AN ABSTRACT OF THE DISSERTATION OF

Mitra Ansariola for the degree of Doctor of Philosophy in Molecular and Cellular Biology presented on January 5, 2018.

Title: Computational Approaches for Understanding Combinatorial Gene Regulation

Abstract approved: _____

Molly Megraw

Gene regulation is a complex mechanism that controls the spatial and temporal expression of genes in a living cell. My dissertation studies focus on two problems. First, tissue-specific gene expression prediction from DNA sequence and chromatin state, and second, the accurate discovery of small over-represented regulatory circuits in gene regulatory networks.

Tissue-specific gene expression prediction is a complex process. In this dissertation, I introduce a generalizable machine learning method to study tissue-specific gene expression prediction in the model plant Arabidopsis thaliana. The results show that the transcription factor binding sites and chromatin states accurately characterize the tissue of expression.

In the context of gene regulatory networks, I introduce IndeCut as a very first method that evaluates the performance of network discovery algorithms. Genomic

networks represent a complex map of molecular interactions which are descriptive of the biological processes occurring in living cells. Identifying the small over-represented circuitry patterns in these networks helps generate hypotheses about the functional basis of such complex processes. IndeCut aids accurate detection of such over-represented patterns.

Computational Approaches for Understanding Combinatorial Gene Regulation

by

Mitra Ansariola

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented January 5, 2018
Commencement June 2018

Doctor of Philosophy dissertation of Mitra Ansariola presented on January 5, 2018.

APPROVED:

Major Professor, representing Molecular and Cellular Biology

Director of the Molecular and Cellular Biology Program

Dean of the Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Mitra Ansariola, Author

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Professor Molly Megraw, for her support during my Ph.D. study. Her advice and guidance have been invaluable both for pursuing my Ph.D. and a successful career. She indeed helped me beyond just developing technical skills, also improving my intellectual thinking and presentation skills which are necessary for a long-term success.

Next, I would like to thank Prof. David Koslicki for his excellent help during my Ph.D. I learned a great deal from his mathematical mentorship on a collaborative project. I also thank Prof. John Fowler as my teacher and committee member for his valuable guidance on the biological aspects of my projects. I thank my Ph.D. committee member Dr. Brett Tyler, previous teachers and mentors Dr. Jeff Chang, Dr. Barbara Taylor, Dr. Sergei Filichkin, and Dr. Maria Ivanchenko. I also would like to thank my friends and colleagues in the Molecular and Cellular Biology program at Oregon State University for our technical discussions, continuous brainstorming, and moments of humor. I am also grateful to my good friends and laboratory colleagues Valerie Fraser, Sarah Alto, Shawn ONeil, and Kai Tao.

Finally, I would like to thank my family, mother and brother in law for their support and inspiration. I thank my father who taught me to work hard for the things that I aspire to achieve. I am so grateful to have such loving and inspirational family members.

This thesis work is dedicated to my husband, Behrooz, who has been a constant source of support and encouragement during the challenges of graduate school

and life. This work is also dedicated to my parents, who have always loved me unconditionally, and to my little son, Behrad, who fills my heart with joy and motivation.

TABLE OF CONTENTS

Page

1 Introduction 1

1.1 Background and Introduction 1
1.1.1 It all starts from transcription. 1
1.1.2 The cascade of gene regulation: what underlies a complex biological process? 5

2 Chromatin state and transcription factor binding sites characterize tissue-specific gene expression in Arabidopsis root and leaf 9

2.1 Background 10
2.2 Results 13
2.2.1 Transcription start site usage and chromatin state in root and leaf tissue 13
2.2.2 TFBS locations and chromatin state accurately predict the tissue of expression 20
2.2.3 Region of erichments (ROEs) capture essential TFBS locations for tissue-specific prediction 22
2.2.4 Information compressed vs. information dispersed 25
2.2.5 Feature Analysis 31
2.2.6 "Hard-coded" and "soft-coded" promoters suggest variability in mechanism for tissue specification 36
2.3 Materials and Methods 40
2.3.1 Sample preparation and sequencing 40
2.3.2 Sequence processing and alignment 40
2.3.3 PWM redundancy detection 41
2.3.4 TSS peak detection and annotation 42
2.3.5 Region of Enrichments 43
2.3.6 Feature generation 43
2.3.7 Model feature scaling 44
2.3.8 Hard coded vs. soft coded promoters 45
2.3.9 Model training and testing 46
2.3.10 Differential expression analysis 47
2.4 Summary 48

TABLE OF CONTENTS (Continued)

Page

3 IndeCut evaluates performance of network motif discovery algorithms 50
3.1 Background . . . 51
3.2 Results . . . 55
3.2.1 IndeCut evaluates the performance of network motif discovery algorithms . . . 55
3.2.2 IndeCut indicates the number of samples required to achieve reproducible results . . . 64
3.2.3 Explaining performance differences found by IndeCut . . . 70
3.3 Methods . . . 75
3.3.1 Definitions . . . 75
3.3.2 How does IndeCut work? . . . 76
3.3.3 Mathematical details of IndeCut . . . 78
3.3.4 Compute Relationship Between Number of Samples and Cut norm Estimates . . . 91
3.3.5 Networks and graphs . . . 92
3.3.6 Description of examined network motif discovery algorithms 93
3.3.7 Description of edge switch graphs . . . 95
3.4 Summary . . . 96

4 Conclusion 103

Bibliography 108

LIST OF FIGURES

Figure		Page
2.1	Distribution of overlapping peaks between root and leaf tissue.	15
2.2	Distribution of open chromatin regions in promoter sequences of examined transcripts.	16
2.3	An example transcript with TSS-seq, DNase-seq, RNA-seq alignmnets along with the processed peaks.	18
2.4	A schematic view of promoter structure and available dataset for chromatin state, transcription initiation and TF bindings.	19
2.5	One to one Comparison between top 200 features in ROE and Tiled models (FWD strand).	27
2.6	One to one Comparison between top 200 features in ROE and Tiled models (REV strand).	28
2.7	An example PWM with weak ROE signal in forward strand.	30
2.8	Distribution of feature weights in ROE model.	31
2.9	Distribution of feature weights in Tiled model.	32
2.10	Top 100 weighted features in Tiled model.	34
2.11	Top 100 weighted features in ROE model.	35
2.12	Sum of feature products for TFBS and chromatin features across training and testing examples (ROE model).	46
2.13	Sum of feature products for TFBS and chromatin features across training and testing examples (Tiled model).	47
3.1	Small uneven graph sampling performance.	56
3.2	Graph sampling performance evaluation on small uneven graphs using IndeCut.	57
3.3	Graph sampling performance evaluation on small even graphs using IndeCut.	58

LIST OF FIGURES (Continued)

Figure		Page
3.4	Graph sampling performance evaluation on small hybrid graphs using IndeCut.	59
3.5	Human TF-miRNA-Gene network sampling performance.	62
3.6	Graph sampling performance evaluation on Ecoli network using IndeCut.	63
3.7	The relationship between cut norm estimates, number of samples and network motif outcome on Drosophila network.	66
3.8	Relationship between the number of samples vs. sampling performance for Human TFGene network.	68
3.9	The ESG graph and cluster-time diagrams for an example even graph.	69
3.10	The ESG graph and cluster-time diagrams for an example hybrid graph.	70
3.11	Constructing an ESG.	71
3.12	An example ESG for degree sequence $R = C = \{2, 1, 1, 2, 1, 1\}$.	73
3.13	An illustrative view of graph sampling strategy outcomes in terms of uniformity and independence.	79

LIST OF TABLES

Table		Page
2.1	TSS-seq and DNase-seq dataset information	17
2.2	Performance of all examines models.	23
2.3	Heavily weighted features analysis and their correlation with RNA-seq dataset	37
2.4	List of hardcode and softcoded promoters in ROE model	39
3.1	Runtime of IndeCut on all examined graphs.	101

LIST OF ALGORITHMS

Algorithm		Page
1	Computing PWM redundancy	42
2	Cut norm lower bound	102

Chapter 1: Introduction

1.1 Background and Introduction

Gene regulation is the process of how the cell controls which genes are turned on (expressed) or off (not-expressed) in its genome. However, gene regulation is not an isolated process independently affecting individual cells. Gene regulatory cascades are typically coordinated among cells to achieve spatial or temporal expression patterning during development, and carry out organismal function including response to the environment The questions which shape my dissertation research studies are as follows. Can we predict the tissue(s) in which a gene will express using its promoter structure? Can we learn the drivers of such gene expression patterns from the information that is encoded in DNA on a genome-wide scale? What are the best ways of studying the complex networks of gene regulatory element interactions that underlie genetic circuits? In these dissertation research problems, a key theme has been the examination of large currently available datasets in order to improve methods for mining these datasets.

1.1.1 It all starts from transcription.

In higher eukaryotes, gene regulation begins at the transcriptional level. Transcription is a process in which information from a gene is copied by an enzyme called

RNA polymerase II (Pol-II) into an intermediate product called a messenger RNA (mRNA). The mRNA may undergo a series of post-transcriptional steps that alter this RNA message, such as splicing or RNA editing, followed by translation of the message into an ultimate product a protein– which is the primary functional entity in the cell. In my dissertation studies, the focus is on transcription, the origin of many spatial and temporal regulatory processes in an organism. From this standpoint, it is critical to analyze the DNA regions in which transcriptional regulation primarily takes place (called promoter regions). Promoters are the genomic regions in the immediate or more distal (within 3 kilobases depends on species) vicinity of gene body. A genes promoter region contains DNA subsequences or elements which are the targets of regulatory interactions with proteins called transcription factors (TFs), as well as RNA polymerase enzymes (Pol-II in the case of protein-coding genes).

Many recent studies have attempted to analyze temporal/spatial gene regulation using computational approaches and available large-scale genomic datasets [24, 28, 32, 35, 56, 60, 66, 72]. Several factors are considered as keys to these analyses. One common factor is the combination of DNA sequence elements present in promoters, the other factor is chromatin state which controls transcription factor access to these DNA sequence elements. Some studies have investigated aspects of chromatin state including modifications to DNA (including methylation), or alterations to histones which make up the nucleosomes around which the DNA is wrapped [27, 57, 60]. A recent study in human cell lines [50] proposed using DNase I hypersensitive sites to predict cell-type specific and housekeeping gene expres-

sion. DNase I is an enzyme that can digest DNA regions that are not occupied by a nucleosome (i.e. open chromatin regions). This study supported a key role for TF binding element accessibility (chromatin state surrounding TF elements) in determining cell-type specific expression. However, relying only on DNase signals without also considering the effect of direct TF-DNA binding doesn't appear to provide a comprehensive understanding of the drivers of cell-type specificity arising from promoter structure. This is because, as mentioned in [27], the DNase I signal (like other chromatin features) may be affected by transcription rather than primarily causal of transcription. Another study used the number of in vivo TF binding sites in gene promoters to predict expression breadth (expression level of genes across different cell-lines)[28]. They could distinguish the subset of TFs involved in the regulation of housekeeping genes as compared to highly specialized genes. This group of methods, which rely on high throughput chromatin intercommunication mechanisms to identify the TF binding sites, faces an important challenge. Genome-wide techniques such as chromatin immunoprecipitation followed by microarrays or sequencing (ChIP-chip and ChIP-seq) have also been used as essential tools in identifying precise TFBS locations. These datasets are very difficult to acquire across multiple cell or tissue types for many TFs. This is because ChIP experiments require high-quality antibodies or tagged protein, which is not always available for the TF(s) of interest. Also, TFs must be assayed individually, which requires many independent ChIP experiments in order to identify combinatorial patterns of TF binding. The methods using only ChIP datasets for TFBS information therefore focus only on well-characterized TFs, and do not

include many other TFs in the modeling process.

The limited number of available high resolution and large-scale datasets in many species is another important challenge in genome-wide tissue-specific gene expression prediction studies. Most of the relevant computational analyses have been performed in mammals, which have a rich repository of large-scale datasets, particularly for human cell lines; the ENCODE [17] and FANTOM5 [39] projects provide a comprehensive gene expression profile for human cell lines and tissue types. Other species, including plant species, lack such datasets. Under these circumstances, the question of what specifies a complete set of fundamentally causal factors for tissue-specific gene expression has not been satisfactorily answered despite 30 years of investigation [27]. In this dissertation study, I used a generalizable computational approach to predict the tissue in which a gene will express using promoter sequence content and chromatin state in the model plant Arabidopsis thaliana. My lab generated a unique dataset providing a high-resolution genome-wide snapshot of plant's expression profile in root and leaf tissues. These datasets capture transcription start locations (TSS-seq), chromatin state (DNase-seq) [18], and gene expression (RNA-Seq) in each tissue. I used a set of 413 TF binding domain representations (PWMs) obtained from databases containing only domains characterized by experimental techniques (including protein-binding microarrays (PBMs)[68] and SELEX). Each family of TFs has at least one member in this set. I investigated whether TF binding sites or chromatin state alone can predict the tissue of expression. I also investigated whether additional predictive value can be obtained by explicitly considering interactions between these two factors (TFs

and chromatin state). Results show that TF binding sites and chromatin state are both strong predictors of the tissue(s) in which a gene will express, and that TF binding site presence and binding site location are predominant explainers of outcome. This approach can easily be applied to other organisms in which the relevant datasets become available. This modeling process is therefore an important step toward understanding the drivers of tissue specificity in other species.

1.1.2 The cascade of gene regulation: what underlies a complex biological process?

Many gene regulatory events in a cell is coordinated with similar events in other cells to regulate a biological behavior in a specific time or condition. Regulatory genes such as TFs and microRNAs are important components of larger interaction networks. In the context of tissue-specific gene expression, this can be envisioned as the cascade of TF-DNA binding events (TF proteins bind to the promoters of other TFs, miRNAs, and non-TF protein coding genes). In regulatory pathways, these interactions control quantity and time of expression as necessary to perform a physiological or biological process such as response to an environmental stimulus. Understanding such interactions and their underlying mechanisms is one of the large open challenges in biology. These different types of interactions can be studied in the context of networks known collectively as "Gene Regulatory Networks". Improving computational techniques for Gene Regulatory Network studies is a second major focus of my dissertation.

Genomic networks represent a complex map of molecular interactions which are descriptive of the biological processes occurring in living cells [23, 46]). Due to the size and complexity of these networks, it is often difficult to infer the physiological function of individual interactions or collections of interactions without additional detailed information about network structure. Because this type of experimentally supported prior information is usually sparse or unavailable, a systematic approach for identifying key sub-components and their functions within a biological system is essential for analysis. From this perspective, it has been shown that the functional essence of a complex genetic network within a cell can often be distilled by thinking of the network as a circuit board composed of small, understandable components that work together to carry out higher-order processes [2, 4, 40, 42, 46, 53, 58, 67, 71]). Network motif discovery is a well-established statistical strategy for performing network analysis from this viewpoint. This strategy compares the frequency of observation of a sub-network within the larger original network to its frequencies in many randomized background networks in order to identify network motifs, which are defined as those sub-networks observed at a significantly higher frequency in the original network. In other words, a network motif is an over-represented sub-structure within a larger network. Network motif discovery tools aid in generating specific testable hypotheses about the behavior and function of a genetic sub-circuit. Although such hypotheses are valuable starting points for understanding the underlying mechanisms of a biological process through analysis of genomic networks, the laboratory validation of a predicted network motif is generally a costly and time-consuming endeavor. To characterize statistical

significance of a given genomic network (here called the original network), network motif discovery algorithms generate random graphs (here called background network generation) while striving to satisfy two conditions. 1) Background networks should preserve a sensible set of biological assumptions constrained by the original network. Computationally, the core component of background network generation is the sampling of a number of networks (for example, 1000 networks) from the set of all possible networks (e.g. 1 million networks) having in-degree and out-degree sequences identical to those of the original biological network. Despite a rich mathematical literature on the subject [3, 8, 15, 16, 21, 30, 34, 44, 45]), practical solutions to this problem remain elusive. This creates a concern for the accurate performance of network motif discovery algorithms on real biological networks, which often contain large source hubs (master regulators) and/or target hubs (heavily regulated nodes) [61, 70]).

To date, no mathematically sound yet computationally practical method is available in order to determine whether a graph sampling method samples uniformly and independently for a large or even moderately-sized network of interest. However, relatively recent advances in the enumerative combinatorics literature [1, 5]) have opened an avenue for the development of solutions to this long-standing problem. Here, we present IndeCut, which assesses the degree of sampling uniformity and independence for network motif discovery algorithms. We also show how IndeCut can provide a way to understand the cause of performance variations among different graph sampling approaches.

A major goal of my dissertation studies was to identify important open ques-

tions in the modeling of gene regulation, and to advance computational approaches in order to address these questions. I am pleased that I received an opportunity to work as a team member in a highly-motivated laboratory that supported my interest in these studies, and that I could achieve great progress toward these goals. My hope is that the findings of my dissertation research will aid the genomics and computational biology community in studying these fundamental scientific questions.

Chapter 2: Chromatin state and transcription factor binding sites characterize tissue-specific gene expression in Arabidopsis root and leaf

Mitra Ansariola, Molly Megraw

In prep

2.1 Background

In higher eukaryotes, gene regulation begins at the transcriptional level. Transcription is a process in which information from a gene is copied by an enzyme called RNA polymerase II (Pol-II) into an intermediate product called a messenger RNA (mRNA). The mRNA may undergo a series of post-transcriptional steps that alter this RNA message, such as splicing or RNA editing, followed by translation of the message into an ultimate product a protein which is the primary functional entity in the cell. In my dissertation studies, the focus is on transcription, the origin of many spatial and temporal regulatory processes in an organism. From this standpoint, it is critical to analyze the DNA regions in which transcriptional regulation primarily takes place (called promoter regions). Promoters are the genomic regions in the immediate or more distal vicinity (within 3 kilobases, depends on species) of a gene body. A genes promoter region contains DNA subsequences or elements which are the targets of regulatory interactions with proteins called transcription factors (TFs), as well as with RNA polymerase enzymes (Pol-II in the case of protein-coding genes).

A longstanding open question regarding tissue-specific gene expression studies is whether (and if so, how) one can predict a genes tissue(s) of expression using promoter structure. This question has not been satisfactorily answered [27] due to the complexity of potentially predictive biological information components including DNA sequence motifs, their binding factors, and associated chromatin properties. Many recent studies have attempted to analyze temporal/spatial

gene regulation using computational approaches and available large-scale genomic datasets [24, 28, 32, 35, 56, 60, 62, 66, 72]. Several factors are considered as keys to these analyses. One common factor is the combination of DNA sequence elements present in promoters; another commonly discussed factor is chromatin state, which controls transcription factor access to DNA binding sequence elements. Some studies have investigated aspects of chromatin state that include modifications to DNA (e.g. methylation), or alterations to histones which make up the nucleosomes around which the DNA is wrapped [60, 66]. However, there are very few studies which build predictive models to examine tissue-specific gene expression using DNA sequence and chromatin state (open vs. closed states). A recent study in human cell lines, [50], proposed the use of DNase I hypersensitive sites to predict cell-type-specific and housekeeping gene expression. DNase I is an enzyme that can digest DNA regions that are not occupied by a nucleosome (i.e. open chromatin regions). This study supported a key role for TF binding element accessibility (chromatin state surrounding TF elements) in determining cell-type specific expression. However, relying only on maximum TF binding affinity in open chromatin regions without considering the location of TF binding sites doesnt appear to provide a comprehensive understanding of the drivers of cell-type specificity arising from promoter structure. The second study ([62]), attempts to distinguish between highly-expressed genes versus lowly expressed genes in each human tissue type using the number of TF binding sites observed within each genes promoter region. The study obtains a successful classifier (AUC ¿ 0.8) for about half of the tissue types. In both studies ([62, 50]), TF binding locations are not

considered, and it is not clear why certain cell or tissue types are associated with very low model performance. Other studies in this area are highly dependent on the organism of study, and do not provide a high performing generalizable model for the prediction of tissue-specific gene expression [56, 28].

In past studies for predicting gene tissue specificity from promoter structure (specifically, from TF binding sites and chromatin state), two main computational approaches have been used: (i) Machine learning strategies, and (ii) combinatorial data mining methods. The input to machine learning models is a collection of features defining the promoter structure for each example (gene), and output is the probability of expression of each example (gene) within each tissue category. Combinatorial data mining has been used for the purpose of extracting the most informative promoter architecture features for use in subsequently applied machine-learning strategies. We use both machine learning and combinatorial data mining to address the central question of this study.

Despite several relevant computational studies, a high performing model has not yet been achieved [27]. Two potential reasons for this are as follows. First, the metrics which have been used for tissue specificity analysis may be problematic. Some studies, as explained earlier, use only relative measurements such as highest or lowest expression within a cell or tissue type, without considering the average expression level of genes across all cells or tissue types. The second potential reason for not achieving a high performing model is that the location of TF binding sites is not considered in the modeling process.

In this study, we propose a generalizable computational approach to predict the

tissue in which a gene will express using promoter sequence content and chromatin state in the case of highly expressed genes in root and leaf tissues in the model plant Arabidopsis thaliana. We investigate whether the TF binding site locations along with chromatin openness can predict the gene expression pattern in two tissues. We use a unique dataset that provides a high-resolution genome-wide snapshot of plants expression profile in root and leaf tissues. These datasets capture transcription start locations (TSS-seq), chromatin state (DNase-seq), and gene expression (RNA-Seq) in each tissue. We use a set of 413 known TF binding domain representations (PWMs) obtained from databases containing only domains characterized by experimental techniques (including protein-binding microarrays (PBMs) and SELEX). We propose two types of models to investigate the relationship between the TF binding sites within a promoter region, their associated chromatin state, and the spatial expression pattern of genes. This modeling process is therefore an important step toward understanding the drivers of tissue specificity in this and other species.

2.2 Results

2.2.1 Transcription start site usage and chromatin state in root and leaf tissue

Genome-wide identification of transcription start locations is an important first step in studying gene regulation patterns in the promoter region. On a genome

scale, transcription factors have shown different binding location preferences under various temporal/spatial conditions [63, 14]. The identification of such binding locations requires accurate knowledge about transcription start site location [63, 49, 37, 14]. Cap analysis of gene expression (CAGE) is a sequencing technology designed specifically to detect the 5 ends of intact mRNA molecules. After mapping sequenced reads (TSS tags) back to a reference genome, CAGE data highlights the specific TSS positions and their usage at nearly single nucleotide resolution. In this study, we use a similar trapping protocol (nanoCAGE-XL [19]) to generate such high-throughput sequence reads in wild-type A.thaliana root and leaf tissue.

Chromatin state is an important factor in regulating the expression of genes under various spatial/temporal conditions [7, 13, 50]. To study the impact of chromatin state on tissue-specific gene regulation, we generated a map of accessible DNA regions using the DNase I SIM (for simplified in-nucleus method) protocol which captures the genomic regions that are not occupied by nucleosomes or other cis-regulatory elements [18].

A comprehensive computational pipeline was generated to process these datasets (see Methods Section 2.3.1). First, TSS-seq reads were mapped to the TAIR10 genome and TSS tag clusters (TSS-Peaks) were identified using the JAMM peak finder [29] in each tissue. The JAMM peak finder was originally developed and tuned for analyzing ChIP-seq data and thus it needed to be re-tuned for detecting TSS peaks. The JAMM peak finders parameters were carefully tuned as described in Methods section 2.3.4. In the next step, TSS peaks were assigned to the nearest transcript in the genome using a TSS peak annotator program (Methods section

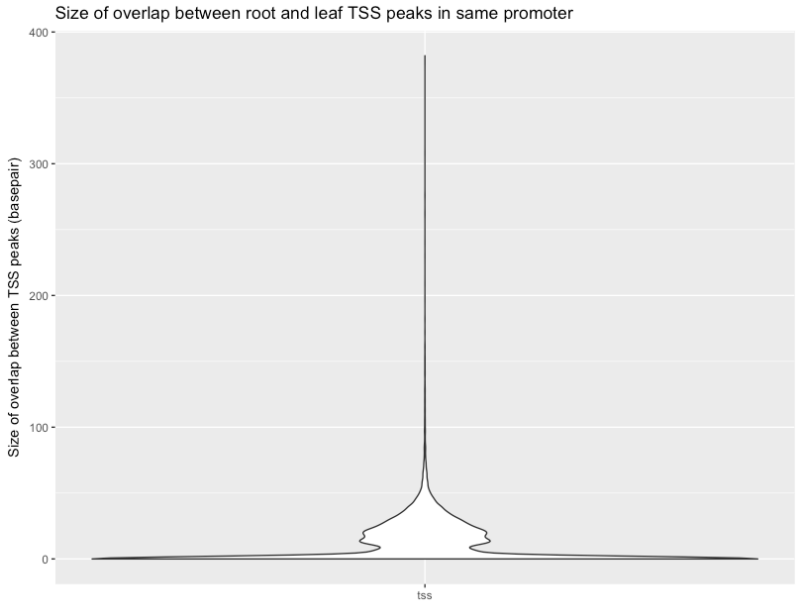

Figure 2.1: Distribution of overlapping peaks between root and leaf tissue.

2.2.1). TSS peaks that fell within 250 base pairs upstream of the translation start site of transcripts and contained more than 50 mapped reads were selected. Table 2.1 shows the number of TSS peaks and transcripts associated with them in each dataset. In summary, 41,778 total TSS peaks were detected in both root and leaf tissues, of which 19, 583 transcripts were assigned to one or more TSS peaks in root and leaf tissues. 12,129 of transcripts were associated with both root and leaf TSS peaks (62%). For 5,230 transcripts (44%), root and leaf TSS peaks were located in different non-overlapping regions according to the genomic locations. Figure 2.1 shows the size of overlapping regions between root and leaf TSS peaks across all promoters containing both TSS peak types (root and leaf).

