treatment if applicable
Treatment times	Time the sample was exposed to a treatment
Nascent protocol	GRO-seq, PRO-seq, NET-seq, GRO-cap, PRO-cap .etc
Library preparation	ligation, circularization, random primed and template switch reverse transcription
Spike in	If applicable, specify which spike in control was used (ERCC, Drosophila, Arabopsis or Luciferase/-Gal4/NeoR/GFP)

Table 4.1: Summary of the metadata manually collected from GEO and SRA.