Faculty of Cognitive Sciences and Human Development

Ensemble Framework for Motif Discovery Based on Data Partitioning

Allen Choong Chieng Hoon

Doctor of Philosophy
2020

Ensemble Framework for Motif Discovery Based on Data Partitioning

Allen Choong Chieng Hoon

A thesis submitted

In fulfillment of the requirements for the degree of Doctor of Philosophy

(Cognitive Science)

Faculty of Cognitive Sciences and Human Development
UNIVERSITI MALAYSIA SARAWAK
2020

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

……

Signature

Name: Allen Choong Chieng Hoon

Matric No.: 12010051

Faculty of Cognitive Sciences and Human Development

Universiti Malaysia Sarawak

Date: 11 September 2020

ACKNOWLEDGEMENT

I would like to thank my supervisor Assoc. Prof. Dr. Lee Nung Kion for his helpful advice, guidance and comments throughout my PhD study. This thesis work contributes as part of the Minister of Education Malaysia, Fundamental Research Grant Scheme-FRGS/SG03(01)/1134/2014(01). I would like to thank the funding body for supporting the presentation of the conference paper I co-authored with Norshafrina Omar.

ABSTRACT

Computational DNA motif prediction is a challenging problem because motifs are short, degenerated, and are associated with ill-defined features. With the advances of genome-wide ChIP analysis technology, computational motif discovery tools are necessary to effectively tackle the large-scale datasets for motifs search. Ensemble of DNA motif discovery methods is one of the most successful approaches for motif discovery. Nevertheless, most of the existing works cannot perform motif searches in ChIP datasets because of the limited input sizes of the classical tools employed in the ensemble. Ensemble approach not only uses the results from the classical motif discovery tools, it also combines the discovered results to produce better results. The merging algorithm contributes to the prediction accuracy of the discovered motifs. The primary contribution of this thesis work is the development of an ensemble method called ENSPART with the novelty of using data partitioning technique on ChIP dataset for DNA motif prediction. The idea is to reduce the search space by portioning the input datasets into subsets and tackle by ensemble of classical motif discovery tools separately. Then, using a proposed merging algorithm, the candidate motifs are merged regardless the different lengths. Three experiments are conducted. ChIP datasets have been downloaded to evaluate the performances of the ENSPART with Receiver Operative Curves and Area Under Curve performance metrics. ENSPART was compared with the genome-wide motif discovery tools MEME-ChIP, ChIPMunk, and RSAT peak-motifs using partitioning technique. The results demonstrate that ENSPART performed significantly better than MEME-ChIP and RSAT peak-motifs in terms of the two performance metrics. Another set of datasets are gathered and sampled without partitioning. ENSPART is compared to its employed classifiers: AMD, BioProspector, MDscan, MEME-ChIP, MotifSampler, and Weeder 2. ENSPART is also compared to

MEME-ChIP, ChIPMunk, and RSAT peak-motifs without partitioning. The results show that ENSPART produces significantly better results than its individual classifiers and also MEME-ChIP, ChIPMunk, and RSAT peak-motifs. Finally, an experiment on the simulated datasets is conducted. ENSPART is compared to GimmeMotifs and MotifVoter which both are also ensemble-based tools. The results show that ENSPART produce significantly higher precision and recall rates than GimmeMotifs and MotifVoter. In conclusion, the ensemble technique is effective for DNA motif prediction, while the ChIP dataset can be tackled effectively using data partitioning techniques. The developed merging technique in ENSPART allows effective merging of same motifs from different data partitions. Such methods are generally applicable to any ensemble techniques that utilised classical motif discovery tools, or more recently, ChIP analysis tools.

Keywords: DNA motif discovery, ensemble method, data partitioning

Rangka Kerja "Ensemble" untuk Ramalan Motif DNA Berdasarkan Pembahagian Data

ABSTRAK

Ramalan motif DNA komputasi adalah sesuatu yang mencabarkan kerana motif yang pendek, merosot, dan dikaitkan dengan ciri-ciri yang tidak jelas. Kemajuan teknologi analisis ChIP yang genomik amat memerlukan kemudahan alat penemuan motif komputasi yang dapat mencari motif daripada data berskala besar dengan berkesan. Penemuan motif DNA kaedah "ensemble" adalah salah satus pendekatan yang paling berjaya untuk penemuan motif. Walau bagaimanapun, sebahagian besar penyelidikan yang sedia ada tidak dapat melakukan pencarian motif dalam dataset ChIP kerana alat-alat klasik yang digunakan dalam bersama bersifat dengan saiz input terhad. Pendekatan "ensemble" bukan hanya menggunakan hasil daripada alat penemuan motif klasik, ia juga menggabungkan hasil yang ditemui untuk keputusan yang lebih berkesan. Algoritma penggabungan menyumbang kepada ketepatan ramalan motif yang ditemui. Sumbangan utama kerja tesis ini adalah pembangunan kaedah "ensemble" yang dipanggil ENSPART dengan menggunakan teknik pemecahan data daripada dataset ChIP untuk ramalan motif DNA. Idea ini adalah untuk mengurangkan ruang carian dengan memasangkan dataset input ke dalam subset dan diatasi dengan alat penemuan motif klasik secara berasingan. Dengan menggunakan algoritma penggabungan yang dicadangkan, motif calon disatukan tanpa mengira kepanjangan yang berlainan. Tiga eksperimen telah dijalankan. Dataset ChIP telah dimuat turun untuk menilai prestasi ENSPART dengan metrik "Receiver Operative Curves" dan "Area Under Curve". ENSPART telah dibandingkan dengan alat penemuan motif MEME-ChIP, ChIPMunk, dan RSAT peak-motifs menggunakan teknik pemecahan. Hasilnya menunjukkan bahawa penggunaan ENSPART lebih berkesan daripada MEME-ChIP dan RSAT peak-motifs dari segi dua metrik prestasi. Satu lagi kumpulan dataset dikumpulkan dan disampel tanpa

pemecahan. ENSPART dibandingkan dengan pengelas yang telah digunakan, iaitu AMD, BioProspector, MDscan, MEME-ChIP, MotifSampler, dan Weeder 2. ENSPART juga dibandingkan dengan MEME-ChIP, ChIPMunk, dan RSAT peak-motifs tanpa pemecahan dataset. Hasilnya menunjukkan bahawa ENSPART menghasilkan keputusan yang lebih baik dan mempunyai kesan ketara daripada pengkelas individu dan juga MEME-ChIP, ChIPMunk, dan RSAT peak-motifs. Akhir sekali, eksperimen dengan dataset simulasi telah dijalankan. ENSPART dibandingkan dengan dua alat yang berasaskan kaedah "ensemble" iaitu GimmeMotifs dan MotifVoter. Keputusan menunjukkan bahawa ENSPART mempunyai kadar ketepatan dan pengingatan yang lebih tinggi daripada GimmeMotifs dan MotifVoter. Kesimpulannya, teknik "ensemble" adalah berkesan untuk ramalan motif DNA, manakala ChIP dataset dapat diatasi secara berkesan dengan menggunakan teknik pembahagian data. Teknik penggabungan dalam ENSPART berkesan dalam menggabungkan motif yang sama hasil daripada pembahagian data yang berlainan. Secara amnya, kaedah-kaedah sedemikian dapat digunakan di mana-mana teknik "ensemble" yang menggunakan alat penemuan motif klasik atau alat analisis ChIP yang baru secara amnya.

Kata kunci: *Penemuan motif DNA, kaedah ensemble, pemecahan dataset*

TABLE OF CONTENTS

		Page
DECLARATION		i
ACKNOWLEDGEMENT		ii
ABSTRACT		iii
ABSTRAK		v
TABLE OF CONTENTS		vii
LIST OF TABLES		xi
LIST OF FIGURES		xiv
CHAPTER 1: INTRODUCTION		1
1.1	Background	1
1.2	Problem statements and motivation	4
1.2.1	Evidences	4
1.2.2	Limited input size	5
1.2.3	Effective merging of large number of intermediate motifs	6
1.3	Research questions	7
1.4	Objectives	8
1.5	Hypotheses	8
1.6	Thesis contributions	9
1.7	Thesis structure	10

1.8 Conclusion 11

CHAPTER 2: LITERATURE REVIEW 13

2.1 Background 13

2.2 Motif discovery problem 14

2.3 Motif representations 16

2.4 Datasets for computational motif discovery 23

2.5 Motif discovery approaches 25

2.5.1 Probabilistic approach 25

2.5.2 Enumerative approach 31

2.6 Other machine learning approaches 41

2.6.1 Unsupervised learning 42

2.6.2 Supervised learning 44

2.6.3 Genetic Algorithm 49

2.6.4 Summary of tools developed for motif discovery 51

2.7 Ensemble approach 54

2.7.1 Summary on ensemble approaches 64

2.8 Performance metrics for motif discovery tools 66

2.8.1 Receiver operating characteristics (ROC) 69

2.9 Conclusion 74

CHAPTER 3: METHODOLOGY 78

3.1 Introduction 78

3.2 Motivation 79

3.3 Method 81

3.3.1 Partitioning 84

3.3.2 Motif discovery 85

3.3.3 Candidate motifs comparison and sorting 88

3.3.4 Labelling and grouping 97

3.3.5 Motifs merging 100

3.4 Datasets 103

3.4.1 Datasets 1 103

3.4.2 Datasets 2 105

3.4.3 Datasets 3 107

3.5 Evaluation metric and tools for comparisons 109

3.6 Conclusion 111

CHAPTER 4: FINDINGS AND DISCUSSION 112

4.1 Introduction 112

4.2 Comparison to genome-wide motif discovery tools 112

4.2.1 Findings 115

4.3 Comparison using unpartitioned datasets 129

4.3.1 Motif discovery tools 129

4.3.2 Findings 130

4.4 Comparison using simulated datasets 138

4.4.1 Evaluation metric 139

4.4.2 Findings 142

4.4.3 Coverage metric 149

4.5 Discussion 152

4.6 Conclusion 157

CHAPTER 5: CONCLUSION 159

5.1 Conclusion 159

5.2 Limitations 162

5.3 Future works 163

REFERENCES 166

APPENDICES 197

LIST OF TABLES

		Page
Table 2.1	IUPAC Notation	17
Table 2.2	Summary table of tools developed for motif discovery.	51
Table 2.3	Comparisons of various ensemble tools.	62
Table 3.1	List of motif discovery tools.	86
Table 3.2	List of scoring functions.	87
Table 3.3	Example of KFV distance score comparison matrix after sorting using E2F4 dataset.	94
Table 3.4	Comparison of KFV distance of k=2, k=3, and k=4 after sorting	95
Table 3.5	KFVs of the pair of motifs using k=2 from E2F4 dataset.	97
Table 3.6	Information of the datasets.	104
Table 3.7	TF datasets and the source collected for the experiment without partitioning.	105
Table 3.8	The average sequence length, total bp count, number of sequences, and the percentages of each nucleotide of the collected TF datasets.	106
Table 3.9	TF and the source of known motif matrices from JASPAR database.	108
Table 3.10	The average sequence length, total bp count, number of sequences, and the percentages of each nucleotide of the simulated TF datasets.	109
Table 4.1	The tools that were ran on different datasets.	113
Table 4.2	Number of motifs discovered from the partitioned datasets.	115
Table 4.3	Comparison of number of motifs before and after merging.	116
Table 4.4	Best AUCs and average AUCs of the discovered motifs with ENSPART.	118

Table 4.5 Comparison of the best AUC and average AUC between ENSPART, MEME-ChIP, ChIPMunk, and RSAT peak-motifs. 121
Table 4.6 ENSPART motifs match with JASPAR 2014 database using Tomtom. 125
Table 4.7 P-values of paired sample t-test on the comparison of the best AUCs of the ENSPART and other algorithms. 127
Table 4.8 The average sequence length, total bp count, number of sequences, and the percentages of each nucleotides of the sampled datasets. 129
Table 4.9 Comparison of best AUCs of the discovered motifs by ENSPART and individual tools. 131
Table 4.10 Comparison of paired sample t-test p-values of the best AUCs from ENSPART and individual classifiers used by ENSPART. 133
Table 4.11 Comparison of AUCs of the discovered motifs by ENSPART, ChIPMunk, MEME-ChIP (online), and RSAT peak-motifs. 135
Table 4.12 Comparison of paired sample t-test p-values of the best AUCs from ENSPART, ChIPMunk, MEME-ChIP (online), and RSAT peak-motifs. 137
Table 4.13 Comparison of individual tools used in ensemble algorithms 139
Table 4.14 Sequence logos of the discovered motifs by ENSPART, GimmeMotifs, and MotifVoter 142
Table 4.15 Comparison of precision and recall rates of the best motifs between ENSPART, GimmeMotifs, and MotifVoter 144
Table 4.16 Comparison of F1 scores of the best motifs between ENSPART, GimmeMotifs, and MotifVoter. 147
Table 4.17 Comparison of paired sample t-test p-values of precisions from ENSPART, GimmeMotifs, and MotifVoter. 148

Table 4.18 Comparison of paired sample t-test p-values of recalls from ENSPART, GimmeMotifs, and MotifVoter. 148

Table 4.19 Comparison of paired sample t-test p-values of F1 scores from ENSPART, GimmeMotifs, and MotifVoter. 148

Table 4.20 Coverage scores of the best motifs discovered by ENSPART, GimmeMotifs, and MotifVoter 151

Table A1 Command invocation on the partitioned datasets. 197

LIST OF FIGURES

		Page
Figure 2.1	Sequence alignment to produce a motif profile	18
Figure 2.2	Example of sequence logo.	21
Figure 2.3	Suffix tree for ACCG	36
Figure 2.4	Generalised suffix tree for ACCG$ and ACTG#	37
Figure 2.5	Stages involved in ensemble learning on motif discovery	55
Figure 2.6	Results of ensemble approach.	65
Figure 2.7	Comparison of ROC generation in machine learning classifier and ROC generation in ensemble-based motif discovery.	74
Figure 3.1	ENSPART framework	83
Figure 3.2	Steps to calculate KFV from given PWM or PFM.	91
Figure 3.3	Alignment process of two PWMs	101
Figure 3.4	A multiple alignment of five PWMs.	102
Figure 3.5	Process of converting discovered motifs to ROC and AUC	110
Figure 4.1	Best three ROCs from ENSPART	117
Figure 4.2	ROCs from MEME-ChIP on whole datasets	120
Figure 4.3	Comparison of sequence logos obtained using CREB dataset.	123
Figure 4.4	Comparison of sequence logos obtained using CTCF dataset.	123
Figure 4.5	Comparison of sequence logos obtained using E2F4 dataset.	123
Figure 4.6	Comparison of sequence logos obtained using FOXA1 dataset.	123
Figure 4.7	Comparison of sequence logos obtained using FOXA2 dataset.	124
Figure 4.8	Comparison of sequence logos obtained using NRSF dataset.	124
Figure 4.9	Comparison of sequence logos obtained using NTERA dataset.	124

Figure 4.10 Comparison of sequence logos obtained using OCT4 dataset. 124

Figure 4.11 Comparison of sequence logos obtained using P53 dataset. 125

Figure 4.12 Comparison of sequence logos obtained using STAT1 dataset. 125

Figure 4.13 Comparison of AUCs from ENSPART and individual tools. 132

Figure 4.14 Comparison of AUCs from ENSPART, ChIPMunk, MEME-ChIP, and RSAT peak-motifs. 136

Figure 4.15 Scatter plot of the precision and recall rates for the three tools: ENSPART, GimmeMotifs, and MotifVoter 145

CHAPTER 1

INTRODUCTION

1.1 Background

Proteins are essential biopolymers in cells as the building blocks of various organs or tissues as well as essential component of enzymes. Proteins are produced through a process known as gene-expression, which involves decoding the information stored in protein coding genes in genomes. The two steps involved are transcription and translation. Transcription is a step that replicates the exact copy of genetic codes in the gene into messenger RNA, where the translation step decodes the information in the messenger RNA into proteins. Transcription factor (TF) proteins control when and to what extent each gene is transcribed. The short sequences (i.e. 6–12 bp or base pair) in a genome that are bound by TFs for regulating gene-expression are called transcription factor binding sites (TFBSs) of motifs which are located in the gene's upstream or downstream. There are various types of motifs in the DNA sequences, such as the promoter, silencer, enhancer, insulator, proximal, and distal regulatory motifs. Predicting transcription factors is essential so that biologists are able to study the various diseases such as cancers (Lanchantin, Singh, Wang, & Qi, 2016; Shlyueva, Stampfel, & Stark, 2014; Whitaker, Nguyen, Zhu, Wildberg, & Wang, 2015). Biologists are able to prepare the sequence regions that are anticipated to contain the TFBSs through wet-lab technology (N. K. Lee & Choong, 2013). However, wet-lab experiments to identify the motifs are costly and time-consuming (D. Wang & Do, 2012). Therefore, computational motif analysis techniques are necessary to predict the candidate motifs before further verification (N. K. Lee, Choong, & Omar, 2016). That would allow rapid analysis of transcription factors binding sites in genomic datasets.

Computational motif discovery is a non-deterministic polynomial-time hard (NP-hard) problem (Rigoutsos & Floratos, 1998) because motifs are short (5–20 bp or base pair) and degenerated (Jin, O'Geen, Iyengar, Green, & Farnham, 2007). Dozens of computational tools have been developed to predict the location of TFBSs (Das & Dai, 2007; Lihu & Holban, 2015; Salekin, Zhang, & Huang, 2017; Tran & Huang, 2014). *De novo* motif discovery tools predict novel motifs representing binding sites using certain algorithms. Given an input DNA dataset which contains the binding sites of a TF and its co-factors, computational tools return the most overrepresented repeating sequence patterns or motifs, in the DNA sequences. The predicted candidate motifs can be verified by biologists for functional roles (Bailey, 2011; Satya & Mukherjee, 2004).

A set of motifs which cooperates together is known as *cis*-regulatory module (CRM) (Klepper, Sandve, Abul, Johansen, & Drablos, 2008). There are several types of CRMs: enhancer, silencer, and insulator (Maston, Evans, & Green, 2006). Enhancers are genomic regions that controls the timing, amplitude, and cell-type specific gene expression (Erwin et al., 2014; C. Wang, Zhang, & Zhang, 2013). Because of the role of these enhancers, they are of great interest to understand the evolution and diseases like cancer (Shlyueva et al., 2014). By studying enhancers, biologists are able to understand the development of DNA in order to foresee the tissue specific activity of regulatory elements (Ghandi, Lee, Mohammad-Noori, & Beer, 2014). Enhancers are usually found at distal location from promoter in non-coding regions (C. Wang et al., 2013). They can be located in megabases away from the target genes (Noonan & McCallion, 2010) or located at other chromosomes (Lomvardas et al., 2006). Furthermore, there is no single type of data that is adequate to identify all the enhancers. As a result, enhancers are difficult to be identified (Erwin et al., 2014).

Traditional motif discovery tools are broadly categorised into probabilistic and enumerative approaches. Probabilistic approach uses position weight matrix (PWM) to represent the probability of the nucleotides (A, C, G, and T) occur in the data sequence (Das & Dai, 2007; N. K. Lee & Wang, 2011). Probabilistic approach implements stochastic method such as expectation maximization (EM) and Gibbs sampling (Das & Dai, 2007) while enumerative approach performs exhaustive matching of the nucleotides that are commonly enumerated as A, C, G, and T (Das & Dai, 2007; Kuksa & Pavlovic, 2010; Sandve & Drabløs, 2006).

Artificial intelligence (AI) approaches are also been widely used for DNA motif discovery. Applying AI techniques in motif discovery requires different definition of the problem. For instance, clustering algorithms can be employed to cluster k-mers in a set of DNA sequences based on the k-mers similarities, self-organizing map (SOM) (Mahony, Benos, Smith, & Golden, 2006) has been employed in motif discovery. Furthermore, a motif that is represented as PWM can be assumed as the population in Genetic Algorithm (GA). By using GA, the motifs can be optimised and discovered (L. Li, 2009; L. Li, Liang, & Bass, 2007; F. F. M. Liu, Tsai, Chen, Chen, & Shih, 2004; Z. Wei & Jensen, 2006) in motif discovery.

Motif discovery can be considered as a machine learning task (Brazma, Jonassen, Eidhammer, & Gilbert, 1998) because discovering the motifs is extracting the general rules from the dataset. The sequences that contain the motifs are the positive sequences, while the background sequences are the negative sequences. Thus, the objective of the motif discovery is to identify the motifs through the training from these positive and negative datasets. Recently, supervised learning, especially deep learning, has shown good results in bioinformatics in recent years (Alipanahi, Delong, Weirauch, & Frey, 2015; Eser & Churchman, 2016; Kelley, Snoek, & Rinn, 2015; Qin & Feng, 2017; J. Zhou & Troyanskaya,

2015).

Ensemble approach is a machine learning that uses multiple classifiers to produce a new classifier (Hu, Li, & Kihara, 2005). Unlike hybrid algorithm, ensemble approach does not combine the algorithms to produce a new algorithms. Ensemble approach can be applied in motif discovery by using multiple *de novo* motif discovery tools to discover the candidate motifs. Hence, each individual motif discovery tool or individual classifier can retain its strength. Ensemble approaches have been employed in many previous works and demonstrated excellent performances (Hu et al., 2005; Hu, Yang, & Kihara, 2006; Jin, Apostolos, Nagisetty, & Farnham, 2009; Kuttippurathu et al., 2011; Romer, Kayombya, & Fraenkel, 2007; Wijaya, Yiu, Son, Kanagasabai, & Sung, 2008; Yanover, Singh, & Zaslavsky, 2009). In ensemble approaches, the results from each classifier are combined to produce better results discovered by individual classifier. This allows ensemble learning superior to a single *de novo* motif discovery tool. Furthermore, the ensemble approach is flexible to employ different *de novo* motif discovery tools.

1.2 Problem statements and motivation

1.2.1 Evidences

Motif discovery is a NP-hard problem, because the motifs are short and degenerated (Jin et al., 2007). While many tools have been proposed, the ensemble approaches have shown better overall performances (Hu et al., 2005; Jin et al., 2009; Kuttippurathu et al., 2011; van Heeringen & Veenstra, 2011; Wijaya et al., 2008). One of the reasons is ensemble approaches utilised multiple types of motif discovery algorithms and therefore can predict motifs of different characteristics in dataset. They showed good performance on motif

discovery, but existing methods are not designed for large-scale genomic datasets. In addition, these motif discovery tools can only accept limited size of inputs with few hundreds sequences (Zambelli, Pesole, & Pavesi, 2013). In the post ChIP sequencing (ChIP-seq) era, the genome-wide transcription factor binding region datasets have become available. Those datasets typically have hundreds to multiple thousands of sequences. While there are dozens of standalone motif discovery tools have been proposed to enable motif discovery in the large-scale datasets (Haudry, Ramialison, Paten, Wittbrodt, & Ettwiller, 2010; Kulakovskiy, Boeva, Favorov, & Makeev, 2010; Shi et al., 2011; Bailey, 2011), it is hypothesised the ensemble technique has an edge in term of sensitivity and specificity. There are evident in many past studies (Hu et al., 2005, 2006; Jin et al., 2009; Kuttippurathu et al., 2011; Romer et al., 2007; Wijaya et al., 2008; Yanover et al., 2009) that ensemble approaches performed significantly better than any single tool alone. This owing to the fact that different tools might be able to search for motifs with different characteristics, for example, short versus long motifs, conserved versus weakly conserved, or dependent and non-dependent between nucleotides in motifs. Nonetheless, there is a lack of study that demonstrates the potentiality of ensemble approach on motif discovery towards the large-scale genomic datasets. Most existing ensemble approaches developed pre-ChIP-seq era are deemed infeasible due to the limitation of the individual motif discovery tools to search for motifs in the complex, large search space, and the requirement of high memory resource.

1.2.2 Limited input size

While there have been several ensemble methods (e.g. ChIPMotifs, GimmeMotifs, and CompleteMOTIFS) developed for the ChIP-seq dataset motif analysis, they have restricted

the input sizes for searching to ensure the result can be completed within reasonable time. Therefore, it is necessary to propose an ensemble method that can accept the input with large sizes and at the same time is able to employ the pre-ChIP-seq motif discovery tools.

1.2.3 Effective merging of large number of intermediate motifs

There are two common models can be used to represent the motifs: profile and consensus. Because of ensemble approach uses multiple tools, it is not restricted to accept only certain motif models produced by the individual tools. This increases the technical challenge of designing an ensemble method, because merging different motif model representation is not a straight forward task. The merging of the motif requires common representation, which involves conversion of one model to another. Besides that, similar or identical motifs would be merged. A measurement is necessary to compute the similarity of the motifs. Furthermore, merging condition needs to be defined, so that only when the condition is fulfilled, the motifs should be merged. The merging algorithms will also determine characteristic of the final output. For example, let $merge(a, b) = (a + b)/2$ as a merging function, where a and b are two similar motifs. In order to merge a group of similar motifs, $merge(a, b)$ will be called repetitively until all similar motifs are merged. This indicates that the motif being merged last has the largest weight on the final output. Contrarily, let $merge(L) = (\sum l)/n$ as a merging function, where L is a list of similar motifs, $\sum l$ is the summation of similar motifs, and n is the number of motifs. By using this formula, every similar motif has equal weight on the final output. Therefore, different merging algorithm will produce motifs differently. Existing merging methods are only suitable for merging small number of motifs from the whole input set. With large dataset, more intermediate motifs could be discovered and hence

a more effective merging method is needed.

To address the problems identified in the existing approach, we are motivated by the use of data partitioning approaches in clustering. Data partitioning is a promising solution as it divides the search space into smaller search space. By partitioning the datasets, it not only allows large-scale datasets to be scanned, but it also allows the usage of traditional motif discovery tools on the large-scale datasets. Traditional motif discovery tools were proved to be useful for the pre-ChIP-seq era datasets. This also implies that ensemble approach with data partitioning is potential to solve large-scale datasets motif discovery problem by using any individual motif discovery tools, as long as discovered motifs from the partitioned datasets are able to be merged.

1.3 Research questions

The followings are the research questions derived from problem statements:

i. How to discover the motifs using pre-genomic era or pre-ChIP-seq individual motif discovery tools on the large-scale datasets?

ii. What is the algorithm to combine the discovered motifs from individual motif discovery tools regardless the difference of motif representations?

iii. Does ensemble approach based on data partitioning and multiple merging has better performance comparing to existing ensemble approaches and genome-scale motif discovery approaches?

1.4 Objectives

The main goal of this thesis work is to develop an effective novel ensemble framework for motif analysis of ChIP-seq, that has better motif discovery performance in terms of accuracy comparing to contemporary motif discovery tools.

The specific objectives towards the goal are as follows:

i. To discover motifs by utilising pre and post ChIP-seq era motif discovery tools in the ensemble framework.

ii. To merge similar motifs and produce new motifs without strongly diminishing the quality by using a novel motifs merging algorithm.

iii. To develop a novel ensemble approach framework that performs better than several contemporary ensemble-based motif discovery tools.

Identifying motifs in the genome-wide datasets involves a large search space. This thesis work proposes a framework to partition the genome-wide datasets to smaller samples. Consequently, motif discovery tools will be able to discover the motifs from the subsets. Moreover, by using ensemble approach, the results from each tool will be combined to produce new candidate motifs.

1.5 Hypotheses

This study involves three hypotheses:

i. The proposed ensemble framework that employs novel partitioning technique has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

ii. The proposed ensemble framework that employs novel merging technique has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

iii. The proposed ensemble framework is able to perform significantly better than several contemporary ensemble-based motif discovery tools.

1.6 Thesis contributions

In this thesis work, an ensemble framework for DNA motif discovery called Ensemble Framework Based on Data Partitioning (ENSPART) is developed. The aim of ENSPART is to permit motif analysis on ChIP dataset using multiple motif discovery tools — pre- or post-ChIP era tools. Existing ensemble approaches limit the input sizes as the individual tools employed in the ensemble cannot run on large-scale dataset which is a typical case in the current motif analytic research. ENSPART circumvented that issue by employing a simple data partitioning technique to enable running of classical motif discovery tools on the ChIP dataset.

ENSPART is distinct from existing ensemble methods because each tool does multiple runs to generate many candidate motifs. That increases the chances of finding true motifs. Another novelty is ENSPART performs multiple merging on a big pool of candidate motifs generated by individual tools. While most methods employed alignment method to determine similar motifs in the candidate pool, ENSPART uses alignment free method because of its fast computation of pair-wise motif similarity.