Open chromatin genomic coordinates were compiled from [18]. Figure 2.2 shows

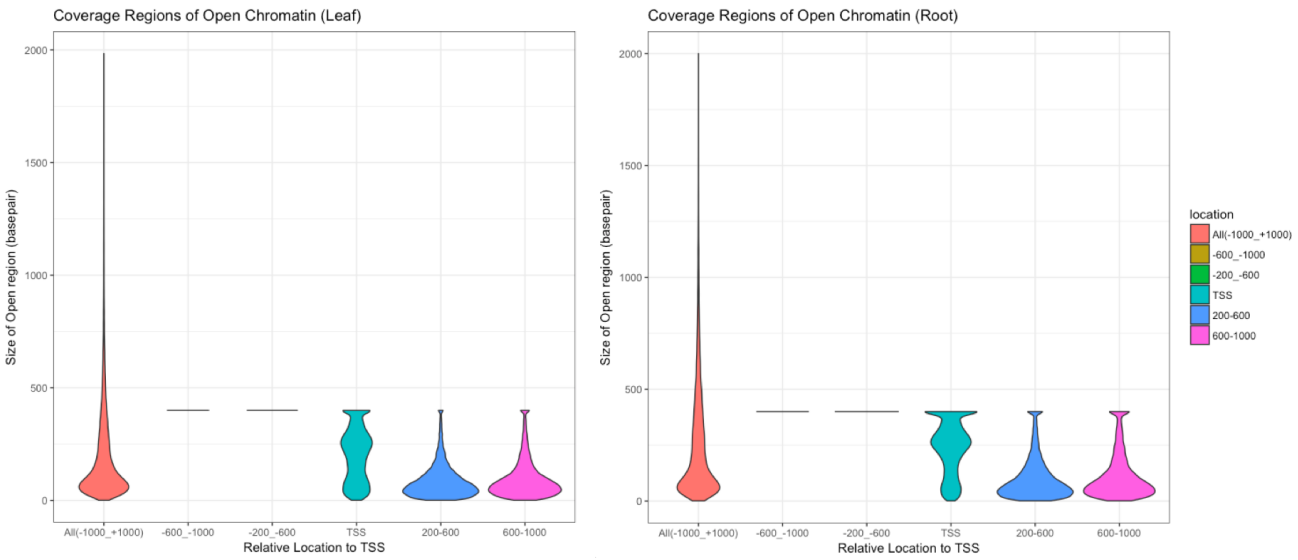

Figure 2.2: Distribution of open chromatin regions in promoter sequences of examined transcripts.

the distribution of open chromatin locations within 6-kb TSS-centered regions across the genome. This figure shows that chromatin tends to be mostly open within 1-kb upstream of the TSS. This pattern of openness is comparable between root and leaf tissues. Table 2.1 1 shows that about 98% of the promoters contain open chromatin regions, whereas 2% to 3% of promoters that express in each tissue are in an entirely closed chromatin state.

Table 2.1: TSS-seq and DNase-seq dataset information

Dataset	**ROOT**	**LEAF**	**TOTAL**
No. of TSS Peaks	19,158	22,620	41,778
No. of Transcripts Having Associated TSS Peak	14,768	16,930	19,583
No. of Promoters Having OC Region	40,682 (97% of total peaks)	40,742 (98% of total peaks)	-
No. of Tissue-specific Transcripts (fold-change >4 and qval <0.05)	1,096	981	1,177

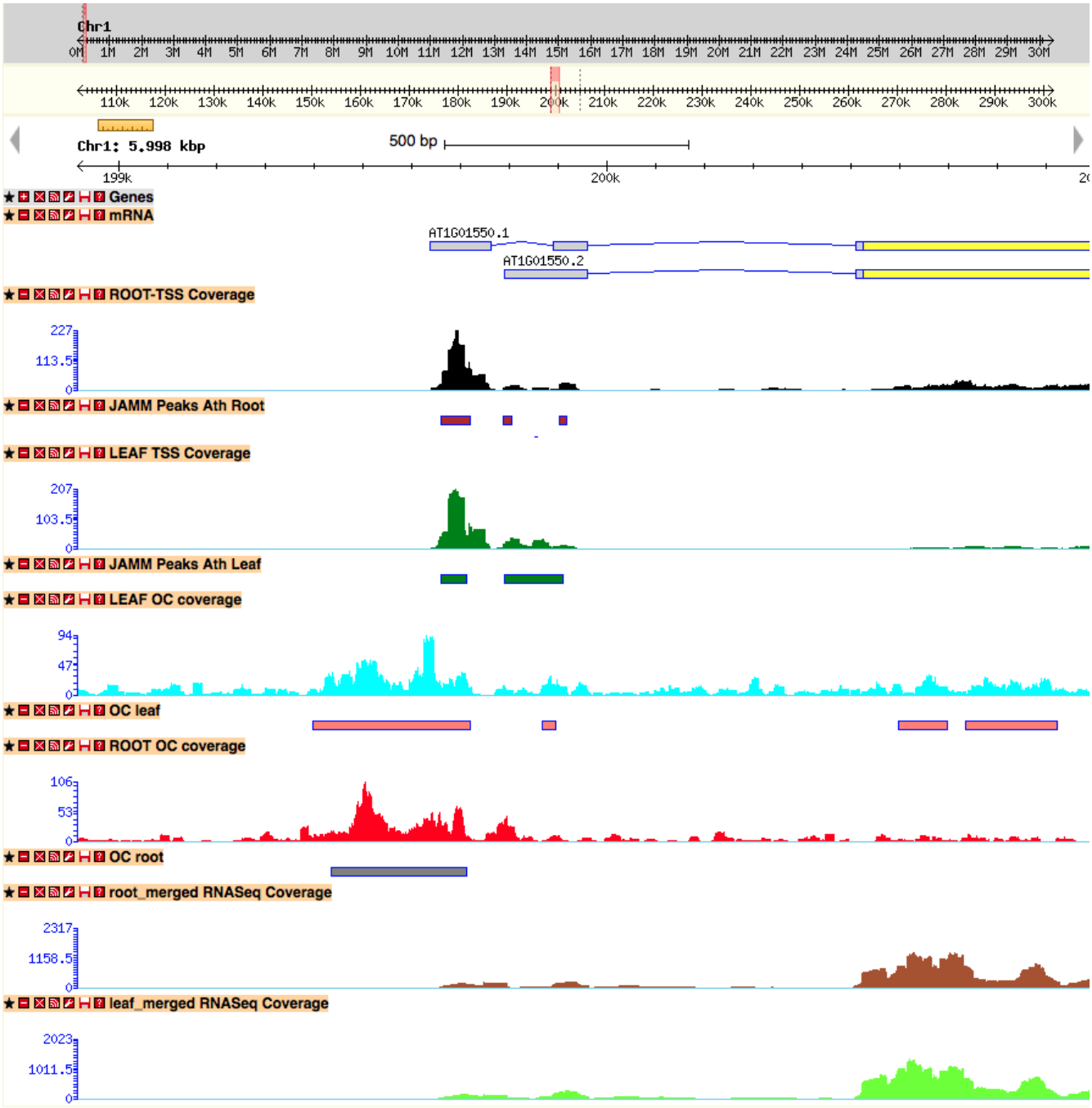

Figure 2.3: An example transcript with TSS-seq, DNase-seq, RNA-seq alignmnets along with the processed peaks.
This snapshot is taken from GBrowse web software which shows the processed and un-processed examples in all datasets used in this study.

We used expression profiles of A.thaliana genes in root and leaf tissues in order to identify tissue-specific promoters. RNA-Seq libraries were prepared from 7-day old root and leaf tissues with three replicates. Transcript abundance was computed using Kallisto [52]; normalized mean expression was computed using three replicates for each tissue. The differential expression analysis was performed using Sleuth [52]; differentially expressed transcripts between the two tissues were identified (expression in root as compared to expression in leaf). Transcripts with a fold change of 4 or larger were considered as tissue-specific. From 27,918 transcripts obtained from RNA-Seq data, only 1,177 transcripts were tissue-specific according to this definition (fold change greater than 4 and q-value <= 0.05, see Table 2.1).

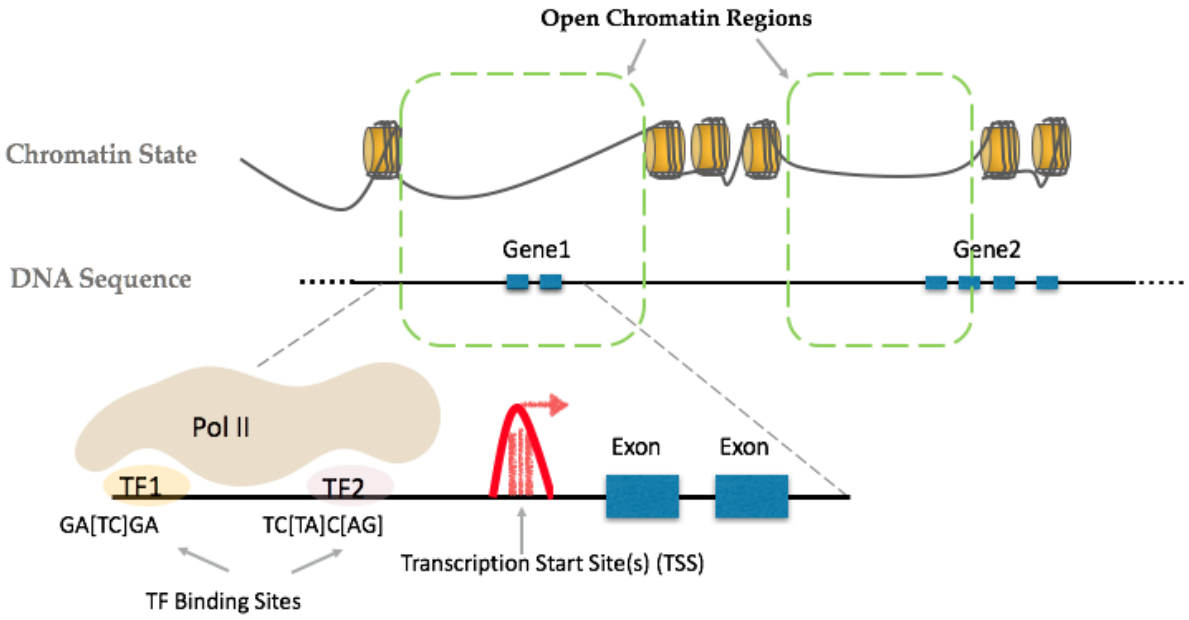

Figure 2.4: A schematic view of promoter structure and available dataset for chromatin state, transcription initiation and TF bindings.

2.2.2 TFBS locations and chromatin state accurately predict the tissue of expression

TFs have been recognized as playing a fundamental role in gene regulation. TFs influence gene expression by binding to short DNA sequence motifs (typically between 6bp and 20bp) in the proximal promoter or in distal regulatory regions [27, 59, 50]. Tissue-specific gene regulation is a complex biological process in which a cascade of TF binding events drives conditional/temporal expression patterns. Some transcription factors have been shown to be expressed only under very specific conditions, whereas housekeeping genes or constitutively expressed genes display low variability in their temporal expression patterns. It is well established in the literature that functional TF binding is much more likely to occur in DNA regions which are not occupied by nucleosomes. In previous studies that attempt to predict gene expression patterns from sequence, computational approaches have frequently used features contained within fixed-size promoter sequences (a region of fixed size around a genes annotated TSS is used). However, predictions about the tissue or cell-type in which a gene will express have not been particularly precise or successful in human cell lines, despite the rich repository of large-scale datasets for chromatin state, transcription start sites, and gene expression across cell lines and tissue types.

In this study, we investigated whether a plant-specific model which is built using DNA sequence and chromatin state can reveal principles of tissue specific gene regulation in the extreme cases of differentially expressed genes. An important

aspect of this study is that all datasets (RNA-Seq, TSS-Seq, and DNAse-I Seq) were derived from the roots and leaves of the same plants; samples of different tissues from the same animal organism are often difficult or impossible to obtain, thus these datasets provide a unique scientific opportunity. Figure 2.4 shows a schematic view of the regulatory events which we incorporate into our modeling process. The role of transcription factor binding sites and chromatin state in tissue-specific gene expression has been supported by many past studies, as mentioned earlier in the introduction section. Here, we use promoter sequence extracted from highly expressed genes in each tissue category along with the chromatin state within those regions to model the gene expression pattern between two-tissue categories (see Section 2.2.1 for data processing procedure).

In order to identify putative transcription factor binding site (TFBS) locations in promoter sequences, position weight matrices (PWMs) −TF binding domain representations− in A.thaliana were compiled from the TRANSFAC, CIS-BP, and AGRIS databases [41, 68, 20]. Redundant PWMs with similar binding profiles were identified and removed from the final set (see Methods section 2.3.3). A total of 413 PWMs were obtained, of which each TF family had at least one PWM present. A standard log-likelihood TFBS scanning technique was used to approximate DNA binding affinity [43]. The log likelihood scores show the affinity of binding for a PWM as compared to background nucleotide composition. The log likelihood scores show the affinity of binding for a PWM as compared to background nucleotide composition. In order to model the relationship between promoter structure (TF binding site locations and chromatin state) and tissue-specificity, two

separate models were generated as follows. One is the "ROE" model in which the TF binding affinity is computed only in enriched TF binding regions, whereas in the alternative "Tiled" model that TF binding affinity has completed in fixed-sized arbitrary sub-regions within the 1kb upstream and 500 bp downstream of the TSS mode locations. Methods section 2.3.6 details each model.

2.2.3 Region of erichments (ROEs) capture essential TFBS locations for tissue-specific prediction

The ROE models features are generated based on the concept of Regions of Enrichment (ROE). An ROE (see Methods section 2.3.5) is a segment of genomic sequence where binding sites for a particular TF are preferentially located with respect to many observed TSSs; it represents a location of putative biological relevance for TF binding [43, 47]. We identified a set of at least 349 PWMs exhibiting Regions of Enrichment on at least one DNA strand. TF binding affinity scores were computed for each PWM within the ROE regions. Chromatin features, in which the openness of each TF binding region is represented by a number between 0-1, are calculated as described in Methods section 2.3.6. TFBS score values are scaled between 0-1 using the scaling approach described in Methods section 2.3.7. Among $41,196$ promoters, only $2,028$ examples were associated with differentially expressed transcripts according to the RNA-Seq dataset. Among these transcripts, $2,028$ promoter examples were used to train and test the ROE model.

With the purpose of designing a highly interpretable model that effectively

Table 2.2: Performance of all examines models.

Model Type	**auROC**	**auPRC**
ROE Model	93%	95%
Tiled Model	93%	94%
TFBS only (ROE Model)	87%	88%
TFBS only (Tiled Model)	87%	87%
OC only (ROE Model)	90%	91%
OC only (Tiled Model)	90%	90%

selects among many sequence features to provide a minimal set for optimal classification performance, we performed model training and 5-fold cross-validation (Methods section 2.3.9) using L2-regularized logistic regression. The output of the model is a probability that the promoter under examination is expressed in root or leaf. The ROE model was trained using 80% of examples. Model performance was measured by the area under the ROC curve (auPRC) and area under the precision-recall curve (auPRC) metrics on an independent subset of examples that was not used to train the model. The performance outcome (auROC = 93% and auPRC = 95%) shows that ROEs capture essential TFBS locations for tissue-specific prediction (Table 2.2).

It is important to examine and compare the way in which the ROE model captures tissue-specific regulatory information encoded in the promoters of highly differentially expressed genes. Thus, we examined an alternative model type, the Tiled model, in order to investigate this question. In the Tiled model approach, the region located 1-kb upstream to 500bp downstream of TSS mode location was

divided into 15 non-overlapping windows, or tiles, of 100bp in width. The full set of PWMs (413 PWMs) was used to compute the sum of log-likelihood scores within each tile and strand (30 TFBS features for each PWM). The percentage overlap between each tile and open chromatin regions within the promoter were computed ($24,780$ features total).

L2-regularized logistic regression was used to train the model with 5-fold cross-validation. The output of the model is a probability that the promoter under examination is expressed in root or leaf. Model performance was measured by the area under the ROC curve (auROC) and area under the precision-recall curve (auPRC) on an independent subset of examples that was not used to train the model (see Methods Section 2.3.9). Results show that Tiled and ROE models both achieve a comparable performance (see Table 2.2).

Identically to the ROE model, L2-regularized logistic regression was used to train the model with 5-fold cross-validation. The output of the model is a probability that the promoter under examination is expressed in root or leaf. Model performance was measured by the area under the ROC curve (auROC) and area under the precision-recall curve (auPRC) on an independent subset of examples that was not used to train the model Results show that the Tiled and ROE models both achieve comparably high performance (see Table 2.2).

We then investigated the impact of chromatin features on each models performance by dropping chromatin features from the training set. Both models were trained using only the TFBS features. The results show a small drop in performance for both the ROE and Tiled models on the independent held-out test set −

see Table 2.2). This suggests that TFBS site presence is a predominant explainer of tissue specific gene expression prediction. We also examined the effect of using only chromatin features by dropping all TFBS features from each model. Interestingly, the resulting models still perform very well even in the absence of TFBS features. For the ROE model, achieving a high performance by using chromatin-only features can be explained by the fact that TFBS enrichment location information is encoded into the chromatin features. However, for Tiled model, a more detailed investigation is required, which is performed as follows.

2.2.4 Information compressed vs. information dispersed

A comparable performance for both ROE and Tiled models triggers the following questions. First, what is different and what is in common between the two models? Second, do these two models disagree on the predictors of tissue specific gene expression? Answering such questions lead us to perform an analysis comparing the two modeling approaches, which can be thought of as information compressed (ROE) vs. information dispersed (Tiled) models. Both ROE and Tiled models achieved comparably high performance by all measures (auROC and auPRC). We performed follow up experiments in order to understand what makes both models successful in predicting the the tissue of expression, and whether the two models select different sets of features as being important to successful prediction.

Using an L2-regularized logistic regression approach in both Tiled and ROE models has a major benefit in that model features are highly interpretable. Fea-

ture weights used by the model are directly proportional to their importance in the models success. Using this concept, the 200 top-weighted features from each model (ROE, Tiled) were compared. Each feature is distinguished by the following information. The PWM name, the coordinates into which this feature falls with respect to TSS, and feature type (chromatin vs. TFBS). For example, a feature such as "M0376_1.02_FWD_2" quantifies TF binding affinity associated to PWM "M0376_1.02" within the second ROE sub-window on the FWD strand (in this case, FWD indicates the same strand as the genes considered; REV indicates the opposite strand).

A direct comparison between the highly-weighted features (top 200) in the Tiled vs. ROE models provides important insights about promoter structure. Figures 2.5 and 2.6 show visualization of this comparison. In these maps, red lines represent the ROE models features and green lines represent the Tiled models features. The y-axis represents the PWMs with at least one feature present in the top 200 feature lists of both models. The width of the red or green lines (intervals) represents the actual genomic coordinates of the feature. The Overlap between the ROE and Tiled feature intervals indicates that both models agreed in selecting the same PWM and region as important (i.e. contained within the 200 top-weighted features). Many agreements and disagreements on feature importance by this definition are observed. Studying such similarities and differences aids in understanding new aspects of promoter structure and tissue specificity. As shown in Figures 2.5 and 2.6, ROE and Tiled model agree on many common PWMs (60% common PWMs) and among these common PWMs there is a lot

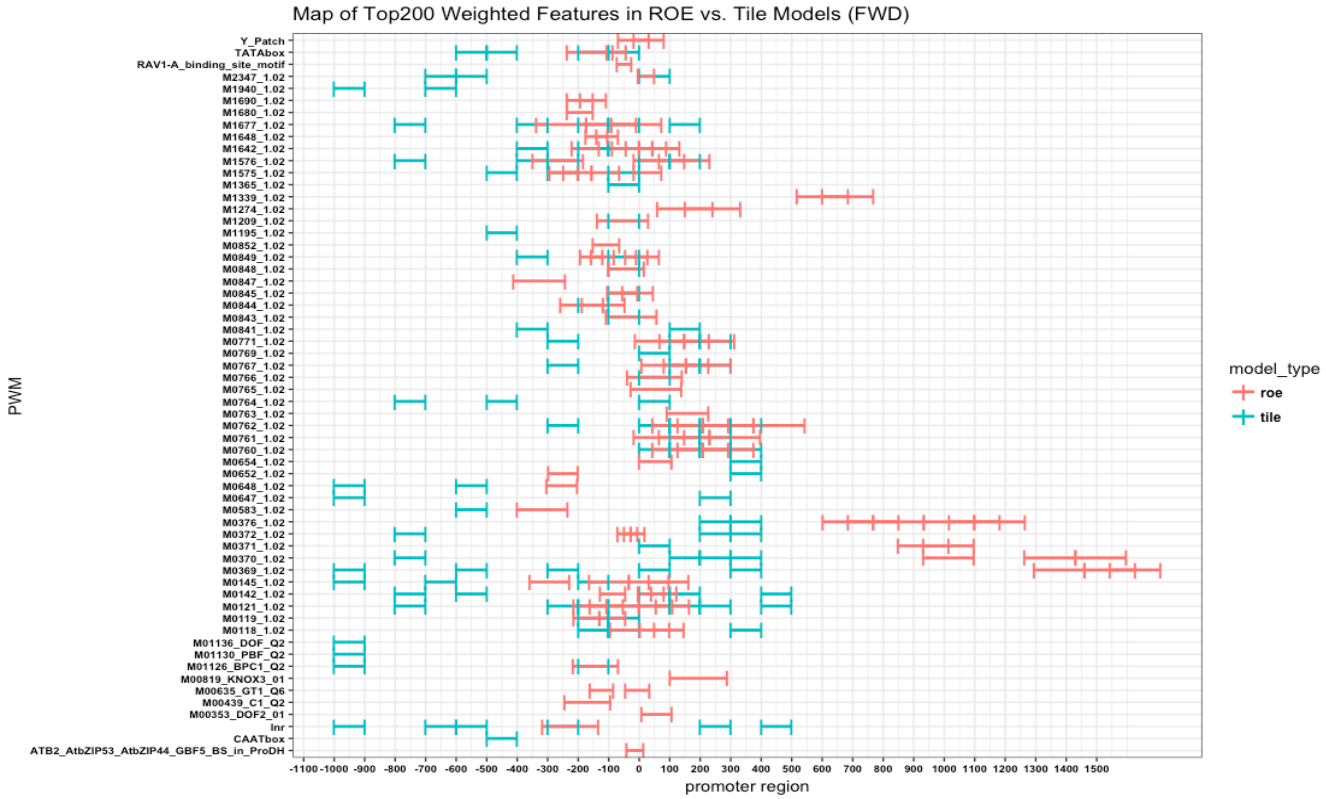

Figure 2.5: One to one Comparison between top 200 features in ROE and Tiled models (FWD strand).

of agreement on the location as well (55% of 60% features represent a common location). the ROE and Tiled models agree on many common PWMs (60% common PWMs), and among these common PWMs there is a great deal of agreement on the location as well (55% of 60% features represent a common location). The logistic regression method for two-class classification (used in both the Tiled and ROE models) assigns numerical values to the outcome variables: expressed in root is represented by 0, and expressed in leaf is represented by 1. In the modeling process, this means that negatively weighted features (those with a minus sign) can

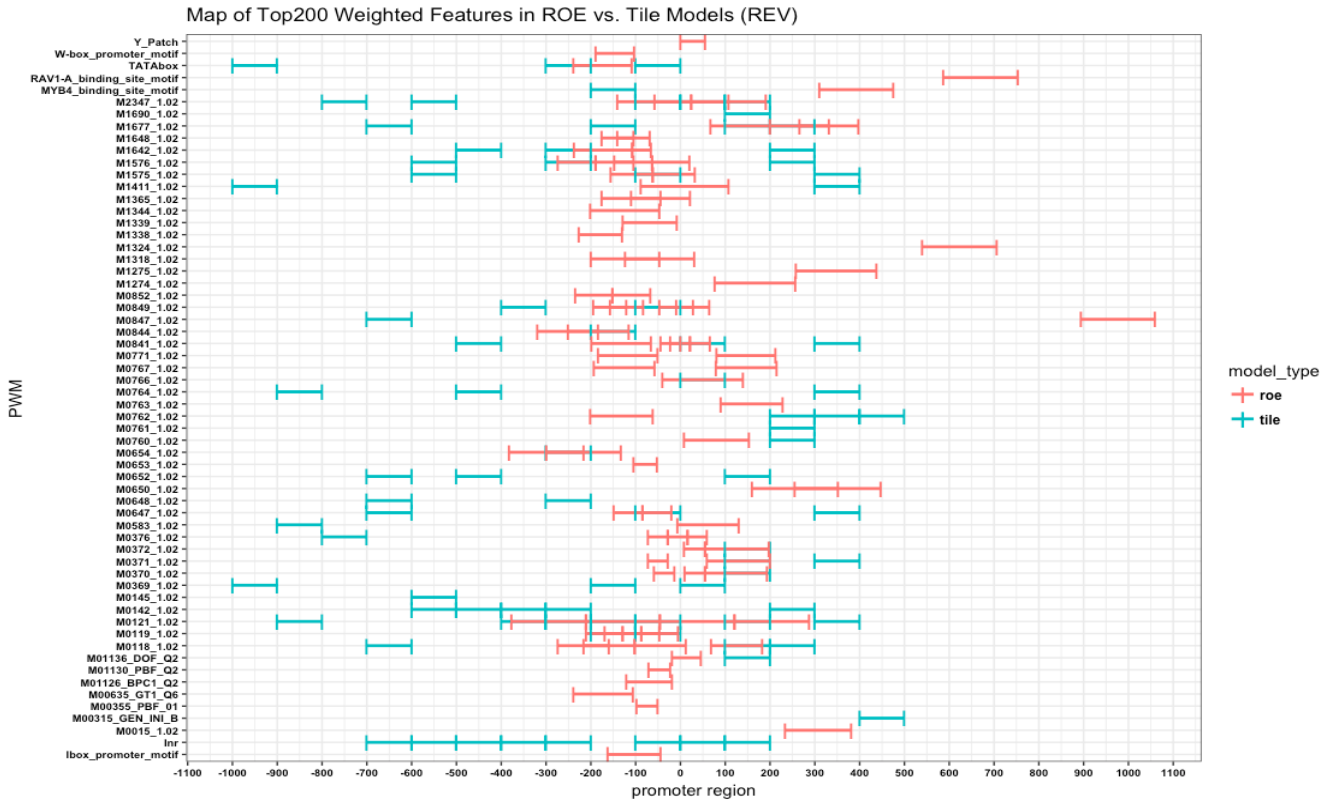

Figure 2.6: One to one Comparison between top 200 features in ROE and Tiled models (REV strand).

be interpreted as contributing to "rootness" whereas positively weighted features (those with a plus sign) can be interpreted as contributing to "leafness" in terms of the tissue-of-expression prediction. From this standpoint, we observed all of the overlapping features between the Tied and ROE models agree on the same sign.

In the next step, disagreements between the two models were analyzed. We observed three general types of disagreement between the two models. As mentioned above, some PWMs in our set do not have any region of enrichment within the examined promoter set. As a result, their corresponding features are absent from

the ROE model, but present in the Tiled model. Examples of disagreements falling into this category are M0764 on the "FWD" (same) strand, and M1690, M0369, Inr on the "REV" (opposite) strand. The second type of model disagreement is associated with PWMs that tend to have many binding sites over a broad variety of locations within promoter regions. These PWMs do not have a particularly well-defined ROE, and there may be many locations in which sites could substantially influence the tissue of expression; the Tiled model explores this possibility. Examples of features falling into this second category include M0370, M0371, M1365, M0372, M0369 on the same strand; an example of this type of enrichment is shown in Figure 2.7.

For the last type of disagreement, there is no clear explanation. M1904 provides an example of such a case. This PWM has a clear ROE, and represents a binding domain shared by multiple transcription factors with different expression profiles in root and leaf (some are root specific some are leaf specific). We observed that such examples are heavily weighted in both the ROE and Tiled models, but with substantially different relative rankings. Since we are comparing only top 200 weighted features, this cutoff provides one possible explanation− a feature may be quite important to both models, but fall on either side of our selected cutoff (i.e. M1940 is ranked at 335th in ROE model vs. 135th in Tiled). Another hypothesis is that some PWMs may contribute to tissue-specific gene expression at a location (or locations) where binding sites for this PWM are not enriched.

These results support the concept that promoter structure encodes information required for gene regulation in two ways. First, there is enough information com-

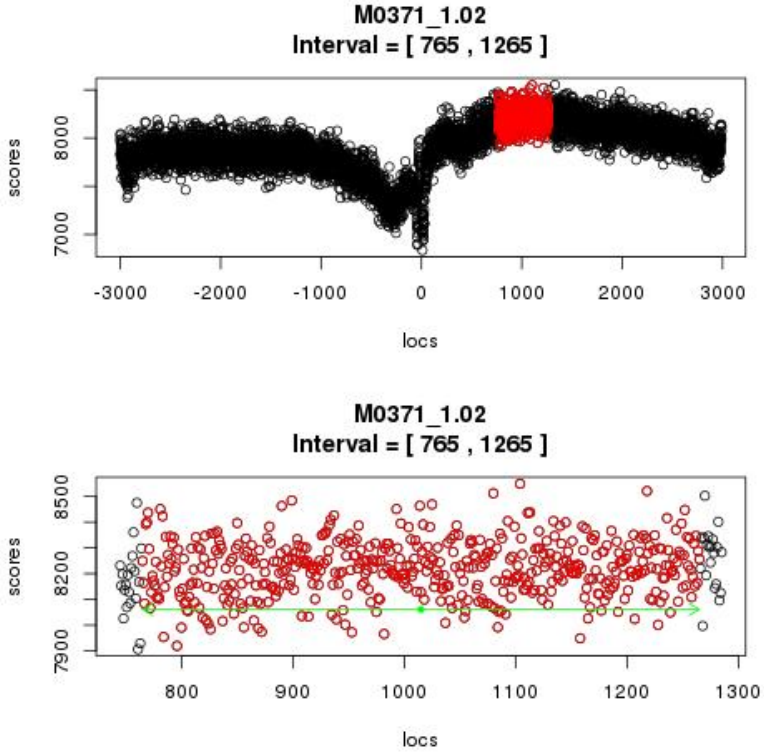

Figure 2.7: An example PWM with weak ROE signal in forward strand.

pressed into ROE regions that these locations can be used for accurate classification between the two tissues, at least at the extremes (classifying highly differentially expressed genes); and second, some TFs likely dont have a single specific locational preference for functional binding, or they have multiple preferred locations dispersed within a promoter region.

2.2.5 Feature Analysis

We used L2-logistic regression to classify promoters, given their features, as likely to express in root vs likely to express in leaf. A classifier tunes the input feature coefficients (weights) in order to achieve a model that can correctly predict the observed outcome (for examples on which the model was not trained) with high accuracy.

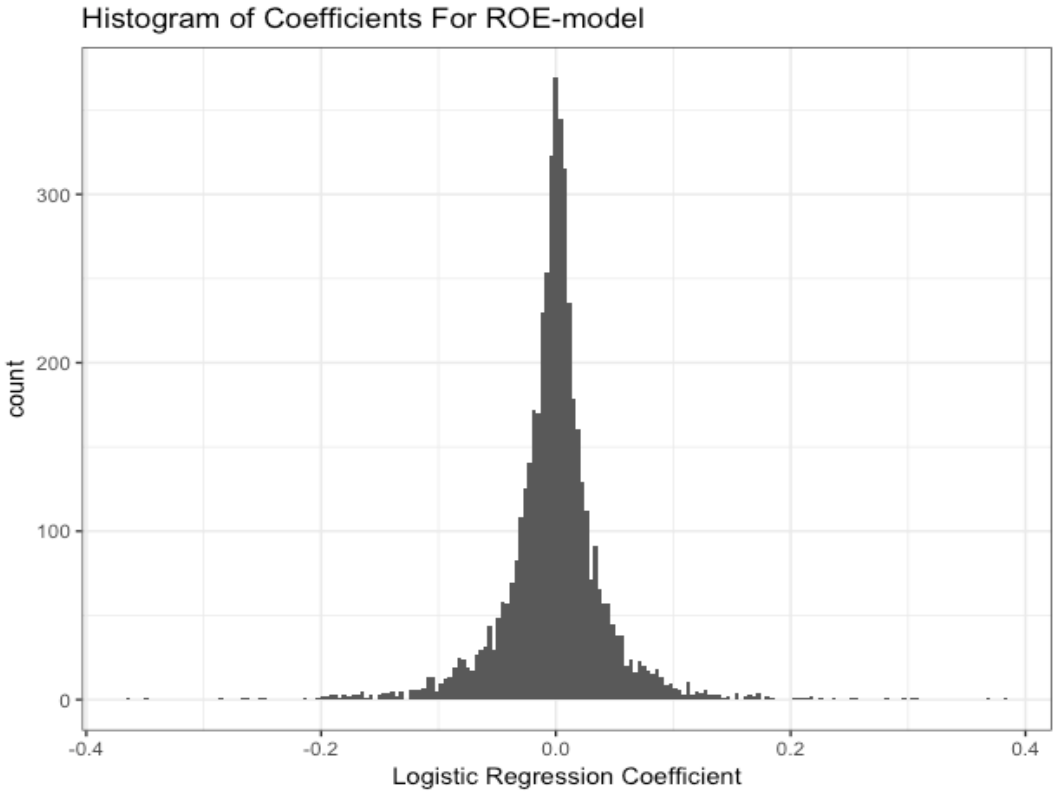

Figure 2.8: Distribution of feature weights in ROE model.

Given a list of model weights obtained from the best performing model (the model that has the highest success in its predictions), we investigated the possi-

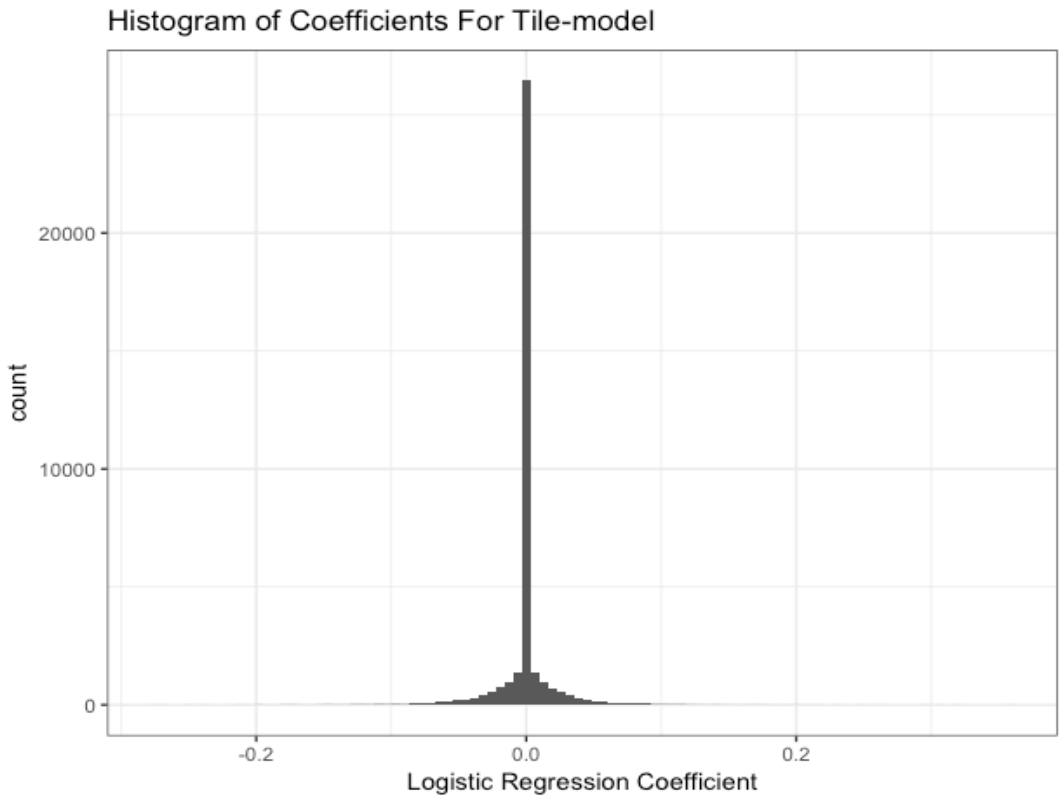

Figure 2.9: Distribution of feature weights in Tiled model.

ble biological meaning of top weighted features using literature search and gene expression profiles obtained from the RNA-seq dataset (Methods Section 2.3.10). The "top 100" features with the largest absolute coefficient values were extracted from both Tiled and ROE models. Figures 2.8 and 2.9 show the distribution of the model weights in each model. The x-axis represents the PWMs corresponding to each feature and y-axis represents the model weight. As described above, negative coefficients contribute to "rootness" which is shown in red, and positive weights contribute to "leaf-ness" which is shown in green. Examining the top 100

most heavily weighted features in the ROE model (Figure 2.8) provides important observations about the most likely influence of each factor on tissue-specific gene expression. For example, features such as M0376_1.02 contribute to both root and leaf predictions, whereas other features such as M0119_1.02 contribute to root prediction substantially more than to leaf prediction. In contrast, M1318_1.02 contributes much more strongly to leaf prediction as compared to root prediction.

In the next step, we investigated whether the model weights agree with what has been observed to date in the literature regarding the role of specific TFs in determining tissue-specificity. We used the RNA-Seq dataset to extract the expression level of the TF associated with each feature; in some cases, a PWM may represent the binding domain of more than one TF. For example, M2347_1.02 represents the binding domains of two genes AT2G45660 (higher expression in leaf as compared to root with fold change of 2.35) and AT4G10480 (root-specific expression with fold change of 3.7). We also performed a literature search in order to find more information about the biological functions of each gene or PWM. These analyses are described below.

We first gathered information about the possible biological function(s) associated with each highly weighted feature. Table 2.3 shows the gene information and expression profiles of several top-weighted features in the ROE model. The Feature column shows the feature names used in the model; as described above, each feature name encodes the PWM name, strand (FWD vs. REV), and ROE subwindow location (see Methods for ROE features). Model weight displays the feature coefficient assigned by L2-logistic regression as described above, and the Gene

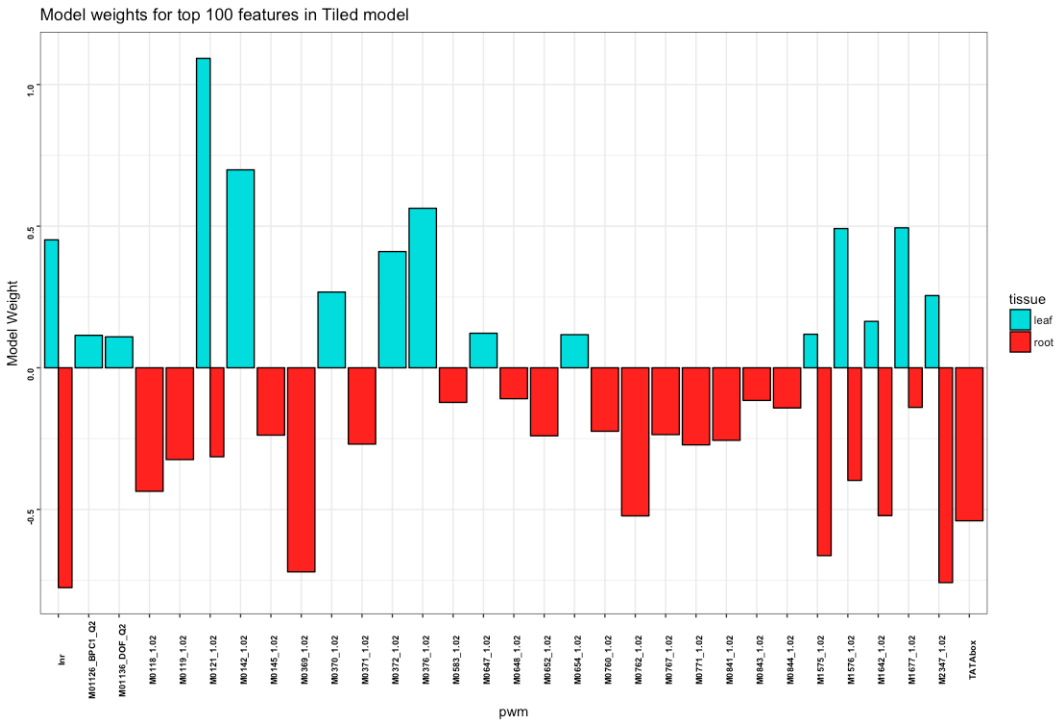

Figure 2.10: Top 100 weighted features in Tiled model.

ID represents the gene associated with the corresponding PWM in the feature. Other columns show expression level of the gene in root and leaf tissue according to RNA-Seq data. As described earlier in the introduction and data preparation sections 2.2.1 and 2.3.10, RNA-Seq, TSS-Seq, and DNase-Seq library samples are obtained from the same plants, creating an accurate snapshot of promoter state at the time that gene expression was measured . The features associated with PWM M0376_1.02 (Table 2.3) are weighted heavily by the ROE model (top two features in the ROE model), and Figure 2.11 indicates its importance in distinguishing "leaf expressing" promoters from root expressing promoters. This PWM is associated with AT5G06070, a gene for which RNA-Seq dataset shows no expression in root

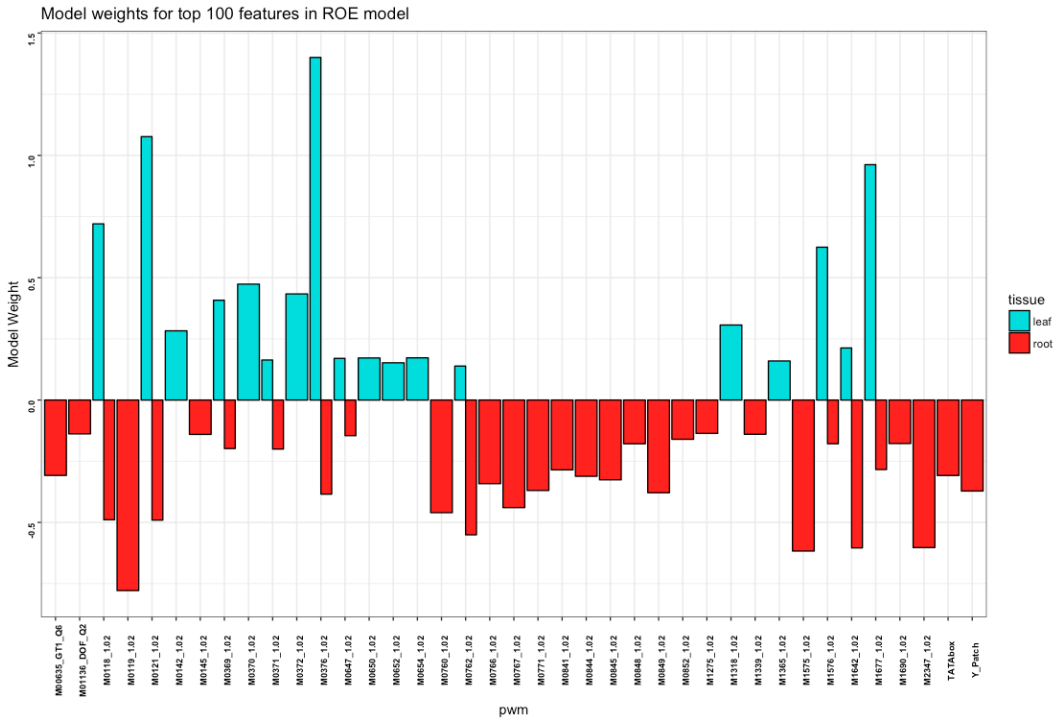

Figure 2.11: Top 100 weighted features in ROE model.

or leaf in the observed time. However, according to the related literature, this gene is known for strong expression in inflorescences and flowers, and weak expression in siliques, seedlings, and roots. In flowers, it is expressed in petal primordia and their precursor cells. It is also expressed in the lateral root caps and the basal cells of lateral roots [36] These results agree with the ROE models weight sign for this feature.

Another example of a highly weighted feature is M1576_1.02_FWD_7 (the 10th highest weighted feature in ROE model), which is weighted strongly for distinguishing "leaf" specific promoters. This feature is associated with a PWM that represents two TF genes (AT2G26580, and AT4G00180). According to RNA-Seq

dataset both genes are highly expressed in leaf (fold change of 5.15 and 4.44, respectively).

In summary, analyzing heavily weighted features of the models didnt show any disagreement between the literature and what model predicts. All indications are that the models provide a valuable source of information in order to generate hypotheses about the role of TFs in the regulation of tissue-specific gene expression. The modeling process in this study provides locational information for TFs involved in tissue-specific gene expression prediction, which provides an essential source of information for future biological experiments including laboratory knock-down experiments.

2.2.6 "Hard-coded" and "soft-coded" promoters suggest variability in mechanism for tissue specification

The "hard-coded/soft-coded" experiment explores the idea that tissue specificity of some promoters may be determined primarily through TFBS site presence ("hard-coded"), and for other promoters it may be primiarly determined through the openness of sites in the promoter ("soft-coded"). We hypothesized that there may exist specific promoters in these categories with unique TFBS or chromatin state compositions. We examined correctly classified promoters in both ROE and Tiled models to identify such promoters (see Methods section 2.3.8). Using a simple mathematical formula, the sum of TFBS feature products (normalized TFBS log-likelihood scores multiplied by model weight) and the sum of chromatin feature

Table 2.3: Heavily weighted features analysis and their correlation with RNAseq dataset

.

Feature	**Model Weight**	**Gene ID**	**LEAF normalized mean expression**	**ROOT normalized mean expression**	**Avg fold change**
M0376_1.02_REV_4	0.346	AT5G06070			
M0118_1.02_REV_1	0.30	AT1G14900	107.19	58.63	-0.7
M1677_1.02_REV_3	-0.2	AT2G06200	0.4	5.4	2.17
M1576_1.02_FWD_7	0.2	AT2G26580	29.14	0.15	-5.15
		AT4G00180	33	0.2	-4.44
M2347_1.02_REV_4	-0.2	AT4G11880	0.05	1.7	3.76
		AT2G45660	8.6	0.9	-2.35
M0371_1.02_REV_2	0.17	AT2G41940	34	0.2	-6.3
M0652_1.02_FWD_1	0.14	AT4G38000			
M0145_1.02_FWD_1	-0.13	AT5G62260	2.1	14	1.6
M1275_1.02_REV_7	-0.12	AT2G30130	0.04	21	5.12
M1648_1.02_FWD_3	0.1	AT4G18390	0.00000003	0.5	-3.7

This table shows the expression level information for genes associated with each feature. Some examples among top 100 features weighted by L2-logistic regression model that have correlation with expression level of their genes in each tissue type are listed.

products (percent chromatin openness multiplied by model weight) were computed for all tissue-specific promoters. The distributions of these scores were analyzed; promoters in which the sum of TFBS feature products fell above the 90th percentile of the distribution of the TFBS scores, and the sum of chromatin feature products fell below the 10th percentile of the chromatin feature score distribution, were considered as hard-coded. By contrast, promoters in which the sum of chromatin feature products fell above the 90th percentile of the chromatin feature score distribution, and the sum of TFBS feature products fell below the 10th percentile of the scores in the TFBS score distribution, were considered as soft-coded. Table 2.4 shows the hard-coded and soft-coded promoters found in the ROE model (12 hard-coded vs. 11 soft-coded promoters were found). This experiment was repeated in Tiled model as well (results are not shown). In the Tiled model, 15 hard-coded and only 5 soft-coded promoters were found. Tiled and ROE models showed 3 shared hard-coded promoters.

Table 2.4: List of hardcode and softcoded promoters in ROE model

Promoter ID	Coding Type	Class	OC percentile	TFBS Percentile	Gene
AT1G13300.1 Chr1_4553922	hard-code	0 (root)	5	95	HRS1 (GARP family transcription factor); nitrate/phosphate signalling in root
AT1G54940.1 Chr1_20478679	hard-code	0 (root)	10	90	GUX4 (GLUCURONIC ACID SUBSTITUTION OF XYLAN 4, a xylan glucuronosyltransferase); cell wall organization
AT2G02610.1 Chr2_707673	hard-code	0 (root)	5	95	Cysteine/Histidine-rich C1 domain family protein
AT2G17590.1 Chr2_7650197	hard-code	0 (root)	5	95	Cysteine/Histidine-rich C1 domain family protein
AT3G04330.1 Chr3_1142742	hard-code	0 (root)	10	90	Kunitz family trypsin and protease inhibitor protein; present in lateral root cap
AT4G13235.1 Chr4_7675295	hard-code	0 (root)	10	90	EDA21 (EMBRYO SAC DEVELOPMENT ARREST 21); involved in embryo sac development, defense against fungal pathogen
AT4G33120.1 Chr4_15975080	hard-code	0 (root)	10	90	S-adenosyl-L-methionine-dependent methyltransferases superfamily protein; methyltransferase activity at plasma membrane
AT5G54230.1 Chr5_22014930	hard-code	0 (root)	5	95	MYB49
AT5G54230.1 Chr5_22014974	hard-code	0 (root)	5	95	MYB49
AT1G56520.2 Chr1_21175962	hard-code	1 (leaf)	10	90	Disease resistance protein (TIR-NBS-LRR class) family
AT3G50270.1 Chr3_18632922	hard-code	1 (leaf)	10	90	HXXXD-type acyl-transferase family protein; located in chloroplast
AT5G42760.1 Chr5_17145949	hard-code	1 (leaf)	10	90	Leucine carboxyl methyltransferase
AT1G55670.1 Chr1_20800394	soft-code	1 (leaf)	90	10	Photosystem I subunit G; electron transport during photosynthesis
AT1G55670.1 Chr1_20800400	soft-code	1 (leaf)	90	10	See above.
AT1G76080.1 Chr1_28546385	soft-code	1 (leaf)	90	10	CDSP32 (CHLOROPLASTIC DROUGHT-INDUCED STRESS PROTEIN 32 kDa); Probable thiol-disulfide oxidoreductase involved in resistance to oxidative stress
AT2G06520.1 Chr2_2585258	soft-code	1 (leaf)	90	10	PSBX (PHOTOSYSTEM II SUBUNIT X); photosynthesis
AT3G21055.1 Chr3_7374003	soft-code	1 (leaf)	90	10	PSBTN (PHOTOSYSTEM II SUBUNIT T 5 kDa); photosynthesis
AT3G54050.1 Chr3_20014101	soft-code	1 (leaf)	95	5	CFBP1; Involved in the regulation of photosynthetic electron flow and sucrose synthesis. Its activity is critical for normal plant development and important for the regulation of a wide range of metabolic processes.
AT4G02420.1 Chr4_1063406	soft-code	1 (leaf)	90	10	LECRK-IV.4 (L-TYPE LECTIN RECEPTOR KINASE IV.4); protein phosphorylation; defense against bacteria, oomycetes.
AT4G19500.2 Chr4_10622717	soft-code	1 (leaf)	90	10	nucleoside-triphosphatase/transmembrane receptor/nucleotide binding/ATP binding protein; signal transduction;
AT4G19500.2 Chr4_10622720	soft-code	1 (leaf)	90	10	See above.
AT5G10380.1 Chr5_3264660	soft-code	1 (leaf)	90	10	E3 ubiquitin-protein ligase RING1; positive regulation of programmed cell death in response to fungal pathogen; mRNA is cell-to-cell mobile.
AT5G10380.1 Chr5_3264698	soft-code	1 (leaf)	90	10	See above.

2.3 Materials and Methods

2.3.1 Sample preparation and sequencing

7-day old Arabidopsis seedlings were grown in a chamber at 21 °C under a 12 hour light cycle (50% humidity, and 250 mol/m2/s light intensity). The seedlings were divided into three batches for TSS-Seq, DNase-Seq, and RNA-Seq procedures. For each batch, roots and leaves of 1 week old seedlings were dissected using a surgical blade, flash-frozen in liquid nitrogen, and stored at −80 °C. TSS-Seq was carried out for both root and leaf samples as described in [19] using the nanoCAGE protocol in conjunction with the HiSeq-2000 sequencing platform. Chromatin from isolated nuclei was digested with DNase I, and DNase-seq libraries were prepared for both root and leaf samples as published in [18]. RNA-Seq was carried out for both root and leaf samples and sequenced on the Illumina HiSeq-2000 sequencing platform (three replicates for each sample).

2.3.2 Sequence processing and alignment

The TAIR10 reference genome was used for sequence alignments. CapFilter software [19] was used to pre-process all TSS sequence files prior to alignment (extra Gs "GGG", that are library artifacts were removed from the beginning of the reads). For RNA-seq, DNase-seq, and TSS-seq sequences (Cap filtered reads in TSS-seq), all reads were aligned to the TAIR10 reference genome, using Bowtie version 2.0 [] with the parameter settings '-v 0 -m 1 -a −−best −−strata' (only one

mismatches allowed along with uniquely mapped reads). Sample depth analysis was performed for all the sequences. Sequencing depth analysis was carried out using method descibed in [47].

2.3.3 PWM redundancy detection

Position Weight Matrices (PWMs) for TFs in *Arabidopsis thaliana* downloaded from TRANSFAC [65], JASPAR [55], AGRIS [20], and CIS-BP [68] databases. We developed a software program in the Python language in order to compute the element-wise distance between PWM pairs. In the first step, this program computes the element-wise distance (absolute difference) between PWM pairs with the same dimension (Binding motifs with the same length). The PWM pairs with less than or equal distance of a certain threshold (0.09) were marked as redundant (see Definition 1).

Definition 1. *for each PWM pair:* $P1_{N,4}$ *and* $P2_{M,4}$ *where* $M = N$*; compute*

$$D_{P1,P2} = \max |P1 - P2| \tag{2.1}$$

if $D_{P1,P2} < 0.09$*; then* $P1$ *and* $P2$ *are similar.*

In the next step, PWM pairs for which the dimension difference was less than three were compared in order to identify PWMs that share the same core consensus sequence as follows (Algorithm 1).