This thesis work also contributed to comprehensive evaluation of ENSPART using real and simulated datasets. The evaluation results have contributed to a better method to

design ensemble method for motif discovery. In specific, the results showed that, running classical motif discovery tools on ChIP dataset remain practical by data partitioning approach and effective merging of large number of redundant candidate motifs. While many new genome-wide motif analysis tools claimed that classical motif discovery tools are no longer applicable in the current data landscape, our method demonstrated that they are still relevant by using the method such as in ENSPART. In our evaluation using three (3) sets of datasets (two ChIP-seq and a simulated), ENSPART performed marked improvement in comparison to existing ensemble approaches as well as genome scale motif discovery tools. Evaluation results also showed ENSPART performed better in terms of sensitivity and specificity rates than individual tools it uses. Interestingly, ENSPART performed better than the state of the art ensemble tool MotifVoter with significant improvement.

1.7 Thesis structure

The next chapter is the Literature Review on previous studies related to DNA motif discovery. The chapter defines motif discovery problem and motif representation using different models. Various motif discovery tools with different approaches including ensemble approach are discussed. Finally, evaluation methods of the motif discovery tools are explained.

Chapter 3 is the methodology of the proposed motif discovery ensemble framework. The chapter covers the proposed ensemble framework, ENSPART, and provides detailed explanation of the algorithm. This chapter also describes the datasets that are collected for the experiments. Lastly, evaluation metric and tools that are used for performance comparisons are explained.

Chapter 4 is the findings of three experiments conducted in this thesis work. The chapter

contains three findings: (i) comparison of ENSPART to genome-wide motif discovery tools, (ii) comparison of ENSPART to various motif discovery tools using unpartitioned datasets, and (iii) comparison of ENSPART to ensemble-based motif discovery tools using simulated datasets. Discussions on the major findings are also provided.

Chapter 5 concludes the thesis with thesis contributions, summary of the findings, limitations of the thesis, and potential future works.

1.8 Conclusion

Transcription factors (TFs) play an important role in regulating gene expression. The transcription process requires TFs to bind to a short (6–12 bps) DNA region, which is called transcription factor binding sites (TFBSs). By studying TFBSs, the biologists are able to verify various diseases and improve medical solutions. Discovering TFBSs is a process known as motif discovery, is a NP-hard problem because the pattern is short and degenerated. The problem is being exaggerated when the next generation sequencing (NGS) such as ChIP-seq is introduced. This is because ChIP-seq produces large-scale datasets for genomic analysis. Various *de novo* motif discovery tools and algorithms were developed to discover the motifs. There are multiple approaches being employed for motif discovery. The two major approaches for motif discovery are probabilistic and enumerative. The probabilistic approach calculates the probabilities of DNA nucleotides occurrences and represented in a matrix form. Enumerative approach uses word enumeration with A, C, G, and T for the DNA nucleotides and performs exhaustive matching. Ensemble learning was introduced to use various *de novo* motif discovery tools as individual classifiers to discover the motifs. The results from the individual classifiers are combined. Previous studies have demonstrated that using ensemble approach can improve the prediction performance. However, large-scale

datasets are not able to be scanned by classic motif discovery tools that were developed before NGS era. As a result, a novel ensemble approach with partitioning of the datasets is proposed. Furthermore, ensemble approach is commonly employed with combining similarity and clustering on the results. Hence, each ensemble-based motif discovery tool is expected to have distinctive results and the prediction accuracies of each ensemble-based tool are able to be compared. Moreover, the ensemble approach should perform better than or at least at the same level as its individual classifiers (Hu et al., 2005).

CHAPTER 2

LITERATURE REVIEW

2.1 Background

In biology, a genome is a complete set of genetic material of an organism which consists of deoxyribonucleic acid (DNA). DNA carries the genetic information of an organism and is composed of nucleotides. There are four nucleotides: A (adenine), C (cytosine), G (guanine), and T (thymine). DNA plays important role in gene expression, that is, the process to synthesize the gene products, especially proteins. Transcription is the initial step in gene expression. It is a process of creating a complementary ribonucleic acid (RNA) copy of a sequence of DNA. Similar to DNA, RNA consists of four types of nucleotides: A, C, G, and U (uridine). The transcription requires a transcription factor (TF) to bind to a region of the DNA called promoter. Transcription factor is a protein that binds to the DNA region to initiate the transcriptional regulation.

Transcription factors regulate the transcription of genetic information from DNA to messenger RNA (mRNA). A short sequence segment in a DNA sequence that is bound by TF is called transcription factor binding site (TFBS) or regulatory motifs, such as promoters. TFBSs are the small fragments (i.e. 5–20 bp or base pair) that are statistically overrepresented in a genome (Lihu & Holban, 2015). They are located in the gene's upstream (5'-end) or downstream (3'-end) closely to the transcription start site (TSS) of proximal promoter region or further region as enhancers and silencers (Pavesi, Mauri, & Pesole, 2004). A motif is a conserved pattern which can be found in more than one sequence of DNA. Motif can also be known as repetitive pattern that appears in the DNA sequences (Zaslavsky & Singh, 2006).

Over-representation of a motif indicates that the pattern of the combination of the nucleotides (A, C, G, and T) has a frequency of occurrences significantly higher than the background set. Because motifs are important in the gene-expression regulation, the scientists are interested to discover these binding sites for biological research.

2.2 Motif discovery problem

Motif discovery problem can be formulated in several ways but in general the aim is to predict candidate motifs and their locations in the input sequences. The input sequences are typically obtained through wet-lab experiments that are believed to be enriched with binding regions of a TF protein of interest. Nevertheless, other than the primary motifs of the TF, there are possibly motifs of co-factors. Therefore, motif discovery tools typically return several candidate motifs in a run.

In a more technical way, motif discovery can be defined as a problem to find the short and similar sequence by given a set of DNA sequences with a common biological function (Zambelli et al., 2013). The problem can be formulated as searching for motif length k and its instances. Typically, the lengths of motifs to search are specified by the users. Once the lengths are specified, computational motif algorithms would search for motif instances in the input DNA sequences.

A motif can be represented as a string pattern or a matrix. String pattern is known as consensus, while matrix is known as motif profile which can be expressed as position weight matrix (PWM). Because motifs are conserved, their instances appear to be like each other but not identical. The instances of a motif are usually overrepresented in the input sequences but rarely in the background sequences that do not contain motifs. Therefore, over-representation

of a motif can be computed by its scoring in the input sequences against the background sequences. Following Lee's (2018) definition, we formally define motif discovery problem as:

> Given a set of DNA sequences S_F containing potential binding sites of a transcription factor protein, predict their locations in S_F to reveal the motif patterns, described by a suitable computational model, return a list of top scored length k motif patterns together with their instance locations.

In the definition above, the motif patterns with length k are assumed unknown and are automatically determined by the algorithm. Due to random evolutionary events such as the mutation, insertion, and deletion, the motif patterns are not conserved. However, this formulation is NP-hard that large motif lengths such as $k > 15$ bp cannot be solved within feasible computation time. Moreover, the search space grows exponentially when k increases.

A TF may bind to binding site "AAATCGGG" or its degenerated form "AAATTGGG" with only one position with distinct nucleic acid. To identify such motifs, all possible variants of the nucleic acids need to be considered. For example, the complexity of MEME algorithm is $O(N^2 \cdot L^2)$ where N is the number of input sequences and L is the sequence length. As a result, the time complexity will increase exponentially with the large datasets.

Other than defining computational motif discovery as a search problem, it can be defined as a multiple alignment problem as well (Bailey & Elkan, 1994). Alignment of genomic sequences from orthologous or paralogous species can identify regions that are conserved.

Through conversation analysis, the potential binding regions can be identified. Nevertheless, alignment techniques have been shown to miss out many true binding sites (N. K. Lee et al., 2018).

In order to search for candidate motif locations, *de novo* motif discovery tools have been developed. They are used for identifying motifs without prior knowledge of their motif appearance and locations (Z. Wei & Jensen, 2006). Motif discovery tools search through DNA sequences and find patterns that appear frequently and evolutionary conserved. The results from the search are candidate motifs that are most overrepresented using some statistical scores (Sandve & Drabløs, 2006).

2.3 Motif representations

There are two common ways to represent a motif (Hashim, Mabrouk, & Al-Atabany, 2019; Stormo, 2000): consensus (or word enumeration, or oligos) based on International Union of Pure and Applied Chemistry (IUPAC), and profile matrix. The most prevailing profile matrix is position weight matrix (PWM), which is also known as position specific scoring matrix (PSSM) (Linhart, Halperin, & Shamir, 2008; Sandve & Drabløs, 2006).

The consensus representation is using the DNA nucleotides to represent the motifs as a string, $S = (s_1 s_2 s_3 \cdots s_l)$ where $s_i \in \Sigma$. According to evolutionary theory, motifs are not fully conserved because of the genetic operations such as mutation, deletion, and insertion. As a result, motif patterns must be presented with the ambiguity of the nucleotides. These ambiguous codes are usually based on IUPAC notation or regular expression. For instance, a pattern can be translated to IUPAC notation (Haudry et al., 2010; Ji & Wong, 2006) as AARGTTAT, where R is a “purine”, which can be either A or G. On the other hand, the same

pattern can be represented using regular expression as AA[AG]GTTAT, which the third word can be either A or G.

Table 2.1 shows the relation of IUPAC ambiguity notation to the regular expression.

Table 2.1: IUPAC Notation

IUPAC Notation	Description	Regular expression
W	weak	[AT]
S	strong	[CG]
M	amino	[AC]
K	keto	[GT]
R	purine	[AG]
Y	pyrimidine	[CT]
B	not A	[CGT]
D	not C	[AGT]
H	not G	[ACT]
V	not T (and U for RNA)	[ACG]
N	any	[ACGT]

An alternative to consensus representation is profile (Das & Dai, 2007). A profile is constructed from multiple sequence alignment (Gribskov, McLachlan, & Eisenberg, 1987). The sequences are aligned by similarity then the frequency of the nucleotides according to the position is calculated. The PSSM is also known as profile (Brejová et al., 2000; Gribskov et al., 1987; Wasserman & Sandelin, 2004). Consensus is different from profile as the

former uses words to denote the most frequent nucleotide in each position, the latter involves sequence alignment and building the matrix such as PSSM.

Figure 2.1 shows the sequence alignment of binding sequences for the construction of motif profile.

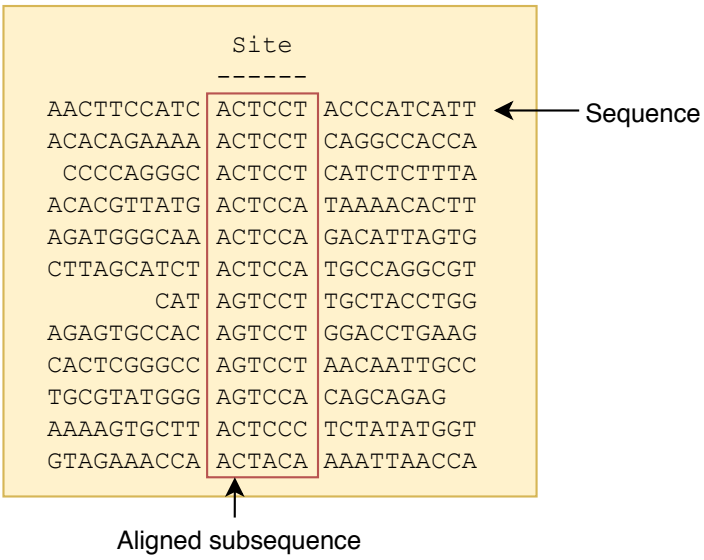


Figure 2.1: Sequence alignment to produce a motif profile. The sequences are aligned so that the subsequences are produce identical or similar pattern. The discovered pattern is used to construct the profile.

After alignment and trimmed into equal length, the position frequency matrix (PFM) (Stormo, 2000) can be calculated by frequency of the nucleotides as follows:

$$\begin{array}{c} A \\ C \\ G \\ T \end{array} \begin{bmatrix} 12 & 0 & 0 & 1 & 0 & 5 \\ 0 & 8 & 0 & 11 & 12 & 1 \\ 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 12 & 0 & 0 & 6 \end{bmatrix}$$

The PFM entries can be defined as Equation 2.1:

$$m_{k,j} = \sum_{i=1}^{N} I(X_{i,j} = k) \qquad \text{Equation 2.1}$$

where $i \in (1, \ldots, N)$, N is the length of motif, $j \in (1, \ldots, 4)$, $k \in \Sigma$, and $I(a = k)$ is the indicator function.

The indicator function $I(a = k)$ is defined as Equation 2.2:

$$I(x = k) = \begin{cases} 1 & \text{if } x = k, \\ 0 & \text{if } x \neq k. \end{cases} \qquad \text{Equation 2.2}$$

The element $m_{1,1}$ with the value 12 means that the frequency of nucleotide A at position 1 is 12. The element $m_{2,2}$ with the value 8 means that the frequency of nucleotide C at position 2 is 8, and $m_{3,2}$ means the frequency of nucleotide G at position 2 is 4. Hence, the probability of nucleotide C to appear at position 2 of the motif is 0.67 probability, while nucleotide G is 0.33. Furthermore, $f_{b,i}$ can be used to denote the frequency of the nucleotide b at position i.

The PFM can be converted to the position probability matrix (PPM) by normalising the entries in each position of the matrix in Equation 2.3:

$$P = \frac{1}{N} M \qquad \text{Equation 2.3}$$

where M is the PFM and N is the number of aligned binding sites.

As a result, the matrix above will be converted into the following PPM:

$$\begin{array}{c} A \\ C \\ G \\ T \end{array} \begin{bmatrix} 1.00 & 0.00 & 0.00 & 0.08 & 0.00 & 0.42 \\ 0.00 & 0.67 & 0.00 & 0.92 & 1.00 & 0.08 \\ 0.00 & 0.33 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.00 & 0.00 & 0.00 & 0.50 \end{bmatrix}$$

According to Stormo (2000), the PWM is a log-scale matrix. Given W as the PWM, the matrix entries $W_{i,j}$ can be defined in Equation 2.4.

$$W_{i,j} = \log_2 \frac{p(b,i)}{p(b)} \qquad \text{Equation 2.4}$$

where $p(b)$ is the background probability of nucleotide b and $p(b,i)$ is the corrected probability of base b in position i of the alignment of binding sites. $p(b,i)$ is defined in Equation 2.5,

$$p(b,i) = \frac{f_{b,i} + s(b)}{N+1} \qquad \text{Equation 2.5}$$

where the $s(b)$ is a pseudocount function, and N is the total number of nucleotides at the binding site. If the frequency of a nucleotide is 0, this will produce undefined result when applying logarithm to the value. Therefore, a pseudocount is used in the formula to compensate the zero occurrences of the nucleotides (Wasserman & Sandelin, 2004). The pseudocount function is defined in Equation 2.6,

$$s(b) = \frac{1}{4}\sqrt{N} \qquad \text{Equation 2.6}$$

where the value $1/4$ is used by assuming that the background probability of each nucleotide

Σ is equal.

The consensus representation can be converted into PWM (Sumazin et al., 2005) and vice versa (Gao, Liu, & Ruan, 2017). PWM is the most widely used representation (Nishida, Frith, & Nakai, 2009; Sinha, 2006) method due to its preservation of the motif variations. The consensus representation causes information loss because it only captures the information of the most dominant nucleotide(s) in each position of a motif (Wasserman & Sandelin, 2004). For instance, in the case of "AA[AC]GTTAT", it only depicts the most frequent nucleotide(s) but failed to indicate the relative frequency of all nucleotides. In actual fact, some of the less frequent nucleotides are important for TF protein binding as well. As a result, consensus representation fails to describe the quantitative information of the binding sites (Schneider, 2002).

The motif profile can often be visualised using a sequence logo (Schneider & Stephens, 1990). It is the most widely used visual representation for TFBSs in the past 20 years (Gao et al., 2017; N. K. Lee & Oon, 2013). Figure 2.2 shows an example of a sequence logo visualisation.

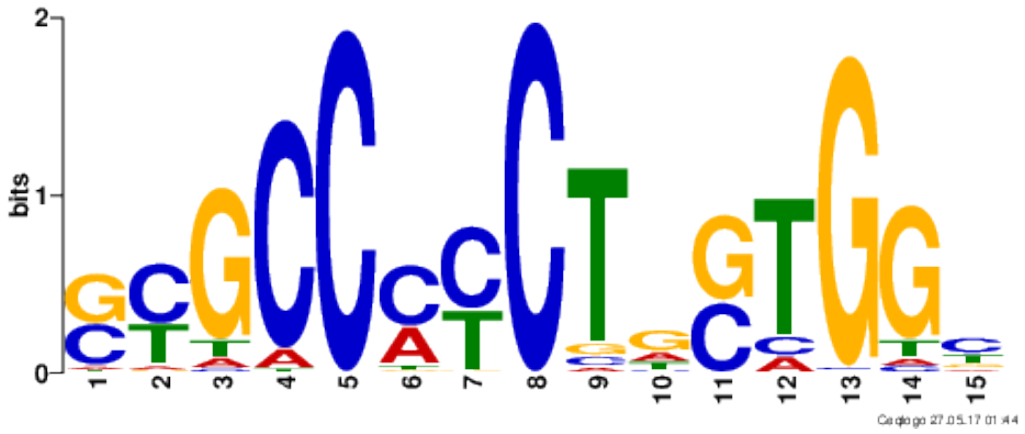


Figure 2.2: Example of sequence logo.

The height of the letter is calculated by using Equation 2.7,

$$\text{height} = f_{b,i} \times R_i \qquad \text{Equation 2.7}$$

where R_i is the amount of information presented at position i. R_i is defined in Equation 2.8,

$$R_i = 2 - (H_i + e_n) \qquad \text{Equation 2.8}$$

where H_i is the uncertainty of position i and e_n is the correction factor for n letters. H_i is defined in Equation 2.9,

$$H_i = -\sum_{b=a}^{t} f_{b,i} \times \log_2 f_{b,i} \qquad \text{Equation 2.9}$$

e_n is defined in Equation 2.10 according to MEME,

$$e_n = \frac{1}{\ln 2} \times \frac{3}{2n} \qquad \text{Equation 2.10}$$

A sequence logo depicts two important conservation information of a motif. Firstly, the sequence logo depicts the conservation level in each multiple-alignment of a motif in bits. A nucleotide with 2 bits indicates the maximum conservation of the DNA sequence. For example, the fifth and the eighth positions of "C" in Figure 2.2 are 2-bit height, which indicates that letter "C" has full conservation in fifth and eighth positions. Besides that, the total height of the stacked letters indicates the conservation level measured by using information content concept at the specific position (N. K. Lee & Oon, 2013). Secondly, the sequence logo also illustrates the relative frequency, $f_{b,i}$ in Equation 2.9, of the four nucleotides Σ represented by each height. Sequence logo, while is useful for analysis of

the characteristics of a motif, it is inappropriate to be used as evaluation metric because it does not provide quantitative information that is necessary for accurate comparison between motifs (N. K. Lee & Oon, 2013).

2.4 Datasets for computational motif discovery

Besides the algorithms involved in the motif discovery, different types of datasets are also concerned by the researchers. Decades ago, there were limitations in the technology used to sequence the whole genome. Hence, the datasets involved are usually small eukaryotic datasets. The earliest motif discovery tools targeted these datasets. For instance, MEME command-line tool by default only reads the small size datasets which are less than 100,000 bp, and GAPWM by default only reads 5,000 sequences.

With the advancements of the immunoprecipitation technology, chromatin immunoprecipitation (ChIP), ChIP with paired end tags (ChIP-PET), ChIP-on-chip (or ChIP-Chip) which is coupled with microarray, and ChIP-seq (ChIP sequencing) which the ChIP is combined with massively parallel DNA sequencing have been proposed. ChIP-on-chip and ChIP-seq are next generation sequencing (NGS) methods used by researchers to produce genome-wide datasets with the regions bound *in vivo* by a selected TF as several thousands of bp down to 300 bp long (Zambelli, Ré, & Pavesi, 2009; Zambelli et al., 2013). Classic *de novo* motif discovery tools were designed to input a few hundred sequences of co-regulated genes only. However, with the NGS techniques, the input sequences become shorter and centred on the actual TFBS around 50–200 bp around the peak. The regions of interested that are larger than actual TFBSs are the perfect case study for motif discovery, because by using genomic regions, it is able to lead to the better results using promoter analysis (Zambelli et al., 2013). As a result, NGS like ChIP-seq have rapidly

become the standard in motif discovery.

There are several annotated motif databases available online. One of the most commonly used DNA databases is JASPAR which contains the TFBSs data in PWM format (Sandelin, Alkema, Engström, Wasserman, & Lenhard, 2004). JASPAR provides an API for users and developers so that they can use their tools to check the results by comparing to the JASPAR database to identify true motifs. Similarly, TRANSFAC (Wingender, 2008) database contains the transcription factors and the profile information for both DNA and protein. JASPAR and TRANSFAC databases are the well-known databases to be used for comparing the true motifs with the discovered candidate motifs to evaluate the performance of the algorithm.

ChIPBase (K. R. Zhou et al., 2017; Yang, Li, Jiang, Zhou, & Qu, 2013) is another database that provides the thorough annotation and discovery of the transcription factor binding maps from ChIP-seq data. The source of human datasets of ChIPBase are collected from Encyclopedia of DNA Elements (ENCODE) (Consortium, 2012) and Gene Expression Omnibus (GEO) (Barrett et al., 2009). ENCODE project organises the mapped regions of transcription and the transcription factor association. On the other hand, GEO is a public repository for high-throughput gene expression data. It contains functional genomic data including transcription factor binding. By using the database, we are able to access curated ChIP-seq datasets and download them for the experiments.

2.5 Motif discovery approaches

2.5.1 Probabilistic approach

Expectation Maximization (EM)

Probabilistic approach motif discovery uses stochastic methods for sequence alignment. The goal of "local multiple alignment" (Smith & Waterman, 1981) is to locate short patterns of local segment of sequences that are conserved sequences. Examples of probabilistic algorithms proposed to solve the local alignment include Expectation Maximization (EM) (C. E. Lawrence & Reilly, 1990) and Gibbs sampling (C. Lawrence et al., 1993).

Multiple EM for Motif Elicitation (MEME) (Bailey & Elkan, 1995) is an unsupervised learning based on the expectation maximization (EM) (C. E. Lawrence & Reilly, 1990). According to Bailey and Elkan (1995), identifying the motifs can be expressed as given a dataset of sequences by assuming that it contains a single motif, then locating the starting position of the motif in each sequence and describing the motif. EM is able to solve this problem as when given the length of the motif, it estimates the probability $P_{i,j}$ of the motif starts in position j of the sequence i. The $P_{i,j}$ is then used to estimate the frequency f of the nucleotides A, C, G, and T, in each column of the motif. The estimation is repeated until the changes of f is very small. There are two limitations of EM. Firstly, the difficulty of choosing the initial value of the f. Secondly, EM assumes that each sequence of the dataset contains exactly one motif. MEME algorithm solves the two limitations. MEME runs repetitively with different starting points derived from the subsequence. The starting point which produces the highest likelihood will be chosen as the motif. The steps are repeated several times to discover further motifs. Finally, motifs are probabilistically removed after they are found to ensure that other motifs can be discovered in the same set of sequences. MEME

algorithm assumes that each discovered motif nearly forms by at least one subsequence in the dataset. Compared to other algorithms, MEME can estimate the motif width by itself, without providing motif width as a parameter (Hu et al., 2006). Third order Markov model is used as model for the background sequences. There are three types of sequence models supported by MEME: one occurrence per sequence (OOPS); zero or one occurrence per sequence (ZOOPS); and two-component mixture (TCM). The OOPS model assumes that there is exactly one occurrence of a motif in a sequence. ZOOPS is a generalised OOPS that assumes zero or one occurrence of a motif in a sequence. Lastly, TCM model assumes that there are zero or more non-overlapping candidate motifs occurred in each sequence. According to Bailey, Williams, Misleh, and Li (2006), in order to successfully discover the motifs using MEME, the input sequences have to be prepared. The sequences should be less than 1000 bp long and contains only a few background sequences. It is not suitable for genome-wide motif discovery. Additionally, it suffers computational problems on ChIP-chip or ChIP-seq datasets because these datasets have thousands of binding regions for a single TF (Ma et al., 2012; Shi et al., 2011).

MEME Suite is a web service to offer a unified portal for online motif discovery (Bailey et al., 2009). GLAM2 is a generalised gapless Gibbs sampling algorithm (Frith, Saunders, Kobe, & Bailey, 2008). MEME Suite integrates three motif scanning tools: MAST sequence homology search (Bailey & Gribskov, 1998), Find Individual Motif Occurrences (FIMO) (Grant, Bailey, & Noble, 2011), and GLAM2SCAN (Frith et al., 2008). The MEME algorithm is enhanced with the Gapped Local Alignment of Motifs (GLAM2) algorithm to solve the gap problem of the motifs. In addition, Discriminative Regular Expression Motif Elicitation (DREME) (Bailey, 2011) discovers short motifs without gap for large-scale datasets and Tomtom (Gupta, Stamatoyannopoulos, Bailey, & Noble, 2007) which is used for

comparing similarities between the motifs using E value to the databases of the motifs. FIMO is a tool that scans DNA or protein sequences according to the motifs in PWM or PFM format. The tool calculates log-likelihood ratio score for each motif. The log-likelihood scores are converted to the p-values and estimate the false discovery rates (FDRs). FIMO is able to show the locations of the top-scoring occurrences of the target motif. Hence, it is able to be used for precision and recall calculation.

MEME-ChIP (Machanick & Bailey, 2011) that targets for ChIP datasets is also a part of MEME Suite. MEME-ChIP is a computational pipeline that trims the sequences to 100 bp, then uses DREME to discover short motifs and 600 randomly selected sequences are scanned by MEME. MEME-ChIP also runs AME (McLeay & Bailey, 2010) on all trimmed sequences to calculate the statistical enrichment using known motifs in the JASPAR database. MEME-ChIP is a script that utilises the MEME program with the parameters that can scan the input datasets in a faster manner. Firstly, MEME-ChIP generates the Markov model from the FastA file. Then it invokes the other utilities such as fasta-most, fasta-dinucleotide-shuffle, fasta-subsample, and psp-gen, to calculate the parameters from the file and shuffle the sequences in the FastA file. Next, MEME is invoked to perform the motif discovery with the re-ordered data sequences. Finally, DREME is also invoked from MEME-ChIP to discover short motifs which cannot be found by MEME (Machanick & Bailey, 2011).

ChIPMunk (Kulakovskiy et al., 2010) is developed to target on ChIP-Seq datasets. It employs a greedy optimisation strategy combined with EM (Lihu & Holban, 2015; Zambelli et al., 2013). The basic algorithm of ChIPMunk calculates the discrete analog of Kullback-Leibler divergence as the discrete information content (KDIC). It also implements zero-or-one-occurrence-per-sequence (ZOOPS) mode to search the local alignment that has

maximum KDIC and constructs the PWM. The weights of each sequence position forming the sequence profiles are assigned to the initial sequence data. The PWM is optimised by rebuilding the alignment from the words of maximal PWM scores and from new motif occurrences. The convergence is achieved after the profile values are substituted by their maxima within a sliding window, while the window length is equal to the motif length. The advantage of ChIPMunk is that it can process tens of thousands of data sequences without truncating long DNA segments. According to Kulakovskiy et al. (2010), ChIPMunk can identify the true motifs with same or even better quality as MEME and the study also showed that it is faster than MEME in motif discovering.

Gibbs sampling

Other than EM, Gibbs sampling (C. Lawrence et al., 1993) is another commonly used stochastic search technique. Gibbs sampling uses iterative sampling for "local multiple alignment". It employs the Markov Chain Monte Carlo (MCMC) approach so that every step is based on random sampling and the results are depending on the results of the previous steps like EM. The main difference of Gibbs sampling and EM is that Gibbs sampling takes a weighted sample from the subsequences, while EM takes a weighted average across all subsequences (D'haeseleer, 2006). Gibbs sampling searches a relatively small amount of ungapped patterns from each input sequences with specified width. There are two evolving data structures are maintained. First, a set of probabilistic model variables $[q_{i,1}, \ldots, q_{i,n}]$ is used to describe the residue frequencies at each position of a single pattern. The variables are also accompanied by background frequencies $[p_1, \ldots, p_n]$. Second, a set of variables is used to describe the probabilistically inferred position of the patterns a_k of each sequence. The algorithm is initialised using a random starting position of a single sequence. Then, it

iteratively updates the pattern for $q_{i,j}$ and p_j. Next, it calculates the probabilities Q_x from $q_{i,j}$ and P_x from p_j. Consequently, it can calculate the weight for the segment x in Equation 2.11 as follows:

$$A_x = Q_x / P_x \qquad \text{Equation 2.11}$$

Finally, the new a_k can be obtained. The fundamental concept iteratively constructing accurate pattern then the locations.