Algorithm 1 Computing PWM redundancy

for each PWM pair: $P1_{N,4}$ and $P2_{M,4}$ where $|M$ -$N| < 3$ and $M >= N$;

for $i = 0$ to $|M - N|$; do

$d_i = D(P1_{i:N}, P2)$

if $d_i < 0.09$; then $P1$ and $P2$ are similar.

terminate.

2.3.4 TSS peak detection and annotation

After aligning and cap filtering the TSS reads in root and leaf samples, JAMM' peak finder [29] was tuned to identify TSS read clusters. Two parameters control the resolution of JAMM: Fragment Length (F) (default is 250) and Bin Size (b). The bin size is computed automatically based on F unless it is manually set. For TSS datasets in our study, the fragment size and bin size are both set to 10. The output of JAMM is a list of peaks along with their genomic coordinates. The bedtools software suite was used in order to retrieve the number of aligned reads in each peak region for peak annotation. An R script was developed to process the aligned reads within peak regions, and peak information such as the number of aligned reads in peak, TSS peak mod location, mod read count, and peak shape (narrow, broad, or weak), was computed. Peak shape detection was performed as described in [51, 52]. TSS peaks were assigned to the closest annotated transcript (according to TAIR10 annotation) location using a software program. TSS peaks which fell within 250 bps upstream of the annotated translation start site and

containing more than 50 TSS peak read counts were selected.

2.3.5 Region of Enrichments

TSS peaks found in root and leaf were unified ($\approx$ 42,000 peaks) and 6-kb promoter sequences centered at TSS peak modes were extracted from reference TAIR10 FASTA file. Using TFBSScan software package available at `http://megraw.cgrb.oregonstate.edu/software/TFBS\_Scanner\_Suite`, each PWM was scanned over an 6-kb region centered around the TSS peak mode, identifying the regions where each TF binding site was most likely to occur across all promoter regions extracted from all TSS peaks. ROEs were defined on both strands by identifying the highest scoring region for each PWM across all promoter examples. PWMs with cumulative log-likelihood score peaks up to 2 kb from the TSS mode were considered (similar to studies in [47, 48]). The regions of enrichment (ROEs) for each PWM were computed using an updated version of the ROEFinder software (developed in-lab).

2.3.6 Feature generation

Each TFBS associated feature represents an approximation of cumulative binding affinity that a particular TF has for a specific genomic region. Each open chromatin feature represents a percentage of nucleotides within the associated region that are open. In the ROE model, each ROE region as determined in section 2.3.5 is divided

into five overlapping sub-widows and two flanking windows as described in [43]. The TFBS features are cumulative log-likelihood scores for each ROE sub-window on the same and opposite strands (as the gene in consideration). The chromatin features are computed as the percentage overlap between the open regions and each ROE sub-window. Only log-likelihood scores greater than 0 are considered as potential binding sites and contribute to the sum, therefore the minimum value for a TFBS feature is 0 (this is a case where none of the nucleotides in the region represent a potential binding site with a greater-than-zero log-likelihood score). In the Tiled model, the entire region from 1-kb upstream to 500bp downstream of the TSS mode is divided into non-overlapping windows of 100bp in width. The TFBS features were computed as cumulative log-likelihood scores within each tile for both strands. The Tiled model open chromatin features are computed as a percentage overlap between the open regions and each tile.

2.3.7 Model feature scaling

Open chromatin features as described in section 2.3.6, share the same 0-1 range and are interpretable as an "openness proportion" without modification. As described in the Feature Generation 2.3.6 , this is computed as a sum of log-likelihood scores over all nucleotides in the region, where each nucleotide is taken as the starting point of a potential TF binding site; the log-likelihood score is computed at this site using (1) the PWM associated with the TFs binding domain, and (2) a local background nucleotide distribution model. The maximum possible value

for a TFBS feature is region length L multiplied by the maximum possible log-likelihood value $PWM_{scoreMax}$ of any binding site (i.e. the score of the PWMs consensus sequence); this is a theoretical case in which every nucleotide in a region represents the consensus sequence of the PWM. To scale TFBS features such that each feature conceptually approximates a binding affinity proportion, we compute $PWM_{scoreMax}$ using each PWM and a universal background model reflecting the nucleotide content of the Arabidopsis genome. Each TFBS feature is then scaled by 1/PWMscoreMax. Finally, all TFBS feature values are scaled by 10, in order to put the mean TFBS feature value on the same order of magnitude as the mean OC feature value (i.e. put both feature types on the same approximate order of magnitude for the regularization algorithm).

2.3.8 Hard coded vs. soft coded promoters

Promoter examples which were correctly classified with a high probability (0.9) were selected and examined for the soft-coded/hard-coded experiment as follows. The set of TFBS and chromatin feature values of each promoter were extracted, and for each set the following formula was applied.

$$tfbsSOP_{promoter_i} = \sum tfbsScore_j W_j \tag{2.2}$$

$$ocSOP_{promoter_i} = \sum ocScore_j W_j \tag{2.3}$$

W is model weight vector.

The distribution of the sum of products for TFBS and chromatin features were computed,and 5% tails of the histograms were considered for detecting hard-code/softcoded promoters. Promoters for which $tfbsSOP$ fell above the 95th percentile and $ocSOP$ fell below the 5th percentile were labeled as hard-coded. The soft-coded promoters where analogously determined (see Figures 2.12 and 2.13).

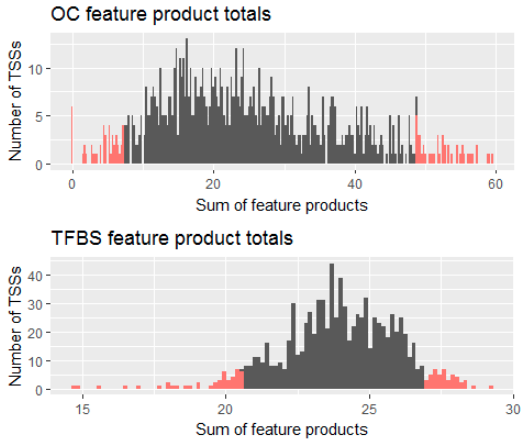

Figure 2.12: Sum of feature products for TFBS and chromatin features across training and testing examples (ROE model).
Pink regions are the focus of finding candidate promoters for hardcoded and soft-coded experiment.

2.3.9 Model training and testing

We used pythons scikit-learn [51] library to implement a L2-regularized logistic regression program. The model was implemented in the Python language using cross-validation. For cross-validation, a range of parameter values was examined, and the average of parameter values resulting best performance across the folds

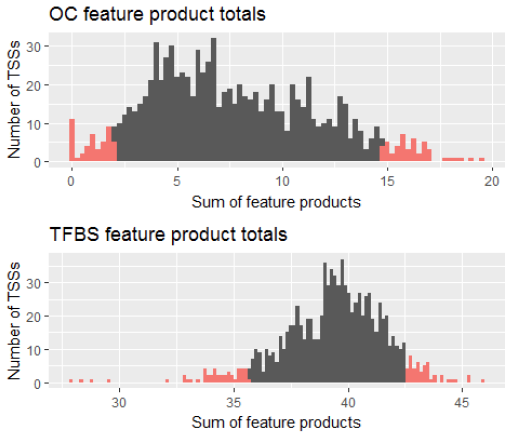

Figure 2.13: Sum of feature products for TFBS and chromatin features across training and testing examples (Tiled model).
Pink regions are the focus of finding candidate promoters for hardcoded and soft-coded experiment.

was selected for final testing.

2.3.10 Differential expression analysis

After mapping RNA-seq reads to the TAIR10 genome, first, the abundance of individual transcripts was determined using the Kallisto software package [12]. Next, differentially expressed transcripts were identified using the sleuth tool [52], which is designed for the analysis of differential transcript abundance values produced by Kallisto, with three replicates from each sample tissue.

2.4 Summary

In this study, we showed that a plant-specific model which is built using DNA sequence and chromatin state can accurately predict the tissue of expression for highly differentially expressed genes. We used a unique plant dataset including TSS-seq, DNase-seq, and RNA-seq data in order to construct our model. We used TSS-seq to identify accurate transcription start locations and their TF binding site locations. DNase-seq was used to compute the openness of TFBS binding regions. RNA-seq data was used to select differentially expressed promoters for training the model. We proposed two different models, ROE and Tiled, in order to model the relationship between promoter structure and tissue-specific gene expression in the model plant Arabidopsis thaliana, using root and leaf tissues. The ROE model was constructed on the assumption that most of the transcription factors have binding location preferences on the genome with respect to TSS location. From this perspective, TFBS features were computed within ROE regions. Those PWMs that did not have an ROE region were discarded from ROE model. In contrast, in the Tiled model, TFBS features were generated by computing log-likelihood scores for non-overlapping window regions (100bp in width) located within a 1-kb upstream to 500bp downstream region of the TSS. Our results showed that TF binding sites and chromatin state accurately predicted the tissue-specific gene expression in both ROE and Tiled models.

Removing chromatin features didnt reduce the models performance significantly. We also examined the idea of hard-coded and soft-coded promoters. We

found examples of hard-coded promoters in which TFBS features were the predominant predictor of tissue-specific expression, and soft-coded examples in which chromatin features were heavily weighted by the model and were influential for tissue-specific prediction.

Since the initial set of PWMs downloaded from multiple databases contained many matrices which were highly similar or sometimes identical, I used a software program to compute the similarity between the PWMs and remove redundant PWMs. This improved both the performance of the models and interpretation of the feature weights. This is because redundant PWMs with similar binding domains can lead to collinearity among the features and as a result, the weights assigned to each PWM may be distributed to similar PWMs, with comparatively few PWMs attaining significant coefficients.

Chapter 3: IndeCut evaluates performance of network motif discovery algorithms

Mitra Ansariola, Molly Megraw, and David Koslicki

Bioinformatics Journal

https://doi.org/10.1093/bioinformatics/btx798

3.1 Background

Genomic networks represent a complex map of molecular interactions which are descriptive of the biological processes occurring in living cells [23, 46]. Due to the size and complexity of these networks, it is often difficult to infer the physiological function of individual interactions or collections of interactions without additional detailed information about network structure. Because this type of experimentally supported prior information is usually sparse or unavailable, a systematic approach for identifying key sub-components and their functions within a biological system is essential for analysis. From this perspective, it has been shown that the functional essence of a complex genetic network within a cell can often be distilled by thinking of the network as a "circuit board" composed of small, understandable components that work together to carry out higher-order processes [42, 2, 71, 40, 58, 46, 4, 67, 53]. Network motif discovery is a well-established statistical strategy for performing network analysis from this viewpoint. This strategy compares the frequency of observation of a sub-network within the larger original network to its frequencies in many randomized background networks in order to identify network motifs, which are defined as those sub-networks observed at a significantly higher frequency in the original network. In other words, a network motif is an over-represented sub-structure within a larger network.

Network motif discovery tools aid in generating specific testable hypotheses about the behavior and function of a genetic sub-circuit. For example, in the case of a gene regulatory network, a bi-stable switch coupled with a noise-damping

circuit may be necessary to tune the expression of developmental transcription factors involved in body-plan patterning at a specific stage of development [42, 2, 65]; thus this circuit may appear as a motif in networks constructed from tissue samples in developing organisms. Although such hypotheses are valuable starting points for understanding the underlying mechanisms of a biological process through analysis of genomic networks, the laboratory validation of a predicted network motif is generally a costly and time-consuming endeavor. For example, validating a candidate regulatory sub-network containing a specific transcription factor, a microRNA, and a protein coding gene would typically require a series of procedures such as electrophoresis mobility shift assays and generation of reporter constructs, involving months of labor and thousands of dollars in supplies. This highlights the need for accurate network motif discovery procedures in order to acquire a biologically meaningful outcome.

To characterize statistical significance of a given genomic network (here called the "original network"), network motif discovery algorithms generate random graphs (here called "background network generation") while striving to satisfy two conditions. 1) Background networks should preserve a sensible set of biological assumptions constrained by the original network. For example, if the original network contains a node type (e.g. transcription factor) that can target itself as well as other genes in the original network, then this property can be preserved in the generated background networks. 2) The background networks generated should provide a truly representative sample of all possible such networks. That is, for statistical purposes, the generation method should not favor the production of certain types

of networks over others. While there are a variety of choices that a researcher may make about network property preservation, it is clearly crucial to generate an unbiased sample of background networks which preserve these properties- thus avoiding inaccuracy resulting from the background network generation procedure itself.

Computationally, the core component of background network generation is the sampling of a number of networks (for example, 1000 networks) from the set of all possible networks (e.g. 1 million networks) having in-degree and out-degree sequences identical to those of the original biological network. Networks are usually thought of as graphs, and this sampling process is known as "graph sampling." Ideally, graph sampling would be unnecessary; one would simply generate all possible graphs in the sample space, count the number of times a particular sub-graph of interest was observed, and then calculate an exact P-value by comparing this count to the number of times it was observed in the original network. A very small P-value would indicate significant over-representation, and thus a network motif. Unfortunately, for networks of realistic biological size −even a few hundred nodes and edges − the size of the sample space is enormous (over trillions of graphs). Furthermore, there is not even any known closed-form formula for computing just the number of graphs in the sample space. Thus, graph sampling is a practical necessity but presents a challenge in its own right, as one must sample in an unbiased manner from a set of unknown size. To date, no method has been given to estimate even the number of samples required in the background network generation process.

Despite a rich mathematical literature on the subject [45, 3, 21, 8, 15, 44, 30, 16, 34], practical solutions to this problem remain elusive. Even so, several network motif discovery tools with different underlying graph sampling strategies are currently available [33, 64, 26]. Theoretical results that ensure uniformity have been obtained, but only when an arbitrarily large number of samples is allowed [25]. Practical performance evaluation has been restricted to small "test graphs" where samples spaces can be empirically enumerated by producing all possible graphs in the space. On such graphs, it has been shown that depending on graph topology, the same sampling strategy can have very different performance outcomes in terms of uniform and independent sampling [42]. For example, while d-regular graphs rarely pose a problem, small graphs with highly irregular or "uneven" degree sequences frequently cause difficulty [42, 9, 25]. This creates a concern for the accurate performance of network motif discovery algorithms on real biological networks, which often contain large source hubs ("master regulators") and/or target hubs (heavily regulated nodes) [61, 70].

To date, no mathematically sound yet computationally practical method is available in order to determine whether a graph sampling method samples uniformly and independently for a large or even moderately-sized network of interest. However, relatively recent advances in the enumerative combinatorics literature [1, 5] have opened an avenue for the development of solutions to this long-standing problem. In this study we present IndeCut, which assesses the degree of sampling uniformity and independence for network motif discovery algorithms. We also show how IndeCut can provide a way to understand the cause of performance variations

among different graph sampling approaches.

3.2 Results

As previously described, IndeCut uses the cut norm to assess how uniform and independent a network motif discovery algorithm's sampling regime is (with larger cut norm values indicating non-uniform or non-independent sampling). In this section, we assess the performance of a selection of such network motif discovery algorithms.

3.2.1 IndeCut evaluates the performance of network motif discovery algorithms

Two different types of graphs are examined: 1) small graphs with topologies that typically occur in biological networks, and 2) realistic graphs from the literature with a large number of nodes and edges. We selected four network motif discovery approaches from the recent literature: FANMOD (Fast Network Motif Detection) [69], DIA-MCIS (Diaconis Monte Carlo Importance Sampling) [22], WaRSwap (Weighted and Reverse Swap sampling) [42], and CoMoFinder (Coregulatory network Motif Finder) [38]. Each of these algorithms represents a fundamentally different strategy for network motif discovery background network generation.

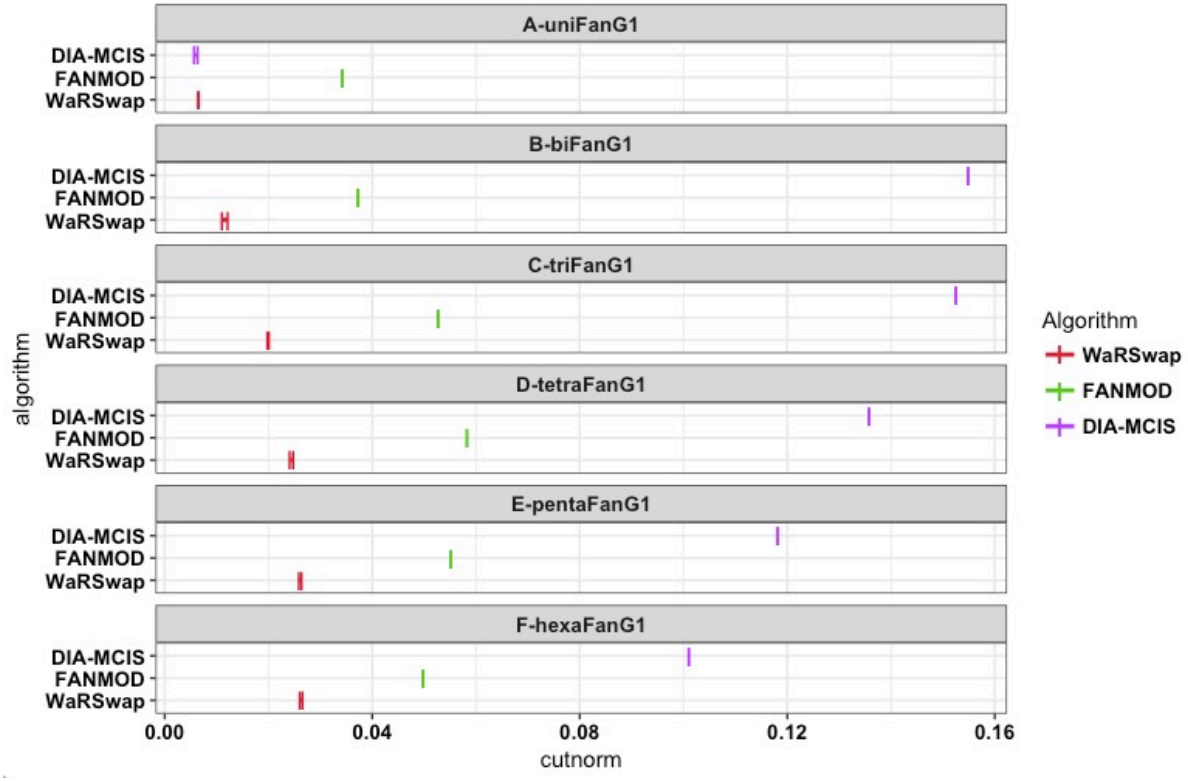

Figure 3.1: Small uneven graph sampling performance.
For each small uneven graph and algorithm, 5000 graphs were generated and the cut norm estimates for each algorithm were computed using IndeCut. The vertical lines represent lower and upper bounds returned by the cut norm estimation with the true (NP-hard) value lying in this interval. A cut norm interval that is far from zero represents less uniform and independent sampling. With the exception of uniFanG1, the cut norm estimates for CoMoFinder were much larger than 0.16, and hence are not shown for ease of comparison (see Figure 3.2 and 3.3 for detailed results).

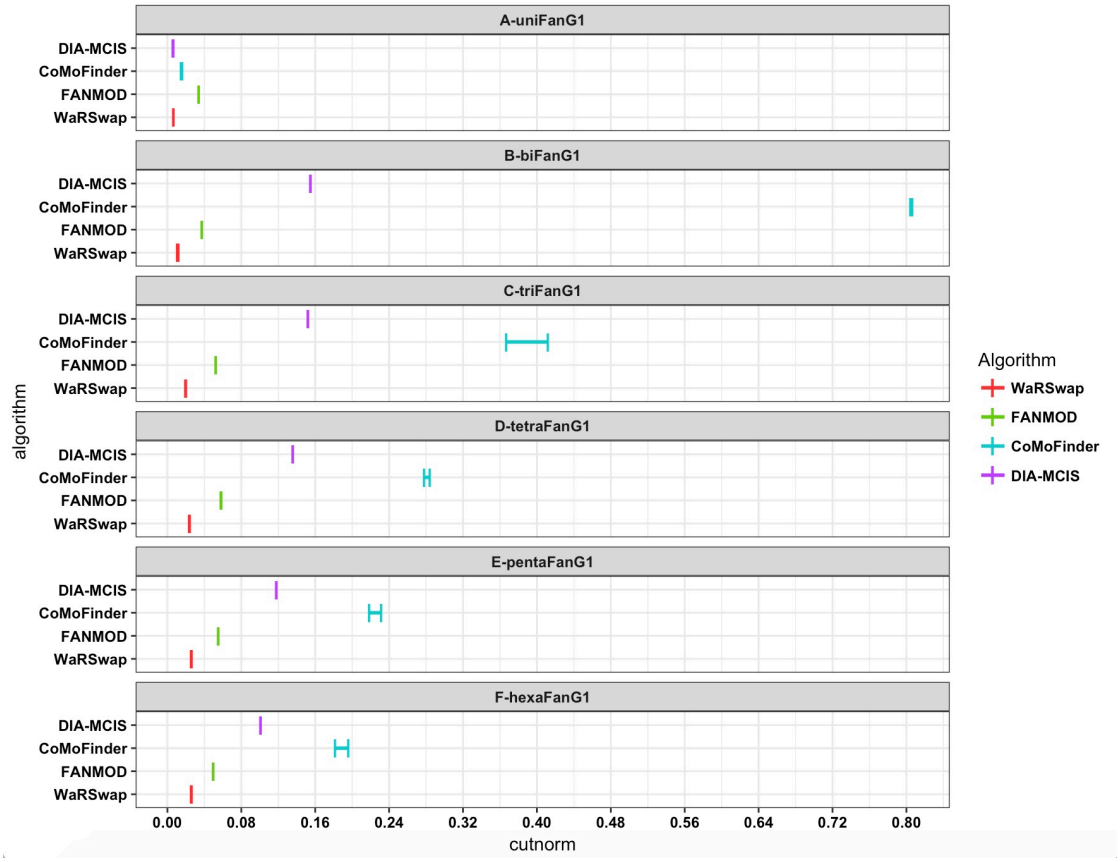

Figure 3.2: Graph sampling performance evaluation on small uneven graphs using IndeCut.
This figure shows the cut norm estimates for all four examined algorithms: WaRSwap, CoMoFinder, DIA-MCIS, and FANMOD. For each graph and algorithm, 5000 graphs were generated. The cut norm estimates for each algorithm were computed using IndeCut. The vertical lines represent lower and upper bounds returned by the cut norm estimation with the true (NP-hard) value lying in this interval. A cut norm interval that is far from zero represents less uniform and independent sampling.

3.2.1.1 Small graph collection

Three classes of small graphs were created to consider three distinct topological properties: 1) "uneven" (irregular) graphs containing "hub" nodes with large in-

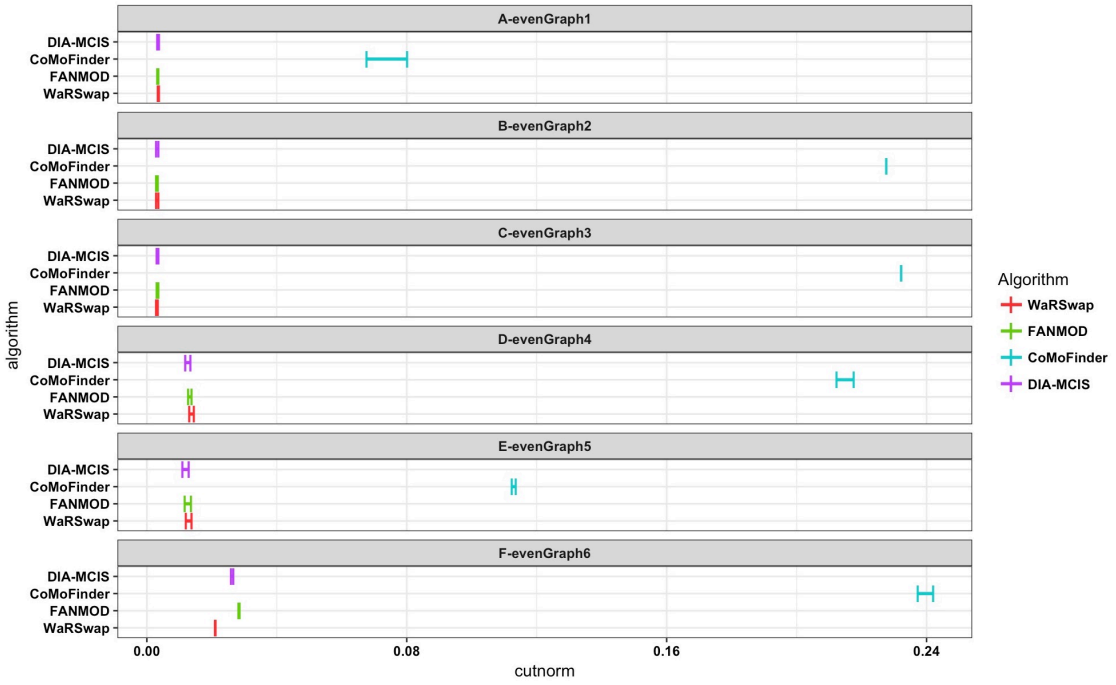

Figure 3.3: Graph sampling performance evaluation on small even graphs using IndeCut.
This figure shows the cut norm estimates for all four examined algorithms: WaRSwap, CoMoFinder, DIA-MCIS, and FANMOD. For each graph and algorithm, 5000 graphs were generated. The cut norm estimates for each algorithm were computed using IndeCut. The vertical lines represent lower and upper bounds returned by the cut norm estimation with the true (NP-hard) value lying in this interval. A cut norm interval that is far from zero represents less uniform and independent sampling.

degree or out-degree as compared to the other nodes in the graph. 2) "even" (regular) graphs with even (d-regular) or nearly even degree sequences. 3) "hybrid" combinations of even and uneven graphs. These graphs mimic the properties of large biological networks on a smaller scale and enable us to examine how IndeCut evaluates the sampling performance of different algorithms on specific graph

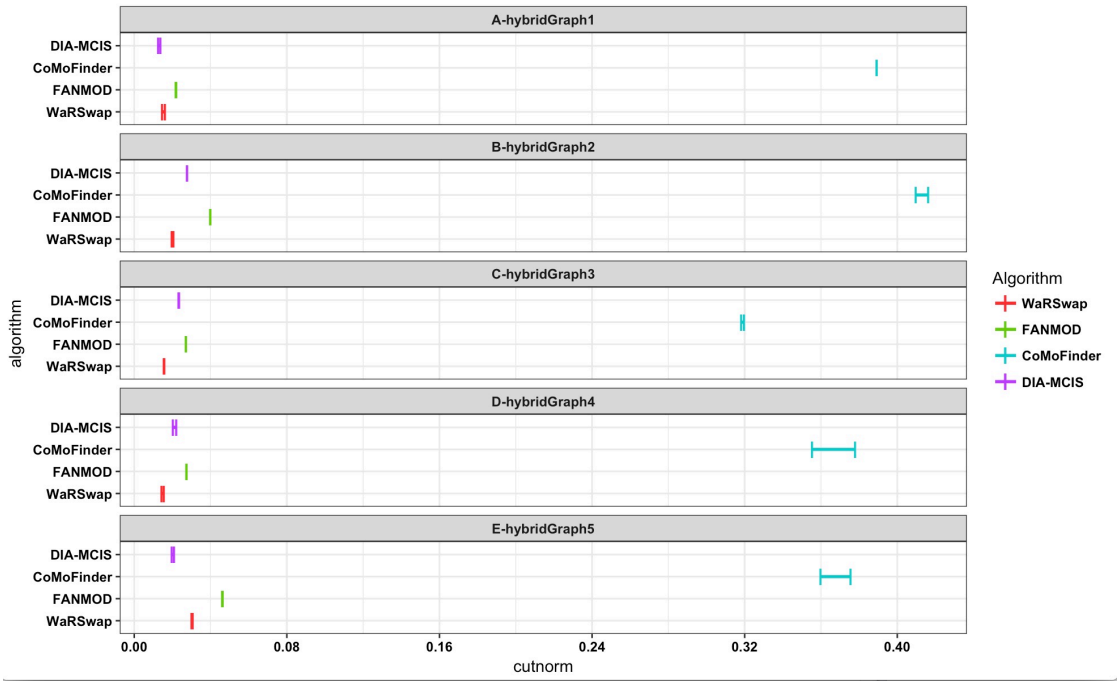

Figure 3.4: Graph sampling performance evaluation on small hybrid graphs using IndeCut.
This figure shows the cut norm estimates for all four examined algorithms: WaRSwap, CoMoFinder, DIA-MCIS, and FANMOD. For each graph and algorithm, 5000 graphs were generated. The cut norm estimates for each algorithm were computed using IndeCut. The vertical lines represent lower and upper bounds returned by the cut norm estimation with the true (NP-hard) value lying in this interval. A cut norm interval that is far from zero represents less uniform and independent sampling.

structures.

For the uneven class, we created six hub-containing graphs. The first graph (uniFanG1) is an example of a simple hub-containing bipartite graph in which each layer (source and target layers) has a single hub node. We created biFanG1 by duplicating/joining two uniFanG1 graphs. We repeated this process (attaching

a uniFanG1 to an existing graph) to generate the "Fan" series of graphs. These graphs allow us to understand how IndeCut captures the performance of each algorithm on graphs with an increasingly large degree of unevenness, a topology type which is known to pose difficulties to many algorithms [42]. For the class of regular graphs, three d-regular and three near d-regular graphs with a different number of nodes and edges were created. Fig. 3.1 shows the cut norm estimates for each graph and algorithm within the three classes of small graphs.

In Fig. 3.1, as the degree of unevenness for graphs increases from A to F, one observes decreasing performance (in the case of FANMOD and WaRSwap) or comparatively poor performance (in the case of DIA-MCIS). This is in contrast to the performance on the nearly regular graphs, where most of the methods have comparably strong performance. On the "hybrid" graphs, sampling performance varies widely among the methods. The hybrid graphs highlight the necessity of IndeCut in determining the performance of each algorithm, particularly when the degree sequence of a graph yields no intuition with regard to the anticipated performance of any given method. This is in agreement with the previously observed trend that hub-containing graphs are highly problematic to many algorithms, whereas regular graphs are typically less troublesome [42]. These results confirm that many algorithms have difficulty sampling from bipartite graphs containing large hubs, while most of the algorithms have comparably strong sampling performance when operating on a sample space of even or near-even graphs. We conclude that different graph topologies can produce vast performance differences when using the same algorithm and that there exists a wide variation in performance between

algorithms.

3.2.1.2 Real-world biological networks

In order to understand how these topologies interact in real biological graphs of interest, we examined two published genomic networks with different degree sequences and scales. First, we analyzed a well studied, medium sized E-coli regulatory network (≈ 400 nodes and ≈ 600 edges) with a mixed degree sequence and two node types: transcription factors (TFs) and protein-coding genes (Genes). This network has been used as a case study by several network motif discovery studies including those published in conjunction with the FANMOD and CoMoFinder programs [69, 38]. Figure 3.6 shows the performance of each algorithm on E-coli network.

Secondly, we analyzed a large human regulatory network ($\approx 15,000$ nodes and $\approx 150,000$ edges) containing three different node types (TFs, miRNAs, and protein-coding genes) that was used as a case study in CoMoFinder's publication [38]. This network contains TFs that are "master regulators," and thus has large source hubs.

As mentioned in Section 3.3.3, IndeCut uses bipartite graphs as input, so networks were broken into component bipartite graphs (TF→TF, TF→Gene in the Ecoli network and TF→TF, TF→Gene, TF→miRNA, miRNA→TF, and miRNA→Gene in the human network). Figure 3.5 depicts the resulting cut norm estimates for each algorithm on the large human network and demonstrates the variability among

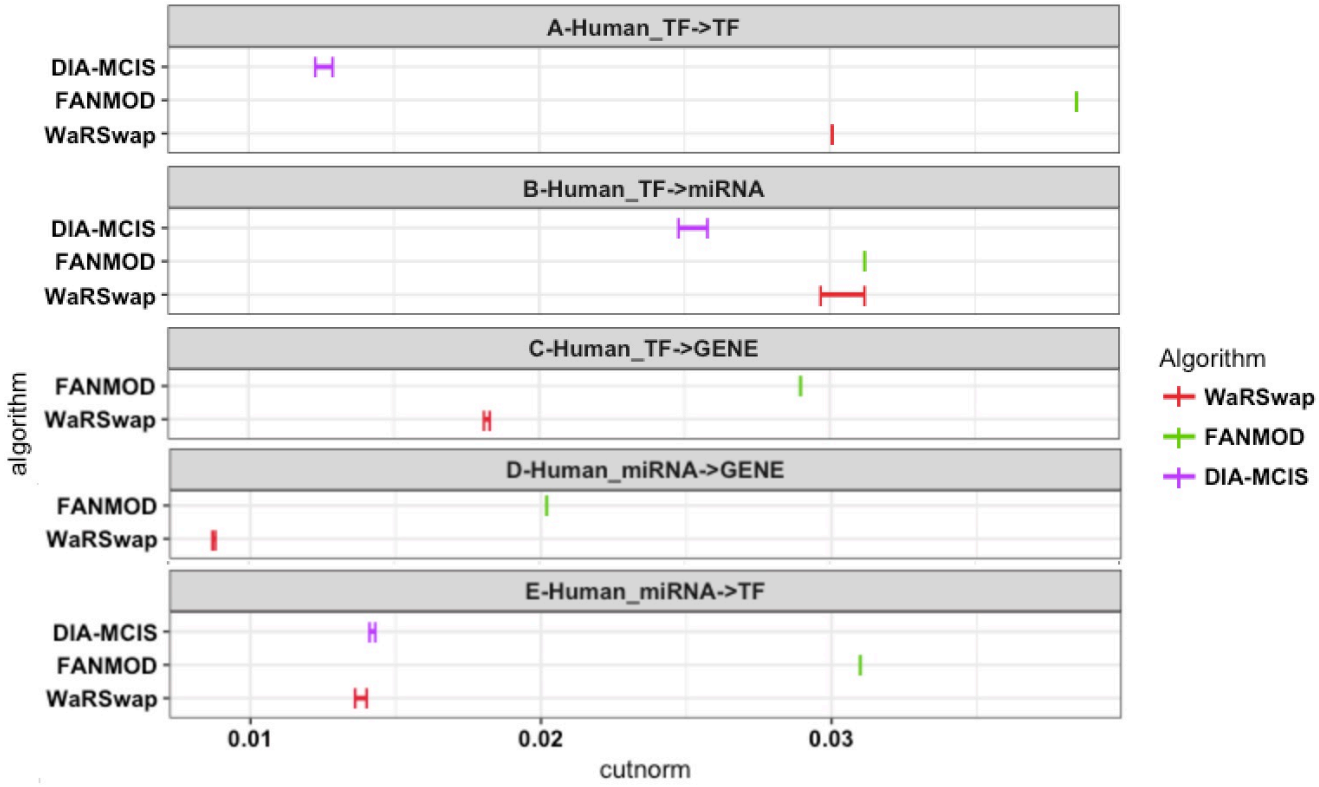

Figure 3.5: Human TF-miRNA-Gene network sampling performance. A total of 5000 graphs were generated by each algorithm and the cut norm estimates were computed using IndeCut. The vertical lines represent lower and upper bounds returned by the cut norm estimation with the true (NP-hard) value lying in this interval. A cut norm interval that is far from zero represents less uniform and independent sampling. The cut norm estimates for CoMoFinder were much larger than 0.04, and hence were removed for ease of comparison. In panels C and D, results for DIA-MCIS are absent since this algorithm does not operate on graphs with more than 2,035 nodes.

the considered algorithms. This highlights the importance of evaluating a network motif discovery algorithm on a network of interest, particularly when considering costly and time-consuming experimental validations. These results indicate that at least one algorithm for every topology was able to achieve strong performance (near-uniform sampling performance). However, some graphs caused major performance difficulties for one or more algorithms. This highlights the importance of evaluating the performance of network motif discovery algorithms on biological networks of interest, particularly when considering costly and time-consuming experimental validations.

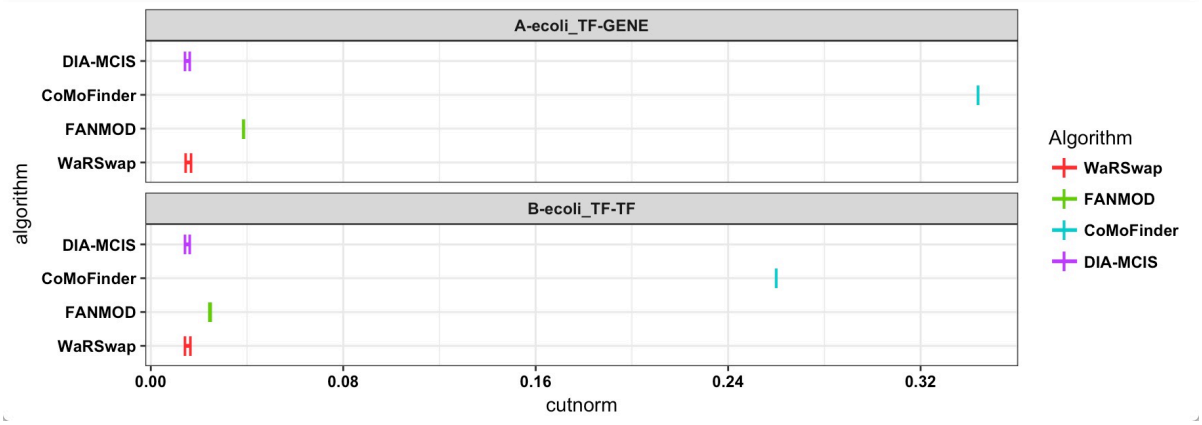

Figure 3.6: Graph sampling performance evaluation on Ecoli network using IndeCut.
This figure shows the cut norm estimates for all four examined algorithms: WaRSwap, CoMoFinder, DIA-MCIS, and FANMOD. For each graph and algorithm, 5000 graphs were generated. The cut norm estimates for each algorithm were computed using IndeCut. The vertical lines represent lower and upper bounds returned by the cut norm estimation with the true (NP-hard) value lying in this interval. A cut norm interval that is far from zero represents less uniform and independent sampling.

IndeCut evaluates graphs on the order of several thousand nodes and tens of thousands of edges. IndeCut presents a very practical method for making an informed network motif discovery algorithm choice on biological networks of study. Table 3.1 provides IndeCut's observed run time on each graph and algorithm. Table 3.1 provides IndeCut's run time on each graph and algorithm, which on smaller networks is no more than 5 minutes but increases considerably as the network size increases.

3.2.2 IndeCut indicates the number of samples required to achieve reproducible results

For a very large network with hundreds or thousands of nodes and edges, running a network motif discovery program − even with the minimum number of samples recommended in the user manual − generally takes days to month. To date, there has been no method to provide any indication of the number of sample graphs necessary for a reproducible result, but we demonstrate that IndeCut can be used for this purpose.

In general, the larger the number of graphs sampled, the more accurately a program can "characterize" the nature of the entire background network sample space, leading to better performance. Within a certain range of sample sizes, adding more graphs to a sample may result in a large performance increase. However, it is expected that beyond a certain sample size, performance increase per additional graph sampled will start to plateau (reach a point of diminishing returns). Here

we use IndeCut to evaluate how the performance of a sampling algorithm improves as the number of graphs in a sample increases. We examine where a performance plateau occurred for each graph and algorithm. Furthermore, we provide an example from the literature that illustrates the advantage of using IndeCut in this fashion.

These results show that overall, the approximate number of samples that is needed to approach best possible performance for each algorithm varies based on the topological features of the examined graph. For some algorithms, good performance is achievable even on topologies with extremely uneven degree sequences, but as expected, in all cases this comes at a cost of an increased number of samples. On the other hand, for more even degree sequences, a large number of samples is sometimes unnecessary.

We selected a published work [54] that reports network motifs in a Drosophila regulatory network. The authors have used FANMOD [69] to detect enriched 3-node network motifs in 100 sampled networks. To examine the reproducibility of the reported motifs, we ran FANMOD on the original network 50 times, where for each time, 5000 samples were generated and the significance of 3-node subgraphs was computed for different subsets of samples (100, 200, 300, ..., 5000 samples). Those 3-node subgraphs with p-value less than 0.01 and Z-score greater than 2.0 were considered in our analysis to be network motifs (threshold were not reported in the original publication of [54]. The performance plot in Fig. 3.7 shows that even for a moderately sized and relatively even graph such as the TF$\rightarrow$TF layer extracted from the original network, at least $\approx$ 1000 samples are required to reach

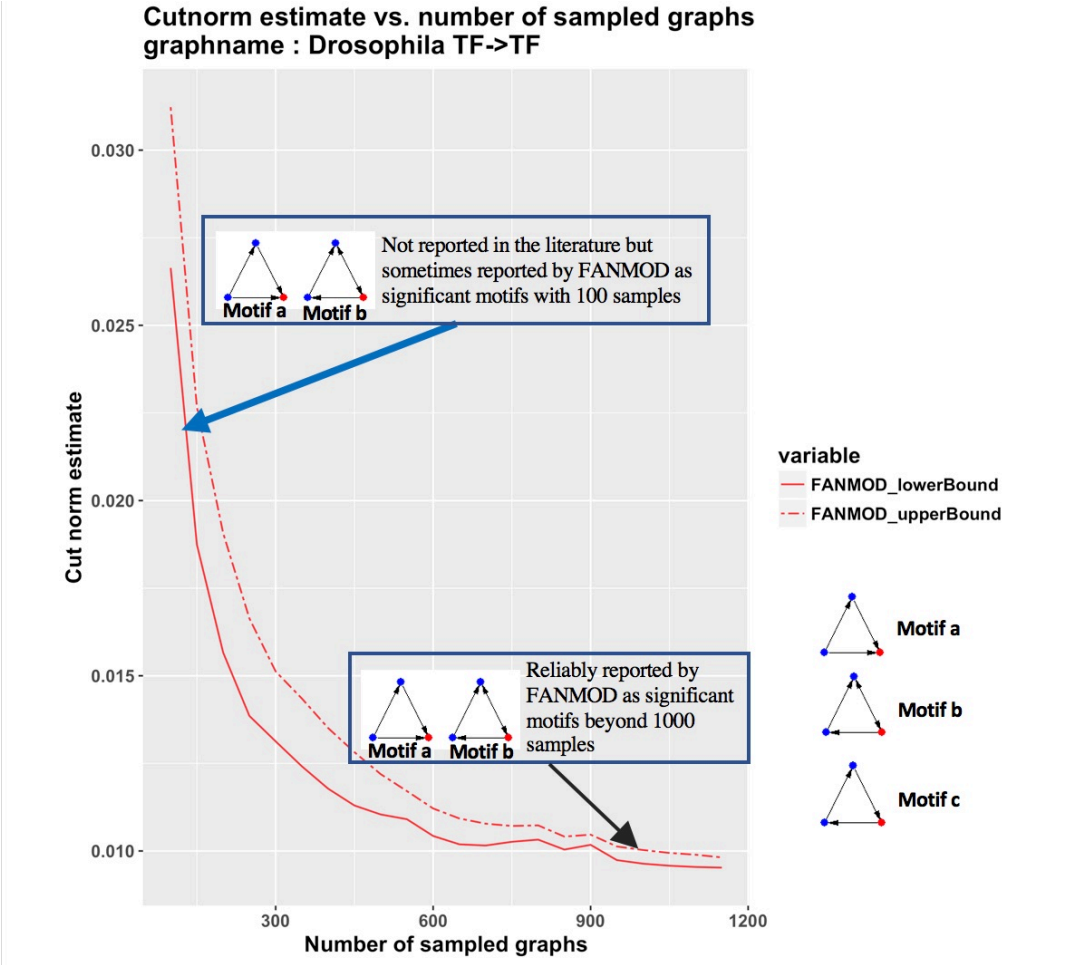

Figure 3.7: The relationship between cut norm estimates, number of samples and network motif outcome on Drosophila network.
FANMOD was run on the Drosophila network [54] for 50 iterations. Motifs a and b are not reported in [54]. These motifs were both found to be significant in a small proportion of trials at 100 sampled graphs (blue arrow). Motifs a and b are reliably detected beyond 1000 samples. Motif c was reported in [54] but was never observed as significant in any trials.

a performance level that is close to the best possible performance of the algorithm. However, taking only 100 samples as in the original analysis of [54] can lead to motifs being reported as significant due only to the relatively few number of graphs that were sampled. Indeed, in our results, motif5 in Fig7B of [54] (motif c shown here in Fig. 3.7) was never observed in any iteration. This is likely due to the relatively low number of iterations (100 iterations) that were used to run FANMOD in the original analysis. We also observed two significant network motifs at higher iterations, both of which were missed in the original work. Fig. 3.7 shows that with more than 1000 samples these two motifs are detected consistently, but with a smaller number of samples they do not reliably appear.

We used IndeCut to compute the relationship between the number of samples and the sampling performance of FANMOD on this network. The performance plot in Fig. 3.7 shows that even for a moderately sized and relatively even graph such as the TF→TF layer extracted from the original network, at least ≈ 1000 samples are required to reach a performance that is close to the best possible performance of the algorithm. However, taking only 100 samples in the original analysis [54] is disturbingly vulnerable to reporting motifs that are artifacts of insufficient sampling, as well as to missing highly significant network motifs.

It is completely understandable that given the long run-times required by many motif finding software implementations and no guidance on sufficient sampling, a relatively small number of samples was chosen by [54]. However, our results show the importance of making an informed choice —enabled now via IndeCut —for the number of background sample graphs required for each algorithm and input net-

work. In large real-world biological networks, we observe that a "blanket policy" of generating a fixed number of graphs may not achieve reasonable performance for a given algorithm and graph topology (for example, FANMOD in its user manual recommends ≈ 1000 samples whereas Figure 3.8 shows that at least 2000 samples are required to achieve reasonable performance in the considered case). We demonstrated that even for small to medium sized networks, understanding the appropriate number of samples is essential to attaining an accurate and reproducible outcome for each network and algorithm.

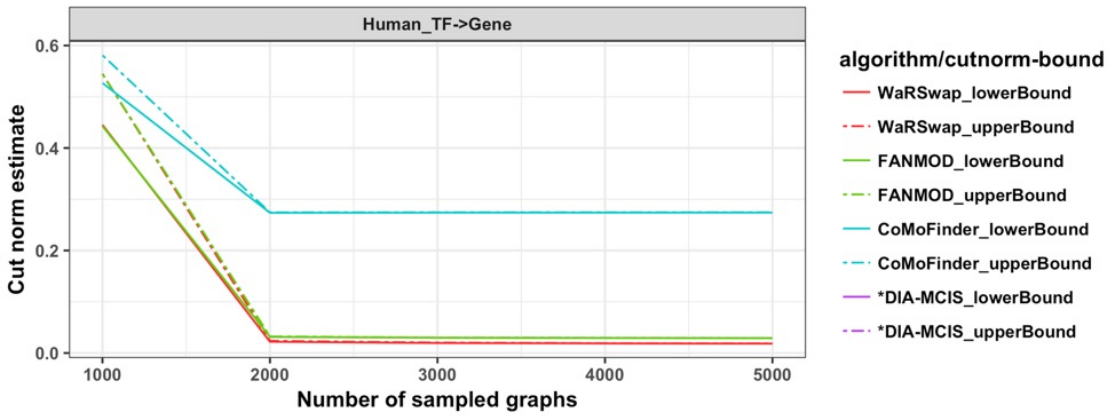

Figure 3.8: Relationship between the number of samples vs. sampling performance for Human TFGene network.
All 5000 samples previously generated by each algorithm for the Human TFGene network were collected and subsampled into five sets (1000, 2000, , 5000 samples in each set, respectively). IndeCut was used to compute the cut norm estimates (lower and upper bounds) for each set of samples and algorithms. Cut norm values closer to zero represent a more uniform/independent sampling. This network has 9,055 nodes and 25,748 edges. *The cut norm estimates for DIA-MCIS are absent because this algorithm is not able to operate on networks with more than 2,035 nodes.

IndeCut provides a practical solution for researchers using network motif discovery packages. Even if the initial package chosen is deemed inappropriate for accuracy given run-time, an alternative package or parallelized algorithm version can be evaluated.

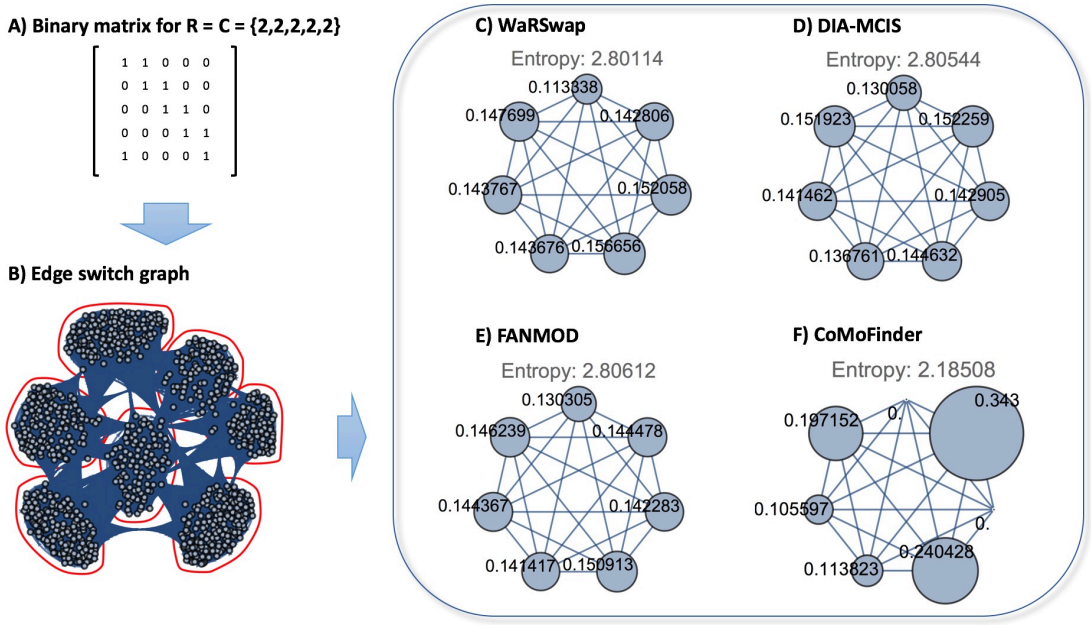

Figure 3.9: The ESG graph and cluster-time diagrams for an example even graph. A) The zero-one matrix representation of an even graph with degree sequence of R=C={2,2,2,2,2}. B) The ESG graph corresponding to the graph in part A. Running the graph clustering algorithm on the ESG graph detects seven different clusters. C-F) The cluster-time diagrams for each examined algorithm were computed and visualized.

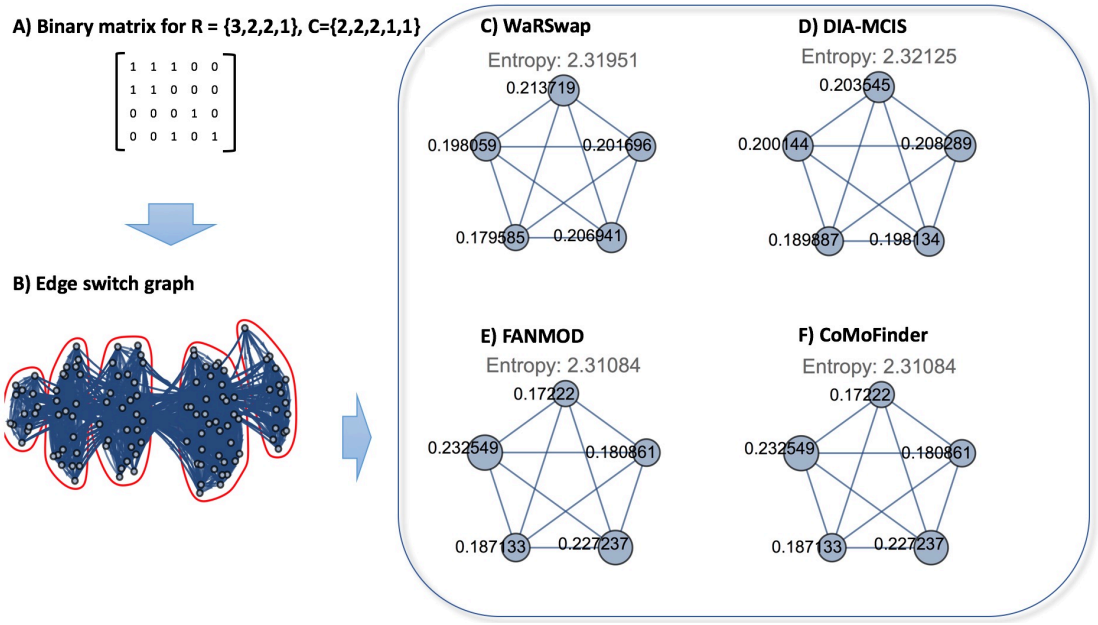

Figure 3.10: The ESG graph and cluster-time diagrams for an example hybrid graph.
A) The zero-one matrix representation of an uneven graph with degree sequence of R={3,2,2,1}, C={2,2,2,1,1}. B) The ESG graph corresponding to the graph in part A. Running the graph clustering algorithm on the ESG graph detects five different clusters. C-F) The cluster-time diagrams for each examined algorithm were computed and visualized.