One tool that uses Gibbs sampling is Aligns Nucleic Acid Conserved Elements (AlignACE) (Hughes, Estep, Tavazoie, & Church, 2000; Roth, Hughes, Estep, & Church, 1998). AlignACE allows multiple types of motifs to be found because it uses iterative masking of the discovered TFBSs in the positive sequences. The maximum a posteriori (MAP) score is used to measure the over-representation motif relative to the random occurrence in background sequences. Furthermore, AlignACE automatically clusters similar motifs together. W-AlignACE was proposed with improved Gibbs sampling algorithm (X. Chen, Guo, Fan, & Jiang, 2008) in order to handle ChIP datasets. Therefore, W-AlignACE employs a sequence weighting scheme on AlignACE algorithm to find the PWM. The sequence weighting scheme assigns a weight proportional to the logarithm fold change of the mRNA expression of downstream gene. Consequently, the maximization of the combination of the binding sites and the expression data can be observed and the PWM is discovered.

MotifSampler (Thijs et al., 2001) is another tool that uses Gibbs sampling method. It improves the Gibbs sampling with two modifications. It uses a higher-order Markov chain background model to estimate the number of motifs in a sequence. The probability generated

by background model is denoted as $P = P(S|B_m)$, where S is the site sequence and B_m is the background model. The probability of a binding site x with length l is denoted as $Q_x = P(S|\theta)$ where θ is a position probability matrix of nucleotides of the binding site x. Consequently, a weight A_x can be calculated as in Equation 2.11. The distribution of the normalized weights A_x is updated to find the alignment vector that maximizes the ratio of the target binding site to the background probability.

BioProspector (X. Liu, Brutlag, & Liu, 2001) uses the Monte Carlo method, which involves repeated random sampling to identify the significance of the discovered motif from the score distribution. Furthermore, Gibbs sampling has been improved to allow discovering gapped motifs. BioProspector implements two thresholds, T_H and T_L, to the threshold sampler process. All the non-overlapping segments of the sequence s with score $\theta > T_H$ are added to the motif and also the positions added to alignment a_s. The segments with the score $T_L \leq \theta \leq T_H$, one segment will be chosen with probability proportional to $A_x - T_L$ where A_x is calculated in Equation 2.11. This allows the motif discovery to achieve convergence quicker when sampling only the segments with the score within $[T_L, T_H]$.

GibbsST (Shida, 2006) is another tool incorporating the Gibbs sampling method. In addition, GibbsST improved the motif discovery performance by combining thermodynamic method, namely simulated tempering, with Gibbs sampling. Simulated tempering is an algorithm related to simulated annealing but uses dynamic temperatures (Marinari & Parisi, 1992). Simulated tempering is proposed because Gibbs sampling is prone to trap in the local optima. By using simulated tempering, the Gibbs sampling efficiency is improved. Simulated tempering is a robust solution which can be applied to bioinformatics problems to escape from local optima. GibbsST was compared with classic Gibbs sampling. Results have

shown that the performance coefficient of GibbsST is significantly superior to classic Gibbs sampling. Moreover, classic Gibbs sampling achieved extremely poor convergence to the global optimum as the initial values are randomly selected.

In probabilistic approach, methods such as EM, stochastic Gibbs sampling, and Hidden Markov Models are commonly used. However, they have limitations due to sensitivity to the initial setting of the parameters. Moreover, these methods may be trapped in the local optima because they are local search (Chan, Leung, & Lee, 2007).

2.5.2 Enumerative approach

The enumerative method (Das & Dai, 2007; Sinha & Tompa, 2003) is also known as the deterministic method (Brazma et al., 1998; Sandve & Drabløs, 2006), word-based method (Ichinose, Yada, & Gotoh, 2012), or consensus-based method (Pavesi, Mauri, & Pesole, 2004). Enumerative methods are more scalable compare to probabilistic approach (Das & Dai, 2007), such as expectation maximization (EM) (Bailey & Elkan, 1994) and Gibbs sampling (C. Lawrence et al., 1993). This is because enumerative method guarantees global optimum by exhaustively counting all the patterns in a given set of sequences to detect the ones that are overrepresented from the background frequencies (Ettwiller, Paten, Ramialison, Birney, & Wittbrodt, 2007; Ichinose et al., 2012).

This method identifies the motifs in the form of oligos, regular expressions, or mismatch expressions (Sandve & Drabløs, 2006). Oligo or oligonucleotide is a string of a certain length of permutation on a set of alphabets. Oligo can be represented as words or strings (Rombauts, Florquin, Lescot, & Van de Peer, 2003). According to Defrance, Janky, Sand, and van

Helden (2008), oligo can be represented by a 4-letter alphabet Σ or 15-letter alphabet as IUPAC code in Table 2.1, for instance AANNGAATTGK. On the other hand, regular expression is a pattern describes a set of strings of characters. Regular expression allows wildcards but not variable length gaps (Bailey, 2011). An example of regular expression is [AT][AT]C{2,3}GC, which means the first and second words be either A or T, then C can occur two or three times, followed by G and C. Mismatch expression can be denoted as (l, d) where l is the l-mer and d is the number of mismatch. The mismatch of the words can be evaluated by using Hamming distance such as AACCGT and ACCCGA has two mismatches on the second word and the last word.

The enumerative method typically relies on exhaustive enumeration of the words to guarantee global optimum by covering the space of all possible motifs for a specific motif model description (D'haeseleer, 2006). It is possible to discover the best solution. However, when the length of the pattern increases, the running time grows exponentially (Brejová et al., 2000). Enumeration of the solution space for all possible oligo combination is computationally impractical (Zambelli et al., 2013), for example DREME limits the motif search to maximum 8 bp (Bailey, 2011) and Weeder limits to 12 bp (Pavesi, Mereghetti, Mauri, & Pesole, 2004; Ichinose et al., 2012). Though different heuristic methods are used to prune the search space or using the mismatches restriction, they do not solve the problem in general (Pavesi, Mauri, & Pesole, 2001). Besides that, it can only identify short patterns and less degenerated (Pavesi et al., 2001).

Mismatch Tree Algorithm (MITRA) (Eskin & Pevzner, 2002) is one of the tools that uses mismatch expression. It uses a mismatch tree data structure to split the pattern space into disjoint subspaces with a given prefix. It reduces the pattern discovery into smaller

sub-problems. The pairwise similarities are used to build a graph where the vertices are the l-mers, while the edges are the connection of two similar l-mers. Based on WINNOWER (Pevzner & Sze, 2000) approach, MITRA prunes the non-signals edges from the graph so that the signals are remained. MITRA uses trie (prefix tree) traversal to discover the motifs in the input strings. It is able to discover both monad and composite pattern in the DNA sequences. Monad pattern is a unit of the word in DNA and composite patterns are a group of monad patterns relatively close to each other. Monad pattern can be expressed as a single continuous block of sequence with the form of $(l, d) - k$ where l is the length of the pattern, d is the maximum number of the mismatch, and the k is the minimum number of times the pattern repeated (Satya & Mukherjee, 2004). On the other hand, dyads are two units of the pattern and the spaced dyads for the gapped motifs. MITRA can be extended with insertion and deletion operations. MITRA has been tested with monad motifs from Buhler and Tompa (2002), dyad signals from Gelfand, Koonin, and Mironov (2000), and composite regulatory signals from GuhaThakurta and Stormo (2001). The results showed that MTIRA was able to recover the dyad signal and the composite pattern which are not able to be found by CONSENSUS, MEME, GibbsSampler, and ANN-Spec.

A motif algorithm for detecting enrichment in multiple species (Amadeus) was developed for genome-wide *de novo* motif discovery (Linhart et al., 2008). Amadeus is a pipeline of filters where each phase refines the list of candidate motifs from the previous phase. There are several phases in the algorithm: preprocess, mismatch, merge, greedy, postprocess, and pairs analysis. The preprocess phase evaluates all the k-mers to determine the most promising motifs. The mismatch phase changes a k-mer to a list of k-mers by introducing degenerate positions into the k-mers. The merge phase combines two similar motifs together recursively so that no new high-scoring similar pairs exist. The greedy phase involves calculation of a

PWM from every motif, then optimises the PWM using an iterative process similar to greedy EM. The process is repeated until the score does not improve any more. The final phase, postprocess, removes all the redundant motifs. Human datasets (CREB, E2F, NANOG, NRF1, TP53, SOX2, and SRF) and mouse datasets (Foxp3, Mef2, Myod, and Myog) were tested. Amadeus was compared with AlignACE, MEME, YMF, Trawler, and Weeder by using Tompa's (2005) benchmark study. The results showed that Amadeus significantly outperformed other tools with 62% success in terms of motif recovery rate on the known motifs. Amadeus also discovered two novel motifs ACTACAWYTC and CTCGCGAGAT. It was also demonstrated that Amadeus is the fastest tool among the other tools while AlignACE and MEME are extremely slow with the large datasets.

Seed-driven algorithm is also used in enumerative motif discovery. It evaluates the seeds from a very simple pattern, then grows possible seeds to full motifs (Sandve & Drabløs, 2006). For instance, TEIRESIAS (Rigoutsos & Floratos, 1998) is a seed-driven exhaustive algorithm. TEIRESIAS has two operation phases. They are scanning and convolution. The scanning phase will scan for the patterns which will be used in convolution phase. During the scanning phase, the patterns are broken into multiple non-maximal and overlapping pieces. Therefore, in the convolution phase, the pieces are convolved together to recover the original patterns. The phases are repeated as the patterns gradually grow larger until all maximal patterns are generated.

Motif Discovery scan (MDscan) (X. S. Liu, Brutlag, & Liu, 2002) is an enumerative deterministic greedy algorithm targets on genome-wide ChIP-array datasets. MDscan algorithm combines the advantage of word enumeration and PWMs. MDscan uses the words sampled from the input sequences as seed pattern to scan for their occurrences with

mismatch in subset of the input sequences. Those matches words of a seed pattern are used to form PWM and then it is scored by using the MAP function with third order Markov chain background model. The 10–50 candidate motifs from the highest scores are used for the next iteration, in which each PWM of a motif is used to scan for matching words in the remaining input subset. A matched word is added to the PWM if it improves the MAP score. MDscan performance was compared to BioProspector, AlignACE, and CONSENSUS (Hertz & Stormo, 1999) on the yeast datasets Ste12, Gal4, Rap1, Mbp1, Swi4, Swi6, Fkh1, Fkh2, Ndd1, Mcm1, Ace2, and Swi5. The study showed that MDscan was able to predict motifs that failed to discovered by BioProspector, AlignACE, and CONSENSUS. For instance, SCB motif from Swi4 and Swi6, SFF motif from Fkh1, MCM1 motif from Fkh2, MBF motif from Swi6. Due to the heuristic nature of the algorithm, MDscan is approximately 35 times faster than BioProspector and 400 times faster than AlignACE.

A suffix tree is a data structure that represents a string in a tree structure. A suffix tree T for a string with length n, $S = s_1, s_2, \ldots, s_n$ has exactly n leaves numbered from 1 to n. Each internal node has minimal two children except the root node. Each edge is labelled with non-empty substring of S. Two edges leaving from the same node have different first characters in the substring. The concatenation of the substring according to the path from the root to any leave spells out the suffix $S_i, \ldots, S_n$. The following is an example of a suffix tree of word enumeration,

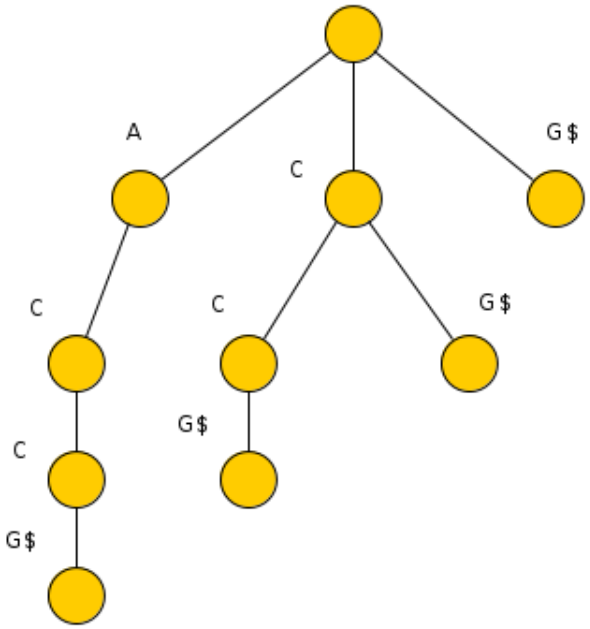


Figure 2.3: Suffix tree for ACCG

Figure 2.3 uses the "$" as termination character to denote the end of the string. Furthermore, a generalised suffix tree is constructed by a set of words instead of single word. And each word uses different terminator.

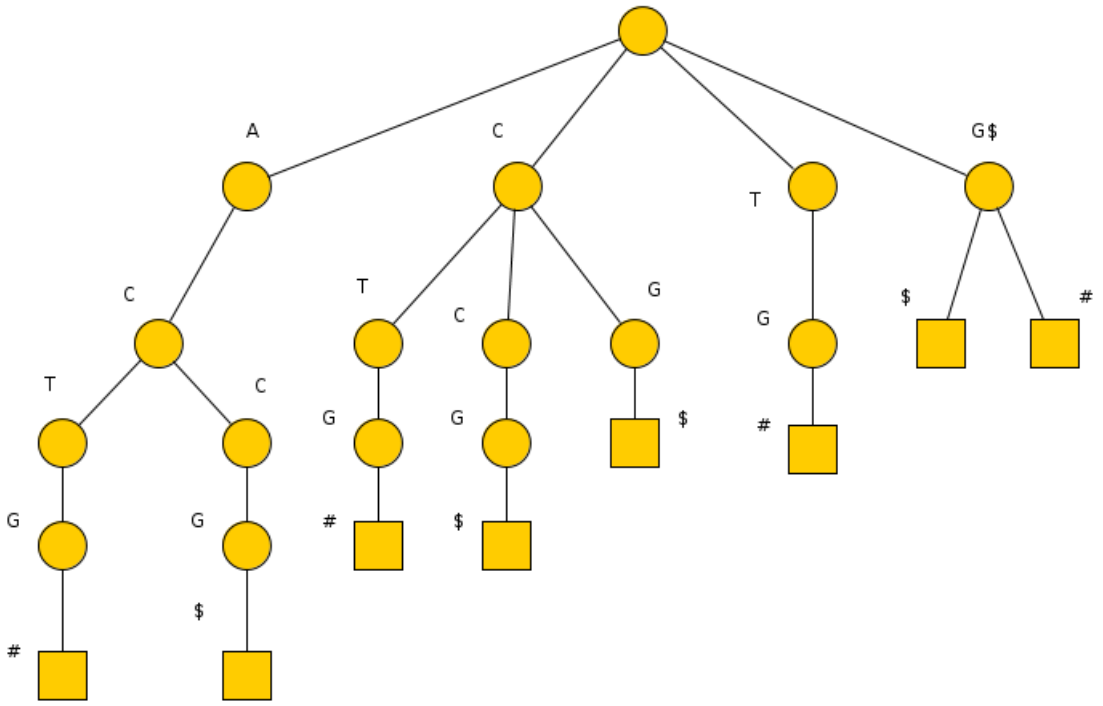


Figure 2.4: Generalised suffix tree for ACCG$ and ACTG#

For instance, Figure 2.4 shows a generalised suffix tree for both ACCG and ACTG, yet ACCG has the terminator character "$" and ACTG has the terminator character "#". After the generalised suffix tree is constructed based on the data sequences, the algorithm can search through the path for the mismatches.

Weeder (Pavesi et al., 2001) uses suffix tree structure for fast counting match hits of a candidate motif represents as a string. The algorithm does not require the input of the exact length of the patterns. The Weeder algorithm starts from the root node by expanding the path recursively. If the number of pattern has mismatch value greater than e (such as 4), then the algorithm will "weed out" the path. The algorithm is sped up by enforcing some restrictions on the positions in which mutations are allowed. Weeder was evaluated by using Pevzner and Sze (2000) challenge to find a signal in a sample of 20 sequences of the fixed

length. The samples are selected from independent and identically distributed sequences. The results were compared to CONSENSUS, Gibbs sampler, MEME, WINNOWER and SP-STAR (Pevzner & Sze, 2000). Weeder showed that it has the highest probability of finding a signal (0.89–0.95) of length 15 with 4 mismatches compared to other algorithms.

Suffix tree is also used in RISOTTO to maximize the extensibility information (Pisanti, Carvalho, Marsan, & Sagot, 2006). The algorithm is written in C language and it implements the maximal extensibility information to RISO (Carvalho, Freitas, Oliveira, & Sagot, 2005). The algorithm is to improve the pattern enumeration for searching the long motifs. RISOTTO uses depth-first search to traverse through the motif tree. Maximal extensibility performance was evaluated using randomly generated uniform distribution of the DNA letters with planted structured motifs.

Trawler is a tool that uses suffix tree approach to examine the large pattern sets (Ettwiller et al., 2007). The standard normal approximation to the binomial distribution is used for evaluation and z-score is used for statistical significance. It evaluates the z-score for a random set of sequence obtained from the background compared to the z-score for the sample sequences. Trawler does not have any expectation on the number of motifs to be discovered. It clusters the similar motifs together to form degenerated traits of the binding sites. The benchmarking datasets used by Trawler are Tompa's (2005) 52 datasets, E2F1 and E2F4, SOX2, OCT4, NANOG, NOTCH1, CREB1, 203 yeast datasets, and so forth. Trawler was compared to AlignACE, MDscan, MEME, Weeder, and some other tools. The results showed that none of the tools were able to the discover the correct motif of NOTCH1 except Trawler and Weeder.

Trawler_standalone (Haudry et al., 2010) was designed and optimised to analyse chromatin immunoprecipitation (ChIP) datasets. To discover the motifs, it firstly searches for the overrepresented motifs. Then, the motifs are clustered into families using either strongly connected component, SOM or k-means clustering. The motifs are mapped to the sample sequence. Finally, the results are presented with a user-friendly interface.

Enumerative method is able to discover both ungapped and gapped motifs. However, discovering gapped motifs is complicated due to the highly degenerate positions in a motif which increases the search space (C. Y. Chen et al., 2008). Regulatory Sequence Analysis Tools (RSAT) (Defrance et al., 2008; Thomas-Chollier et al., 2008) are used to detect over-representation or under-representation of the oligonucleotides or dyads (spaced pairs). A space dyad consists of two words that are separated by spacers (H. Li, Rhodius, Gross, & Siggia, 2002). RSAT is able to predict the motifs in the *cis*-regulatory modules and also discover phylogenetic footprints in promoters of orthologous genes.

C. Y. Chen et al. (2008) developed a method for both gapped and ungapped motif discovery. The method uses a ranking system which assesses preferential occurrence of the motif, the number of positions, and the degree of evolutionary conservation of the potential motifs. A position based ranking algorithm and a conservation-based ranking algorithm are proposed to join the information from intermediate motif patterns. The position concurrence is the source of signal density on the true transcription factor binding sites. This is because the true motifs can evolve into different forms through evolution.

AMD (Shi et al., 2011) is another enumeration-based method. AMD is successful in identifying both gapped and ungapped motifs and fast. The tools such as MEME and Weeder

are computational intensive with the ChIP-chip or ChIP-seq datasets. Thus, AMD solves running time problem for the large datasets. AMD can efficiently determine the long motifs in large datasets. Moreover, the motif discovery performance was similar to Amadeus, as Amadeus is outperformed than MEME, AlignACE, YMF, Weeder, and Trawler. AMD uses step-wise motif discovery method. There are five processes involved: (1) core motif filtering; (2) degeneration; (3) extension; (4) refinement; and (5) redundancy removal. AMD has the high success rates on yeast and metazoan data, which includes mouse, human, fly, and worm, with 38–45% and 44–50% respectively. It also achieves the highest success rate (79%) in terms of all motifs tested by using 0.75 cut-off score, where Amadeus, Weeder, MEME, and MDscan achieves success rates 69%, 43%, 7%, and 0% respectively. This research discovered that AMD is a very fast algorithm compared to MEME, AlignACE, and Weeder. Nevertheless, the outputs of AMD do not show the location of the binding sites, only the consensus and the PWM.

An efficient combinatorial algorithm was developed to solve the search complexity problem which is independent of the size of the alphabet (Kuksa & Pavlovic, 2010). Firstly, it finds a set of candidate motifs. Then, the intersections of the candidate neighbourhoods are constructed. The set is represented with "stems" or patterns with wildcards. Common patterns are discovered by pruning the stems that does not appear in all input sequences. This is to construct a set of candidates that contains motifs. After the candidates are constructed, selection algorithm based on the Hamming distance between pairs of patterns is used. The sort is used so that same k-mers are grouped together. Then the algorithm will scan to create a list of sequences which the motifs appear. Finally, the algorithm will generate the stem, which is independent of the alphabet size. This is done by considering the distance between the pair of k-mers. The algorithm reduces computational complexity comparing to MITRA.

Furthermore, there is an enumerative method used with bipartite graph for motif discovery in yeast (Kellis, Patterson, Birren, Berger, & Lander, 2004). A bipartite graph is built to connect annotated candidates with the predicted candidates based on the similarity of the sequence. The algorithm was developed based on graph theoretic framework. The similarities between the genes are represented as a bipartite graph connecting genes of two species. The edges connecting two genes are weighted. Then, the graph is separated iteratively to construct subgraphs. The edges which are less than 80% of the maximum weight edge are removed. The algorithm is based on exhaustive enumeration and testing the short sequence patterns to discover strongly conserved motifs. As opposed to Bipad, Kellis et al. (2004) uses bipartite graph for motif discovery, yet Bipad is discovering the bipartite pattern using greedy methods to discover the bipartite alignment (Bi & Rogan, 2004).

Hegma (Ichinose et al., 2012) proposed to solve the large-scale motif discovery by applying DNA Gray code and equiprobable oligomers to solve the clustering problem and oligomer bias. A Gray code is a binary number coding system that the adjacent numbers differ by one single bit. The Gray code is applied to DNA sequence using quaternary numbers instead of binary numbers. Using the DNA Gray code, it will produce an ordered tree structure. As a result, a depth-first search algorithm can be used to search through the Gray code. Equiprobable oligomers are the oligomers which have an approximately equal background probability so that they can be combined naturally with DNA Gray code. The benchmark tests showed that Hegma outperformed Weeder Web (Pavesi, Mereghetti, et al., 2004).

2.6 Other machine learning approaches

DNA motif discovery is considered a machine learning task because it consists of extracting a general rule from the training set which contains both positive and negative datasets (Brazma

et al., 1998). DNA motif discovery involves two problems: classification problem and conservation problem. The purpose of a classifier is to identify the "true" sequences from the "false" sequences. Machine learning can generally be divided into two categories: supervised and unsupervised. Supervised learning requires the training data that contains the labelled output. The labelled output is the known true positive of the data. The supervised learning algorithm is used to train the machine so that it can be given the input and infers or predicts the output accurately after the training. On the other hand, unsupervised learning uses the unlabelled training data which has no desired output. Therefore, unsupervised learning will train the machine so that it is able to cluster the data into groups by similarity.

In bioinformatics, one unsupervised learning algorithm is the self-organizing map (SOM) (Kohonen, 1998) neural network. SOM is able to be used in motif discovery because it clusters the similar sequences together. The sequences are represented in PWMs. The purpose of the SOM training is to adapt the weights vector of every node to form a topological map which has the spatial locations of the nodes correspond to the intrinsic features. Compared to unsupervised learning, supervised learning algorithms are more widely used in bioinformatics. For instance, Time-Delay Neural Network (TDNN), Support Vector Machine (SVM), and Deep Learning. Deep Learning is the state-of-the-art machine learning in motif discovery, because it is able to handle genome-wide datasets and perform high accuracy prediction.

2.6.1 Unsupervised learning

SOMBRERO (Mahony et al., 2006) uses self-organizing map (SOM) neural networks for motif discovery. SOMBRERO uses PWM to represent the motif. A log-likelihood ratio of

a DNA string is used to evaluate the string similarity to a motif. Since general structure of an SOM is a 2D-network of interconnected nodes, SOMBRERO uses a PWM at each node to represent the features in the input sequences. During training, each motif evolves to depict a distinct feature of the data. The nodes will influence each other and produce similarity in the adjacent nodes. The PWM is updated iteratively at each node. SOMBRERO is able to discover multiple distinct motifs in a single dataset and distinguish the weak motif signals from the large or noisy datasets. SOMBRERO has been compared with MEME and AlignACE using ten yeast genomic sequence sets from Promoter Database of *S. cerevisiae*. Each sequence length is minimal 500 bp and contains at least one instance of certain motif. The results showed that SOMBRERO has the best performance with the lower false positive rate. The datasets of *Drosophila* were also tested. The datasets contain five TFBSs. SOMBRERO showed superior performance, with the lowest false negative rate on three out of five motifs.

SOMBRERO node models are unable to represent the motif and the background sequences simultaneously, because the PWM is designed to represent motif only. This limitation is overcome by SOM based Extraction Algorithm (SOMEA) (N. K. Lee & Wang, 2011). SOMEA uses a hybrid node model which is composed of PFM and static Markov chain to enhance the two types of sequence signal. Markov chain is used to represent the background because the background noises are considered as generated by Markov process (N. K. Lee & Choong, 2013). The two models are weighted to reflect the distribution of foreground and background k-mers assigned to a node. SOMEA was compared to SOMBRERO, MEME, AlignACE, and Weeder with eight datasets (one TF from *E. colli*, six TFs from *Homo sapiens*, and one TF from *S. cerevisiae*). SOMEA performs significantly better than SOMBRERO in terms of recall rates, F-measure values, and discrimination ability. However, MEME is better

than SOMEA in terms of average F-measure values in four out of five of the datasets.

Due to the randomness of the background sequences in DNA datasets, SOM has limitations in forming dense clusters. N. K. Lee and Choong (2013) proposed preprocessing step to filter the noise before using SOM for motif discovery. Firstly, a threshold value is obtained by using mismatch filtering function. Secondly, k-mers from the input are used as seeds to group the potential patterns. Some patterns cannot be grouped are assumed as noises, hence they are filtered. Another strategy is using a subset of the input sequences to generate the gapped motifs instead of using k-mers as seeds. The preprocessing filtering on the background sequences showed a promising result and it improves the prediction accuracy.

2.6.2 Supervised learning

Chromatin Signature Identification by Artificial Neural Network (CSI-ANN) (Firpi, Ucar, & Tan, 2010) is a framework that performs classification on enhancers using histone modification datasets. It applies Fisher discriminant analysis (FDA) to extract features which separate the two classes. Then time-delay neural network (TDNN) is used for classification. The traditional back propagation algorithm is very slow and prone to be trapped in the local minima. Thus, CSI-ANN implements the Particle Swarm Optimization (PSO) on the TDNN training. PSO deploys a population of solutions to explore the search space of optimal fitness values for the TDNN. Through PSO, the optimal architecture for the TDNN can be found and used for the classification. CSI-ANN was tested with histone modification data from ENCODE. The study showed that CSI-ANN achieved a 66.3% positive prediction value, or 11.6% more than HMM method (Won, Chepelev, Ren, & Wang, 2008).

SVM (Cortes & Vapnik, 1995) has been used in various biological applications (Furey et al., 2000; Hirose, Shimizu, Kanai, Kuroda, & Noguchi, 2007; Y. Zhang et al., 2011). It is widely used in bioinformatics because of its good performances, the ability to handle the high-dimensional and large datasets, and flexibility to model the diverse source of biological data (Ben-Hur, Ong, Sonnenburg, Schölkopf, & Rätsch, 2008). SVM adopts the concept that the input vectors are non-linearly mapped to high-dimensional feature space (Cortes & Vapnik, 1995). The decision surface is constructed in the feature space. The goal of the SVM is to find the decision boundary that maximally distinguishes the positive and negative training data. A larger margin of separation of the positive and negative classes can reduce generalisation errors (Ben-Hur et al., 2008).