3.2.3 Explaining performance differences found by IndeCut

In this section, we aim to explain the performance differences among the motif finding algorithms that we found with IndeCut in Section 3.2.1. In particular, we use the performance outcomes from IndeCut to analyze why certain graph topologies have been historically challenging for some classes of algorithms. We show that in cases of graphs with uneven degree distributions (characteristic of biological

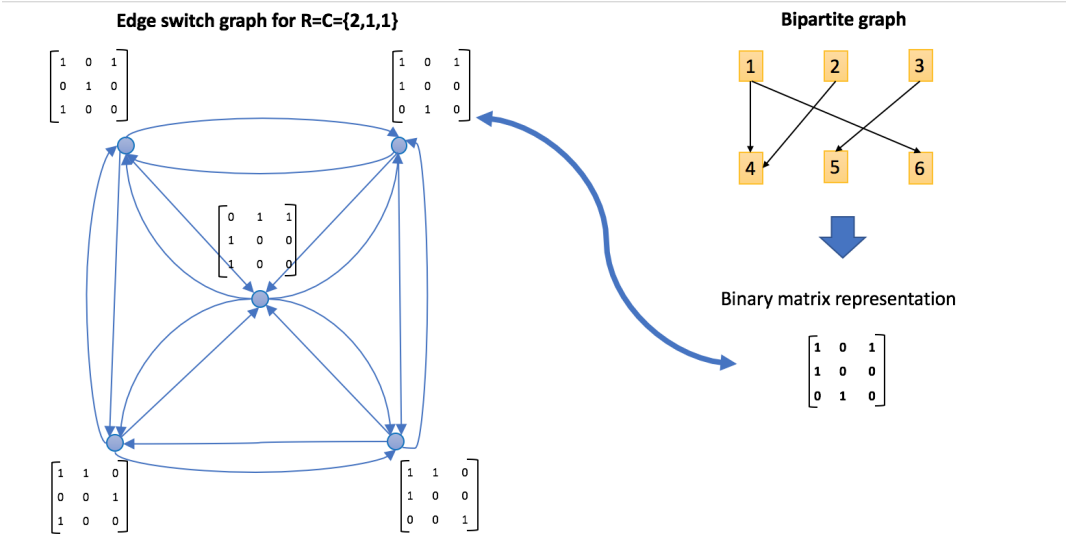

Figure 3.11: Constructing an ESG.
An initial bipartite graph (top right) with degree sequence of $R = C = \{2, 1, 1\}$ produces a sample space containing five different graphs which are represented as nodes in the ESG (left). The zero-one matrices represent the edge configuration of each node. An edge connects two nodes (graphs) which can be converted to each other by performing one edge switch.

networks), network motif discovery algorithms based on the graph randomization strategy known as "edge-switching" are vulnerable to highly non-uniform and/or non-independent sampling. Thus, this strategy is prone to spurious results on these networks. We use the concept of an edge-switching graph (ESG) to show why this is the case. In essence, edge-switching algorithms produce a sampling bias by spending a majority of time sampling graphs that can be reached from the starting graph via a small number of edge-switches. Fig. 3.11 depicts the construction of a 5-node ESG given a degree sequence of $R = C = \{2, 1, 1\}$.

Fig. 3.12B shows an ESG constructed from an in-degree sequence of $R = \{2, 1, 1, 2, 1, 1\}$ and an out-degree sequence of $C = \{2, 1, 1, 2, 1, 1\}$. The sample

space of this graph has 5400 elements. After running a graph clustering algorithm, 10 separate clusters were detected in the ESG. We executed each algorithm on the given degree sequence to produce 10,000 sample graphs per algorithm. We then calculated the number of times each algorithm returned a graph falling within each of the clusters (normalized by cluster size). This indicates how each of the examined algorithms samples its space with respect to these clusters (an equal number of graphs sampled within each cluster indicates a more uniform sampling method). Fig. 3.12C-F shows "cluster-time" diagrams, visualizing how much time each algorithm has spent in each cluster. In a cluster-time diagram, nodes represent clusters (in the ESG), and the size of each node represents the fraction of graphs sampled in the corresponding cluster compared to all graphs sampled (i.e. the total "time" the algorithm spends in a given cluster). The larger a node appears, the more time that has been spent sampling from the associated cluster by a given algorithm. A method that samples uniformly will result in a cluster-time diagram with nearly equal node sizes. We take the fraction of time spent in each cluster and compute the entropy to summarize the "evenness" of the sampling regime (larger numbers are better). The entropy value for each algorithm is noted above the corresponding cluster-time diagram in Fig. 3.12C-F (See Figures 3.9 and 3.10 for more examples).

Algorithms based on edge-switching, such as FANMOD and CoMoFinder, generally spend a substantially uneven amount of time in different clusters. Briefly, edge-switching algorithms start with the input graph (the original biological network) and then perform a series of edge-switching operations, resulting in one background graph in the sample. Each series of switching operations corresponds

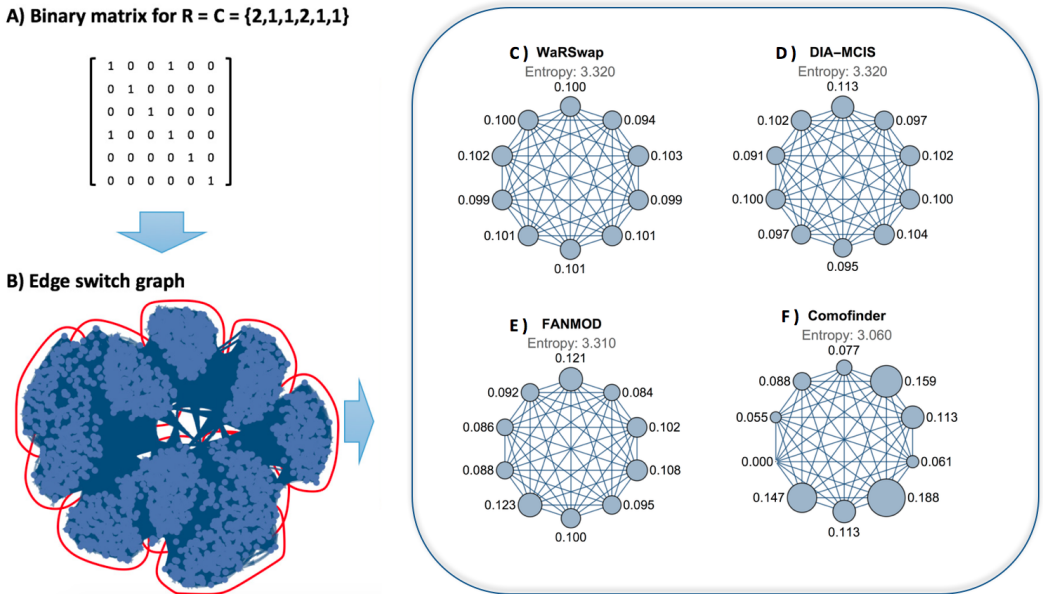

Figure 3.12: An example ESG for degree sequence $R = C = \{2, 1, 1, 2, 1, 1\}$. A) The 0-1 matrix of the initial graph. B) ESG corresponding to this degree sequence (blue dots represent graphs in sample space) with detected clusters outlined in red. C-F) Cluster-time diagrams for each examined algorithm; nodes represent clusters in the ESG with node size indicating the fraction of times a given algorithm sampled graphs in that cluster.

to a path in the ESG. In general, real biological networks have sample spaces in which some graphs in the space are "easy" to reach (few switch operations required) from the initial input network, while others are more "difficult" to reach in the sense that one must select a rare sequence of edge switches in order to reach these graphs.

FANMOD's strategy selects sequences of edge-switching operations without any condition on the number of times that the same pair of edges can be selected for a switch. CoMoFinder also selects sequences of edge-switching operations, but disallows revisiting the same pair of edges. Effectively, when traversing a path in an ESG away from the initial graph using CoMoFinder's strategy, the number of paths available to reach a destination graph from the current state is limited as compared to FANMOD's strategy.

WaRSwap and DIA-MCIS do not use edge-switching, but rather generate each sample graph by placing edges between source and target nodes using a weighted sampling scheme (thus there is no direct relationship between a sampled background graph and the initial input graph in terms of a path in the ESG). In Fig. 3.12, the size of the cluster nodes is nearly even for WaRSwap and DIA-MCIS, and the corresponding entropy values are higher as compared to FANMOD and CoMoFinder; hence the sampling is more uniform. However, there are performance differences between WaRSwap and DIA-MCIS on large hub-containing graphs (Fig. 3.1) that result from different weighted sampling strategies. In the case of certain highly uneven graphs, the static weighting strategy of DIA-MCIS appears to be susceptible to undersampling of rare graphs (those with few/no hub-

hub connections). Overall, either DIA-MCIS or WaRSwap appear preferable to an edge-switching method on graphs containing uneven degree sequences.

3.3 Methods

3.3.1 Definitions

A graph $G = (V, E)$ is a structure describing the relationships between elements in a *vertex set* V through a set of (directed) *edges* $(v_i, v_j) \in E$ where $v_i, v_j \in V$. In this work, we define a network as a two-layered or *bipartite* graph G containing m source nodes $\{S_1, .., S_m\}$ and n target nodes $\{T_1, .., T_n\}$ where a single directed edge connects a source node to a target node. The number of edges coming into a node is called its *in-degree* and the number of edges coming out from a node is called its *out-degree*. In a bipartite graph G, source nodes and target nodes have zero in-degrees and zero out-degrees, respectively. The structure or topology of a bipartite graph G is described by its in-degree and out-degree sequences.

A bipartite graph G can be represented as a binary matrix $A \in \{0, 1\}^{m \times n}$. When $A_{i,j} = 1$, there is a directed edge from S_i to T_j, and $A_{i,j} = 0$ means there is no edge between them. The row sums $R = (r_1, \ldots, r_m)$ and column sums $C = (c_1, \ldots, c_n)$ of matrix A represent the out-degree and in-degree sequences of G, respectively. Collectively, they are referred to as the degree sequences of a graph. Hence, we have that $\sum_i A_{i,j} = C_j$ and $\sum_j A_{i,j} = R_i$.

3.3.2 How does IndeCut work?

This section provides a high-level summary of how IndeCut works, with more mathematical detail contained in Section 3.3.3. An ideal graph sampling strategy would produce samples from the set of all possible graphs (the *sample space*) that are perfectly uniform and independent. In the case of perfect sampling, each sample would have a sample average that is identical to the true average (mean of all elements in the sample space). From this perspective, violation of uniformity and independence can be quantified by measuring how far the *sample average* is from the *true average*. Fig. 3.13 provides an abstracted visualization of this concept illustrating how the distance between the sample average A and true average (centroid E) can be used to assess uniform and independent sampling. Sampled graphs, described mathematically as zero-one matrices, are represented as gray dots inside the sample space of all graphs with the prescribed in and out-degrees. The point E represents the centroid, or true average, of the entire sample space and the point A represents the average of sampled matrices for a hypothetical graph sampling strategy.Fig. 3.13A shows that a uniform and independent sampling method produces samples that are "evenly spread" over the sample space, resulting in a centroid matrix E and an average matrix A which are nearly identical. In the case of perfectly uniform and independent sampling, E and A are identical. In Fig. 3.13, a violation of independence (Fig. 3.13B) or uniformity (Fig. 3.13C) results in a difference between E and A. The greater the violation of uniformity and/or independence, the further the centroid E will be from the sample average A. Computing the exact

centroid E by empirically enumerating all graphs in the sample space is generally prohibitive because such spaces are astronomically large (for example, the space of 3-regular bipartite graphs with 10 source nodes and 10 target nodes has more than 10^{26} elements in it). Therefore instead of computing the exact centroid E, we use a related matrix, called the maximum entropy matrix and denoted by Z, which is known to be close to A in terms of its cut norm distance (Definition (Cut Norm) below) when the sampling regime is uniform and independent (this is proved in Theorem 3 of [5]). Thus, an ideal sampling method will have a zero cut norm for Z - A, the matrix representing the difference between Z and A. Similarly, a cut norm bounded significantly away from zero indicates that sampling is either highly non-uniform, highly non-independent, or both. Unfortunately, computing the cut norm for matrices of realistic size is intractable given today's computing hardware capability (MAX SNP-hard). We overcome this barrier by using the ideas of [1] to create an approximation algorithm that returns an interval in which the distance between Z and A is guaranteed to be contained. Comparing these intervals allows us to compare the uniformity and independence of graph sampling strategies.

Here, we present IndeCut as a practical method that determines the degree of uniformity/independence for a sampling method on a given bipartite graph G with fixed in-degrees and out-degrees (and no multiple edges). G is a simple graph wherein source nodes have direct links to the target nodes but not vice-versa. The sample space associated with graph G is defined as the set of all possible valid bipartite graphs that can be produced from G's degree sequence. Each bipartite graph in the sample space can be represented as a zero-one matrix with fixed row

and column sums R and C, respectively.

In summary, IndeCut, performs the following tasks: the sample average matrix A and maximum entropy matrix Z are computed, and then a (typically small) interval is computed along with a guarantee that the cut norm lies in this interval. As a consequence of Theorem 3 in [5], if the cut norm is large (bounded far from 0), then we can be sure that the sampling was not uniform and independent. The further this interval is bounded away from zero, the less uniform and independent the graph sampling technique is. Importantly, no additional information is required by IndeCut to assess a whether a graph sampling strategy performs uniform/independent sampling other than the input graph and sequence of graphs returned by the sampling method.

3.3.3 Mathematical details of IndeCut

Let $\Sigma(R, C)$ be the set of all binary matrices with row-sums $R = (r_1, \ldots, r_m) \in \mathbb{N}^m$ and column-sums $C = (c_1, \ldots, c_n) \in \mathbb{N}^n$. Throughout, we only consider R and C such that for every choice of $1 \leq i \leq m$ and $1 \leq j \leq n$, there exist at least two matrices $L, M \in \Sigma(R, C)$ such that $L_{i,j} = 0$ and $M_{i,j} = 1$. This condition requires the space $\Sigma(R, C)$ to be reasonably large.

We now recount a pertinent definition from [5].

Definition 2 ([5, Theorem 1])**.** *Let*

$$F(\mathbf{x}, \mathbf{y}) = \left(\prod_{i=1}^{m} x_i^{-r_i} \right) \left(\prod_{j=1}^{n} y_j^{-c_j} \right) \left(\prod_{i,j} (1 + x_i y_j) \right)$$

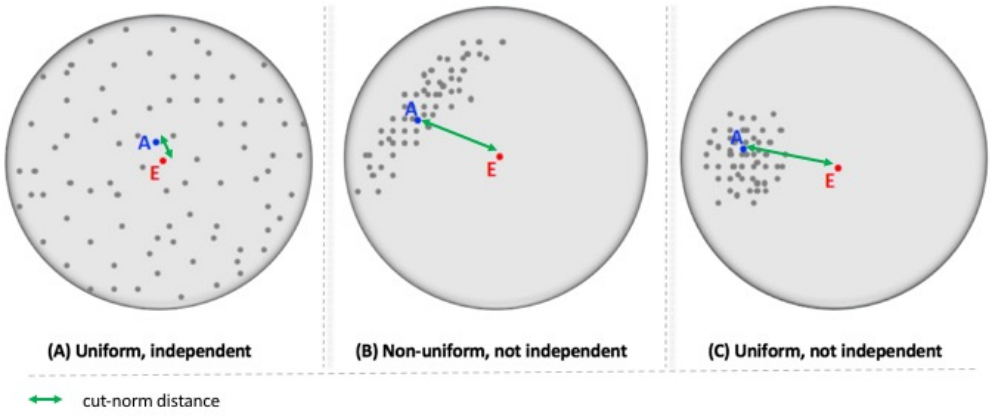

Figure 3.13: An illustrative view of graph sampling strategy outcomes in terms of uniformity and independence.
Each gray circle represents a hypothetical sample space of a graph. Sampled graphs which are the outcome of a hypothetical graph sampling strategy are represented as gray dots inside the sample space of all graphs with the prescribed in and out-degrees. The point A represents the sample average and the point E represents the centroid or true average of sample space. The distance (here characterized with the cut norm) between A and E indicates the degree of uniformity and independence of a produced sample. The further away A is from E, the more confident one can be that points are not sampled uniformly and independently.

for $\mathbf{x} = (x_1, \ldots, x_m)$ *and* $\mathbf{y} = (y_1, \ldots, y_n)$*, and let* $\alpha(R, C) = \underset{\mathbf{x},\mathbf{y}>0}{minimum} \quad F(\mathbf{x}, \mathbf{y})$.

Taking the logarithm of $F(\mathbf{x}, \mathbf{y})$ gives a convex function on $\mathbb{R}^{m \times n}$, so $\alpha(R, C)$ may be efficiently computed. This allows us to define the *maximum entropy matrix*:

Definition 3 (Maximum Entropy Matrix ([5, Lemma 2]))**.** *Let* $\mathbf{x}^*$ *and* $\mathbf{y}^*$ *be the vectors that obtain optimality in the definition of* $\alpha(R, C)$*. Define* $Z \in \mathbb{R}^{m \times n}$ *as*

$$Z_{i,j} = \frac{\mathbf{x}_i^* \mathbf{y}_j^*}{1 + \mathbf{x}_i^* \mathbf{y}_j^*}. \tag{3.1}$$

Ideally, we would not need Z and would have access to the true centroid $E_{i,j} = \frac{1}{|\Sigma(R,C)|} \sum_{M \in \Sigma(R,C)} M_{i,j}$ and this would be compared with the sample average of matrices returned by a motif finding algorithm. Unfortunately, the matrix E is computationally intractable to calculate and there appears to be no way to obtain tight estimates of its entries. In contrast, the matrix Z can be computed to arbitrary precision in an efficient fashion, and Theorem 3 of [5] states that the sample averages are close to Z in terms of the cut norm (see the Material Section Theorem 3 for a rigorous statement to this effect). We can thus leverage this result to use the cut norm and Z to test for violation of uniform/independent sampling.

Definition 4. *Let* $A \in \mathbb{R}^{m \times n}$*. The cut norm is defined by*

$$||A||_{\mathcal{C}} = \underset{\substack{I \subseteq \{1,\ldots,n\} \\ J \subseteq \{1,\ldots,m\}}}{maximize} \quad \left| \sum_{i \in I, j \in J} A_{i,j} \right| \tag{Cut Norm}$$

Let $\mathcal{A}$ represent a given motif finding algorithm (thought of as a binary matrix

valued random variable). Let $(A_i)_{i=1}^N$ be N iterates of this algorithm and define

$$A_{(N)} = \frac{1}{N}\sum_{i=1}^{N} A_i. \tag{3.2}$$

If the sequence $(A_i)_{i\geq 1}$ is a realization of a sequence of independent and uniformly distributed random matrices, then Theorem 3 implies that, with high probability, the norm $||Z - A_{(N)}||_{\mathcal{C}}$ is small. Arguing contrapositively, a large norm implies too few samples were taken (N is small) or else the sampling was not uniform or not independent. We can thus use $||Z - A_{(N)}||_{\mathcal{C}}$ as a measure of the non-uniformity/independence of a motif finding algorithm $\mathcal{A}$: For large N, if one algorithm outputs matrices whose average is closer in the cut norm to Z than that of another algorithm, then the latter algorithm samples the space $\Sigma(R, C)$ in a less uniform/independent fashion.

Unfortunately, computing the cut norm is MAX SNP-hard. However, it is possible to obtain easy to compute upper and lower bounds on the cut norm and the same logic as above applies when comparing these intervals. In particular, we bound the cut norm above by $\frac{1}{4}||\cdot||_{\text{SDR}}$ and below by $\frac{1}{4}||\cdot||^{\text{est}}_{\infty\mapsto 1}$ where $||A||_{\text{SDR}} = \underset{||u_i||_2=||v_j||_2=1}{\text{maximize}} \sum_{i,j} A_{i,j}(u_i \cdot v_j)$ and $||A||^{\text{est}}_{\infty\mapsto 1}$ is the value returned by Algorithm 3.3.3.3. Hence, IndeCut returns an interval estimating the relative cut norm:

$$\textsf{IndeCut}(Z, \mathcal{A}, N) = \left(\frac{||Z - A_{(N)}||^{\text{est}}_{\infty\mapsto 1}}{4||Z||_{\mathcal{C}}}, \frac{||Z - A_{(N)}||_{\text{SDR}}}{4||Z||_{\mathcal{C}}}\right).$$

Note that $||Z||_{\mathcal{C}}$ is straightforward to calculate as all entries of Z are nonnegative.

Finally, while IndeCut uses bipartite graphs to evaluate motif finding algorithm performance, as long as the graphs under consideration can be partitioned into bipartite subgraphs (consisting of layers such as TF-¿TF, TF-¿miRNA, etc.) as is typically the case for genomic networks, IndeCut can evaluate the performance on each layer. Non-uniform sampling on any one such layer implies non-uniform sampling overall.

In this section, we give the mathematical details necessary to support the claim that the cut norm and maximum entry matrix can be used to test for non-uniformity of a bipartite graph sampling algorithm. We begin by recalling the results of [5] that we use and then derive bounds necessary to estimate the cut norm.

3.3.3.1 Results from Barvinok [5]

Let $\Sigma(R, C)$ be the set of all binary matrices with row-sums $R = (r_1, \ldots, r_m) \in \mathbb{N}^m$ and column-sums $C = (c_1, \ldots, c_n) \in \mathbb{N}^n$. Let $\mathcal{P}(R, C)$ be the polytope of matrices with entries bounded between 0 and 1 and with row and column sums R and C respectively. Throughout, we only consider R and C such that for every choice of $1 \leq i \leq m$ and $1 \leq j \leq n$, there exist at least two matrices $L, M \in \Sigma(R, C)$ such that $L_{i,j} = 0$ and $M_{i,j} = 1$. This condition requires the space $\Sigma(R, C)$ to be reasonably large (i.e. the polytope $\mathcal{P}(R, C)$ is non-empty).

We now recount pertinent theorems from [5]. The first gives an estimate of the

number of bipartite graphs with degree sequences R and C: $|\Sigma(R,C)|$.

Theorem 1 ([5, Theorem 1])**.** *Let*

$$F(\mathbf{x},\mathbf{y}) = \left(\prod_{i=1}^{m} x_i^{-r_i}\right)\left(\prod_{j=1}^{n} y_j^{-c_j}\right)\left(\prod_{i,j}(1+x_i y_j)\right)$$

for $\mathbf{x} = (x_1,\ldots,x_m)$ *and* $\mathbf{y} = (y_1,\ldots,y_n)$*. Let*

$$\alpha(R,C) = \underset{\mathbf{x},\mathbf{y}>0}{minimum} \quad F(\mathbf{x},\mathbf{y}).$$

Then

$$\alpha(R,C) \geq |\Sigma(R,C)| \geq \frac{(mn)!}{(mn)^{mn}}\left(\prod_{i=1}^{m}\frac{(n-r_i)^{n-r_i}}{(n-r_i)!}\right)\left(\prod_{j=1}^{n}\frac{c_j^{c_j}}{c_j!}\right)\alpha(R,C).$$

Taking the logarithm of $F(\mathbf{x},\mathbf{y})$ gives a convex function on $\mathbb{R}^{m\times n}$, so $\alpha(R,C)$ may be efficiently computed. This allows us to define the *maximum entropy matrix*:

Definition 5 ([5, Lemma 2])**.** *Let* $\mathbf{x}^*$ *and* $\mathbf{y}^*$ *be the vectors that obtain optimality in the definition of* $\alpha(R,C)$*. Define* $Z \in \mathbb{R}^{m\times n}$ *as*

$$Z_{i,j} = \frac{\mathbf{x}_i^*\mathbf{y}_j^*}{1+\mathbf{x}_i^*\mathbf{y}_j^*}. \tag{3.3}$$

The cut norm is needed to state the next theorem.

Definition 6. *Let $A \in \mathbb{R}^{m\times n}$ and let*

$$||A||_{\mathcal{C}} = \underset{\substack{I\subseteq\{1,\ldots,n\}\\ J\subseteq\{1,\ldots,m\}}}{maximize} \left|\sum_{i\in I, j\in J} A_{i,j}\right|. \qquad \text{(Cut Norm)}$$

Let $S \subset \{(i,j) : i = 1,\ldots,m,\ j = 1,\ldots,n\}$ be a set of indicies. For a $m\times n$ matrix A, let

$$\sigma_S(A) = \sum_{(i,j)\in S} A_{ij}.$$

Note that $||A||_{\mathcal{C}} = \max_S |\sigma_S(A)|$.

We can now state the main result that serves as the justification of IndeCut. Recall our assumption that $|\Sigma(R,C)| \geq 2$ and so $\mathcal{P}(R,C)$ is non-empty.

Theorem 2 ([5, Theorem 3])**.** *Fix numbers $\kappa > 0$ and $0 < \delta < 1$ then there exists a number $q = q(\kappa,\delta)$ such that the following holds. Let R and C be such that $n \geq m > q$ and let $Z \in \mathcal{P}(R,C)$ be the maximum entry matrix. Let $S \subset \{(i,j) : i = 1,\ldots,m,\ j = 1,\ldots,n\}$ be such that $\sigma_S(Z) \geq \delta mn$ and let $\epsilon = \delta\frac{\ln n}{\sqrt{M}}$. If $\epsilon \leq 1$, then*

$$\mathbb{P}\left\{D \in \Sigma(R,C) :\ (1-\epsilon)\sigma_S(Z) \leq \sigma_S(D) \leq (1+\epsilon)\sigma_S(Z)\right\} \geq 1 - 2n^{-\kappa n}.$$

This theorem states that a uniformly sampled binary matrix is close to the maximum entry matrix in terms of the cut norm.

We now re-state this result in terms of the cut norm.

Theorem 3. *Let $Z \in \mathcal{P}(R,C)$ be the maximum entry matrix. Let $(A_i)_{i=1}^N$ be a*

sequence of independent and uniformly distributed random variables on $\Sigma(R,C)$ *and let* $A_{(N)} = \frac{1}{N}\sum_{i=1}^{N} A_i$. *Let* $S \subset \{(i,j) : i = 1,\ldots,m,\ j = 1,\ldots,n\}$ *be such that* $|\sigma_S(Z - A_{(N)})| = ||Z - A_{(N)}||_{\mathcal{C}}$. *Let* $0 < \delta < 1$ *be such that for this* S, $\sigma_S(Z) \geq \delta mn$ *and let* $\epsilon = \delta\frac{\ln n}{\sqrt{M}}$. *Fix* $\kappa > 0$, *then there exists a number* $q = q(\kappa,\delta)$ *such that if* R *and* C *are such that* $n \geq m > q$, *the following holds: If* $\epsilon \leq 1$, *then*

$$\mathbb{P}\left(\frac{\left|\left|\frac{1}{N}\sum_{i=1}^{N} A_i - Z\right|\right|_{\mathcal{C}}}{||Z||_{\mathcal{C}}} \leq \epsilon\right) \geq 1 - 2n^{-\kappa n}. \tag{3.4}$$

Proof. Assuming that

$$(1-\epsilon)\sigma_S(Z) \leq \sigma_S\left(A_{(N)}\right) \leq (1+\epsilon)\sigma_S(Z),$$

since $||Z||_{\mathcal{C}} = \max_{S'} |\sigma_{S'}(Z)|$, this implies that

$$(1-\epsilon)||Z||_{\mathcal{C}} \leq \sigma_S\left(A_{(N)}\right) \leq (1+\epsilon)||Z||_{\mathcal{C}}. \tag{3.5}$$

By hypothesis, $|\sigma_S(Z - A_{(N)})| = ||Z - A_{(N)}||_{\mathcal{C}}$ and so along with linearity of $\sigma_S(\cdot)$, equation (3.5) implies that

$$||A_{(N)} - Z||_{\mathcal{C}} \leq \epsilon||Z||_{\mathcal{C}}.$$

Monotonicity of probability and the conclusion of Theorem 2 then imply that

$$\mathbb{P}\left(\frac{\left|\left|\frac{1}{N}\sum_{i=1}^{N} A_i - Z\right|\right|_{\mathcal{C}}}{||Z||_{\mathcal{C}}} \leq \epsilon\right) \geq 1 - 2n^{-\kappa n}.$$

□

Given appropriate R, C, κ, δ, and ϵ, the contrapositive of this result implies that if $\frac{\left|\left|A_{(N)}-Z\right|\right|_{\mathcal{C}}}{||Z||_{\mathcal{C}}}$ is large, then there is an exponentially small chance that the sequence of random variables is independent and uniformly distributed. This is the justification to use the quantity

$$\frac{\left|\left|A_{(N)} - Z\right|\right|_{\mathcal{C}}}{||Z||_{\mathcal{C}}}$$

as a measure of non-uniformity/independence and forms the mathematical justification of IndeCut. We turn now to looking at how to calculate this quantity in practice.

3.3.3.2 Computing norms

The cut norm $||\cdot||_{\mathcal{C}}$ is difficult to compute (in fact, it is MAX SNP hard [1] for general matrices, but we will be able to relate it to another norm ($||\cdot||_{\infty\mapsto 1}$) that can be approximated with a semidefinite relaxation. We then round the solution of the semidefinite relaxation to get an estimate of $||\cdot||_{\infty\mapsto 1}$ and hence of $||\cdot||_{\mathcal{C}}$. We begin with definitions of the norms of interest.

Definition 7. *Let $A \in \mathbb{R}^{m\times n}$. Define the following norms by*

$$||A||_{\infty\mapsto 1} = \underset{\substack{x_i\in\{-1,+1\}\\ y_j\in\{-1,+1\}}}{maximize} \quad \sum_{i,j} A_{i,j}x_iy_j \qquad (\infty \mapsto 1 \text{ Norm})$$

We denote the semidefinite relaxation of $||A||_{\infty\mapsto 1}$ by $||A||_{\text{SDR}}$:

$$||A||_{\text{SDR}} = \underset{||u_i||_2=||v_j||_2=1}{\text{maximize}} \quad \sum_{i,j} A_{i,j}(u_i \cdot v_j) \qquad (\text{SDR Norm})$$

Note that $||A||_{\text{SDR}}$ can be converted to the following optimization problem:

$$\begin{aligned} ||A||_{\text{SDR}} = \frac{1}{2}\underset{X}{\text{maximize}} \quad & \text{tr}(CX) \\ \text{subject to} \quad & \text{tr}(F_kX) = a_k,\ k = 1,\dots,m+n \\ & X \succeq 0, \end{aligned} \tag{3.6}$$

for

$$C = \begin{bmatrix} 0 & A \\ A & 0 \end{bmatrix},\ F_k = \begin{cases} 1 & \text{if } i=j=k \\ 0 & \text{o.w.} \end{cases},\ \text{and } a_k = 1 \text{ for } k = 1,\dots,m+n.$$

This form allows us to use popular computational packages to compute $||\cdot||_{\text{SDR}}$. We utilize the computational package CSDP version 6.1.0 [10].

It turns out that for the matrices of interest, the norms $||\cdot||_C$ and $||\cdot||_{\infty\mapsto 1}$ are equal up to a factor of 4. Indeed, note the maximum entropy matrix Z defined in equation (3.3) and $A_{(N)}$ defined in the previous section both have row/column sums equal to R and C: $A_{(N)}, Z \in \Sigma(R,C)$. Hence the matrix $Z - A_{(N)}$ has zero row and column sum. This allows us to obtain the well-known [1, 31] relationship

between the norms $||\cdot||_{\infty\mapsto 1}$ and $||\cdot||_{\mathcal{C}}$.

Proposition 4. *If the matrix A has zero row and column sums (i.e. $\sum_i A_{i,j} = \sum_j A_{i,j} = 0$), then $||A||_{\infty\mapsto 1} = 4||A||_{\mathcal{C}}$.*

Proof. For $I \subseteq \{1,\dots,n\}$ and $J \subseteq \{1,\dots,m\}$ the sets achieving the maximum in the definition of $||A||_{\mathcal{C}}$, define $x_i = 1$ for $i \in I$, $x_i = -1$ for $i \notin I$ and $y_j = 1$ for $j \in J$, $y_j = -1$ for $j \notin J$. Then

$$\begin{aligned}
||A||_{\mathcal{C}} &= \sum_{i,j} A_{i,j} \frac{1+x_i}{2} \frac{1+y_j}{2} \\
&= \frac{1}{4}\left(\sum_{i,j} A_{i,j} + \sum_{i,j} A_{i,j} x_i + \sum_{i,j} A_{i,j} y_j + \sum_{i,j} A_{i,j} x_i y_j \right) \\
&= \frac{1}{4} \sum_{i,j} A_{i,j} x_i y_j \\
&= \frac{1}{4} ||A||_{\infty\mapsto 1}.
\end{aligned}$$

□

3.3.3.3 Cut norm estimates

In [1, Section 5.1], an algorithm was presented that computes bounds on $||\cdot||_{\infty\mapsto 1}$. We use a slight modification of this algorithm that gives tighter bounds in practice as follows:

Given a matrix A, let $u_i, v_j \in \mathbb{R}^{m+n}$, for $i = 1,\dots,m, j = 1,\dots,n$ be the optimal vectors obtained from the computation of $||A||_{\text{SDR}}$. Let $g_i \sim \mathcal{N}(0,1)$, $i = 1,\dots,m+n$, be independent standard normal random variables and let $G =$

$(g_1, \ldots, g_{m+n})$. Let $x_i = \text{sign}(u_i \cdot G)$ and $y_j = \text{sign}(v_j \cdot G)$.

Now $\sum_{i,j} A_{i,j} x_i y_j \leq ||A||_{\infty \mapsto 1}$ since $||A||_{\infty \mapsto 1}$ is the maximum value. However, there is a positive probability that $\sum_{i,j} A_{i,j} x_i y_j = ||A||_{\infty \mapsto 1}$. To observe this fact, let $x_i^*, y_j^* \in \{-1, +1\}$ be such that $||A||_{\infty \mapsto 1} = \sum_{i,j} A_{i,j} x_i^* y_j^*$. We can find at least one vector G^* such that $x_i^* = \text{sign}(u_i \cdot G^*)$ and $y_j^* = \text{sign}(v_j \cdot G^*)$ since this reduces to solving a solvable system of linear inequalities due to the u_i, v_j being obtained from eigenvectors of the spectral factorization of X in the optimization procedure (3.6). Given such a G^*, note that for any $a \in \mathbb{R}$, $a > 0$, $x_i^* = \text{sign}(u_i \cdot aG^*)$ and $y_j^* = \text{sign}(v_j \cdot aG^*)$. Hence, with probability at least 2^{-m-n}, a randomly chosen G will result in obtaining the optimal x_i^* and y_j^*. We do not attempt to make a more nuanced estimation of this probability since only bounds are necessary for our purposes.

Repeating the above rounding procedure a number of times and taking the maximum result, we obtain Algorithm 1 which we use to compute the bounds on the cut norm of a matrix A. In practice, we take the number of iterates of Algorithm 1 to be $1,000$. Denote the output of this algorithm with $||A||_{\infty \mapsto 1}^{\text{est}}$. As a result, $||A||_{\infty \mapsto 1}^{\text{est}} \leq ||A||_{\infty \mapsto 1} \leq ||A||_{\text{SDR}}$, so combining these with proposition 4, we have the following estimation of the cut norm:

$$\frac{1}{4}||A||_{\infty \mapsto 1}^{\text{est}} \leq ||A||_{\mathcal{C}} \leq \frac{1}{4}||A||_{\text{SDR}}. \tag{3.7}$$

We apply this estimation to obtain bounds on the quantity of interest:

$$\frac{\left|\left|A_{(N)} - Z\right|\right|_{\mathcal{C}}}{4||Z||_{\mathcal{C}}}. \tag{3.8}$$

We can compare motif finding algorithms in the following fashion: Let $(A_i)_{i=1}^N$ and $(B_i)_{i=1}^N$ be N random binary matrices generated by two algorithms $\mathcal{A}$ and $\mathcal{B}$. If the upper bound for one algorithm (say, $\mathcal{A}$) falls below the lower bound of the other algorithm (say, $\mathcal{B}$), then we can be sure that the cut norm quantity of interest for $\mathcal{A}$ is smaller than for $\mathcal{B}$. As a consequence of the previous section, this implies that we can be more confident that $\mathcal{B}$ samples the space in a less uniform and independent fashion. If the bounds do not overlap, then no conclusion can be made since we cannot guarantee that one cut norm is larger than the other. More rigorously, let $A_{(N)} = \frac{1}{N}\sum_{i=1}^N A_i$ and $B_{(N)} = \frac{1}{N}\sum_{i=1}^N B_i$. If for sufficiently large N, we have that

$$||A_N - Z||_{\text{SDR}} < ||B_{(N)} - Z||_{\infty\mapsto 1}^{\text{est}}$$

then equation 3.7 implies that

$$\frac{\left|\left|A_{(N)} - Z\right|\right|_{\mathcal{C}}}{4||Z||_{\mathcal{C}}} < \frac{\left|\left|B_{(N)} - Z\right|\right|_{\mathcal{C}}}{4||Z||_{\mathcal{C}}}.$$

As a consequence of Theorem 3 and the discussion that followed, we can be confident that $\mathcal{B}$ samples the space $\Sigma(R, C)$ in a less uniform and independent fashion than $\mathcal{A}$. The symmetric case of overlapping bounds $||B_N - Z||_{\text{SDR}} < ||A_{(N)} - Z||_{\infty\mapsto 1}^{\text{est}}$ would imply the reverse conclusion being made about $\mathcal{A}$ and $\mathcal{B}$. If, however, the

bounds overlap:

$$\left(\frac{||Z - A_{(N)}||^{\text{est}}_{\infty\mapsto 1}}{4||Z||_{\mathcal{C}}}, \frac{||Z - A_{(N)}||_{\text{SDR}}}{4||Z||_{\mathcal{C}}}\right) \cap \left(\frac{||Z - B_{(N)}||^{\text{est}}_{\infty\mapsto 1}}{4||Z||_{\mathcal{C}}}, \frac{||Z - B_{(N)}||_{\text{SDR}}}{4||Z||_{\mathcal{C}}}\right) \neq \emptyset,$$

then no conclusion can be drawn as no information is provided about the relative sizes of the cut norms. Hence, we use the quantity:

$$\mathsf{IndeCut}(Z, \mathcal{A}, N) = \left(\frac{||Z - A_{(N)}||^{\text{est}}_{\infty\mapsto 1}}{4||Z||_{\mathcal{C}}}, \frac{||Z - A_{(N)}||_{\text{SDR}}}{4||Z||_{\mathcal{C}}}\right)$$

to compare uniformity/independence of motif finding algorithms.

3.3.4 Compute Relationship Between Number of Samples and Cut norm Estimates

Given the space of all sampled graphs produced by an algorithm $\{G_1, \ldots, G_n\}$, we generated m sets of samples $\{S_1, \ldots, S_m\}$ in which the set S_1 contained the first 100 sample graphs $\{G_1, \ldots, G_{100}\}$, set S_2 contained all of the samples from S_1 plus the next 100 samples $\{G_{101}, \ldots, G_{200}\}$, and so on, until S_m contained all of the sample graphs $\{G_1, \ldots, G_n\}$. We then used $\mathsf{IndeCut}$ to compute cut norm estimates for each set of subsamples S_i, in order to identify an approximate sample size at which the cut norm estimate for S_i became very close to the cut norm estimate for the entire sample space $S_m = \{G_1, \ldots, G_n\}$. Fig. 3.8 shows a visualization of the relationship between the number of samples and the cut norm estimates for a large biological network (TF $\rightarrow$ Gene network extracted from the Human

regulatory network).

In order to help user to estimate a sensible cutoff range for required number of samples for each algorithm and network we provide a plugin in IndeCut software package. This plugin visualizes the relationship between the number of samples and cutnorm estimates (see IndeCut's manual for details).

3.3.5 Networks and graphs

Two sets of graphs were created or selected for this study: 1) Manually constructed "toy" bipartite graphs with sizes ranging from tens of nodes to hundreds of nodes, representing different graph structures, including "even" or "near-even" graphs, "uneven" graphs, and "hybrid" combinations of in/out-degrees, and 2) Real biological networks.

Real networks - Two biological networks were obtained from literature and public databases. An *Ecoli* network representing a medium-size yeast transcriptional network was downloaded from [58]. This network contains two types of nodes: transcription factor (TF), and gene. Two layers of interactions (TFgene, TFTF) were extracted into separate bipartite graphs for application of IndeCut. A *Human* regulatory network was downloaded from http://encodenets.gersteinlab.org/, representing a network with thousands of nodes and edges. This network is used as a case study in the publication of CoMoFinder [38]. This network contains three types of nodes: TF, miRNA, and protein-coding gene. This network comprises five interaction layers: TFTF, TFmiRNA, miRNATF, TFgene, and miRNAgene.

Each of these layers forms a separate input bipartite graph for IndeCut.

3.3.6 Description of examined network motif discovery algorithms

In order to compare the performance of existing network motif discovery algorithms using IndeCut, four different network motif finding algorithms were selected: FANMOD (Fast Network Motif Detection) [69], DIA-MCIS (Diaconis Monte Carlo Importance Sampling) [22], WaRSwap (Weighted and Reverse Swap sampling) [42], and CoMoFinder [38].

FANMOD is a well-known implementation of the edge switching randomization algorithm. The edge-switching method randomly chooses two directed edges (x, y), (u, v) from input graph G and switches their endpoints only if G doesn't already contain either of these new edges (x, v), (u, y). It repeats this procedure for defined number of attempts and reports a random graph G'. An implementation of FANMOD was downloaded from [69] and we added a print statement in the source code "main.cpp" which prints the edges of the randomized graph produced by the method named "randomized_graph" so we can read them as input for IndeCut.

CoMoFinder implements a restricted version of the edge-switching method to detect only K-node motifs containing all node types such as TF, miRNA, and Gene, on given TF-miRNA-Gene regulatory networks. It breaks down the original network into seven different layers (miRNA $\rightarrow$ TF, TF $\rightarrow$ gene, miRNA TF, TF TF, TF $\rightarrow$ miRNA, TF $\rightarrow$ TF, TF $\rightarrow$ gene). Within each layer it chooses two edges (x, y), (u, v) and switches the endpoints if two conditions satisfied: 1)

Neither of the new edge-pairs (x, v), (u, y) exist in the input graph G, and 2) An edge-switch between (x, y), (u, v) is allowed to happen only once, as revisiting a previously performed switch is not allowed (i.e. switching back from a graph containing (x, v) and (u, y) to a graph containing (x, y) and (u, v) is not allowed). CoMoFinder repeats the above-described procedure until either it reaches a stage such that no edge-pair is available to switch, or it has completed a pre-defined maximum number of edge-switching attempts. The original CoMoFinder program [38] was downloaded and modified to print randomized graphs into files for our analysis.

DIA-MCIS is an efficient implementation of an importance sampling algorithm [16] to generate random graphs (self-loops included) from fixed in/out-degree sequences. DIA-MCIS converts an input graph G into a zero-one adjacency matrix M_{m*n} with m rows and n columns where m_{ij} is 1 if node j has a directed link to node i. It then sequentially fills the columns by a weighted-sampling scheme. It starts with first column which represents the first source node with out-degree of $\deg_0$, and assigns $\deg_0$ 1s randomly to m cells (each cell represents a target node). In this process, nodes with higher in-degrees have more chance of selection by source nodes with higher out-degrees. The algorithm updates the row/column sums as proceeds to the next column.

WaRSwap produces randomized background graphs from an input graph by breaking it into layers representing five possible interaction types: TF→TF, TF→miRNA, TF→gene, miRNA→TF, and miRNA→gene. WaRSwap treats each layer as a bi-partite graph G and operates as follows to generate a randomized graph G'. It first

sorts the source nodes in descending order of out-degree, and for each source node S_i it computes the sampling weights for each target node T_j using a weighting formula [42]. The weighting formula corrects the tendency of source nodes with large out-degrees to target nodes with larger in-degrees. WaRSwap places an edge between S_i and T_j if possible, otherwise it enters a specific back-swapping procedure to identify a new target node. We downloaded a Java implementation of WaRSwap from http://megraw.cgrb.oregonstate.edu/software/WaRSwapSoftwareApplication/ and R implementation from http://megraw.cgrb.oregonstate.edu/software/WaRSwap. The WaRSwapApp makes an automated selection of the WaRSwap weighting parameter for the user based on the in/out-degree sequences of the input graph. We modified the R implementation of WaRSwap to include this automated weighting parameter selection.

3.3.7 Description of edge switch graphs

We detail here how the edge switch graphs (ESG's) were created. Given in and out-degrees R and C, we generate all possible bipartite graphs $\{G_1, \ldots, G_N\}$ with in/out-degrees R and C. The edge switch graph G_{ESG} is an undirected graph with vertex set $V = \{G_1, \ldots, G_N\}$ and edge set E defined as follows: for $G_i, G_j \in V$, the undirected edge (G_i, G_j) is an element of E if and only if the graph G_j can be obtained as a result of performing one edge switch on G_i. In more detail, this means that the graphs G_i and G_j have the same vertex set, and identical edge sets, except for one pair of edges (x, y) and (u, v) present in the edge set of G_i but

absent in the edge set of G_j, and one pair of edges (x, v) and (u, y) present in the edge set of G_j but absent in the edge set of G_i.

A graph clustering algorithm known as modularity clustering [11] was then applied to the edge switch graph G_{ESG} to identify clusters that maximize the number of within-cluster edges while minimizing the number of between-cluster edges. Let L be the number of clusters found.

Given a graph sampling algorithm $\mathcal{A}$, the output of $\mathcal{A}$ can be viewed as sampling vertices of the ESG. Define a count vector $count^{\mathcal{A}} \in \mathbb{N}^L$ as a vector indexed by the clusters found above, with $count_i^{\mathcal{A}}$ being equal to the number of times the algorithm $\mathcal{A}$ returned a graph found in cluster i.

A "cluster-time" graph is then created with vertices corresponding to the clusters found above, and edges between two pairs of vertices/clusters if there exists edges in G_{ESG} connecting vertices belonging to these two clusters respectively. The size of the vertex i corresponds to the entry of the count vector $count_i^{\mathcal{A}}$. The entropy of the vector $count^{\mathcal{A}}$ is also calculated to quantify how equally (or unequally) the algorithm $\mathcal{A}$ samples graphs belonging to each cluster: $-\sum_{i=1}^{L} \frac{count_i^{\mathcal{A}}}{\sum_j count_j^{\mathcal{A}}} \log\left(\frac{count_i^{\mathcal{A}}}{\sum_j count_j^{\mathcal{A}}}\right)$. Larger entropy values indicate that the algorithm $\mathcal{A}$ samples each cluster more equally.

3.4 Summary

Over the last two decades, network motif discovery algorithms have been proposed that use several different underlying background graph sampling strategies.

By all agreement in the literature, a uniform and independent background graph sampling method is fundamental for accurate network motif discovery due to subsequent statistical analysis. Evaluation of this condition on networks beyond tens of nodes was previously not possible because there was no proposed way to perform such an evaluation. Methods originating from the field of mathematical algorithms have been proposed that provably sample uniformly for nearly regular graphs [6, 8], or given an arbitrarily large number of samples [25]. However, most biological networks of interest contain at least several hundred nodes and one or more "hubs" (for example, a transcription factor that is a master regulator). Thus, these results guaranteeing uniformity are of limited practical value due to very large sample spaces (and subsequently infeasible computation times required) and/or uneven degree sequences seen in practice. Direct uniformity tests were performed in the study of some algorithms by empirically enumerating all the graphs in a very small sample space. This did lead to the understanding that graphs of uneven degree distribution posed problems for most algorithms. However, these small-graph tests left uncertainty as to how these algorithms would perform in the case of larger biological networks. As a result, despite the surge in popularity of network motif finding with the exciting findings reported by [46] and by [2], reported laboratory validations of predicted network motif instances were subsequently rare to nonexistent in multicellular organisms. We posit that this may in part be due to unforeseen sampling biases and/or using a low number of samples leading to mis-reporting of motifs (as illustrated in Section 3.2.2.

With the IndeCut method, we hope to change this state of affairs by making

it possible to evaluate the performance of any network motif finding algorithm on any network of interest, including biologically realistic networks.

In addition, we used IndeCut to show that the same motif finding algorithm can perform very differently depending on the graph topology. We also used IndeCut to show that algorithm performance plateaus often occur at a number of iterations exceeding the number of samples recommended by the program user manuals and/or default settings. Most importantly, IndeCut demonstrates that in cases of graphs with uneven degree distributions that are characteristic of biological networks, algorithms based on the sampling strategy known as "edge-switching" are vulnerable to non-uniform and/or non-independent sampling. In this case, reported P-values may be inaccurate due to sampling biases. For such algorithms, we found that non-uniform sampling biases can be caused by frequently sampling graphs that can be reached in a few number of edge switches from the original graph.

We used the concept of an edge-switching graph to investigate the performance differences of algorithms based on edge-switching to those based on other sampling techniques and found that non-uniform sampling bias can be caused by frequently sampling graphs that can reached in a few number of edge switches from the original input graph. While we observed that DIA-MCIS and WaRSwap (which are not based on edge-switching) maintained relatively strong performance overall, this varied based on the topology of the input graph. Hence, one can use IndeCut to ensure that, for a particular graph of interest, one selects the algorithm offering the most uniform sampling procedure. Thus, this strategy is prone to spurious results on such networks. Lastly, we used the concept of an edge-switching graph

to show why this is the case. In essence, edge-switching algorithms produce a sampling bias by spending too much time sampling graphs that can be reached from the starting graph via a small number of edge-switches.

Overall, IndeCut's results show that some graph topologies are in fact highly troublesome to some algorithms (such as hub-containing graphs for edge-switch based algorithms), however, sampling performance cannot be anticipated universally for any algorithm or graph. In most cases, we observed that DIA-MCIS and WaRSwap both maintained relatively strong performance in sampling graphs with each example topology type. Nonetheless, algorithm performance variation across network types can be substantial, particularly in the case of the "hybrid" graphs which characterize many real-world networks. These results highlight IndeCut's necessity in order to select an appropriate sampling algorithm for a biological network of interest.

Importantly, IndeCut demonstrates that the fast and popular algorithm FANMOD may not uniformly sample graphs when used with uneven degree sequences. This can lead to a bias in the motifs being reported and can confound laboratory validation of motifs. By providing the community with an informed choice of network motif discovery algorithm, we hope that IndeCut will re-ignite interest in laboratory validation of the fascinating hypotheses that result from network motif discovery outcomes.

The advent of IndeCut has significant implications for network motif finding studies that have been published in the literature over the last decade. These studies use the FANMOD program almost exclusively due to its early software

availability and superior run times as compared with other programs available in the 2000's. Studies performed using an edge-switching technique on small, relatively regular networks are likely justified. However, studies on enormous transcription factor-containing networks are very likely to contain spurious results in the sense that the outcomes obtained are not those that the user intended to measure. Since laboratory validation attempts on specific instances of network motifs are rarely if ever reported in these large network studies, the primary consequence is likely limited to spurious observations or conclusions that are only of general theoretical interest. By providing the community with an avenue to informed network motif discovery algorithm choice in the future, we hope that IndeCut will re-ignite interest in laboratory validation of the fascinating hypotheses that result from network motif discovery outcomes.

Table 3.1: Runtime of IndeCut on all examined graphs.

Graph	Number of nodes	Number of edges	IndeCut run-time
uniFanG1	11	20	10 s
biFanG1	22	40	10 s
triFanG1	33	60	15 s
tetraFanG1	44	80	20 s
pentaFanG1	55	100	35 s
hexaFanG1	66	120	48 s
regularSmallG1	23	22	9 s
regularSmallG2	62	86	21 s
regularSmallG3	31	32	10 s
regularG1	80	800	27 s
regularG2	92	1058	30 s
regularG3	88	968	28 s
Human_TF→TF	174	644	52 s
Human_TF→miR	332	1237	2 min
Human_miR→TF	606	2594	5 min
Human_TF→GENE	9055	25748	4 days
Human_miR→GENE	12185	115421	14 days
Ecoli_TF→TF	140	129	2 min
Ecoli_TF→GENE	365	390	4 min

IndeCut evaluates graphs on the order of several thousand nodes and tens of thousands of edges within a few minutes to a few days using standard hardware. This table provides IndeCut's observed run time on each graph and algorithm. The miRNA→Gene layer in the human network allows us to provide run time given an extreme example with approximately 100,000 edges. To put these run times into perspective, network motif tools typically take several days simply to provide an output for graphs of this size, using a small number of iterations that does not guarantee meaningfully accurate performance (we discuss the number of iterations necessary for optimal performance for each sampling method in the next section). Using a commercial optimization package such as Guorbi or Mosek (in contrast to the open-source package CSDP that we use here) will result in speed improvements to IndeCut. Thus, considering time costs of running network motif finding algorithms themselves as well as the enormous potential laboratory costs of attempting to validate inaccurate results, IndeCut presents a very practical method for making an informed network motif discovery algorithm choice on biological networks of study.