The k-mers based approach has an advantage over PWM based approach (Ichinose et al., 2012). PWM requires large amount of data for optimisation. This approach is simpler as it determines whether the feature is present or absent. The kmer-SVM was developed (D. Lee, Karchin, & Beer, 2011) to predict the enhancers in the genome-wide EP3000 and CREBP datasets. It implements the k-mer frequency vector then uses inner product as the kernel function for two normalized frequency vectors. The results showed that kmer-SVM produced AUC above 0.9 regardless its types of kernel, types of tissue, or length of k-mers. However, when the k-mers length increases, the training will be more vulnerable to the noise (Ghandi et al., 2014). Therefore, gapped k-mers solution is proposed and gkm-SVM is developed. The gkm-SVM uses two parameters for gapped k-mers: l as the whole word length including gaps and k as then length of non-gapped word. Hence, $l - k$ is the number of gaps. The kernel function is the similarity score of the l-mer pairs. The gkm-SVM method applies a tree data structure to count mismatch between two l-mer pairs for the kernel matrix. The results showed that gkm-SVM is able to predict enhancers with higher accuracy consistently

compared to its predecessor, kmer-SVM, which is strictly overfitted when the length of the k-mer is greater than 10, yet gkm-SVM is not influenced by the length of l-mers. However, gkm-SVM achieved accuracy in terms of AUC 0.967 with $l = 14$ and $k = 6$, which is significantly higher than kmer-SVM (AUC 0.912 with $k = 10$).

Nonetheless, D. Lee (2016) stated that larger genomic datasets cause gkm-SVM scarcely to be trained because computing full kernel matrix requires memory resources proportional to n^2. An improved SVM called LS-GKM which can deal with large-scale dataset has been proposed. LS-GKM uses LIBSVM framework (Chang & Lin, 2011) for the integration of kernel functions. While gkm-SVM uses tree data structure and stores the list of pointers to the same matched nodes during tree traversal. LS-GKM improves this by maintaining the list of pointers to the matched bases of the sequence. Furthermore, multi-threading is used to speed up the tree traversal. The improvement of LS-GKM involves software implementation instead of algorithm improvement. Moreover, a radial basis function is employed in the space of the gapped k-mer frequency vectors to improve prediction accuracy.

EnhancerFinder (Erwin et al., 2014) is a tissue-specific enhancer predictor based on SVM. It employs a supervised learning technique, namely multiple kernel learning (MKL) (Kloft, Brefeld, Sonnenburg, & Zien, 2011), which is based on SVM. MKL allows EnhancerFinder to integrate multiple data types into a single discrimination function. The training data is labelled as enhancers and non-enhancers. Hence, the first step of EnhancerFinder is using the classifier to separate the enhancers in the context of interest from non-enhancers. In the second step, EnhancerFinder trains several classifiers to distinguish the candidate enhancers from the previous step. Consequently, the enhancers are classified into heart, limb, forebrain, midbrain, hindbrain, and neural tube. The training data is generated from VISTA Enhancer

Browser (Visel, Minovitsky, Dubchak, & Pennacchio, 2007). The results showed that prediction on H3K27ac, H3K4mel histone marks and p300 datasets achieved AUC 0.61, 0.72, and 0.68 respectively.

DEEP (Kleftogiannis, Kalnis, & Bajic, 2015) is a machine learning framework that employs ensemble of SVMs together with a simple ANN for the final prediction. The training data is gathered from ENCODE repository, FANTOM5 consortium (Andersson et al., 2014), and VISTA Enhancer Browser. As in the ensemble approach, the data are partitioned. Each partition is used to train an SVM model with Gaussian kernel function. The ensemble of SVMs are trained to classify the instances according to cell-lines and tissues. The predictions of the SVM models are finally passed to the simple ANN model derived from CSI-ANN to determine whether the candidate is an enhancer. Various histone marks and cell lines including p300 were used in the study. The findings showed that DEEP was able to achieve 90.2% accuracy based on FANTOM5 (Andersson et al., 2014) consortium datasets and 89.64% accuracy based on VISTA Enhancer Browser datasets (Leslie, Eskin, & Noble, 2002). According to Kleftogiannis et al. (2015), DEEP achieved better performance than CSI-ANN.

EnhancerPred (Jia & He, 2016) was developed for enhancer prediction. It uses bi-profile Bayes and pseudo-nucleotide composition (PseNC) to extract the features from the sequences. Bi-profile Bayes is used because it reflects both positive and negative samples. PseNC is based on pseudo amino acid composition (PseAAC) model (Chou, 2001). The fundamental concept of PseAAC it to avoid losing hidden information in the protein when extracting the features. Hence, PseAAC is in fact a set of discrete numbers derived from the sequence which is able to preserve the pattern information. In EnhancerPred, the PseNC

consists of trinucleotide composition. The radial basis function kernel is used for the SVM to perform the training on the extracted features. EnhancerPred uses a jackknife test to compare the accuracy with another algorithm, iEnhancer-2L (B. Liu, Fang, Long, Lan, & Chou, 2016). Findings have shown that EnhancerPred has better accuracy. To identify the enhancers and non-enhancers, EnhancerPred achieved accuracy of 77.39% and iEnhancer-2L achieved 76.89%. Moreover, to identify strong enhancers and weak enhancers, EnahcerPred achieved accuracy of 68.19% while iEnhancer-2L achieved 61.93%.

Convolutional Neural Network (CNN) is currently most widely used deep learning technique in machine learning and increases the performance of bioinformatics (Alipanahi et al., 2015; Kelley et al., 2015; Zeng, Edwards, Liu, & Gifford, 2016; J. Zhou & Troyanskaya, 2015). According to Min, Lee, and Yoon (2016), CNN can be used to discover useful recurring patterns in one-dimensional genomic sequences and two-dimensional data such as time-frequency matrices of biomedical signals. CNN is appropriate for application in DNA motif discovery, because (1) DNA sequences contain recurrence sequence patterns that are binding sites of TF proteins; and (2) CNN with GPGPU is able to learn large-scale datasets such as genome-wide data sequences more efficiently than traditional machine learning algorithms. For instance, DeepBind (Alipanahi et al., 2015) was proposed to predict sequence specificities of DNA- and RNA-binding proteins using CNN. The sequence specificities of DNA- and RNA-binding proteins are important to identify disease variants. Similar to DeepBind, DeepSEA (J. Zhou & Troyanskaya, 2015) learns DNA sequences from the large-scale chromatin-profiling data including TF binding to predict noncoding variants from the sequence. Besides that, Basset is an open source CNN learning application to learn functional activities of DNA sequences (Kelley et al., 2015). It targets on the genomic cell type datasets.

2.6.3 Genetic Algorithm

Genetic algorithm (GA) (Goldberg, 1989) is a search optimisation that simulates the process of natural evolution based on the evolutionary theory. GA is also being applied in motif discovery (Chan et al., 2007; Chan, Leung, & Lee, 2008; L. Li et al., 2007; L. Li, 2009; Z. Wei & Jensen, 2006).

Genetic Algorithm for Motif Elicitation (GAME) (Z. Wei & Jensen, 2006) is a motif optimisation tool. Nevertheless, GAME utilises numerous randomly generated PWMs as the starting points to discover the motifs. Therefore, GAME does not require the motifs from *de novo* motif discovery tools like BioOptimizer (Jensen & Liu, 2004). GAME uses tournament selection to select the parents for generating new offspring. Furthermore, two operations, ADJUST and SHIFT, are also used to alleviate the problem of premature convergence to local optima. GAME is slower than MEME and BioProspector in the computational speed. However, the performance of the motif site prediction result of GAME is far better than MEME and BioProspector in terms of F-score. For instance, GAME scores 0.77 yet MEME and BioProspector score 0.58 and 0.56 on average of all tested datasets.

GAPWM (L. Li et al., 2007) was proposed to improve the quality of existing PWM by using ChIP dataset. According to L. Li et al. (2007), PWMs in TRANSFAC and JASPAR have many PWMs but identified TFBSs are relatively small. Therefore, maximum-likelihood estimates may be poor because of insufficient data. GAPWM employs scoring function of a motif based on Match (Kel et al., 2003) as defined in Equation 2.12,

$$s(x_1, x_2, x_3, \ldots, x_w) = \frac{\sum_{i=1}^{w} I(i) \cdot f_{b,i} - \sum_{i=1}^{w} I(i) \cdot f_i^{\min}}{\sum_{i=1}^{w} I(i) \cdot f_i^{\max} - \sum_{i=1}^{w} I(i) \cdot f_i^{\min}} \qquad \text{Equation 2.12}$$

where x_i is the base Σ at the position i, w is the motif length, $f_i^{\min} = \min(f_{b,i})$ and $f_i^{\max} = \max(f_{b,i})$ are the minimal and maximal frequency of the base at position i; and $I(i)$ is the information vector as defined in Equation 2.13,

$$I(i) = \frac{1}{\log_{10} 4} \left(\sum_{b=a}^{t} f_{b,i} \cdot \log_{10}(f_{b,i}) + \log_{10} 4 \right) \qquad \text{Equation 2.13}$$

Moreover, Roulette wheel-based selection of the parent is implemented. GAPWM performs mutation on all PWM except the best individual. GAPWM does not apply crossover among the individuals. It sets 0.05 probability of mutation on the first 100 generations to explore larger sampling space. Then, 0.02 probability of mutation on the following generations to find the maximum in the local space. Interestingly, GAPWM adopts area under ROC curve (AUC) as the fitness function. Therefore, by optimising the motifs iteratively is also increasing the AUC value. Human Oct4, mouse Oct4, and human p53 datasets were selected for GAPWM experiment. Among the three datasets, p53 dataset produced steep curve and quickly achieved AUC 0.9. The existing PWMs were optimised and improved on sensitivity and specificity. GAPWM was compared with MEME and found that GAPWM has higher AUC than MEME for all three datasets.

GADEM (L. Li, 2009) is a *de novo* motif discovery tool for search optimisation based on spaced dyads with the EM on ChIP datasets. Space dyads are the two contact sites in *cis*-regulatory elements spaced by non-conserved fixed length (van Helden, Rios, & Collado-Vides, 2000). GADEM uses combination of word enumeration and probabilistic techniques. However, enumerating all the spaced dyads from large datasets for EM algorithm is computationally impractical. Hence, GADEM only uses the overrepresented words such as tri-, tetra-, penta-, and hexamers to reduce search space in the large datasets. The word

enumeration does not count the frequencies but convert the spaced dyads in the data into probability matrices like MEME. Then the matrices are used as the initial models for a probabilistic algorithm as an EM. GADEM uses GA to "evolve" the spaced dyads from which the starting PWMs are obtained. Logarithm of the E-value is used as the fitness score in GADEM. Six genome-wide datasets were experimented: Oct4, p53, two different ERE datasets, CTCF, and STAT1. The result showed that GADEM is able to identify 15–30 motifs with different lengths.

2.6.4 Summary of tools developed for motif discovery

Table 2.2 shows the summary of various tools mentioned above that uses different techniques for motif discovery.

Table 2.2: Summary table of tools developed for motif discovery.

Tool	Description	Authors
MEME	Probabilistic approach. Solves two limitations of EM in motif discovery.	Bailey and Elkan (1995)
MEME Suite	Web service of MEME with extra tools such as GLAM2 and Tomtom.	Bailey et al. (2009)
MEME-ChIP	Improves MEME targets on ChIP datasets.	Machanick and Bailey (2011)
ChIPMunk	Probabilistic approach. Employs greedy optimisation with EM.	Kulakovskiy et al. (2010)
AlignACE	Probabilistic approach. Uses Gibbs sampling algorithm.	Hughes et al. (2000)
W-AlignACE	Improves AlignACE for ChIP datasets.	X. Chen, Guo, et al. (2008)
MotifSampler	Improves Gibbs sampling with high-order Markov chain background model.	Thijs et al. (2001)
BioProspector	Gibbs sampling with Monte Carlo method.	X. Liu et al. (2001)

Table 2.2 continued

GibbsST	Gibbs sampling with simulated tempering.	Shida (2006)
MITRA	Enumerative approach. Uses mismatch tree data structure to reduce the problem into sub-problems.	Eskin and Pevzner (2002)
Amadeus	Genome-wide motif discovery using mismatch algorithm.	Linhart et al. (2008)
TEIRESIAS	Enumerative approach. Uses seed-driven exhaustive algorithm.	Rigoutsos and Floratos (1998)
MDscan	Enumerative deterministic greedy algorithm on ChIP datasets. Combines both advantage of word enumeration and PWMs. Uses MAP scoring function.	X. S. Liu et al. (2002)
Weeder	Enumerative approach. Uses suffix tree.	Pavesi et al. (2001)
Weeder2	Improved Weeder to handle ChIP datasets.	Zambelli and Pavesi (2011)
RISOTTO	Enumerative approach. Uses suffix tree and implements the maximal extensibility information to RISO.	Pisanti et al. (2006)
Trawler	Enumerative approach. Uses suffix tree with standard normal approximation to the binomial distribution and z-score.	Ettwiller et al. (2007)
Trawler_standalone	Improves Trawler for ChIP datasets.	Haudry et al. (2010)
RSAT	Enumerative approach. Discovers for both gapped and ungapped motifs.	Defrance et al. (2008); Thomas-Chollier et al. (2008)
AMD	Enumerative approach. Also able to discover both gapped and ungapped motifs on ChIP datasets.	Shi et al. (2011)
Hegma	Enumerative approach. Solves large-scale motif discovery using DNA Gray code and equiprobable oligomers to solve clustering problem and oligomer bias.	Ichinose et al. (2012)
SOMBRERO	Uses SOM neural network. PWM is used as the node model.	Mahony et al. (2006)
SOMEA	Uses SOM neural network. Uses hybrid node model composed of PFM and static Markov chain.	N. K. Lee and Wang (2011)

Table 2.2 continued

CSI-ANN	Supervised learning. Uses TDNN for classification and PSO on the training.	Firpi et al. (2010)
kmer-SVM	SVM approach. Uses word model instead of PWM model.	D. Lee et al. (2011)
gkm-SVM	SVM approach. Allows gapped k-mers prediction.	Ghandi et al. (2014)
LS-GKM	SVM approach. Improves gkm-SVM with large scale datasets.	D. Lee (2016)
EnhancerFinder	SVM approach. Enhancer prediction using multiple kernel learning.	Erwin et al. (2014)
DEEP	SVM approach. Uses simple ANN for the final prediction.	Kleftogiannis et al. (2015)
EnhancerPred	Enhancer prediction uses bi-profile Bayes and pseudo-nucleotide composition.	Jia and He (2016)
DeepBind	CNN approach. Predicts sequence specificities.	Alipanahi et al. (2015)
DeepSEA	CNN approach. Predicts noncoding variants from the sequences.	J. Zhou and Troyanskaya (2015)
Basset	CNN approach. Predicts functional activities of DNA sequences.	Kelley et al. (2015)
GAME	GA approach. Optimises random generated PWMs to discover the motifs. It is slow but performs better than MEME and BioProspector.	Z. Wei and Jensen (2006)
GAPWM	GA approach. Optimises the PWM of ChIP dataset. Employs Match scoring function.	L. Li et al. (2007)
GADEM	GA approach. Uses EM for initial model then optimised with GA.	L. Li (2009)

Among the tools as shown in Table 2.2, MEME-ChIP, AlignACE, W-AlignACE, MotifSampler, BioProspector, MDscan, and Weeder2 were to be the individual tools in the proposed ensemble approach. The decision to choose these tools was based on the availability of the tools as well as their reliability. Most of the selected tools were also used in existing ensemble-based motif discovery tools. MEME-ChIP is chosen instead of MEME because using MEME to scan large datasets will be very slow (Linhart et al., 2008). Besides that, the

chosen motif discovery tools consists of probabilistic approach (MEME-ChIP, AlignACE, W-AlignACE, MotifSampler, and BioProspector) and enumerative approach (MDscan and Weeder2). Among the chosen tools, MEME-ChIP, W-AlignACE, and Weeder2 are designed for ChIP-seq datasets, while others are not. However, though they are designed for ChIP-seq datasets, the size of the dataset input file is limited. Because of the limitations of the individual motif discovery tools, ensemble approach is a promising solution that can overcome the limitations.

2.7 Ensemble approach

Ensemble learning consists of different classifiers to produce a new classifier which performs better than those classifiers (Opitz & Maclin, 1999; Rokach, 2010). In addition, combining the results from the multiple classifiers via averaging will reduce the uncertainty from the poor performance classifiers (Polikar, 2006). Additionally, in some cases, the data is too large for a classifier for analysis, such as genome-wide data sequences in motif discovery. An ensemble approach combines multiple algorithms to improve the prediction accuracy, which is analogous to making decision by asking advice from multiple experts (Lihu & Holban, 2015). Each algorithm used by ensemble approach is a classifier. Each classifier is possible produce different prediction results. Therefore, combining the prediction results from multiple classifiers is able to reduce the risk poor prediction from a classifier (Polikar, 2006). The outputs of the classifiers are combined using a combination rule. Ensemble approach is also suitable to solve the problem with insufficient amount of data by resampling techniques to draw overlapping random subsets for training different classifiers (Polikar, 2006).

Different motif discovery tools have their own strengths and weaknesses. For example,

Yeast Motif Finder (YMF) (Sinha & Tompa, 2003) is outperformed in yeast dataset motif discovery, while MEME and BioProspector presume that each input sequence has a motif and draw to premature local optima. DREME (Bailey, 2011) limits the search to maximum 8 bp. Researchers combine these motif discovery tools together as an ensemble approach (Hu et al., 2005, 2006; Kuttippurathu et al., 2011; Romer et al., 2007; van Heeringen & Veenstra, 2011).

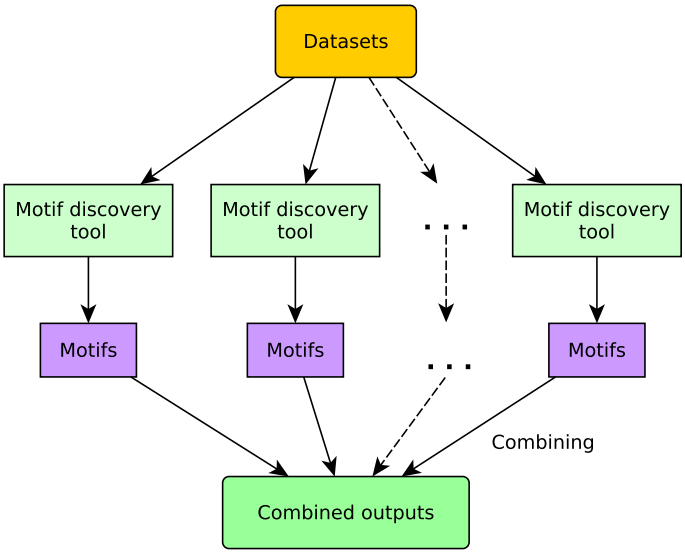


Figure 2.5: Stages involved in ensemble learning on motif discovery. The whole datasets are used for motif discovery by the individual classifiers.

Figure 2.5 shows general stages involved in ensemble learning based on Lihu and Holban (2015), Opitz and Maclin (1999), Polikar (2006), and Rokach (2010) in the motif discovery context. The whole dataset is searched by each tool to discover candidate motifs. All candidate motifs are finally combined to produce the final outputs. Tools that use the framework similar to Figure 2.5 include EMD (Hu et al., 2006), GimmeMotifs (van Heeringen & Veenstra, 2011), MotifVoter (Wijaya et al., 2008), and WebMOTIFS (Romer et al., 2007). There is a drawback of this general framework, when the large-scale datasets

being used, the individual classifiers will be affected by the datasets because the individual classifiers may not able to process large amount of data. EMD is one of the ensemble-based motif discovery tools that uses subsets sampled from the input for each individual tool. The proposed ensemble approach in this study will be different from this framework as described in Chapter 3 Section 3.3.

According to Hu et al. (2005), the benchmark experiments using AlignACE, MEME, BioProspector, MDscan, and MotifSampler were low with 25–35% accuracy for the sequences of 400 nt long, but had at least one TFBS was correctly predicted more than 90% of the time. Consensus Ensemble Algorithm (CEA) (Hu et al., 2005) was proposed by combining the results of multiple predictions from multiple runs of the base algorithms: AlignACE, MEME, BioProspector, MDscan, and MotifSampler. It uses random seeds to run any base motif discovery algorithm several times. The candidate with the highest scores are collected and aligned on the sequence. Next, the number of times the position is estimated will be counted and normalized to calculate the consensus score. Then, positions with consensus scores less than the threshold value are abandoned so that the highest score are remained. Finally, the binding site width of each candidate is adjusted and combined. *Escherichia coli* datasets from RegulonDB were used for the experiment. CEA was compared with AlignACE, BioProspector and MotifSampler. The results showed that AlignACE, BioProspector, and MotifSampler produced very stable prediction accuracy with multiple runs. CEA achieved 6–45% improvement over its base algorithms. Nevertheless, the nucleotide level and binding site level prediction accuracies of CEA remain very low, especially when sequence length increases.

EMD (Hu et al., 2006) is a novel clustering-based ensemble algorithm based on CEA. Five

base algorithms are used as in CEA: AlignACE, MEME, BioProspector, MDscan, and MotifSampler. The differences of CEA and EMD is that the former only combines the multiple runs of the same algorithm instead and only one dataset with short sequence size was experimented. The latter combines the results from prediction systematically across the algorithms. The combination of the results employs a consensus function. Then the consensus function groups the results by their score for the individual algorithm, then groups the results across the different algorithms. The voting is cast to the sequence positions of the predicted sites to preserve the diversity. EMD is tested with two datasets, *Escherichia coli* dataset and the input with additional margins of 800 bp added to both sides of the known sites. Additional margins are added to test scalability of EMD. Projection (Buhler & Tompa, 2002) is used to improve the scalability of EMD. The EMD algorithm involves five stages: (1) It collects the results from individual algorithm; (2) sorts the predicted binding sites by algorithm's statistical score then the sites are grouped based on the equal number; (3) votes are cast to the sequences with the predicted sites; (4) votes are smoothed by using a sliding window with a specific width as parameter; (5) the local peaks of the smoothed voting curve are selected as final binding sites. The findings showed that EMD always performed better than stand-alone motif discovery tools, or at least had the same level of performance as the stand-alone tools. EMD achieved an improvement of 22.4% nucleotide level performance coefficient over the best single algorithm BioProspector.

WebMOTIFS (Romer et al., 2007) is designed for identification of motifs using multiple motif discovery algorithms. It incorporates MEME, AlignACE, MDscan, and Weeder as ensemble approach. It provides web interface for automate motif discovery and analysing DNA sequence. The default parameters for the individual algorithms are used. To avoid reporting discovered motifs that are similar, a clustering algorithm from Harbison et al.

(2004) is implemented. WebMOTIFS also offers an option for the users to use a Bayesian motif discovery tool, THEME (MacIsaac et al., 2006), which is powerful in motif discovery on mammalian species. WebMOTIFS focuses more on the web interface to provide the information to the users and improve the ease of use of the individual algorithms through the web interface.

The CompleteMOTIFs (cMOTIFs) (Kuttippurathu et al., 2011) is a motif analysis pipeline that accepts DNA sequences or genomic coordinates of the region of interest in genomes. It employed three individual motif discovery tools for motif discovery: CUDA-MEME, Weeder, and ChIPMunk. CUDA-MEME is MEME algorithm extended with Compute Unified Device Architecture (CUDA) programming model on graphical processing unit (GPU) which speeds up motif discovery. cMOTIFS also accepts motif profiles in JASPAR or TRANSFAC format for scanning potential binding site locations in input sequences. cMOTIFs was demonstrated on 13 TFs of embryonic stem cell (X. Chen, Xu, et al., 2008) to identify the TFs. The results showed that it was successful identifying the motifs from 12 of 13 TFs from the dataset as the top ranked.

W-ChIPMotifs (Jin et al., 2009) is a web application for motif discovery which is based on ChIPMotifs (Jin et al., 2007). ChIPMotifs uses MEME and Weeder for motif discovery. ChIPMotifs inputs a set of 154 OTC4-binding sequences into MEME and Weeder and get 10 potential motifs with the length of 8–12 bp. Furthermore, 154 sequences are randomised 100 times to generate 15,400 negative control sets. The negative control sets were scanned by Weeder and MEME to get the core scores and PWM scores. A modified bootstrap resampling method was used to infer the optimised PWM scores. Instead of merging, ChIPMotifs uses Fisher test and P-value to define cut-off threshold for the core scores and PWM scores.

Finally, the candidate motifs were screened against TRANSFAC database to get the known and novel motifs. In W-ChIPMotifs, it refines P-value with a Bonferroni correction by multiplying by the number of input sequences. W-ChIPMotifs also adds MaMF (Hon & Jain, 2006) as another individual algorithm for motif discovery. MaMF is a deterministic search algorithm relies on indexing to speed up the search. Besides that, STAMP (Mahony & Benos, 2007) is used to perform phylogenetic footprinting evaluation to determine the discovered motifs. STAMP is able to construct phylogenetic tree and iteratively refine the alignment of the similar motifs until each leaf node contains a single PWM. Furthermore, STAMP is able to match the similar motifs against TRANSFAC and JASPAR databases. This allows W-ChIPMotifs to discover the known motifs. Unfortunately, there is no benchmarking on the performance of W-ChIPMotifs compared with other tools.

MotifVoter (Wijaya et al., 2008) proposed a novel similarity metric to merge motifs discovered by multiple motif discovery tools. Wijaya et al. (2008) claimed that the ensemble-based motif discovery such as WebMOTIFS and EMD though improve the performance of motif discovery, the improvement is insignificant according to Tompa's benchmark. This is because the average sensitivity is only increased by 62% but the average precision is decreased by 15%. MotifVoter incorporates multiple tools: AlignACE, ANN-Spec (Workman & Stormo, 1999), BioProspector, Improbizer (Ao, Gaudet, Kent, Muttumu, & Mango, 2004), MDscan, MEME, MotifSampler, MITRA, SPACE (Wijaya, Rajaraman, Yiu, & Sung, 2007), and Weeder. It consists of two steps: motif filtering and sites extraction. From the results of several individual motif discovery tools, motif filtering will remove the false positive motifs to produce the cluster of high conformity motifs based on motif similarity measure. The similarity measure $sim(x, y)$ is expressed in Equation 2.14,

$$sim(x, y) = \frac{|I(x) \cap I(y)|}{|I(x) \cup I(y)|} \qquad \text{Equation 2.14}$$

where $I(x)$ is the set of binding sites defined by motif x. Thus, two identical motifs will produce $sim(x, x) = 1$ and others produce $0 \leq sim(x, y) < 1$. Next, sites extraction will extract the motifs that have high confidence to be real from the step one. From the extracted sites, MotifVoter uses multiple sequence comparison by log-expectation (MUSCLE) (Edgar, 2004) to generate a PWM to model the motif. MUSCLE is a tool that uses progressive alignment creates multiple alignments of DNA or protein sequences. It is claimed that the alignment speed is faster than alignment tools like CLUSTALW and FFTNS1. MotifVoter is different from EMD on discriminative and consensus criteria. MotifVoter's discriminative criterion requires the selected cluster of motifs shares as much as possible and motifs outside the cluster share none or few binding sites, so that not only similar motifs within the cluster but the motifs outside the cluster also selected to yield higher precision. The consensus criterion requires the cluster to be contributed by each individual tool as much as possible to increase the confidence that the cluster contains good binding sites. MotifVoter was tested on Tompa's benchmark datasets, *E. coli* dataset, and ChIP-Chip datasets (CREB, E2F, HNF4/HNF6, MYOD/MOG, NFKB, NOTCH, and SOX) and compared to three ensemble algorithm: SCOPE (Carlson, Chakravarty, DeZiel, & Gross, 2007), BEST (Che, Jensen, Cai, & Liu, 2005), and EMD. For Tompa's benchmark datasets, MotifVoter outperformed BEST in sensitivity by 54.1% and SCOPE in precision by 226.2%. For the *E. coli*, MotifVoter outperformed EMD in sensitivity by 130.2% and SCOPE in precision by 45.9%. For ChIP-Chip datasets, MotifVoter was able to identify 56 out of 65 motifs.

GimmeMotifs (van Heeringen & Veenstra, 2011) is a motif prediction pipeline using

an ensemble of existing motif discovery tools. It was proposed to predict TFBSs from ChIP-seq data. It employs BioProspector, GADEM, Improbizer (Ao et al., 2004), MDmodule (X. S. Liu et al., 2002), MEME (L. Li, 2009), MoAn (Valen, Sandelin, Winther, & Krogh, 2009), MotifSampler, Trawler, and Weeder. GimmeMotifs uses weighted information content (WIC) similarity metric to cluster the output from these individual tools. The maximum WIC score of motifs pair are calculated. Similar motifs are clustered using an iterative process by comparing all the scores. Two most similar motifs are merged by averaging. Since the process is repeated iteratively, the motif that occurs frequently will have higher influence on the new averaged motif. The process is repeated until the best scoring alignment has the WIC p-value > 0.05. Besides that, the individual algorithms are run in parallel using inter process communication. It was tested on the (i) mouse datasets from X. Chen, Xu, et al. (2008), (ii) another mouse datasets from Valouev et al. (2008), and (iii) human datasets from Jothi, Cuddapah, Barski, Cui, and Zhao (2008). GimmeMotifs outperformed SCOPE and W-ChIPMotifs with AUCs 0.830, 0.613, and 0.824 respectively.