Algorithm 2 Cut norm lower bound

Input:

$A \in \mathbb{R}^{m \times n}$

$c \in \mathbb{N}$

$u_i, v_j \in \mathbb{R}^{m+n}$ (from computation of $||A||_{\text{SDR}}$)

Initialization:

$its = 0$

$bound = 0$

Iterations:

while $its < c$ **do**

  $G = (g_1, \ldots, g_{m+n})$ (random variates of $\mathcal{N}(0, 1)$)

  for $i = 1, \ldots, m$ **do**

    $x_i = \text{sign}(u_i \cdot G)$

  end for

  for $j = 1, \ldots, n$ **do**

    $y_j = \text{sign}(y_j \cdot G)$

  end for

  $temp = \sum_{i,j} A_{i,j} x_i, y_j$

  if $temp > bound$ **then**

    $bound = temp$

  end if

  $its = its + 1$

end while

Output:

$||A||^{\text{est}}_{\infty \mapsto 1} = bound$ (Lower bound)

Chapter 4: Conclusion

In this dissertation, I proposed novel solutions for two investigations into gene regulation. First, a generalizable machine learning approach is proposed to model tissue-specific gene expression from DNA sequence and chromatin states in Arabidopsis thaliana root and leaf tissue. Two models were introduced: (i)'ROE' and (ii)'Tiled'. For both models, transcription start sites (TSSs) and open chromatin regions are identified using available high-resolution TSS-seq and DNase-seq datasets obtained from 7-day wild-type Arabidopsis thaliana root and leaf tissue. Using identified TSS tag clusters in both root and leaf tissues, $41,196$ promoter sequences with the size of 6-kb (centered at TSS peak mode) were extracted.

Chromatin state (open vs. closed regions) are computed using a DNase-seq dataset in both tissues. The genomic regions wherein chromatin was depleted of nucleosomes were marked as "open" and are used for generating related features in both models. It was observed that chromatin is mostly open within a 1-kb upstream region of the TSS across the vast majority of promoters. A few promoters are entirely closed in one of the tissue types (2-3% of all promoters).

Using an RNA-seq dataset, the promoters of genes that were highly differentially expressed between root and leaf tissue were identified (e.g. highly expressed in root as compared to leaf, or highly expressed in leaf as compared to root). These 2,028 differentially expressed promoters were then selected for training and testing

of the model. Transcription factor binding affinities are computed using a comprehensive set of 413 PWMs from reliable databases along with the TFBSScan software package. Note that at least one family member from each TF family was represented within the PWM collection.

Two types of features for each model were generated: (i) TFBS-features which represent the TF binding affinity of each PWM within the designated region, and OC-features which present the openness of the associated TFBS binding region. Studying the top-weighted ROE and Tiled model features showed that both models agree on the importance of many features. However, model differences yield important insights about cis-regulatory element encodings within promoter structure. We observed that compared to the 'Tiled' model, the 'ROE' model achieves the same level of accuracy in predicting the tissue in which a gene is expressed, despite using 50% fewer features (which are compressed into ROE regions). On the other hand, the Tiled model highlights the putative regulatory contributions of TFs which are ignored in the ROE model due to lack of binding location preference across the full set of promoters. Analysis of the top-weighted features in the ROE and Tiled models suggests specific TFs that are playing a biological role in tissue-specific gene expression according to corroborating literature and the RNA-seq dataset. The modeling process yields hypotheses about the biological roles of these TFs in tissue-specific gene expression, which can be further evaluated through computational and biological experiments.

Intriguingly, very high performance for both models could be achieved through the contributions of TFBS sites presence alone; however, a Tiled model that con-

tained only OC features (no specific TFBS site information encoded) also yielded very high performance. This would suggest that TFBS sites and OC features could each be considered to independently encode the majority of information about tissue-specificity within a promoter region. The highest performance was achieved for both models when using both TFBS and OC features, suggesting that TFBS sites and OC regions collectively encode more biological information about tissue specificity than either feature type alone and thus are not entirely redundant. Our feature analysis supports the hypothesis that for most promoters, it is the collection of accessible TF binding sites (sites in OC regions) that is primarily determinate of the tissue of expression.

Our analysis does not suggest in general whether it is the presence of TF sites that causes chromatin to be opened at these sites (in a direct mechanistic sense), or whether it is a particular OC state in a given tissue which then directly determines which sites become accessible. As an outcome of the modeling process, I do find specific cases of promoters that appear to drive gene expression in a particular tissue primarily because of the collection of important TF binding sites they contain; I also find cases where OC state is suggested by the model to be fundamentally important to tissue of expression with little importance assigned to the collection of TF sites present in the promoter. Therefore, it is possible that the causal mechanism or ordering of events is not identical across all promoters.

The ROE model is designed on the assumption that most TFs which play a role in tissue specificity have binding location preferences with respect to transcription start site (TSS) location. Therefore, TFBS scores in the ROE model were com-

puted within ROE regions. Those PWMs that did not have an enriched region (no ROE) were discarded from ROE model. By contrast, the Tiled model used all TFBS scores inside 100bp non-overlapping regions within 1 kb upstream and 500bp downstream of the TSS location in particular, 67 PWMs that did not have ROEs were added to the PWM set for generating Tiled model features. Since the initial full set of PWMs contained many matrices which were highly similar (sometimes nearly identical), I developed a software program to compute the similarity between the PWMs and remove redundant PWMs. This change to the entire PWM set improved the performance of both models as well as aiding interpretation of the feature weights. This is mainly due to the fact that redundant PWMs with similar binding domains can lead to highly correlated feature subsets; as a result, small weights may be assigned to each PWM in a subset even if a single representative PWM would otherwise receive a relatively large regression coefficient.

An important property of the datasets used in this study is that all three sequencing libraries were generated from same plant samples at the same time and under the same conditions. This attribute allows accurate modeling gene expression between two different tissues, with reliable information about transcription start sites and chromatin state along with gene expression level. To the best of my knowledge, as compared to all other relevant studies [50, 62, 27] , the approach proposed in this thesis achieves the strongest performance outcome in modeling promoter structure to predict the tissue in which a gene will express.

Second, in the context of analyzing gene regulatory networks, a novel method IndeCut were proposed which for a very first time, enables the evaluation of net-

work motif discovery outcome in terms of uniform and independent background network sampling. Over the last two decades, network motif discovery algorithms have been proposed that use several different underlying background graph sampling strategies. By all agreement in the literature, a uniform and independent background graph sampling method is fundamental for accurate network motif discovery due to subsequent statistical analysis. Evaluation of this condition on networks beyond tens of nodes was previously not possible because there was no proposed way to perform such an evaluation. Methods originating from the field of mathematical algorithms have been proposed that provably sample uniformly for nearly regular graphs [6, 8], or given an arbitrarily large number of samples [25].

However, most biological networks of interest contain at least several hundred nodes and one or more 'hubs' e.g., a transcription factor that is a master regulator. Thus, the results guaranteeing uniformity are of limited practical value due to very large sample spaces, their infeasible computational time, and uneven degree sequences are observed in practice. Direct uniformity tests were performed in the study of some algorithms by empirically enumerating all the graphs in a very small sample space. This did lead to the understanding that graphs of uneven degree distribution posed problems for most algorithms. However, these small-graph tests left uncertainty as to how these algorithms would perform in the case of larger biological networks. The proposed IndeCut method in this paper, we change this state of affairs by making it possible to evaluate the performance of any network motif finding algorithm on any network of interest, including biologically realistic networks.

IndeCut is used to show that the same motif finding algorithm can perform very differently depending on the graph topology. Importantly, IndeCut demonstrates that the fast and popular algorithm FANMOD may not uniformly sample graphs when used with uneven degree sequences. This can lead to a bias in the motifs being reported and can confound laboratory validation of motifs. By providing the community with an informed choice of network motif discovery algorithm, my hope is that IndeCut re-ignites interest in laboratory validation of the fascinating hypotheses that result from network motif discovery outcomes.

Bibliography

[1] Noga Alon and Assaf Naor. Approximating the cut-norm via grothendieck's inequality. *SIAM Journal on Computing*, 35(4):787–803, 2006.

[2] Uri Alon. Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461, 2007.

[3] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[4] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101–113, 2004.

[5] Alexander Barvinok. On the number of matrices and a random matrix with prescribed row and column sums and 0–1 entries. *Advances in Mathematics*, 224(1):316–339, 2010.

[6] Mohsen Bayati, Jeong Han Kim, and Amin Saberi. A sequential algorithm for generating random graphs. *Algorithmica*, 58(4):860–910, 2010.

[7] Michael A. Beer and Saeed Tavazoie. Predicting Gene Expression from Sequence. *Cell*, 117(2):185–198, April 2004.

[8] Ivona Bezáková, Nayantara Bhatnagar, and Eric Vigoda. Sampling binary contingency tables with a greedy start. *Random Structures & Algorithms*, 30(1-2):168–205, 2007.

[9] Joseph Blitzstein and Persi Diaconis. A sequential importance sampling algorithm for generating random graphs with prescribed degrees. *Internet Mathematics*, 6(4):489–522, 2011.

[10] Brian Borchers. Csdp, ac library for semidefinite programming. *Optimization methods and Software*, 11(1-4):613–623, 1999.

[11] Ulrik Brandes, Daniel Delling, Marco Gaertler, Robert Gorke, Martin Hoefer, Zoran Nikoloski, and Dorothea Wagner. On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, 20(2):172–188, 2008.

[12] Nicolas L. Bray, Harold Pimentel, Pll Melsted, and Lior Pachter. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology*, 34(5):525, May 2016.

[13] R. J. Britten and E. H. Davidson. Gene regulation for higher cells: a theory. *Science (New York, N.Y.)*, 165(3891):349–357, July 1969.

[14] Piero Carninci, Albin Sandelin, Boris Lenhard, Shintaro Katayama, Kazuro Shimokawa, Jasmina Ponjavic, Colin A. M. Semple, Martin S. Taylor, Pr G. Engstrm, Martin C. Frith, Alistair R. R. Forrest, Wynand B. Alkema, Sin Lam Tan, Charles Plessy, Rimantas Kodzius, Timothy Ravasi, Takeya Kasukawa, Shiro Fukuda, Mutsumi Kanamori-Katayama, Yayoi Kitazume, Hideya Kawaji, Chikatoshi Kai, Mari Nakamura, Hideaki Konno, Kenji Nakano, Salim Mottagui-Tabar, Peter Arner, Alessandra Chesi, Stefano Gustincich, Francesca Persichetti, Harukazu Suzuki, Sean M. Grimmond, Christine A. Wells, Valerio Orlando, Claes Wahlestedt, Edison T. Liu, Matthias Harbers, Jun Kawai, Vladimir B. Bajic, David A. Hume, and Yoshihide Hayashizaki. Genome-wide analysis of mammalian promoter architecture and evolution. *Nature Genetics*, 38(6):626, June 2006.

[15] Sourav Chatterjee, Persi Diaconis, and Allan Sly. Random graphs with a given degree sequence. *The Annals of Applied Probability*, pages 1400–1435, 2011.

[16] Yuguo Chen, Persi Diaconis, Susan P Holmes, and Jun S Liu. Sequential monte carlo methods for statistical analysis of tables. *Journal of the American Statistical Association*, 100(469):109–120, 2005.

[17] The ENCODE Project Consortium. A User's Guide to the Encyclopedia of DNA Elements (ENCODE). *PLOS Biology*, 9(4):e1001046, April 2011.

[18] Jason S. Cumbie, Sergei A. Filichkin, and Molly Megraw. Improved DNase-seq protocol facilitates high resolution mapping of DNase I hypersensitive sites in roots in Arabidopsis thaliana. *Plant Methods*, 11, September 2015.

[19] Jason S. Cumbie, Maria G. Ivanchenko, and Molly Megraw. NanoCAGE-XL and CapFilter: an approach to genome wide identification of high confidence transcription start sites. *BMC Genomics*, 16:597, August 2015.

[20] Ramana V. Davuluri, Hao Sun, Saranyan K. Palaniswamy, Nicole Matthews, Carlos Molina, Mike Kurtz, and Erich Grotewold. AGRIS: Arabidopsis Gene

Regulatory Information Server, an information resource of Arabidopsis cis-regulatory elements and transcription factors. *BMC Bioinformatics*, 4:25, June 2003.

[21] Bailey K Fosdick, Daniel B Larremore, Joel Nishimura, and Johan Ugander. Configuring random graph models with fixed degree sequences. *arXiv preprint arXiv:1608.00607*, 2016.

[22] Diana Fusco, Bruno Bassetti, P Jona, and M Cosentino Lagomarsino. Diamcis: an importance sampling network randomizer for network motif discovery and other topological observables in transcription networks. *Bioinformatics*, 23(24):3388–3390, 2007.

[23] Allison Gaudinier and Siobhan M Brady. Mapping transcriptional networks in plants: Data-driven discovery of novel biological mechanisms. *Annual review of plant biology*, 67:575–594, 2016.

[24] Nick Gilbert, Shelagh Boyle, Heike Fiegler, Kathryn Woodfine, Nigel P. Carter, and Wendy A. Bickmore. Chromatin Architecture of the Human Genome: Gene-Rich Domains Are Enriched in Open Chromatin Fibers. *Cell*, 118(5):555–566, September 2004.

[25] Catherine Greenhill. The switch markov chain for sampling irregular graphs. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1564–1572. SIAM, 2015.

[26] Joshua A Grochow and Manolis Kellis. Network motif discovery using subgraph enumeration and symmetry-breaking. In *Annual International Conference on Research in Computational Molecular Biology*, pages 92–106. Springer, 2007.

[27] ukasz Huminiecki and Jarosaw Horbaczuk. Can We Predict Gene Expression by Understanding Proximal Promoter Architecture? *Trends in Biotechnology*, 35(6):530–546, June 2017.

[28] Laurence D. Hurst, Oxana Sachenkova, Carsten Daub, Alistair R. R. Forrest, Lukasz Huminiecki, and FANTOM consortium. A simple metric of promoter architecture robustly predicts expression breadth of human genes suggesting that most transcription factors are positive regulators. *Genome Biology*, 15(7):413, July 2014.

[29] Mahmoud M. Ibrahim, Scott A. Lacadie, and Uwe Ohler. JAMM: a peak finder for joint analysis of NGS replicates. *Bioinformatics (Oxford, England)*, 31(1):48–55, January 2015.

[30] Shalev Itzkovitz, Ron Milo, Nadav Kashtan, Guy Ziv, and Uri Alon. Subgraphs in random networks. *Physical review E*, 68(2):026127, 2003.

[31] Svante Janson. Graphons, cut norm and distance, couplings and rearrangements. Technical report, Department of Mathematics, Uppsala University, 2010.

[32] Arnav Kapur, Kshitij Marwah, and Gil Alterovitz. Gene expression prediction using low-rank matrix completion. *BMC Bioinformatics*, 17:243, June 2016.

[33] Wooyoung Kim, Martin Diko, and Keith Rawson. Network motif detection: Algorithms, parallel and cloud computing, and related tools. *Tsinghua Science and Technology*, 18(5):469–489, 2013.

[34] Oliver D King. Comment on "subgraphs in random networks". *Physical Review E*, 70(5):058101, 2004.

[35] Paula Korkuc, Jos H. M. Schippers, and Dirk Walther. Characterization and identification of cis-regulatory elements in Arabidopsis based on single-nucleotide polymorphism information. *Plant Physiology*, 164(1):181–200, January 2014.

[36] Vivek Krishnakumar, Matthew R. Hanlon, Sergio Contrino, Erik S. Ferlanti, Svetlana Karamycheva, Maria Kim, Benjamin D. Rosen, Chia-Yi Cheng, Walter Moreira, Stephen A. Mock, Joseph Stubbs, Julie M. Sullivan, Konstantinos Krampis, Jason R. Miller, Gos Micklem, Matthew Vaughn, and Christopher D. Town. Araport: the Arabidopsis Information Portal. *Nucleic Acids Research*, 43(D1):D1003–D1009, January 2015.

[37] Boris Lenhard, Albin Sandelin, and Piero Carninci. Metazoan promoters: emerging characteristics and insights into transcriptional regulation. *Nature Reviews Genetics*, 13(4):233, April 2012.

[38] Cheng Liang, Yue Li, Jiawei Luo, and Zhaolei Zhang. A novel motif-discovery algorithm to identify co-regulatory motifs in large transcription factor and microrna co-regulatory networks in human. *Bioinformatics*, 31(14):2348–2355, 2015.

[39] Marina Lizio, Jayson Harshbarger, Hisashi Shimoji, Jessica Severin, Takeya Kasukawa, Serkan Sahin, Imad Abugessaisa, Shiro Fukuda, Fumi Hori, Sachi Ishikawa-Kato, Christopher J. Mungall, Erik Arner, J. Kenneth Baillie, Nicolas Bertin, Hidemasa Bono, Michiel de Hoon, Alexander D. Diehl, Emmanuel Dimont, Tom C. Freeman, Kaori Fujieda, Winston Hide, Rajaram Kaliyaperumal, Toshiaki Katayama, Timo Lassmann, Terrence F. Meehan, Koro Nishikata, Hiromasa Ono, Michael Rehli, Albin Sandelin, Erik A. Schultes, Peter AC t Hoen, Zuotian Tatum, Mark Thompson, Tetsuro Toyoda, Derek W. Wright, Carsten O. Daub, Masayoshi Itoh, Piero Carninci, Yoshihide Hayashizaki, Alistair RR Forrest, and Hideya Kawaji. Gateways to the FANTOM5 promoter level mammalian expression atlas. *Genome Biology*, 16:22, January 2015.

[40] Shmoolik Mangan and Uri Alon. Structure and function of the feed-forward loop network motif. *Proceedings of the National Academy of Sciences*, 100(21):11980–11985, 2003.

[41] V. Matys, O. V. Kel-Margoulis, E. Fricke, I. Liebich, S. Land, A. Barre-Dirrie, I. Reuter, D. Chekmenev, M. Krull, K. Hornischer, N. Voss, P. Stegmaier, B. Lewicki-Potapov, H. Saxel, A. E. Kel, and E. Wingender. TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Research*, 34(Database issue):D108–110, January 2006.

[42] Molly Megraw, Sayan Mukherjee, and Uwe Ohler. Sustained-input switches for transcription factors and microrna s are central building blocks of eukaryotic gene circuits. *Genome biology*, 14(8):1, 2013.

[43] Molly Megraw, Fernando Pereira, Shane T. Jensen, Uwe Ohler, and Artemis G. Hatzigeorgiou. A transcription factor affinity-based code for mammalian transcription initiation. *Genome Research*, 19(4):644–656, April 2009.

[44] István Miklós, Péter L Erdős, and Lajos Soukup. Towards random uniform sampling of bipartite graphs with given degree sequence. *the electronic journal of combinatorics*, 20(1):P16, 2013.

[45] Ron Milo, Nadav Kashtan, Shalev Itzkovitz, Mark EJ Newman, and Uri Alon. On the uniform generation of random graphs with prescribed degree sequences. *arXiv preprint cond-mat/0312028*, 2003.

[46] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[47] Taj Morton, Jalean Petricka, David L. Corcoran, Song Li, Cara M. Winter, Alexa Carda, Philip N. Benfey, Uwe Ohler, and Molly Megraw. Paired-End Analysis of Transcription Start Sites in Arabidopsis Reveals Plant-Specific Promoter Signatures. *The Plant Cell*, 26(7):2746–2760, July 2014.

[48] Taj Morton, Weng-Keen Wong, and Molly Megraw. TIPR: transcription initiation pattern recognition on a genome scale. *Bioinformatics*, 31(23):3725–3732, December 2015.

[49] Ferenc Mller and Lszl Tora. Chromatin and DNA sequences in defining promoters for transcription initiation. *Biochimica Et Biophysica Acta*, 1839(3):118–128, March 2014.

[50] Anirudh Natarajan, Galip Grkan Yardmc, Nathan C. Sheffield, Gregory E. Crawford, and Uwe Ohler. Predicting cell-typespecific gene expression from regions of open chromatin. *Genome Research*, 22(9):1711–1722, September 2012.

[51] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[52] Harold Pimentel, Nicolas L. Bray, Suzette Puente, Pll Melsted, and Lior Pachter. Differential analysis of RNA-seq incorporating quantification uncertainty. *Nature Methods*, 14(7):687–690, July 2017.

[53] Pedro Ribeiro, Fernando Silva, and Marcus Kaiser. Strategies for network motifs discovery. In *e-Science, 2009. e-Science'09. Fifth IEEE International Conference*, pages 80–87. IEEE, 2009.

[54] Sushmita Roy, Jason Ernst, Peter V Kharchenko, Pouya Kheradpour, Nicolas Negre, Matthew L Eaton, Jane M Landolin, Christopher A Bristow, Lijia Ma, Michael F Lin, et al. Identification of functional elements and regulatory circuits by drosophila modencode. *Science*, 330(6012):1787–1797, 2010.

[55] Albin Sandelin, Wynand Alkema, Pr Engstrm, Wyeth W. Wasserman, and Boris Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*, 32(Database issue):D91–94, January 2004.

[56] Florian Schmidt, Nina Gasparoni, Gilles Gasparoni, Kathrin Gianmoena, Cristina Cadenas, Julia K. Polansky, Peter Ebert, Karl Nordstrm, Matthias Barann, Anupam Sinha, Sebastian Frhler, Jieyi Xiong, Azim Dehghani-Amirabad, Fatemeh BehjatiArdakani, Barbara Hutter, Gideon Zipprich, Brbel Felder, Jrgen Eils, Benedikt Brors, Wei Chen, Jan G. Hengstler, Alf Hamann, Thomas Lengauer, Philip Rosenstiel, Jrn Walter, and Marcel H. Schulz. Combining transcription factor binding affinities with open-chromatin data for accurate gene expression prediction. *Nucleic Acids Research*, 45(1):54–66, January 2017.

[57] Nathan C. Sheffield, Robert E. Thurman, Lingyun Song, Alexias Safi, John A. Stamatoyannopoulos, Boris Lenhard, Gregory E. Crawford, and Terrence S. Furey. Patterns of regulatory activity across diverse human cell types predict tissue identity, transcription factor binding, and long-range interactions. *Genome Research*, 23(5):777–788, May 2013.

[58] Shai S Shen-Orr, Ron Milo, Shmoolik Mangan, and Uri Alon. Network motifs in the transcriptional regulation network of escherichia coli. *Nature genetics*, 31(1):64–68, 2002.

[59] Wenqiang Shi, Oriol Fornes, and Wyeth W. Wasserman. Altered transcription factor binding events predict personalized gene expression and confer insight into functional cis-regulatory variants. *bioRxiv*, page 228155, December 2017.

[60] Ritambhara Singh, Jack Lanchantin, Gabriel Robins, and Yanjun Qi. DeepChrome: Deep-learning for predicting gene expression from histone modifications. *arXiv:1607.02078 [cs, q-bio]*, July 2016. arXiv: 1607.02078.

[61] Trevor R Sorrells and Alexander D Johnson. Making sense of transcription networks. *Cell*, 161(4):714–723, 2015.

[62] Leila Taher, Robin P. Smith, Mee J. Kim, Nadav Ahituv, and Ivan Ovcharenko. Sequence signatures extracted from proximal promoters can be used to predict distal enhancers. *Genome Biology*, 14:R117, December 2013.

[63] Shaolei Teng, Jack Y Yang, and Liangjiang Wang. Genome-wide prediction and analysis of human tissue-selective genes using microarray expression data. *BMC Medical Genomics*, 6(Suppl 1):S10, January 2013.

[64] Sterling Thomas and Danail Bonchev. A survey of current software for network analysis in molecular biology. *Human genomics*, 4(5):1, 2010.

[65] Ngoc Tam L Tran, Luke DeLuccia, Aidan F McDonald, and Chun-Hsi Huang. Cross-disciplinary detection and analysis of network motifs. *Bioinformatics and Biology insights*, 9:49, 2015.

[66] Daniel L. Vera, Thelma F. Madzima, Jonathan D. Labonne, Mohammad P. Alam, Gregg G. Hoffman, S. B. Girimurugan, Jinfeng Zhang, Karen M. McGinnis, Jonathan H. Dennis, and Hank W. Bass. Differential Nuclease Sensitivity Profiling of Chromatin Reveals Biochemical Footprints Coupled to Gene Expression and Functional DNA Elements in Maize. *The Plant Cell*, 26(10):3883–3893, October 2014.

[67] Pei Wang, Jinhu Lü, Xinghuo Yu, and Zengrong Liu. Duplication and divergence effect on network motifs in undirected bio-molecular networks. *IEEE transactions on biomedical circuits and systems*, 9(3):312–320, 2015.

[68] MatthewT. Weirauch, Ally Yang, Mihai Albu, Atina G. Cote, Alejandro Montenegro-Montero, Philipp Drewe, HamedS. Najafabadi, SamuelA. Lambert, Ishminder Mann, Kate Cook, Hong Zheng, Alejandra Goity, Harm vanBakel, Jean-Claude Lozano, Mary Galli, Mathew G. Lewsey, Eryong Huang, Tuhin Mukherjee, Xiaoting Chen, JohnS. Reece-Hoyes, Sridhar Govindarajan, Gad Shaulsky, AlberthaJ. M. Walhout, Franois-Yves Bouget, Gunnar Ratsch, LuisF. Larrondo, JosephR. Ecker, and TimothyR. Hughes. Determination and Inference of Eukaryotic Transcription Factor Sequence Specificity. *Cell*, 158(6):1431–1443, September 2014.

[69] Sebastian Wernicke and Florian Rasche. Fanmod: a tool for fast network motif detection. *Bioinformatics*, 22(9):1152–1153, 2006.

[70] Wynand Winterbach, Piet Van Mieghem, Marcel Reinders, Huijuan Wang, and Dick de Ridder. Topology of molecular interaction networks. *BMC systems biology*, 7(1):1, 2013.

[71] Elisabeth Wong, Brittany Baur, Saad Quader, and Chun-Hsi Huang. Biological network motif detection: principles and practice. *Briefings in bioinformatics*, page bbr033, 2011.

[72] Tao Zhang, Wenli Zhang, and Jiming Jiang. Genome-Wide Nucleosome Occupancy and Positioning and Their Impact on Gene Expression and Evolution in Plants. *Plant Physiology*, 168(4):1406–1416, August 2015.