Table 2.3 summarises the different ensemble methods for DNA motif prediction.

Table 2.3: Comparisons of various ensemble tools.

	CEA	**EMD**	**cMOTIFs**	**WebMOTIFS**	**W-ChIPMotifs**	**MotifVoter**	**GimmeMotifs**
Motif discovery tools	AlignACE, BioProspector, MDscan, MEME, MotifSampler	AlignACE, BioProspector, MDscan, MEME, MotifSampler	CUDA-MEME, Weeder, ChIPMunk	AlignACE, MDscan, MEME, Weeder	MEME, MaMF, Weeder	MITRA, Weeder, SPACE, AlignACE, ANN-Spec, BioProspector, Improbizer, MDscan, MEME, MotifSampler	BioProspector, GADEM, Improbizer, MDmodule, MEME, MoAn, MotifSampler, Trawler, Weeder
Other tools	Nil	Nil	Patser scanning tool and STAMP matching	Analysed by THEME	STAMP matching	Nil	Nil
Methodology	Randomly select the base algorithms and run multiple times to get the combined results.	Based on CEA, enhanced with grouping, voting, smoothing.	Combine the top 10 ranked motifs using background models to determine the TFBS.	Cluster the results from individual motif discovery tools.	Bootstrap resampling statistic method and a Fisher test are used to optimise the results from motif discovery tools.	It involves motif filtering and sites extraction using discriminative and consensus criteria.	It uses weighted IC similarity metric for clustering and runs the algorithms in parallel.
Datasets	*E. coli*. Average sequence length 300 nt	*E. coli*. Average sequence length 300 nt	ChIP-seq embryonic stem cell (13 TFs)	ChIP data from *S. cerevisiae*, mouse, and human	E2F4, Oct4, FOXA1, NRSF	Tompa's benchmark dataset, *E. coli*, and other ChIP-Chip datasets	Mouse and human datasets

Table 2.3 continued

Results	It is able to improve 6–45% in both sensitivity and specificity comparing to the individual algorithms.	It is able to improve 22.4% in nucleotide level prediction accuracy comparing to the individual algorithms.	It is able to identify most TFBS of ChIP-Seq dataset as the top ranked. It also improved the ranking using shuffling.	It is able to reveal the expected protein in the human ChIP-chip sequence.	Nil	It is able to discover more than 95% of the binding sites comparing to individual discovery tools.	Better than SCOPE and W-ChIPMotifs and achieved AUC 0.830.
URL	Nil	Nil	http:// cmotifs.tchlab.org	http://fraenkel.mit .edu/webmotifs	http://motif.bmi .ohio-state.edu/ ChIPMotifs	http://compbio .uthscsa.edu/ ChIPMotifs/ index.shtml	https://github.com/ simonvh/ gimmemotifs

2.7.1 Summary on ensemble approaches

According to Lihu and Holban (2015), W-ChIPMotifs and CompleteMOTIFs are restricted by the user input less than 20 megabytes. This limits the potentiality of the large-scale analysis. Furthermore, W-ChIPMotifs is limited to the mouse and human genome, yet the other ensemble-based tools such as GimmeMotifs, cMOTIFs, and RSAT peak-motifs can be used on any genome.

Due to the limitation of the classic tools, it is necessary to restrict the input size by ensemble-based motif discovery tools such as W-ChIPMotifs. This is to avoid the individual tools fail to search the motifs from the large search space. Though large-scale input is useful and theoretically is able to produce the global optimum, it is infeasible as the classic motif discovery tools were not designed for NGS datasets. However, if the datasets are able to be partitioned into smaller subsets, then each individual tool will be able to discover the motifs that are overrepresented in the subsets.

In addition, ensemble approach has an advantage on the performance evaluation according to Hu et al. (2006), as the performance is at least at the same level of the individual tool that is being used. For example, if there are three individual motif discovery tools are used, namely Algorithm A, Algorithm B, and Algorithm C. The discovered motifs are evaluated with the performance score 0.75, 0.8, and 0.95 respectively. If the final score is calculated by averaging, we will not get the score less than 0.75. However, this also indicates that the performance score may not be able to exceed 0.95, unless the discovered motifs are able to be optimised to produce higher evaluation score than from individual motif discovery tools.

Ensemble approach requires sequences as the inputs for the individual tools. With the restriction of the input size, sampling is necessary to reduce the input size and to avoid biased results. Assuming that a dataset, $dataset = \{a, b, c, d, e, f, g\}$, where the $\{a, b, c, d, g\}$ are the true motifs or true positives, and three existing algorithms can find the candidates as $\{a, b, c\}$, $\{a, b, c, e\}$, and $\{a, b, c, d, e, f\}$ respectively. Because of each algorithm has its own strength, each can discover different set of candidates. However, using the ensemble approach, it is possible to find redundant candidates or similar candidates. The similar candidates will increase the accuracy, since they are approximately redundant.

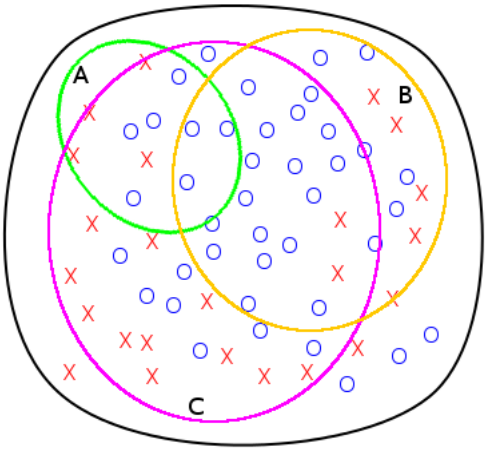


Figure 2.6: Results of ensemble approach.

In the Figure 2.6, it shows the result of ensemble approach. The red crosses represent the negative patterns or candidates, and the blue tiny circles represent the positive patterns. By using ensemble approach, that is using multiple algorithms, they will produce different results. The figure above uses three large circles to represent three different algorithms, namely A, B, and C. Though the algorithms are able to improve the prediction results, by increasing the ratio of positive patterns to the negative patterns, they will cover large overlapped area. Therefore, merging the discovered results and remove the redundancies is

necessary.

The genomic large-scale ChIP-seq datasets are interested to be studied (Zambelli et al., 2013). Nevertheless, most existing approaches employ tools that are developed before the NGS era. Furthermore, large-scale datasets requires longer time to be searched because the search space is exponentially increased. Partitioning is a promising approach on the large-scale datasets so that the problem can be divided into smaller search space. The next challenge is merging or combining the candidate motifs to produce the final outputs. For example, W-ChIPMotifs uses bootstrap re-sampling and Fisher test to get the optimal cut-off of the discovered motifs. MotifVoter uses MUSCLE on the extracted sites to produce the PWM model. GimmeMotifs, WebMOTIFS, and cMOTIFs use clustering on candidate motifs discovered by individual tool. However, cMOTIFs only reports the clustering result. Only GimmeMotifs implements merging on the clustered result to produce new motifs. According to Romer et al. (2007), it is difficult to merge the results from different tools. This is because each individual tool reports the output differently.

2.8 Performance metrics for motif discovery tools

Over-representation of a motif is defined as statistical significance of the motif is assessed from a background model (Zambelli et al., 2013). In enumerative approach, over-representation is measured by comparing actual number of occurrences of a word to the expected number of occurrences (Thijs et al., 2001). While in probabilistic approach, over-representation is measured by comparing the profile model to random expectation from the background model or random model (Hughes et al., 2000).

Though there are many versions of motifs from databases such as JASPAR and TRANSFAC,

the lack of standardised assessment technique causes difficulty for the researchers to benchmark and test the discovered motifs (Kibet & Machanick, 2016). Moreover, there is no reference benchmark set for ChIP-seq datasets (Zambelli et al., 2013). This becomes a challenge for motif discovery on the large datasets with unknown motifs. Tompa et al. (2005) performed the prediction by inserting real motifs into the sequences, but failed to capture the biological condition of TF binding. Similarly, Hu et al. (2005) uses RegulonDB but motif discovery results had low accuracy (15-25%) because of the poor quality annotations in RegulonDB. Unless the TF binding sites are well annotated, motif assessment by predicting binding sites that are inserted or known, cannot be confidently utilised. Nevertheless, novel motifs can be assessed by comparison to "reference motifs" using Euclidean distance or other statistics that measure divergence between two profiles (Kibet & Machanick, 2016). For instance, Thomas-Chollier et al. (2012) proposed motif comparison that uses Pearson's correlation and sum of squared distances. However, the definition of "reference motifs" remains largely subjective (Kibet & Machanick, 2016).

Another motif assessment approach is using scoring function to identify the motifs from the negative background sequences, without known binding sites. compare observed motif scores with the expected scores from the background model. The background model is normally a higher order Markov model. The expected scores are statistics p-value (Takusagawa & Gifford, 2004) or z-score (Sinha & Tompa, 2000). Alternatively, information content (IC) or relative entropy of discovered PWMs (Bailey & Elkan, 1995) can be used for significance measurement. A log-likelihood scoring scheme is also considered as information content (Hertz & Stormo, 1999). In enumerative approach, the evaluation involves calculating two separate values when evaluating motifs, one is the support of a motif and the other one is unexpectedness of a motif (Rigoutsos & Floratos, 1998).

Moreover, hypergeometric (HG) over-representation score (Barash, Bejerano, & Friedman, 2001) is used in Amadeus as the default scoring function (Linhart et al., 2008). Nonetheless, ChIP-seq datasets with different sequence length and different choice of negative background sequences will produce greatly different scoring results. Consequently, these differences lead to variations in results comparison (Kibet & Machanick, 2016).

A good scoring function is important to distinguish real binding sites from background noise (N. K. Lee & Choong, 2013). There are many scoring functions proposed. However, they are applied in different software tools and using different methods and representations. Maximum a posteriori probability (MAP) is a scoring function which is used by Gibbs Sampler, BioProspector, AMD, MDscan, AlignACE (and W-AlignACE), and SOMEA. This scoring function combines the negative entropy of the PWM and the rareness of the PWM based on third order Markov model calculated from all intergenic regions of a genome. MAP is defined in Equation 2.15 as follows:

$$MAP = \frac{\log(x_m)}{w}\left(\sum_{i=1}^{w}\sum_{b=A}^{T} f_{b,i}\log f_{b,i} - \frac{1}{x_m}\sum_{\text{all}s}\log(p_0(s))\right) \qquad \text{Equation 2.15}$$

where w is the width of the motif, x_m is the number of candidate words (m-matches) in the PWM, and $p_0(s)$ is the probability of generating the candidate words from the background model (Friberg, von Rohr, & Gonnet, 2005). The value is important because it takes into account the occurrences of the candidate words from the dataset. Without this value, the score will depend only on the background Markov model, which is not appropriate to determine the motif.

2.8.1 Receiver operating characteristics (ROC)

Receiver operating characteristics (ROC) analysis (Fawcett, 2006) is commonly used in machine learning as a diagnostic method to determine the discrimination of a classifier towards different classes by varying the threshold settings. The discrimination is determined by computing the sensitivity and specificity rates for bins of threshold values in [0, 1]. The fundamental usage of ROC analysis is its application in binary classification problems. The two classes are positive and negative classes. A classifier is usually trained and tested with the samples to classify to which class they belong. A discrete classifier will produce four possible outcomes: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Conversely, a probabilistic classifier will produce scores that represent the degree to which class a sample belongs to. A decision threshold value can be applied to the score so that the value above the threshold is labelled as positive, and otherwise negative. Using the threshold, the probabilistic classifier can be changed into a discrete classifier.

A ROC curve is plotted with true positive rate (TPR) along the vertical axis and false positive rate (FPR) along the horizontal axis. The plot indicates sensitivity and specificity of the prediction. Therefore, the predicted outcomes must be compared with the actual outcomes with the calculation of TPR and FPR.

The followings are the formulae for sensitivity and specificity calculation.

$$\text{sensitivity} = \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\text{P}} \qquad \text{Equation 2.16}$$

$$\text{specificity} = \text{TNR} = \frac{\text{FN}}{\text{TN} + \text{FP}} = \frac{\text{TN}}{\text{N}} \qquad \text{Equation 2.17}$$

Furthermore, the FPR can be also be calculated as,

$$\text{FPR} = 1 - \text{TNR} = \frac{\text{FP}}{\text{TN} + \text{FP}} = \frac{\text{FP}}{\text{N}} \qquad \text{Equation 2.18}$$

Accuracy is calculated as,

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \qquad \text{Equation 2.19}$$

An accuracy of 100% indicates that the prediction sensitivity is 100% and specificity is 100%. Thus, TPR is equal to 1 and FPR is equal to 0. As a result, when plotting the graph, the coordinate (0, 1) is marked in the plot. A discrete classifier will only produce a single point in the ROC space (Fawcett, 2006); thus a discrete classifier cannot produce a curve. However, a ranking or scoring classifier can be used with a threshold to produce continuous points. By connecting the points, the ROC curve can be plotted. ROC provides a visual and numeric summary of a predictions and becomes a primary measurement method in bioinformatics applications (Sonego, Kocsor, & Pongor, 2008). According to Lihu and Holban (2015), ROC can illustrate the problem of *de novo* motif discovery performance that being affected by false positive motifs.

The Area Under Curve (AUC) of the ROC indicates the degree of the separability of a

classification model between classes. AUC has value between 0–1 with larger value indicates better discriminative ability of the model (L. Li et al., 2007; Su, Teichmann, & Down, 2010). The AUC of a classifier is equal to the probability of a classifier to rank a randomly chosen positive instance higher than a randomly chosen negative instance (Fawcett, 2006). AUC is commonly used to measure the quality of extracted motifs (Orenstein & Shamir, 2014; Weirauch et al., 2013). It is used to discriminate foreground that contains the motif from the background (Yao et al., 2014). In motif discovery, the usage of AUC is to evaluate the PFM or PWM models obtained, rather than evaluating the motif discovery tools. It is an indirect inference of the quality of the motif discovery tools based on the quality of the motif models discovered. In contrast to machine learning classifiers, a classifier is able to classify an input as positive or negative, while motif discovery tools are discovering the matrix models that are able to label a given input as positive or negative. Therefore, it is necessary to evaluate a motif discovery tools through the discovered motifs.

In order to plot ROC curve, the foreground dataset can be a dataset with known motifs. For example, Yao et al. (2014) used ENCODE ChIP-seq datasets. The discovered motifs with the best PWM scores of each sequence were measured against the annotated motifs from JASPAR. The background dataset is consists of the randomly chosen sequences from the edge of the peak. The PWM score was used as the threshold for plotting the ROC curve.

Similarly, Deep Motif Dashboard (DeMo Dashboard) (Lanchantin et al., 2016) uses 108 ChIP-seq TF datasets from ENCODE. The negative datasets are generated by shuffling the nucleotides of the positive datasets but preserving dinucleotide frequencies. It compares the discovered motifs to JASPAR motif of the TFs of interest (57 out of 108 TF datasets). Tomtom is used to match the similar motifs that are ranked by p-value. The AUC is computed

by TFBS classification of the discovered motifs.

The above examples (Lanchantin et al., 2016; Yao et al., 2014) construct the ROC by using PWM z-score or p-value, based on the annotated motifs through database such as JASPAR and a background dataset. On the other hand, Energy Matrix Quality Improvement Tool (EMQIT) (Smolinska & Pacholczyk, 2017) uses ROC curves and 10-fold cross-validation to improve motif prediction. Smolinska and Pacholczyk (2017) selected datasets from TRANSFAC as positive datasets, and randomly selected promoters from human genes as the negative datasets. EMQIT counts the number of properly detected motifs in the positive dataset as TP, and the number of motifs detected in the negative dataset as FP for every PWM. By using 10-fold cross-validation method, nine of ten subsets were used as training set, and the remaining was used as test set. A ROC curve is constructed for each PWM. The PWM with the largest AUC is selected and used to scan the test set using Match.

GimmeMotifs also evaluates the discovered motifs using AUC of the ROC. GimmeMotifs computes the ROC of the discovered motifs by comparing the validation set and a background set. The validation set contains the sequences that are not used for motif prediction. There are two types of background sets being used: (i) randomly generated sequences with similar dinucleotide frequencies from first order Hidden Markov model of the input data, and (ii) randomly selected genomic sequences by considering the position of the peaks relative to the transcription start site (TSS). Z-score is used for the best match of the motif to the sequences of the input set. Consequently, the ROC can be plotted and the AUC can be computed.

GAPWM adopts ROC as part of the motif discovery algorithm. GAPWM calculates two sets of Match scores (Equation 2.12) of the discovered PWM by using two datasets. The two

datasets are a positive dataset which contains the motif and negative dataset as a background. Let S_{pos} denotes a set of scores from positive dataset, and S_{neg} as a set of scores from negative dataset. Thus, $S = (s_1, s_2, s_3, \ldots, s_n)$, where s_i is a Match score for sequence i. By using positive dataset, TPR can be calculated with $\frac{\text{TP}}{\text{P}}$, where P is the total number of sequences in the positive dataset, and TP is the number of sequences with the score greater than a cut-off value, $s_i >=$ cut-off. Similarly, FPR is calculated by using negative dataset. The cut-off values are 5000 values generated by using linear decrement from $\max(S_{neg})$ to $\min(S_{neg})$. Therefore, as the cut-off values decreased gradually, the TPR and FPR can be computed and the ROC curve is plotted. GAPWM uses AUC of the ROC as the fitness score for motif discovery. Besides that, choosing a negative dataset is challenging because the background should be similar to real background of the actual motifs (L. Li et al., 2007). Hence, L. Li et al. (2007) randomly selected 1500 sequences from coding regions of the human and mouse as the negative datasets. The advantage of GAPWM is the computation of ROC can be performed without known motifs or annotated motifs.

Figure 2.7 shows a comparison of ROC generation for classifier and unsupervised motif discovery tools. The key difference is a classifier's ROC is generated through the test dataset; while for a motif discovery tool, its performance is evaluated through the motif models, represented as PWM or PFM, its predicted. Furthermore, the ROC for the motif models obtained require the input (positive) and the negative dataset. In DNA motif discovery, since the locations of the binding sites are unknown, it is infeasible to utilise conventional performance metrics that utilise class labels. Hence the sensitivity and specificity rates can only be estimated through positive and negative datasets (D. Wang & Lee, 2009).

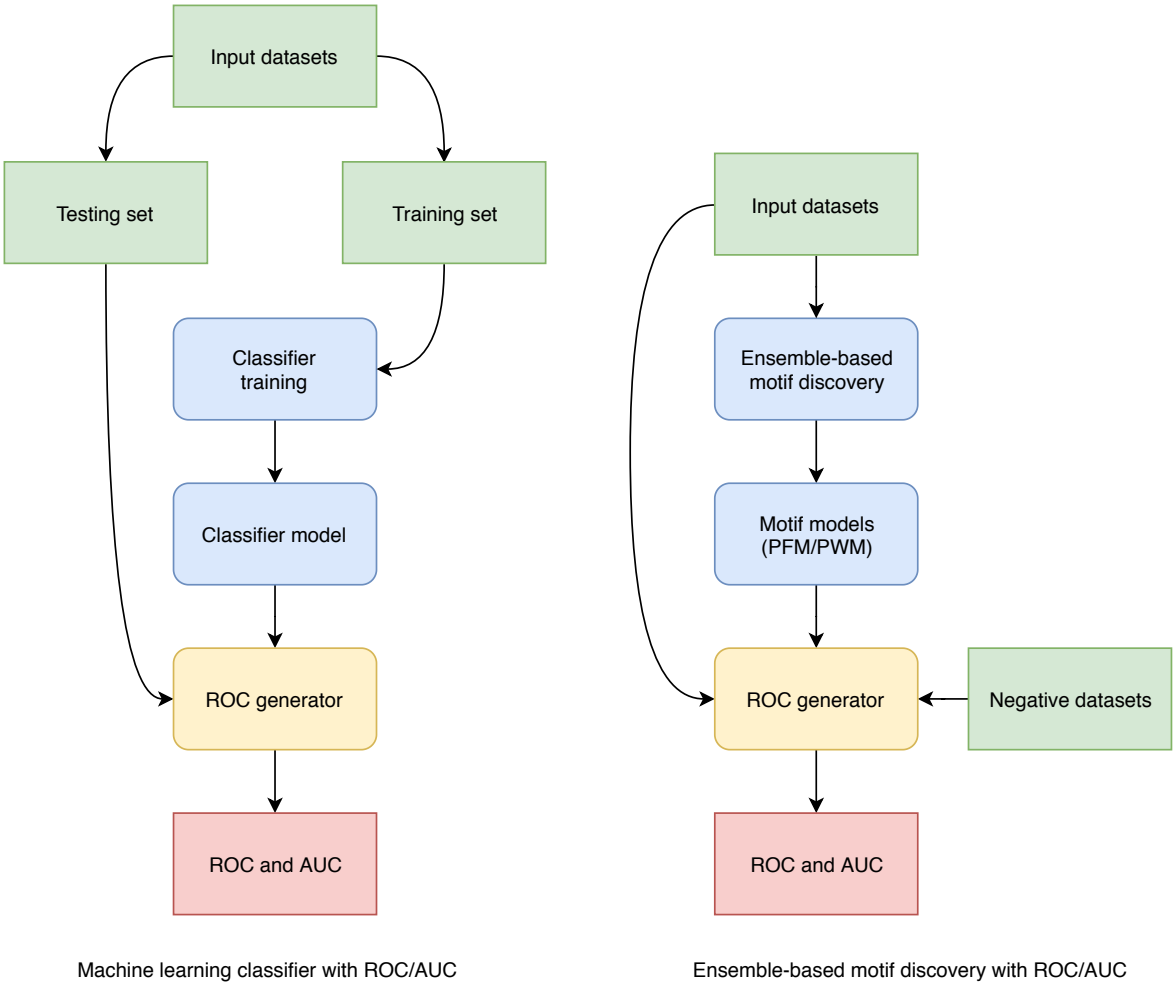


Figure 2.7: Comparison of ROC generation in machine learning classifier and ROC generation in ensemble-based motif discovery.

2.9 Conclusion

Motif discovery can commonly be divided into probabilistic and enumerative approaches. Probabilistic approach uses stochastic methods such as EM and Gibbs sampling to discover motifs. Profile model such as PWM is the most commonly used model in probabilistic approach (Sandve & Drabløs, 2006). For instance, EM involves expectation step to estimate the scores of the PWM, and maximisation step to refine the PWM to maximise the scores over several iterations. MEME is the most popular motif discovery tool that employs

EM. Similarly, Gibbs sampling randomly initialises motif positions in the input sequences. Then, one sequence is randomly selected and PWM for all the other sequences by using the positions that were initialised is computed. Next, the probability of every position of the selected sequence is calculated by using the PWM. The best position will be used for the next iteration. The iterations will be stopped when the PWM cannot be improved. AlignACE, MotifSampler, and BioProspector are the example motif discovery tools based on Gibbs sampling. Though probabilistic motif model such as PWM is more expressive than consensus, probabilistic approach does not guarantee global optimal motifs (Das & Dai, 2007).

Enumerative approach uses consensus model to represent motifs Theoretically, enumerative is able to discover global optimal motifs, yet it is infeasible in large dataset motif discovery, because it is time-consuming (Hashim et al., 2019). Enumerative approach is a word-based approach, that exhaustively enumerates all possible motifs. Suffix tree is a notable algorithm in enumerative approach. It can enumerate all patterns by using $O(N)$ time and $O(N)$ space where N is the length of a sequence. The goal of the algorithm is to discover the longest commonly repeat pattern in the suffix tree, which is the motif. Weeder, RISOTTO, and Trawler are example motif discovery tools that employ suffix tree. Enumerating all possible motifs by using suffix tree is time-consuming, thus Weeder speeds up the algorithm by narrowing down the valid patterns. However, this leads to low efficiency for the long motifs. On the other hand, MDscan combines the advantage of enumerative approach for search strategy and matrix model for scoring.

GA is also being used in motif discovery. The advantage of GA is the design of solution domain can focus on how potential motifs being encoded into GA chromosomes, because

the GA is able to explore the search space according to the defined fitness function. GA is able to optimise PWM with higher sensitivity, but it is not able to fully solve the local minimum problem because lack of understanding about binding sites specificity (N. K. Lee et al., 2018). GAME, GAPWM, and GADEM are the motif discovery tools that employs GA.

SVM is commonly applied in enhancer prediction instead of *de novo* motif discovery. Enhancers are *cis*-regulatory modules (CRMs) consists of a set of TFBSs (motifs) which works together. SVM is a supervised learning. Hence, enhancer prediction requires training by using positive and negative datasets for the training (Ghandi et al., 2014; D. Lee et al., 2011; D. Lee, 2016). Besides that, deep learning such as CNN is also a supervised learning. CNN has demonstrated good performance in bioinformatics. Deep learning is a representation learning that allows the machine to discover the representations in the raw input without the needs to explicitly provide engineered input features (LeCun, Bengio, & Hinton, 2015). Unlike SVM which requires internal representation or feature vector of the input data for the training purpose, deep learning overcomes the requirement of internal representation by learning the features automatically. Though CNN is able to discover motifs with $4 \times m$ matrix similar to PWM (Alipanahi et al., 2015), it may be problematic to adopt CNN as motif learning methods (Zhu, Zhang, & Huang, 2017). This is because CNN is not able to differentiate two sets of different matrices, as long as the matrices are able to lead to same decision function (Zhu et al., 2017).

The review found that the ensemble approaches outperformed single tool in DNA motif discovery. They have the advantage of discovering more motifs with different characteristics. The owing to that each individual tool employed in ensemble has distinct design strengths

that can enlarge the success of discovering true motifs in a dataset. The development of ensemble algorithm for motif discovery has two key components: (i) the merging of the results from multiple individual tools; (ii) the selection of candidate motifs in the final results. First issue is on how to merge motifs produced by different tools. These motifs might have different representations and lengths. The technical challenge is these motifs might only be partially matched a sub-motifs of the true motifs. In addition, there might be hundreds of those motifs, matching them in which order is a challenge and what is the similarity or dissimilarity function should be used. The second challenge is during the merging process, how many levels of merging is needed. In addition, after the merging, which scoring function can be used to select the true motifs. The scoring function clearly involve a discriminative function that can distinguish random motifs from the true ones.

Furthermore, existing ensemble approaches are not able to cope the large-scale datasets, because each individual classifier has the limitation on the large-scale datasets. The traditional motif discovery tools were developed before the NGS era. Therefore, they are not able to effectively handle large-scale genomic datasets by default. A solution is necessary to overcome the limitation of the input size, so that the traditional motif discovery tools are able to be applied in ensemble approach for large-scale datasets.

In this work, the data partitioning and novel motif merging algorithm are proposed to solve the two main issues in existing ensemble approaches. The data partitioning method can reduce the input dataset size for each individual tools in ensemble approach. Moreover, the merging algorithm can merge hundreds of motifs return by individual tools through multiple merging.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter presents a novel ensemble framework known as, ENSPART, which employs several existing *de novo* motif discovery tools to discover certain motifs. Because of NGS techniques, the genome-wide large-scale datasets are available. Furthermore, it is important to predict the transcription factors such as enhancers in order to study various diseases using genome-wide datasets. However, the time complexity increases dramatically with the large dataset size. ENSPART employs the data partitioning technique so that the search space size can be efficiently explored by any motif discovery tools within a reasonable computational time. For example, a full size dataset that is scanned by a motif discovery tool requires eight to ten hours to complete, while using partitioned dataset, the scanning process is able to complete in one to two hours.

This study is not focusing on the computational time, but rather using traditional motif discovery tools in ensemble approach, to solve motif discovery problem from the large-scale datasets. Hence, the computational time is not part of the study. This study illustrates the potentiality of ensemble approach employing the traditional classifiers to solve the DNA motif discovery problem. As the key to the design of ENSPART is how to combine the results produced by multiple tools, the novel multiple stages motif merging algorithm utilising the alignment free method for effective motif comparison is being introduced.

Firstly, this chapter briefly describes the method of ENSPART. Secondly, the datasets and partitioning of the datasets are explained. Next, the motif discovery tools used in ENSPART

are discussed. Finally, the comparison of the motifs, grouping and merging of the similar motifs are also explained.

3.2 Motivation

Most existing ensemble methods (see Section 2.7) employ multiple classical motif discovery tools for DNA motif discovery. Classical tools are those proposed mainly in the 1990s before the ChIP and ChIP motif analysis era (simply ChIP era). Motif analysis before the ChIP era mainly targeted a small number of genes from the prokaryotic species or eukaryotic species. The algorithms thus were not designed to scale with dataset sizes in terms of computational speed and performance. Hence, the classical tools have limited use for motif analysis of ChIP dataset which usually have hundreds of thousands of DNA sequences. For examples, MEME only allows maximum of 100,000 bp input sequences; Weeder 5,000 sequences; and AlignACE approximately 50,000 bp input sequences. The massive amounts of data causes the traditional tools to produce too many positives and long execution times (Lihu & Holban, 2015). Kulakovskiy et al. (2010) tested MEME with ChIP-seq datasets and proved to be inefficient due to the large datasets.

Due to the limitation on input sizes of classical tools, existing ensemble approaches such as ChIPMotifs and CompleteMOTIFs limit the input data sizes. Traditional *de novo* motif discovery tools like MEME, BioProspector, MDscan, and AlignACE were developed before the ChIP era, it is impractical for using these tools to discover the motifs from the ChIP datasets. Therefore, an improved method is needed to make use of these tools for discovering motifs in the ChIP datasets.

While there are tools that are meant for analysing ChIP datasets, most employ heuristic

methods and enumeration techniques are used in the search algorithms. That is, subset of the input dataset is used to generate seeds k-mers that guide the enumeration search. Those methods rely heavily on the quality of the seed generated and selected. In addition, the seeds obtained are determined by the subset used and ranking functions for the selection.

In order reduce the large search space due to large dataset sizes of ChIP datasets, ensemble learning is a promising approach. For example, W-ChIPMotifs, GimmeMotifs, and CompleteMOTIFs are the ensemble-based motif discovery tools developed for the ChIP-seq. However, these tools do not demonstrate the solution on the search space problem. W-ChIPMotifs restricts the input size and this limits the analysis of the large scale datasets. Similarly, CompleteMOTIFs and GimmeMotifs do not pre-process the inputs for the individual tools. This indicates that, if a very large input dataset is given to CompleteMOTIFs and GimmeMotifs, the search time is still increasing because it depends on the employed individual tools.

Because ensemble learning adopts the divide-and-conquer approach and it is workable with the insufficient amount of data (Polikar, 2006), partitioning the input data to the smaller subsets is possible to reduce the search space. Consequently, by using the partitioned datasets, the traditional tools are assumed to be more efficient to discover the motifs comparing to using the whole amount of data.

The discovered candidate motifs represent the local optima because the datasets are partitioned. Hence, the discovered candidate motifs from the subsets need to be combined to produce the final output. However, each motif discovery tool produces different outputs, combining the results require a common model for the discovered motifs. PWM is more

expressive than consensus because PWM contains the probability of the occurrences of each nucleotide. The final discovered motifs should produce higher prediction performance than candidate motifs from individual tools. This is because the features of the candidate motifs are combined to produce the final outputs.

3.3 Method

This study proposes a novel ensemble method ENSPART using a data partitioning technique coupled with classical motif discovery tools. The data partitioning technique partitions the input data set into different non-overlapping partitions, which are then tackled separately by different classical motif discovery tools. Since ChIP-seq dataset is high confidence sequences that are enriched with binding sites of TF of interest, it is reasonable to assume each DNA sequence has one or multiple occurrences of binding sites. Therefore, sampling the dataset into several subsets, the primary motif will remain overrepresented in the subsets. This indicates that, sampling the dataset will not strongly affect occurrences of the motifs. Hence, after partitioning the dataset to smaller subsets, we can assume that the primary motifs remain high frequency of occurrences in distinct subsets. The same motifs that appear across the subsets should be merged according to the similarity of the pattern. Therefore, the key to our method is how to merge these similar motifs obtained from different input subsets. ENSPART employs an alignment-free method to compute the similarity between candidate motifs obtained from different subsets for merging. Alignment-free sequence comparison is defined as any sequence comparison method that do not use or produce alignment at any step of algorithm (Zielezinski, Vinga, Almeida, & Karlowski, 2017). It has the advantage of fast computation and does not have difficulty to merge motifs with different length. Pair-wise alignment method like basic local alignment search tool (BLAST) (Altschul et al.,

1997) requires window shifting between two motifs to obtain the best alignment positions. Hence, comparing to alignment-based method, alignment-free method is computationally less expensive. An alignment-free method is desired as it is computational efficient and remove the difficulty of finding optimal alignment between two motifs of different lengths.

Figure 3.1 shows the framework of ENSPART, which consists of several consecutive steps. Firstly, the datasets are partitioned into non-redundant subsets randomly. Each subset is scanned for motifs by using several classical motif discovery tools. Each tool run three (3) times with different set of parameters. The description is covered in 3.3.1. Running the tools more than three times and gathering more candidate motifs not necessary producing better results. This is because the same tools will produce similar or same candidate motifs. This will only increase more redundant candidate motifs. Table A1 in Appendix shows the parameters being used for each tool. The discovered candidate motifs are pair-wise compared to produce a similarity matrix. The alignment-free method is used for similarity function between any pair of motifs. Then, elements in a matrix are sorted according to their similarity values. The elements are labelled for grouping. The grouped elements are merged to produce new motif. All the new motifs are collected and repeat the comparison, sorting, labelling, and merging steps for three times to reduce the redundant or similar motifs. Finally, the merged motifs are evaluated by using AUCs. The explanation of the parameter to merge the similar motifs is covered in Section 3.3.3 and Section 3.3.4.

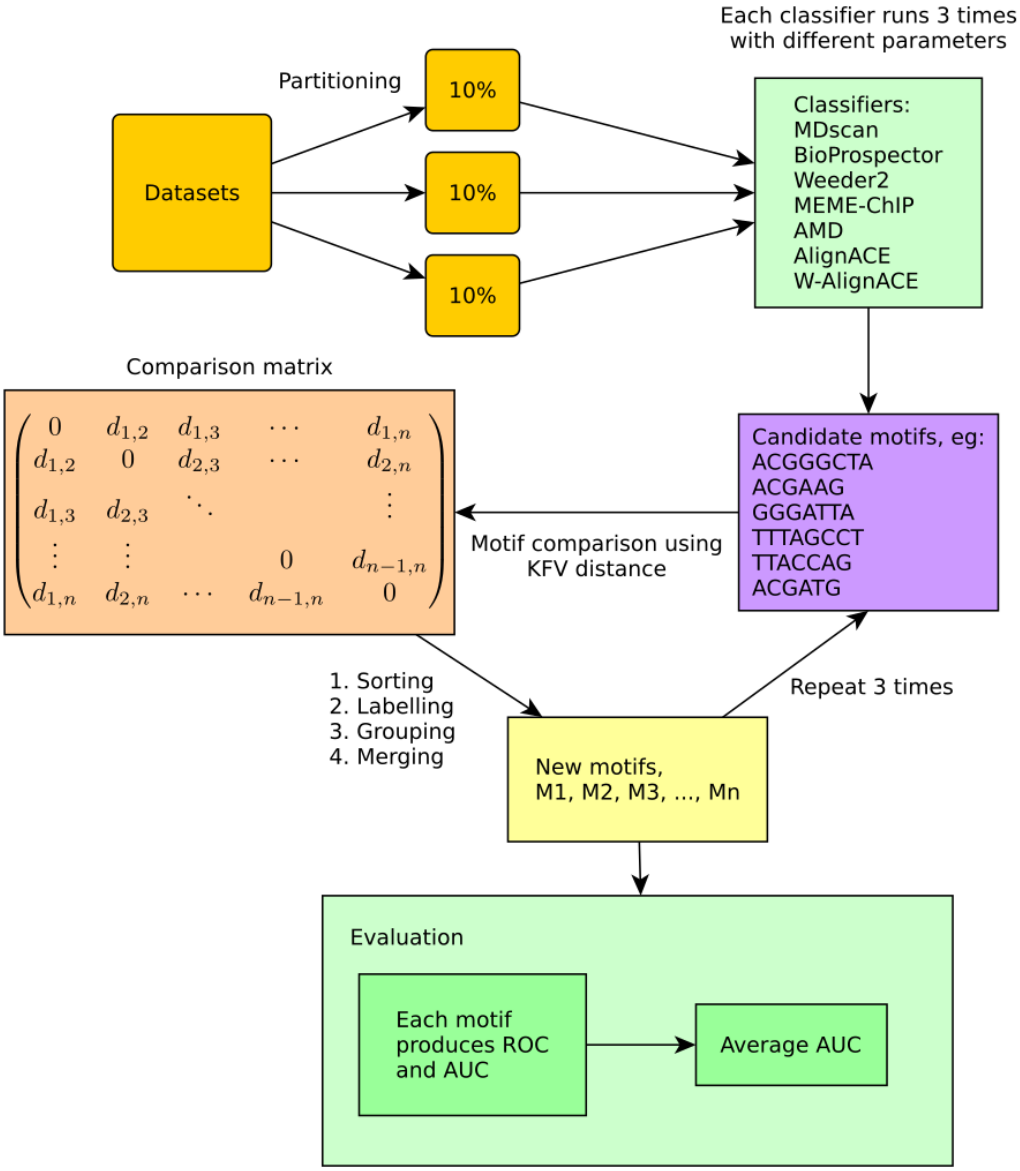


Figure 3.1: ENSPART framework. Datasets are partitioned into three (3) non-overlapping subsets. Seven motif discovery tools are used as individual classifiers. Each tool runs three (3) times for motif scanning with different set of parameters as shown in Appendix 1. The discovered candidates are compared using KFV values. The threshold of the k-mer frequency vector (KFV) similarity is using the empirical value 0.27, which was discovered through observation on a preliminary experiment. Similar motifs are merged by averaging to reduce the redundancy. The steps of merging process is repeated three (3) times to produce the final output. Finally, the final candidate motifs are being evaluated using ROC and AUC.

3.3.1 Partitioning

Existing ensemble methods are employing whole input set for motif discovery tools and merge the results obtained. In contrast, ENSPART employs data partitioning which partitions the collected datasets in FastA format into non-overlapping subsets before each is searched for candidate motifs by set of selected motif discovery tools. The rationale of this method is to reduce the size of the datasets so that each partition can be searched for the motifs by the individual classical motif discovery tools. Partitioning the datasets in data mining is a common method as it reduces the training set sizes for the available memory of the machine (Chawla, Eschrich, & Hall, 2001). However, the fundamental difference between partitioning datasets for data mining and partitioning datasets for motif discovery is that the former is for training, the latter is for motif discovery. This is because the motif discovery tools uses the dataset as the input to discover the motifs as the output. The number of output depends on the individual motif discovery tools, which is covered in Chapter 4 (Section 4.2.1).

The ChIP dataset consists of high-confidence sequences that are potentially enriched with primary motifs instances. Thus, after partitioning, the primary motifs are expected to be abundance in the subsets. In the implementation, 10% of the input sequences were selected randomly without replacement. Three (3) partitions were generated with 10% of sequences each. While more partitions are possible, we limit it to three partitions. The 10% is an arbitrary value and it can be adjusted based on the total sequences available. According to Zia and Moses (2012), adding more sequences does not reduce false positive rate significantly. Moreover, Hu et al. (2005) also stated that, increasing the number of sequences does not certainly improve the prediction accuracy, and using fewer number of sequences allows

to reduce the running time significantly when scanning with the time-consuming motif discovery tool. Therefore, 30% of the whole dataset is chosen as a heuristic decision for this study, and the motif discovery works well with the 30% of the dataset. Theoretically, larger size of partitions can yield better result because the partition size is closer to population size of the dataset. However, further empirical study is necessary to test the significance of the result by using parameter with larger partition size.

The partitioning is performed by shuffling the DNA sequences to guarantee that the order of the sequences are randomised. Each partition (i.e. 10% of the total) is sampled randomly without replacement.

3.3.2 Motif discovery

Each partition is searched for candidate motifs with different algorithms. Each tool is run with three (3) different sets of parameters (Appendix 1) on each partition to increase the possibility to discover wider range of motifs. Different sets of parameters will produce different results. The parameters such as length of the discovered motifs, iterations of the tool (MDscan), minimum gap (BioProspector), and seed value (MEME-ChIP) can be adjusted. The decision of running the tools three times and how to define the parameters are based on heuristic because we have no prior information regarding the candidate motifs in the dataset. Running the tool multiple times increases the chances of obtaining more true motifs but at the same time would increase the number of redundant motifs. All discovered motifs obtained by each tool are saved for the next step.

In the ENSPART, seven (7) motif discovery tools are chosen: MDscan, BioProspector,

MEME (MEME-ChIP), Weeder, AlignACE, AMD, and MotifSampler. These are the most popular tools used in previous ensemble tools as well as individual tools. They are consisted of a good mix of tools using probabilistic technique (BioProspector, MEME-ChIP, AlignACE, MotifSampler) and enumeration technique (MDscan, AMD, Weeder2). The tools were downloaded form the source shown in Table 3.1.

Table 3.1: List of motif discovery tools.

Tool	URL
MDscan	http://motif.stanford.edu/distributions/mdscan/
BioProspector	http://motif.stanford.edu/distributions/bioprospector/
MEME Suite	http://meme-suite.org/
Weeder 2	http://159.149.160.51/modtools/downloads/weeder2.html
AlignACE	http://arep.med.harvard.edu/mrnadata/mrnasoft.html
W-AlignACE	http://www.ntu.edu.sg/home/ChenXin/Gibbs (not available)
AMD	http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0024576
MotifSampler	http://bioinformatics.intec.ugent.be/MotifSuite/motifsampler.php

The motifs returned by each tool is validated through their respective scoring function. Table 3.2 shows what scoring functions that are used to select the candidate motifs. While the scoring functions do not ruled out completely false positives, running multiple times of every tool on the same dataset and merging would ensure only the true motifs are prevailed due to the over-representation property. That is, primary motifs will appear more frequently amongst the produced results.

While ENSPART framework is relying on the results of other tools, the key is how to merge and filter the discovered candidate motifs. After the merging, the merged motifs are scored using a scoring function for ranking purpose. This will ensure the true motifs are selected in

the final results.

Table 3.2: List of scoring functions.

Tool	Score function
MDscan, AMD	MAP score (Equation 2.15)
BioProspector	$n \times \exp\{[\sum_i \sum_j q_{i,j} \times \log(q_{i,j}/p_j)]/w\}$, where n is the number of aligned segments in the motif, $q_{i,j}$ is the probability of nucleotide j at the position i, p_j is the probability of the nucleotide j at the background, w is the length of the motif
Weeder	$\lvert P\rvert \cdots N_P - 2\sum_{i=1}^{k} d(P, P_i)I(P, i)$, where P is the pattern, P_i is the best instance of P in the sequence i, d is the distance, $I(P, i)$ is 1 if P appears in sequence, else 0, N_p is the total number of sequences P
AlignACE	Group specificity score
MEME, MotifSampler	Log-likelihood score

MEME was chosen as it is one of the popular tools been used in many existing ensemble approach (Hu et al., 2005, 2006; Jin et al., 2009; Kuttippurathu et al., 2011; Romer et al., 2007; Wijaya et al., 2007). MEME Suite (Bailey et al., 2009) is a software toolkit that offers motif discovery, motif-motif database, motif-sequence database searching, and assignment of function. MEME-ChIP is one of the tools available in MEME Suite (Machanick & Bailey, 2011). In this study, it was found that MEME is too slow for a personal computer to run on the partitioned. As an alternative, MEME-ChIP is used instead because it is faster. Weeder and AMD were chosen because both tools are fast. AMD is also able to discover gapped motifs. AlignACE and MotifSampler were selected because they implement Gibbs sampling. Therefore, the selected tools cover probabilistic approach (EM and Gibbs sampling) and

enumerative approach (suffix tree and greedy algorithm).

Table A1 in Appendix shows the invocation of each individual tool with different parameters that were being applied in ENSPART. A script was written to parse the outputs of these motif discovery tools with the same format which contains some information especially PWM and the consensus discovered. The purpose of the parsers was to convert the output format from the individual motif discovery tools become compatible to each other. Once all outputs are parsed by the script, the parsed putative motif information were saved in one large file.

3.3.3 Candidate motifs comparison and sorting

K-mer frequency vector (KFV) (M. Xu & Su, 2010) is a alignment-free method to compare TFBS motifs. KFV is used in ENSPART to make comparisons among the discovered motifs. KFVs can compare the similarity of motifs with different lengths using either their PWM or PFM representation. The use of profile-based representation ensures maximum information of the motifs are utilised for the comparison purpose. The KFV generates a vector for each motif which in turn can be used to compute the similarity distance between motifs. KFVs can be used as a distance measure using suitable similarity metric function that satisfies the positivity, symmetry, and triangle inequality.

Some similarity functions are Euclidean distance, Pearson correlation coefficient (PCC), Mahalanobis distance, Kullback-Leibler (KL) discrepancy, and angle metrics. The Euclidean distance would produce a value in the range of $[0, \infty]$, while the others are in the range of $[0, 1]$.

The following describes how the KFV of a motif is constructed. To generate the KFV of a motif, the length of k-mers is required. The k-mers are short sequences of segments of length k bp (Choong & Lee, 2017). The larger the k-mers, the slower that the conversion from PWM or PFM to KFV. In order to construct KFV, firstly, given a positive integer $k > 0$, a set of words from the permutation of a set of Σ is generated. For instance, if $k = 2$, the set is $S = \{AA, AC, AG, AT, CA, CC, CG, \ldots, TT\}$. Each of the element of the set S will be converted to a matrix. For instance:

$$M_{AA} = \begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

where the first row of the matrix is for the word A, followed by C, G, an T. Then, likelihood that describes the k-mer is calculated as Equation 3.20 (M. Xu & Su, 2010):

$$P_K = \sum_{i=1}^{l-k+1} \prod_{j=1}^{k} col_j(B)^T \cdot row_{i+j-1}(\overline{M}) \qquad \text{Equation 3.20}$$

where M is the PFM of the motif $M = \{f_{ij}\}^{4\times l}$, f_{ij} is the count of nucleotide $i \in \Sigma$ at position $j = 1..l$; k is the length of the k-mers; l is the motif length; $\overline{M}$ is the normalized PFM, $\overline{M} = \{f_{ij}^{4\times l} \backslash N\}$, N is the total number of aligned sites in the motif; B is the binary matrix $B = \{b_{ij}\}^{4\times k}$. $b_{ij} = 1$ is nucleotide i is at the position j of the k-mer, otherwise $i = 0$.

For example, if a candidate motif has a PFM

$$M_{AAACGT} = \begin{pmatrix} 7 & 5 & 8 & 0 & 1 & 0 \\ 1 & 2 & 0 & 7 & 0 & 0 \\ 0 & 1 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 1 & 1 & 8 \end{pmatrix},$$

then for each column of the matrix of the set S, for instance M_{AA}, the column is transposed. It is then multiplied by the column from M_{AAACGT}, and a matrix with the size 1×1 is calculated. The value of the matrix is divided by total number of aligned sites. The calculation is repeated and finally the sum of product of these columns is calculated. The result is a likelihood of k-mer AA described by M_{AAACGT}.

The KFV is constructed by using the likelihood as Equation 3.21 (M. Xu & Su, 2010):

$$V_M = (P_{K_1}, P_{K_2}, \ldots, P_{K_{4^k}}) \qquad \text{Equation 3.21}$$

where K_1 is the AA for $k = 2$, K_2 is AC, K_3 is AG, …. Hence, if $k = 2$, a vector with 16 elements will be produced.

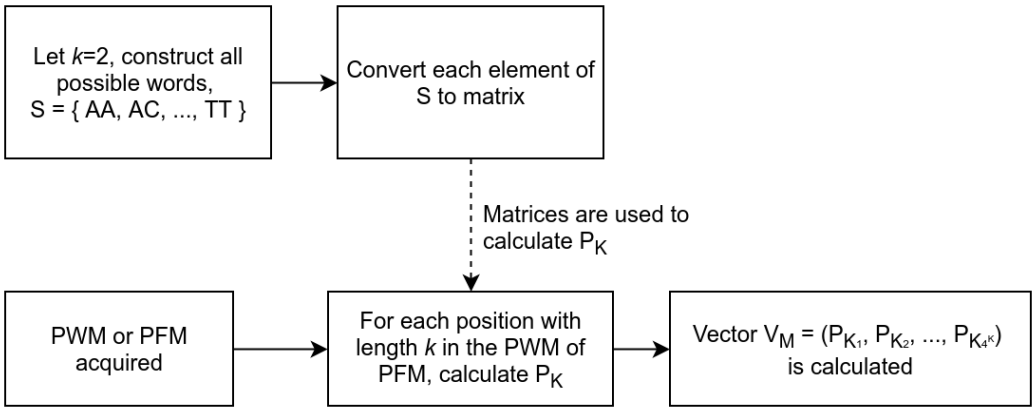


Figure 3.2: Steps to calculate KFV from given PWM or PFM.

Figure 3.2 summarises the steps involved to construct a KFV from PWM or PFM.

Both PWM and PFM can be converted to KFV with the same result. The distance score used by this study is Pearson correlation coefficient distance (PCC) as shown in Equation 3.22:

$$d_{PCC} = 1 - PCC(a,b) = 1 - \frac{\sum(a_i - \overline{a})(b_i - \overline{b})}{\sqrt{\sum(a_i - \overline{a})^2 \sum(b_i - \overline{b})^2}} \qquad \text{Equation 3.22}$$

where a and b are the KFVs. When distance score is 0, that means the two motifs are identical. The advantage of KFV is the calculation of the similarity ignoring the different lengths of the PWMs. When the distance of the two KFVs is within a threshold, the PFMs are assumed to match. The matched pair of TFs are assumed true positive, while the mismatched pair are assumed true negative. Using this, similar PFMs can be grouped together.

Because the orientation of the motifs is unknown, the distance score of a pair of motifs has to consider both the forward strand and reverse complement. That is important because it will effectively reduce the redundant motifs that produced by different tools in a different strand. The two distance scores are computed as $d(a,b)$ and $d(a,b')$ where d is the distance function,

a and b are the KFVs and the b' is the KFV based on the reverse complement of b. The final score for a pair of motifs is given by Equation 3.23.

$$d(a, b) = \min(distance(a, b), distance(a, b*)) \qquad \text{Equation 3.23}$$

ENSPART is different from the previous studies as it involves partitioning of the datasets. In order to merge the results from individual motif discovery tool, there are two possible approaches. The first approach assumes that each result is an individual result ignoring the tool used; then these results are merged as the result of ensemble approach. The second approach is to merge the results based on the algorithm, then only merge these results as the results of ensemble approach.

However, either approach has a problem, namely how to merge the results together. We need to define a threshold value for the similarity of a pair of motifs so that both motifs can be merged. For example, the motifs found by an algorithm "AACCGGTT" and "AACCGGGT" can be considered as one potential motif, as they are similar with only one nucleotide (the 7th nucleotide) is different. On the other hand, if there is another motif as "ATATCGCT", this motif should not be merged with the previous ones, as they are very different. This should be considered as another potential motif. Nonetheless, it is difficult to define whether two motifs are similar, for instance "AAAACCCC" and "AGAGCTCT", because 50% of the nucleotides are matched and 50% are mis-matched. There is no definitive way to determine a pair of patterns is same motif or distinctive motif. Besides that, by adopting the first approach of merging, it indicates that every discovered candidate motif has same weight for the merging. Consequently, the number of discovered motifs from an individual tool will influence the

final result. For example, Tool A discovered 10 motifs and Tool B discovered 1 motif. After the merging of the motifs, it can be assumed that the final result mostly comes from Tool A. On the other hand, the second approach that performs the merging of the motifs according to each tool will produce the final result evenly according to the tool. Nevertheless, the second approach requires two levels of merging: (i) merging the motifs according to each tool, and (ii) merging the motifs from step (i) to produce the final motifs.

In this study, the first approach mentioned above is used in which all the results from obtained by different tools from different partitions are merged at once. Before merging, all the produced motifs are labelled with a unique positive integer. These motifs are compared using their KFV to produce a square symmetric matrix. Then, the elements of matrix are sorted by the sum of KFV distance score of the elements in the row. Each row is calculated with the sum of the KFV distance score, then the rows are sorted according to the sum in the ascending order. Consequently, the least difference candidate motifs are sorted on the top row of the matrix. The matrix is transposed and sorted again. As a result, the matrix maintains symmetric. The motifs that are more similar are sorted at the left-top of the matrix; the motifs which are more distinct from each other are at the right-bottom. The purpose of the sorting is to make sure that similarity of the motif pairs are arranged in the ascending order, so that left-top elements in the matrix have higher precedence when labelling and grouping

are performed. An unsorted comparison matrix can be expressed as Equation 3.24.

$$\begin{pmatrix} 0 & d(1,2) & d(1,3) & \cdots & d(1,n) \\ d(1,2) & 0 & d(2,3) & \cdots & d(2,n) \\ d(1,3) & d(2,3) & \ddots & & \vdots \\ \vdots & \vdots & & 0 & d(n-1,n) \\ d(1,n) & d(2,n) & \cdots & d(n-1,n) & 0 \end{pmatrix} \qquad \text{Equation 3.24}$$

where $d(i,j)$ is KFV distance of motif i and motif j. $d(i,i)$ will produce 0 because they are identical motifs.

Algorithm 1 Algorithm to create KFV distance comparison matrix.

```
Represent the KFV results in a distance matrix
For each row of the matrix, calculate the sum of KFV distance score
Re-order (sort) the rows according to the sum of KFV score
Transpose the matrix
Re-order (sort) the rows according to the sum of KFV score
```

Algorithm 1 shows the steps of creating KFV distance comparison matrix.

Table 3.3: Example of KFV distance score comparison matrix after sorting using E2F4 dataset. The first row and the first column are the motif labels. 0 indicates that the two compared motifs are identical.

0	1	2	3	4	5	6
1	0.0000	0.0000	0.1176	0.1176	0.1176	0.7140
2	0.0000	0.0000	0.1176	0.1176	0.1176	0.7140
3	0.1176	0.1176	0.0000	0.0000	0.0000	0.7493
4	0.1176	0.1176	0.0000	0.0000	0.0000	0.7493
5	0.1176	0.1176	0.0000	0.0000	0.0000	0.7493
6	0.7140	0.7140	0.7493	0.7493	0.7493	0.0000

Table 3.3 shows an example of comparison matrix with six (6) motifs.

Table 3.4: Comparison of KFV distance of k=2, k=3, and k=4 after sorting. The distance values are taken from the first column of the matrix.

k=2		k=3		k=4	
Motif label	Compared to M. 645	Motif label	Compared to M. 181	Motif label	Compared to M. 181
645	0	181	0	181	0
675	0.0225	416	0.0486	618	0.0856
445	0.0170	645	0.0478	416	0.0826
684	0.0250	658	0.0281	471	0.1050
236	0.0291	236	0.0899	167	0.1046
644	0.0168	445	0.0388	420	0.0944
591	0.0325	618	0.0562	651	0.0889
471	0.0243	219	0.0673	658	0.0730
219	0.0223	471	0.0690	644	0.0757
651	0.0120	212	0.0601	645	0.0945
618	0.0379	656	0.0379	184	0.0657
241	0.0142	644	0.0433	176	0.0845
181	0.0168	185	0.0671	212	0.1028
624	0.0409	420	0.0609	694	0.0871
218	0.0327	473	0.0764	473	0.1020
641	0.0249	694	0.0475	236	0.1345
185	0.0213	651	0.0482	208	0.0972
662	0.0219	184	0.0457	206	0.0889
171	0.0293	218	0.0509	445	0.0898
164	0.0177	214	0.0816	222	0.0914

The KFV distances of the motifs obtained from E2F4 dataset for $k = 2$ to 3 were computed. According to M. Xu and Su (2010), the accuracy of the similarity gained discriminative performance starting from $k = 2$ and achieved the maximal overall accuracy when $k = 4$. However, the goal of the merging to merge according to the threshold value rather than merging most similar motifs, therefore larger k is not necessary. Table 3.4 shows the comparison of KFV distance of $k = 2$, $k = 3$, and $k = 4$ after comparison matrices are sorted. The columns "Motif label" refer to the candidate motifs that have been discovered by the tools used by ENSPART. The columns "Compared to Motif n" refer to the PCC distances that are calculated when comparing Motif m of the row to the Motif n. Therefore, from the table we can see that the Motif 645 compares to Motif 185, it produces distance 0.0213 when $k = 2$ was used; 0.0478 when $k = 3$ was used; and 0.0945 when $k = 4$ was used. However,

the sorting order showed that the Motif 645 and Motif 181 are on the top. Motif 236 and Motif 236 are at the same position for both $k = 2$ and $k = 3$. Furthermore, motifs such as 471 and 219 appear in proximity.

KFV of E2F4 dataset with 128 sequences and average length 543 bp wad computed on a computer with 2.5GHz Intel Core i7-6500U CPU and 8GB memory. In terms of computation time, computing KFV for $k = 2$ requires approximately 22 seconds, while $k = 3$ and $k = 4$ requires 106 seconds and 497 seconds respectively. The time used for calculating is exponential when using $k = 4$. Thus, it is infeasible for the experiment by using $k = 3$ and $k = 4$. Similarly, M. Xu and Su (2010) uses different computer specifications, with the dataset 124 sequences and average length 10.6 bp, the computation time used were 2 seconds, 7 seconds, and 21 seconds for $k = 3$, $k = 4$, and $k = 5$ respectively. The result showed that computation time for $k = 4$ is 350% slower than $k = 3$, while our result showed that $k = 4$ is 460% slower than $k = 3$.

In this study, $k = 2$ is used for the KFV because as it is shown, it is sufficient to discover the similarity between the motifs.

An observation was performed on the results taken from E2F4 dataset to decide the similarity threshold value. Table 3.5 shows part of the results taken from E2F4 dataset. The motifs in Table 3.5 are shown as consensus but the calculations of KFV are based on their PWMs. Based on the observation on the E2F4 dataset, the first pair, GGTGGACCTC and GGTGGACCTC have exactly the same consensus pattern, their PCC value is not 0 since their PWMs are not identical. For merging of motif pairs, the similarity threshold is heuristically set at 0.27. This value is chosen because based on the observation on the E2F4 dataset results,

the difference of the consensus can be clearly noticed when the PCC distance is greater than 0.27 as shown in Rows 8, 9, and 10. This value works well in all of our experimental evaluations.

Table 3.5: KFVs of the pair of motifs using k=2 from E2F4 dataset.

No.	Motif 1	Motif 2	PCC distance
1	GGTGGACCTC	GGTGGACCTC	0.001588087
2	GGTGTCCGCCGA	GGTGTCCGCTGA	0.018972666
3	CCCAGGAGGAGGCAATGCC	GGAGGCAGAG	0.114541419
4	GGGGGGTGAG	GTTGGT	0.230132783
5	GGTCCAGGCA	CTGAGTCCCA	0.245219499
6	CAGTGCTGAG	GGAGGCAGAG	0.254222876
7	GGAGACAAGG	GCCAGGGGACAGAGGAGCCCTG	0.266092136
8	GAAGGAAAGA	CCTGAAGA	0.271311867
9	GGAGGGTCCA	CGGAGGAGGA	0.280612390
10	GGTGTGGGCCCT	GGGCTACAGGGGATCTCAGCAG	0.298690672

3.3.4 Labelling and grouping

Once the comparison matrix is produced, the candidate motifs are labelled for the grouping. Each motif is assumed to be a member of one of the groups. Labelling is applied before the grouping. This is because a candidate motif can be similar to multiple candidate motifs. Hence, each motif needs to be grouped to the most similar candidate. In order to find the most similar candidate, each motif is labelled and updated incrementally.

Since the comparison of the KFV produces square symmetric matrix, the labelling process only works on the lower triangular of the matrix. The first stage of the labelling traverses the matrix row by row and labels the position with 1 if the PCC value is less than the threshold value 0.27. The second stage traverses the matrix column by column, to mark any position

with the PCC value greater than the smallest value in the column to 0. This is to make sure that each column will have only one "1". An example of zero-one-matrix produced after labelling of four motifs is shown below.

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

In the example above, the fourth motif will be grouped twice with the motif in the second and third column. That is undesired; therefore, in the next step, it requires some pruning to ensure each motif is only assigned to a group. The pruning step begins by traversing by row starting from the bottom-right of the matrix. For each position marked with 1, namely $d(i, j)$, will compare to the KFV distance values of each position, namely $d(j, k)$, which is 1. The position that has less value will be marked with 0, so that there will remain only one position marked as 1. Algorithm 2 shows the steps for labelling by marking the similarity as 1.

Algorithm 2 Algorithm of labelling the motif for grouping (Part 1).

```
initialise all values in the matrix to 0
// Mark similarity as 1
for all rows do
    for all columns do
        if kfv ≤ threshold then
            mark 1
        end if
    end for
end for
```

Algorithm 3 shows the pseudocode to mark the label as 0 by rows.

Algorithm 3 Algorithm of labelling the motif for grouping (Part 2).

```
// Unmark the similarity to get only 1 pair most similar in the row
for all columns do
    get the smallest distance value
    for all rows do
        if the value > the smallest distance value then
            mark 0
        end if
    end for
end for
```

The third part of the algorithm (Algorithm 4) marks the label as 0 by columns.

Algorithm 4 Algorithm of labelling the motif for grouping (Part 3).

```
// Unmark similarity by comparing the pairs in column
declare cursor A_{i,j}
for i start from bottom dotop row
    for j start from right doleft column
        if cursor A = 1 then
            declare cursor B_{j,k}
            for k = j − 1 doleft column
                if B = 1 then
                    compare A and B
                    if B distance value > A distance value then
                        B=0
                    else
                        A=0
                        break
                    end if
                end if
            end for
        end if
    end for
end for
```

Once labelling is completed, a square matrix is produced which contains binary data. The matrix is then converted to a list of groups, where each group contains similar motifs that

have the PCC distance value less than 0.27. The similar motifs in each group will be merged and produce new motifs. Algorithm 5 shows the steps to convert the labelled square matrix into a list of groups.

Algorithm 5 Algorithm to convert labelled square matrix to groups.

```
1:  declare List to store the list of groups of IDs
2:  declare Loners to store standalone motif IDs
3:  for each row of matrix (where matrix is the square labelled matrix) do
4:      if there is no label with 1 then
5:          // Store to loner when the label appears first time in the loop
6:          store to Loners
7:      else
8:          // Store at the group and remove from loner
9:          declare temporary Group
10:         store current row ID to the Group
11:         for each column of the current row do
12:             if the label is 1 then
13:                 if the column ID appears in Loners then
14:                     delete the ID in Loners
15:                 end if
16:                 store the column ID to the Group
17:             end if
18:         end for
19:         store the Group to List
20:     end if
21:     // Make all loners as single element lists
22:     for all Loners do
23:         store each Loner to List
24:     end for
25: end forreturn List
```

3.3.5 Motifs merging

Similar motifs will be merged to produce new motifs. The aim of the merging is to remove redundant results as they do not contribute to the performance. Moreover, each group is assumed to represent a new motif.

To merge the motifs, their PWMs are aligned by window shifting method. Window shifting refers to moving one of the PWMs from left to right, which is also used by N. K. Lee and Wang (2011). Then for each position, sum-of-square error is calculated. Hence, two PWMs are shifted one position at a time and the position with smallest sum-of-square error will be selected.

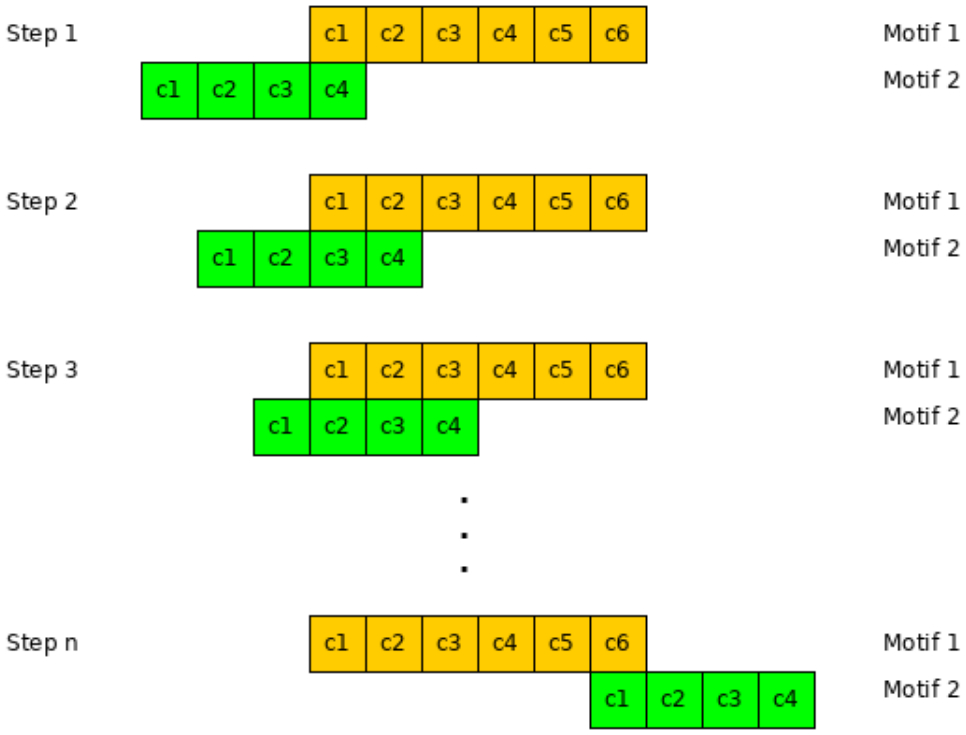


Figure 3.3: Alignment process of two PWMs. c_n is the column n of the PWM. Motif 2 is window shifted from left to right.

Figure 3.3 illustrates the alignment process. For the extended positions during the alignment, the value 0 is used for the PWM. Value 0 is more appropriate by assuming no word exists at the left and right of X. Thus, after spanning the extra columns to the X, a new matrix with a longer length L is produced. The new motif length is calculated as Equation 3.25.

$$L = L_x + (L_y - 1) \times 2 \qquad \text{Equation 3.25}$$

where x and y are the motifs. The left side and the right side length is $L_y - 1$ each.

The sum of squared error (SSE) is used for computing the similarity score of two aligned PWM positions. SSE of each shifted position between two PWMs is computed. Supposing that A and B are two PWMs of two distinct motifs, Equation 3.26 shows the SSE formula:

$$SSE = \sum_{i=1}^{l} \sum_{j=1}^{4} (A_{ij} - B_{ij})^2 \qquad \text{Equation 3.26}$$

where i is the PWM column, j is the position of nucleotide, Σ, of the PWM, and l is the $\min(L_A, L_B)$, L is the motif length. The shifted position with the least SSE is the best aligned position for the merging.

A group possibly contains more than two motifs. Because the KFV distance comparison matrix is sorted, the upper motifs are used as the base for the merging. After the alignment process, the aligned PWMs will be merged to produce a new motif. Figure 3.4 shows an example of the group.

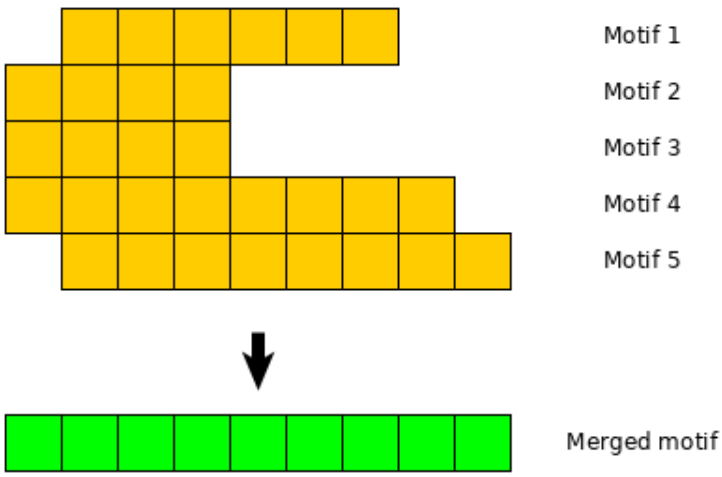


Figure 3.4: A multiple alignment of five PWMs.

The merging of these PWMs involves averaging of the matrices. The merging can be applied to both PWM or PFM.

By using this merging method, similar or redundant PWMs will be combined and the number of candidate motifs is reduced. Because the labelling and grouping assume the candidate motifs existed in one group only, after merging they may still similar motifs that potentially can be merged. Hence, the steps for labelling, grouping, and merging are repeated for three times to reduce the redundant candidate motifs.

3.4 Datasets

Three (3) sets of datasets were collected for motif discovery. In this study, the three sets of datasets are named as Datasets 1, Datasets 2, and Datasets 3.

3.4.1 Datasets 1

Datasets 1 were collected for ENSPART to compare with genome-wide motif discovery tools. Several datasets were collected from http://compbio.uthscsa.edu/ChIPMotifs/examples .shtml. They included mammalian ChIP datasets used by W-ChipMotifs. E2F4 (X. Xu, Bieda, & Jin, 2007), OCT4 Ntera (Jin et al., 2007), forkhead box protein (FOXA1) (Y. Zhang et al., 2008), and NRSF (Valouev et al., 2008). E2F4 is one of the binding pattern members of E2F family. E2F family is involved in regulating a number of critical cellular and organismal functions (X. Xu et al., 2007). Neuron-restrictive silencer factor (NRSF) (Jin et al., 2009) is also known as repressor element-1 silencing transcription factor (REST). MEME was used in motif discovery of the NRSF (Johnson, Mortazavi, Myers, & Wold, 2007).

Besides that, the following datasets were also collected: p53 (C. L. Wei et al., 2006), SOX2 (Boyer et al., 2005), Oct4 (X. Chen, Xu, et al., 2008), CREB (X. Zhang et al., 2005), forkhead box protein A2 (FoxA2) (Tuteja, White, Schug, & Kaestner, 2009), CCCTC-binding factor (CTCF) (Kim et al., 2007), and signal transducer and activator of transcription protein 1 (STAT1) (Jothi et al., 2008). Oct4 and SOX2 work with other transcription factors to active or repress cell (Pesce & Schöler, 2001). Both Oct4 and SOX2 were evaluated in GAPWM, GADEM, and CompleteMOTIFS. The cyclic-AMP response element-binding protein (CREB) family stimulates gene expression, responding to the cyclic-AMP (cAMP). Besides that, a background dataset of human is also collected from Amadeus website (http://acgt.cs.tau.ac.il/allegro/download.html) in FastA format.

Table 3.6: Information of the datasets.

Dataset	Number of sequences	Average length of sequence	File size
CREB	2342	1141	3.3M
CTCF	13804	816	12.7M
E2F4	128	543	76K
FOXA1	2119	357	828K
FOXA2	4051	218	1.3M
NRSF	1657	283	528K
NTERA	154	553	92K
OCT4	7776	461	4.7M
P53	542	1186	669.9K
STAT1	27470	246	8.1M

Table 3.6 shows the number of sequences, average length of sequence, and the file size of datasets.

3.4.2 Datasets 2

Datasets 2 were used for ENSPART to compare with other tools without partitioning. Table 3.7 shows the sources of the datasets for the experiment.

Table 3.7: TF datasets and the source collected for the experiment without partitioning.

Name	Source	URL
CEBPA	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=CEBPA&sample_id=HUMHG01588
	GEO	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM678395
NRSF	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=REST&sample_id=HUMHG05279
	ENCODE	https://www.encodeproject.org/files/ENCFF107EWI/
CTCF	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=CTCF&sample_id=HUMHG03289
	GEO	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM1354438
E2F4	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=E2F4&sample_id=HUMHG05520
	ENCODE	https://www.encodeproject.org/files/ENCFF000XDD/
FOXA1	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=FOXA1&sample_id=HUMHG05243
	ENCODE	https://www.encodeproject.org/files/ENCFF167BKY/
P53	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=TP53&sample_id=HUMHG03164
	GEO	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM1294876
STAT1	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=STAT1&sample_id=HUMHG04175
	ENCODE	https://www.encodeproject.org/files/ENCFF000XLM/
CREB1	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=CREB1&sample_id=HUMHG05801
	ENCODE	https://www.encodeproject.org/files/ENCFF000PGL/
KLF4	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=KLF4&sample_id=HUMHG03393
	GEO	https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM1387028
MYCN	ChIPBase	http://rna.sysu.edu.cn/chipbase/motif_browse.php?organism=human&assembly=hg38&protein=MYC&sample_id=HUMHG05835
	ENCODE	https://www.encodeproject.org/files/ENCFF542GMN/

The datasets collected from ENCODE and GEO are usually BED format. BED format is required to be converted into FastA file as the input for ENSPART. BED is a text file format that contains genomic locations (van Heeringen & Veenstra, 2011). An example of the BED content is as follows:

```
chr1    132263342    132263582
chr1    132309626    132309866
chr2    132315060    132315680
```

Firstly, one can use RSAT web service (http://rsat.sb-roscoff.fr/retrieve-seq-bed_form .cgi) to convert the BED file to FastA directly. Secondly, one can download the chromosome FastA files from UCSC Genome Bioinformatics website (http://hgdownload .soe.ucsc.edu/downloads.html#human), then uses "fastaFromBed" (or getfasta) utility from BEDTools (Quinlan & Hall, 2010) to extract the FastA data based on the BED file.

All the selected datasets were *homo sapiens* TFs. The TFs were randomly selected. Each dataset was sampled to 500 sequences to reduce the search space. This will also reduce the search time when running with different algorithms.

Table 3.8 shows the information of the collected datasets.

Table 3.8: The average sequence length, total bp count, number of sequences, and the percentages of each nucleotide of the collected TF datasets.

TF	Average length	Total bp count	Number of sequences	% of A	% of C	% of G	% of T
CEBPA	474.1	27412895	57818	29.5	20.4	20.5	29.6

Table 3.8 continued

CREB1	1205.3	11987432	9946	26.9	22.9	23	27.1
CTCF	196.9	6064340	30806	28.2	21.6	21.8	28.4
E2F4	868.2	908089	1046	27.3	22.6	22.7	27.3
FOXA1	267.6	2355219	8802	29.5	20.5	20.6	29.4
KLF4	766.2	13698041	17879	28.1	21.8	21.8	28.2
MYCN	372.9	3517871	9434	19.6	30.4	30.5	19.5
NRSF	455.4	8815536	19358	23.2	26.7	26.9	23.2
P53	484.7	562688	1161	29.8	20.1	20.2	29.9
STAT1	880.5	4938045	5608	28.4	21.6	21.6	28.4

3.4.3 Datasets 3

Datasets 3 were collected for ENSPART to compare with two other ensemble based motif discovery tools: GimmeMotifs and MotifVoter. Datasets 3 were simulated datasets created by "rMotifGen" (Rouchka & Hardin, 2007). The advantage of using simulated datasets is that the actual binding sites locations of each sequence are already known. rMotifGen is a tool to generate random DNA and protein sequences. It can generate the sequences with various properties and levels of conservation of the target motif. Users are able to define the motifs according to the PWM. Moreover, rMotifGen can also simulate the mutation of the DNA sequences. The tool is able to help the researchers to test the performance of motif discovery algorithms. In this study, rMotifGen is used to generate the simulated DNA sequences based on true motifs from JASPAR database. Nevertheless, the generated datasets do not contain exact motif from the input. Five (5) known TFs were chosen to generate the simulated sequences. They were CTCF, E2F4, FOXA1, NRSF, and P53. The TFBS matrix were retrieved from JASPAR database. Table 3.9 shows the source of the TFs collected for the experiment.

Table 3.9: TF and the source of known motif matrices from JASPAR database.

Name	URL
CTCF	http://jaspar.genereg.net/matrix/MA0139.1/
E2F4	http://jaspar.genereg.net/matrix/MA0470.1/
FOXA1	http://jaspar.genereg.net/matrix/MA0148.1/
NRSF	http://jaspar.genereg.net/matrix/MA0138.2/
P53	http://jaspar.genereg.net/matrix/MA0106.2/

The parameters for rMotifGen to generate the simulated datasets were

i. Number of sequence = 100

ii. Sequence length = 200

iii. Percentage of nucleotide A, C, G, T = 25%, 25%, 25%, 25%

iv. Percentage of a sequence to contain a motif = 100%

As a result, for each sequence of a TF, it is expected to contain a motif as positive strand. This allows us to use the discovered motifs to locate the binding site and to identify whether it hits the location of the known location. If a motif hits the binding site location, then the motif is the true positive. Example of a sequence header of a generated FastA file,

```
>rMotifGen_RandSeq_1 1 57
```

The header denotes that the generated sequence is named as “rMotifGen_RandSeq_1”, and the true motif is located at position 57 of the sequence. Hence, once a motif is discovered, FIMO can be used to locate the motif on the given FastA file. Then, discovered motif can be identified whether it hits the position of the actual motif.

Table 3.10 shows the information of the simulated datasets.

Table 3.10: The average sequence length, total bp count, number of sequences, and the percentages of each nucleotide of the simulated TF datasets.

TF	Average length	Total bp count	Number of sequences	% of A	% of C	% of G	% of T
CTCF	213.07	21307	100	26.1	25	23.9	25
E2F4	205.09	20509	100	24.7	25.6	25.8	23.9
FOXA1	205.09	20509	100	25.7	25.2	25	24.1
NRSF	215.07	21507	100	24.7	25.6	24.5	25.2
P53	209.07	20907	100	24.5	25.5	25.2	24.8

3.5 Evaluation metric and tools for comparisons

The quality of motifs obtained is evaluated by using the ROC and AUC as presented in Section 2.8.1. ROC and AUC are common evaluation metrics being used in motif discovery (D. Lee et al., 2015; D. Lee, 2016; Min et al., 2016; van Heeringen & Veenstra, 2011).

"Rocpwm" is used to generate ROC and AUC of candidate motifs. The tool is bundled with GAPWM (L. Li et al., 2007). Rocpwm is chosen because it is able to calculate the ROC by using the PWM model. Rocpwm employs scoring function of a motif based on Match (Kel et al., 2003). By default, rocpwm can only handle 500 sequences of foreground dataset. For our use, the source codes are modified to handle 30000 sequences. The command requires dataset that contains the true motifs (that is the dataset has been searched for the motifs) and a dataset that contains the background sequences.

In addition, "fasta-dinucleotide-shuffle" is a tool from MEME Suite (Bailey et al., 2009) that is able to shuffle the nucleotides that the dinucleotide frequencies are explicitly

preserved. It is able to create background dataset from foreground dataset. It is part of the MEME-ChIP (Machanick & Bailey, 2011) algorithm. Dinucleotide shuffling from the positive sequences is a common method for negative datasets generation (Bailey, 2011; Kibet & Machanick, 2016; Levitsky et al., 2007; Qin & Feng, 2017). Alipanahi et al. (2015) and Zeng et al. (2016) used "fasta-dinucleotide-shuffle" to generate the negative sets for the experiment. Hence, the tool is also used to prepare the background sequences in this study.

By using rocpwm, each motif will generate an ROC which in turns is used to compute the AUC. Consequently, the accuracy of the discovered motifs can be measured. The higher AUC value implies the motif is more discriminative between the input and background sequences. Therefore, the motifs discovered by ENSPART and other benchmarking tools are able to be compared by using AUC. Figure 3.5 shows the process of discovered motifs measurement using ROC and AUC.

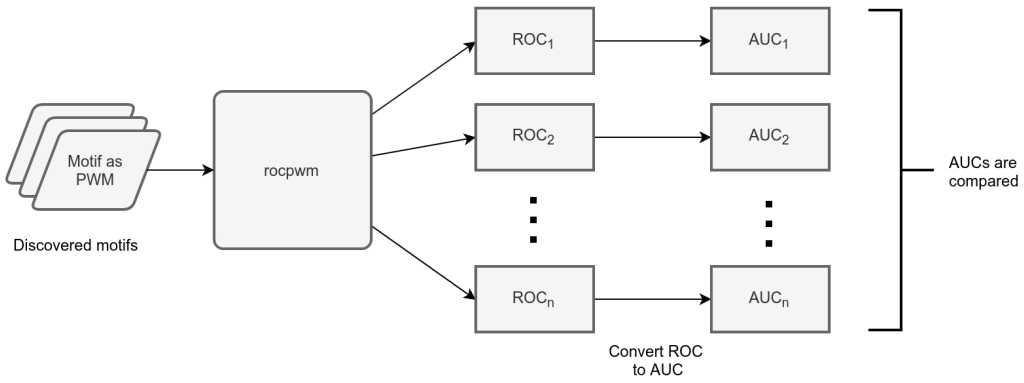


Figure 3.5: Process of converting discovered motifs to ROC and AUC. Finally the AUCs are used for performance comparison.

3.6 Conclusion

ENSPART is a novel motif discovery framework employs ensemble approach. It uses various traditional *de novo* motif discovery tools as its classifiers. They are MDscan, BioProspector, Weeder2, MEME-ChIP, AMD, AlignACE, W-AlignACE, and MotifSampler. In order to discover motifs from the large-scale datasets, ENSPART uses 30% sample of the dataset, and partitions the samples into three subsets. This study assumes that 30% sample size is sufficient to discover the motifs as from the whole size dataset. This is because, ensemble approach is suited to solve the problem with incomplete amount of data. Each individual classifier scans for each subset with different parameters. The results or candidate motifs from each classifier are collected. Next, ENSPART uses KFV for motif similarity comparison, with a threshold value of 0.27. The advantage of KFV is the comparison can be performed on the motifs of any length. The similar motifs are grouped and merged by averaging. The merging process can be performed multiple times to reduce the redundant results. The prediction accuracy of the discovered motifs are evaluated using ROC and AUC. Finally, the ROC is calculated according to Match (Kel et al., 2003).

CHAPTER 4

FINDINGS AND DISCUSSION

4.1 Introduction

This chapter presents the findings of the three different experiments on ENSPART framework. Three experiments with different datasets on ENSPART were being conducted.

i. ENSPART that employs partitioning of non-overlapping subsets from whole datasets.

ii. ENSPART that scans for the motifs from the fixed size dataset without partitioning.

iii. ENSPART that scans for the motifs of the simulated datasets with the known binding sites location.

For each experiment, the method of the simulation is presented. The different parameters used in ENSPART and comparative tools are presented as well. The comparison results are analysed and discussed.

4.2 Comparison to genome-wide motif discovery tools

In this simulation, Dataset 1 as described in Section 3.4.1 is employed. The purpose of this comparison is to compare ENSPART with genome-wide motif discovery tools, namely MEME-ChIP, ChIPMunk, and RSAT peak-motifs. We would like to ascertain the performances of ensemble tool in comparison to non-ensemble based tools.

Table 4.1: The tools that were ran on different datasets.

	MDscan	BioPropsector	MEME-ChIP	Weeder2	AlignACE	W-AlignACE	AMD
CREB	✓	✓	✓	✓	✓		✓
CTCF	✓	✓	✓	✓			✓
E2F4		✓	✓	✓	✓	✓	✓
FOXA1	✓	✓	✓	✓		✓	✓
FOXA2	✓	✓	✓	✓		✓	✓
NRSF	✓	✓	✓	✓	*[1]	✓	✓
NTERA	✓	✓	✓	✓	✓	✓	✓
OCT4	✓	✓	✓	✓			✓
P53	✓	✓	✓	✓	✓	✓	✓
STAT1	✓	✓	✓	✓			✓

[1] NRSF is partially scanned by AlignACE.

Table 4.1 shows the individual algorithms run on the datasets. During the experiment, it was found that AlignACE was too slow to scan through the partitioned dataset. Thus, only W-AlignACE was used for the following datasets as it is theoretically faster than AlignACE since it targets on ChIP datasets. However, the CREB dataset was scanned with AlignACE instead because W-AlignACE failed to complete the CREB scanning. It was possibly caused by the number and the length of the sequences. Then, OCT4, CTCF, and STAT1 were scanned without AlignACE or W-AlignACE because neither one completed the scanning. Each run of W-AlignACE took more than one or two hours. Furthermore, NRSF was partially scanned with AlignACE. It was only scanned for the first partition with the first run. The other runs were terminated due to time constraints. Though motif discovery process is failed like using AlignACE and W-AlignACE, it will not affect the result because motif discovery results from other tools are able to support the final output. Moreover, each tool was run three times with different parameters. The discovered candidate outputs were sufficient to draw the final outputs in ENSPART.

All experiments were run on computer with 4x Intel (R) Core (TM) i7–3687U CPU @

2.10GHz processors and 4GB memory. The operating system was Arch Linux, which was chosen because of the efficient compilation of the source code of the existing motif discovery tools.

To perform comparative studies, several tools were selected. MEME-ChIP was used as one of the benchmarking tools because it is based on MEME and able to deal with the large-scale datasets. Besides MEME-ChIP, RSAT peak-motifs (Thomas-Chollier et al., 2008), ChIPMunk (Kulakovskiy et al., 2010) were also used for benchmarking because they are able to handle the large-scale datasets like ChIP sequence.

Firstly, MEME-ChIP was used for the whole unpartitioned datasets. Then, the three partitioned datasets were combined to produce 30% of the whole datasets. The 30% datasets were then being searched by MEME-ChIP, RSAT peak-motifs, and ChIPMunk for candidate motifs.

Hence, the candidate motifs of the MEME-ChIP, RSAT peak-motifs, and ChIPMunk were parsed and converted to the Motif objects as stated previously, then converted to PWM in the GAPWM format.

In addition to running those tools with the partitioned datasets, the whole (unpartitioned) datasets were also used as input to MEME-ChIP for motif discovery.

This experiment responds to the two hypotheses:

i. The proposed ensemble framework that employs novel partitioning technique has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

ii. The proposed ensemble framework that employs novel merging technique has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

This is because, this experiment uses partitioning and merging techniques to discover the motifs, and compare the results to MEME-ChIP, ChIPMunk, and RSAT peak-motifs.

4.2.1 Findings

Firstly, the performance of ENSPART was evaluated by using the ten (10) datasets (Table 3.6). The seven (7) selected tools were run on each of the partitioned data subsets. Each tool was run three (3) times. The discovered motifs for each run was collected and the numbers of motifs predicted are shown in Table 4.2.

Table 4.2: Number of motifs discovered from the partitioned datasets.

Datasets	MDscan	BioProspector	Weeder2	MEME-ChIP	AMD	AlignACE	W-AlignACE	Total
CREB	30	45	172	45	55	0	0	347
CTCF	30	45	174	45	55	0	0	349
E2F4	30	45	161	45	55	111	262	709
FOXA1	30	45	152	45	66	0	496	834
FOXA2	30	45	165	45	42	0	325	652
NTERA	30	45	156	45	81	111	165	633
NRSF	30	45	169	45	25	3	290	607
OCT4	30	45	168	45	61	0	43	392
P53	30	45	171	45	50	28	534	903
STAT1	30	45	165	45	60	0	0	345

Noted that in Table 4.2, AlignACE produces no result for several datasets because AlignACE cannot handle the large dataset. For instance, CREB has large number of sequences and the average sequence length is long.

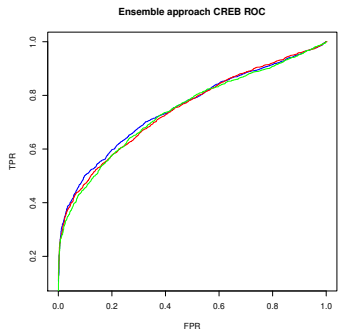
After obtaining the candidate motifs, next step was to merge those motifs using the KFV as similarity measure. Table 4.3 shows the number of motifs obtained in each dataset after the triple-merging.

Table 4.3: Comparison of number of motifs before and after merging.

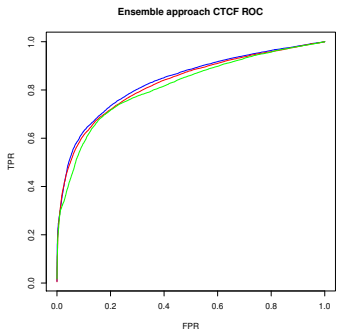
	Before merge	Merge 1st time	Merge 2nd time	Merge 3rd time	Reduced (%)
CREB	347	190	113	73	79.0
CTCF	349	181	109	68	80.5
E2F4	709	375	216	138	80.5
FOXA1	834	450	255	139	83.3
FOXA2	652	348	193	116	82.2
NRSF	607	322	181	107	82.4
NTERA	633	336	194	114	82.0
OCT4	392	191	109	71	81.9
P53	903	483	273	161	82.2
STAT1	345	170	102	64	81.4

After the merging of the candidate motifs, Table 4.3 shows that the number is reduced approximately to 79–83%. This indicates that a large number of the candidate motifs are variation of the same motifs.

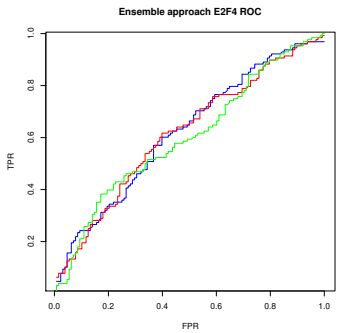
The ROC of each candidate motif was computed using "rocpwm" as explained in Section 2.8.1. Match (Kel et al., 2003) algorithm is used to compute information scores of the discovered motifs against foreground and background datasets. By using cut-off values, TPR and FPR can be calculated to plot the curve. The ROC represents the accuracy of the discovered motifs. This also means that, an ideal motif is expected to have the foreground sequences distinctive from the background sequences and produce large AUC. The ROC curves were also plotted using the best three motifs.



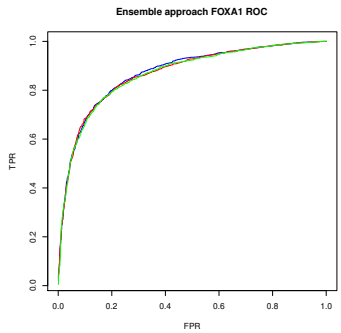
(a) CREB



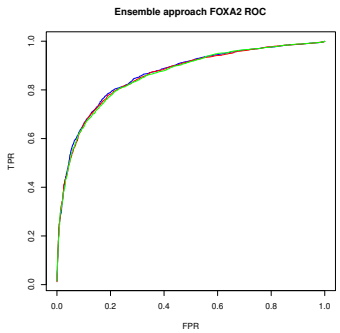
(b) CTCF



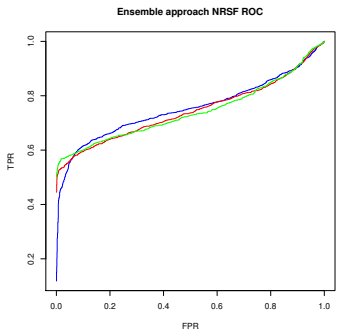
(c) E2F4



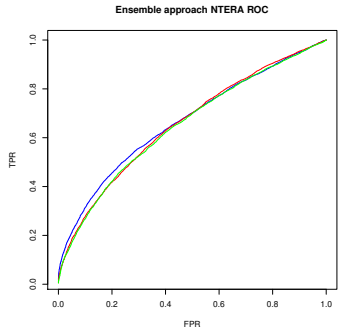
(d) FoxA1



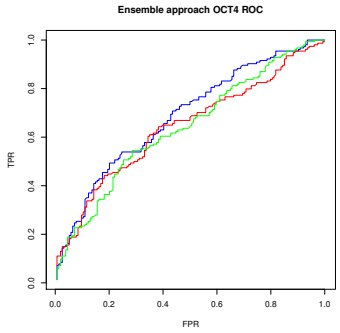
(e) FoxA2



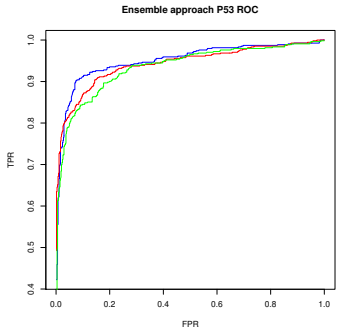
(f) NRSF



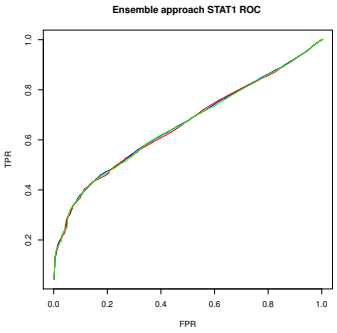
(g) OCT4 Ntera



(h) Oct4



(i) p53



(j) STAT1

Figure 4.1: Best three ROCs from ENSPART. Each curve is generated from a discovered PWM from ENSPART. Some of the charts show the curves that are similar. They may indicate same motifs which can be merged again. The curves of E2F4 and Oct4 are not overlapping, which means that they have distinctive PWMs.

Figure 4.1 shows the best three (3) ROC curves of the discovered motifs from the ten datasets. Each curve is plotted according to a motif predicted using Match algorithm. The larger area under the curve (or AUC), indicates the discovered motif has higher accuracy in the prediction. From the figure, we can notice that the curves were close to each other. This indicates that best three motifs are possibly their different variations, which can potentially be merged again. Notably CREB, CTCF, FoxA1, FoxA2, OCT4 Ntera, and STAT1 shows a strong similarity of the three best motifs, because the curves are overlapping. This indicates that the motifs are commonly discovered across the individual classifiers. On the other hand, the curves of E2F4, NRSF, Oct4, and p53 are dispersed, but they have similar shape. This indicates that the motifs are very different in terms of PWMs, but they possibly refer to same motif. Moreover, the motifs are possibly degenerated, causing the individual tools to discover motifs as different and dissimilar PWMs.

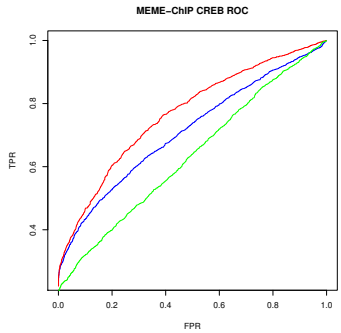
Table 4.4: Best AUCs and average AUCs of the discovered motifs with ENSPART.

	Number of motifs	Best AUC	Average AUC
CREB	73	0.7537	0.5941
CTCF	68	0.8384	0.5642
E2F4	138	0.6191	0.5223
FOXA1	139	0.8721	0.6127
FOXA2	116	0.8643	0.6119
NRSF	107	0.7532	0.6040
NTERA	114	0.6670	0.5651
OCT4	71	0.6847	0.5290
P53	161	0.9499	0.6564
STAT1	64	0.6607	0.5654

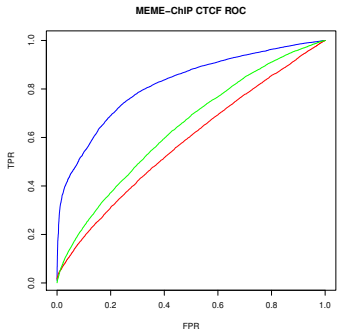
Table 4.4 shows the best AUCs and average of AUCs from the proposed ensemble approach. Best AUC is the highest AUC score from the discovered motifs from ENSPART. True motif

models are expected to have many positive hits in the input dataset, but has little hits in the negative datasets. Therefore, by identifying the best motif in terms of ROC, the quality of the obtained motif models can be compared by different tools. These are the motifs that are high confidence to be true motifs in the input dataset.

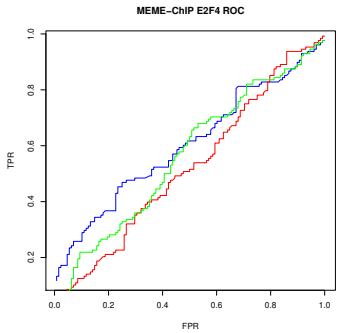
MEME-ChIP was used as the benchmark for the ensemble approach studied in this study. MEME-ChIP web service scanned for three motifs by default. The entire datasets were used for the motifs discovery using MEME-ChIP web service in order to discover the candidate motifs globally. The scanning by using MEME-ChIP web service was performed once, because the default parameters are used. Each discovered motif was converted to GAPWM output format to produce the ROC. The ROC results were plotted as in the Figure 4.2. In comparison to Figure 4.1, Figure 4.2 shows three distinctive curves for each dataset. This indicates that three discovered motifs by MEME-ChIP have very dissimilar pattern. MEME-ChIP discovers multiple motifs from the input datasets. Therefore, each discovered motif will have distinctive PWMs. In contrast to MEME-ChIP, ENSPART collects outputs from multiple classifiers. Hence, it is possible that ENSPART discovers similar motifs. The curve that has the largest AUC resemble the curves in Figure 4.1. Each dataset shows almost three distinct curves. This is because MEME-ChIP is run once to discover multiple potential motifs. From Figure 4.1 and Figure 4.2, they show that ENSPART is able to produce larger area under ROC on datasets such as E2F4, NRSF, and OCT4. This implies that, ensemble approach that utilises different motif discovery tools by ENSPART is able to discover the motifs that have better AUC scores, than using a single motif discovery tool such as MEME-ChIP. In addition, ENSPART uses smaller sample from the datasets, yet it is able to produce results that are equivalent or better than MEME-ChIP.



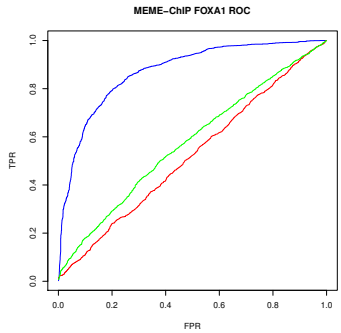
(a) CREB



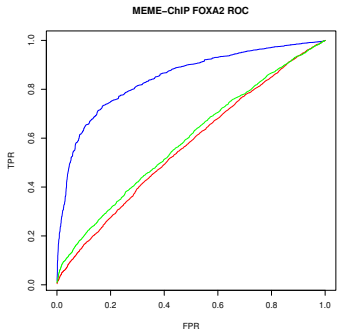
(b) CTCF



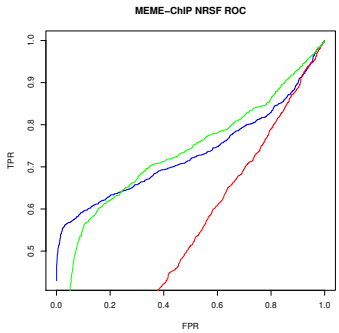
(c) E2F4



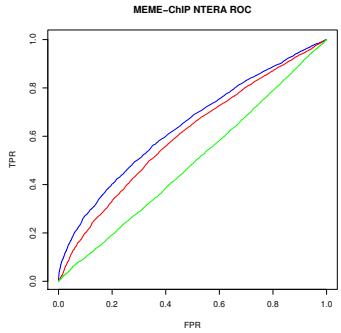
(d) FoxA1



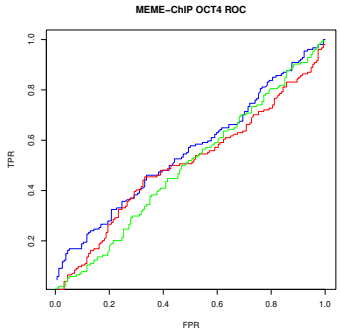
(e) FoxA2



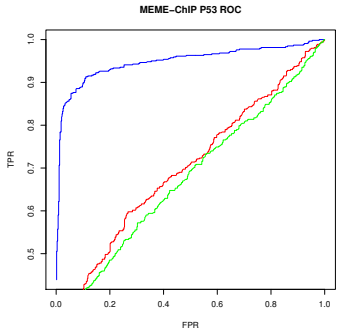
(f) NRSF



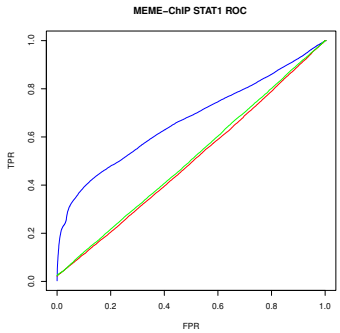
(g) OCT4 Ntera



(h) Oct4



(i) p53



(j) STAT1

Figure 4.2: ROCs from MEME-ChIP on whole datasets. Each curve is a discovered PWM from MEME-ChIP. In contrast to ENSPART, the curves are non-overlapping because they are discovered by one single tool, while ENSPART uses multiple tools as ensemble approach, which will discover similar PWMs among the tools.

30% of the datasets that were used for ensemble approach with partitioning were combined and scanned by MEME-ChIP, ChIPMunk, and RSAT peak-motifs. The results are shown and compared in Table 4.5. The average is calculated by averaging the AUCs from the discovered motifs. ChIPMunk does not have average because it only discover one motif.

The best AUCs of ENSPART and the best AUCs of MEME-ChIP with whole datasets are derived from the ROCs as shown in Figure 4.1 and Figure 4.2. For each dataset, the number of motifs discovered by MEME-ChIP is three (3), with only one for ChIPMunk. The number of motifs predicted by RSAT peak-motifs varies with different datasets. Table 4.5 shows that the best AUC values of ENSPART are better than MEME-ChIP (whole set) on 8 out of

Table 4.5: Comparison of the best AUC and average AUC between ENSPART, MEME-ChIP, ChIPMunk, and RSAT peak-motifs. MEME-ChIP was used on both whole (100%) datasets and 30% of the datasets. Other tools were used 30% of the datasets. There are five (5) best motifs reported by MEME-ChIP and one (1) best motif reported by ChIPMunk. The column "No. of Motifs" under RSAT peak-motifs (30%) is the number of motifs discovered by according to each dataset.

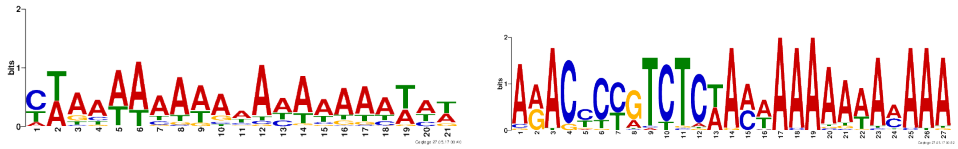
	ENSPART (30%)		MEME-ChIP (Whole)		MEME-ChIP (30%)		ChIP-Munk (30%)	RSAT peak-motifs (30%)		
	Best	Avg.	Best	Avg.	Best	Avg.	Best	Best	Avg.	No. of Motifs
CREB	0.7537	0.5941	0.7671	0.7042	0.6943	0.6656	**0.8054**	0.7306	0.6464	20
CTCF	**0.8384**	0.5642	0.8195	0.6804	0.6067	0.5694	0.8381	0.6720	0.6059	20
E2F4	0.6191	0.5223	0.6018	0.5627	0.5245	0.4813	**0.6525**	0.5954	0.5006	11
FOXA1	**0.8721**	0.6127	0.8717	0.6554	0.5966	0.5639	0.8665	0.8345	0.7390	20
FOXA2	**0.8643**	0.6119	0.8446	0.6666	0.5819	0.5438	0.8469	0.8308	0.7587	20
NRSF	0.7532	0.6040	0.7334	0.6610	0.5351	0.4988	0.7452	**0.7653**	0.6428	20
NTERA	**0.6670**	0.5651	0.6424	0.5791	0.6116	0.5717	0.6351	0.6026	0.5342	20
OCT4	**0.6847**	0.5290	0.5541	0.5181	0.6799	0.5762	0.6493	0.6011	0.5506	20
P53	0.9499	0.6564	0.9473	0.7747	0.7202	0.5639	**0.9515**	0.8358	0.6214	20
STAT1	0.6607	0.5654	0.6675	0.5580	0.5650	0.5420	**0.6732**	0.6653	0.5993	20
Average	0.7663		0.7450		0.6116		0.7664	0.7134		

10 of the input datasets, but slightly lower for the CREB and STAT1 dataset. In addition, MEME-ChIP with 30% of the datasets has poorer AUCs in comparison to both ENSPART and MEME-ChIP (whole set). By comparing ENSPART to ChIPMunk, the latter has better best AUCs on CREB, E2F4, P53, and STAT1 datasets. In addition, ENSPART has better motif AUC values than RSAT peak-motifs in 8 out of 10 of the datasets, except for NRSF and STAT1.

The average AUC values are computed from all discovered motifs of each TF dataset. Because of ENSPART uses partitioning, some discovered motifs may not globally appear in the whole dataset. But during the partitioning with random selection, the pattern appeared in the partition by chance. As a result, averaging the AUCs from these discovered motifs will reduce the average AUC values.

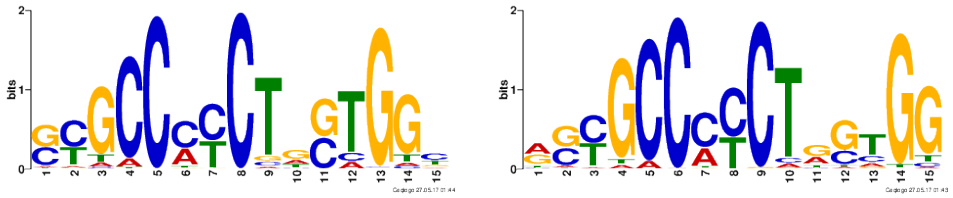
In summary of Table 4.5, it shows that on average, ENSPART has a best AUC value 0.7663, which is higher than MEME-ChIP on both whole datasets and 30% of the datasets. ENSPART also scores better than RSAT peak-motifs. However, ChIPMunk performed slightly better than ENSPART in the average of the best AUC.

Figure 4.3 to Figure 4.12 show the sequence logos of the best motifs predicted by ENSPART and MEME-ChIP. Sequence logo would allow us to compare the conservation characteristics of motifs obtained.



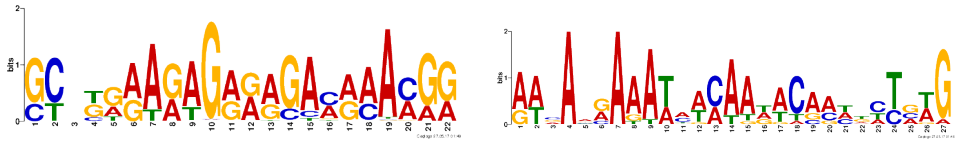
(a) (b)

Figure 4.3: Comparison of sequence logos obtained using CREB dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



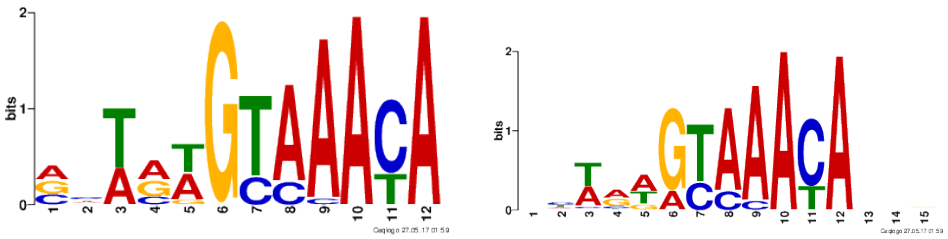
(a) (b)

Figure 4.4: Comparison of sequence logos obtained using CTCF dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



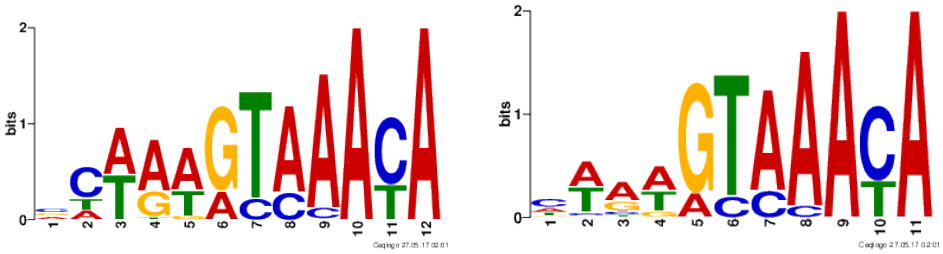
(a) (b)

Figure 4.5: Comparison of sequence logos obtained using E2F4 dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



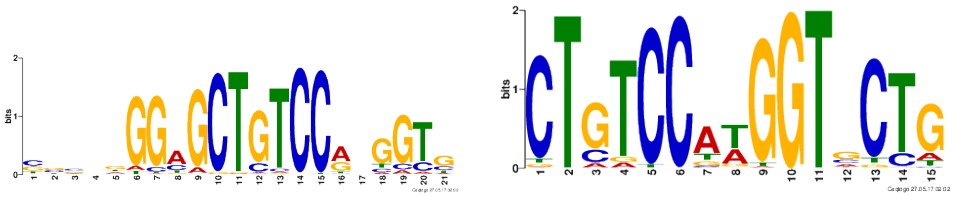
(a) (b)

Figure 4.6: Comparison of sequence logos obtained using FOXA1 dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



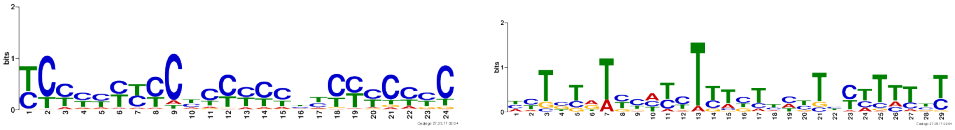
(a) (b)

Figure 4.7: Comparison of sequence logos obtained using FOXA2 dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



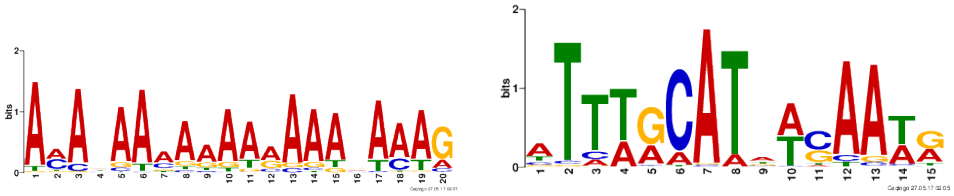
(a) (b)

Figure 4.8: Comparison of sequence logos obtained using NRSF dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



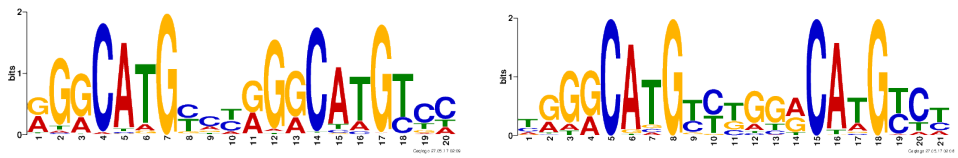
(a) (b)

Figure 4.9: Comparison of sequence logos obtained using NTERA dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



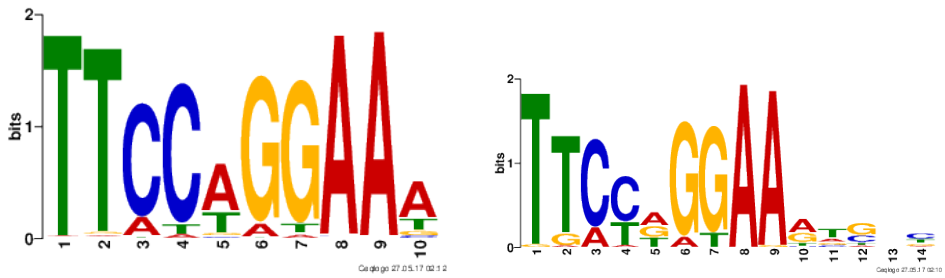
(a) (b)

Figure 4.10: Comparison of sequence logos obtained using OCT4 dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



(a) (b)

Figure 4.11: Comparison of sequence logos obtained using P53 dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.



(a) (b)

Figure 4.12: Comparison of sequence logos obtained using STAT1 dataset. (a) Motif predicted by ENSPART; (b) Motif predicted by MEME-ChIP.

The discovered motifs have also been scanned with Tomtom (Gupta et al., 2007) to match with JASPAR 2014 database. Table 4.6 shows the best matches.

Table 4.6: ENSPART motifs match with JASPAR 2014 database using Tomtom. P-value is the probability that the match occurred by random chance according to the null model. The upper sequence logo of each row belongs to motif from JASPAR, while the lower sequence logo of each row belongs to the best motif discovered by ENSPART.

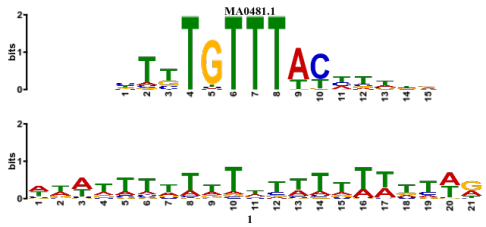
Name	JASPAR Name	p-value	Logo
CREB	FOXP1	4.87e-05	

Table 4.6 continued

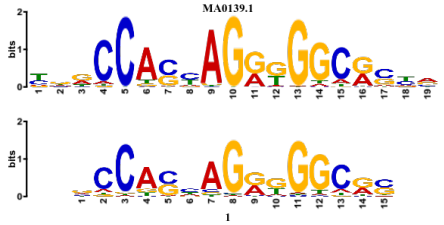
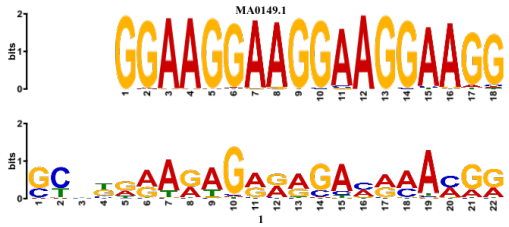
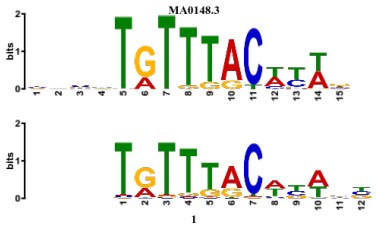
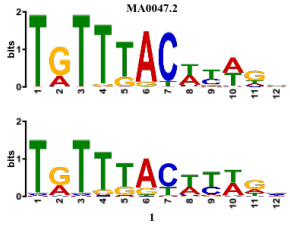
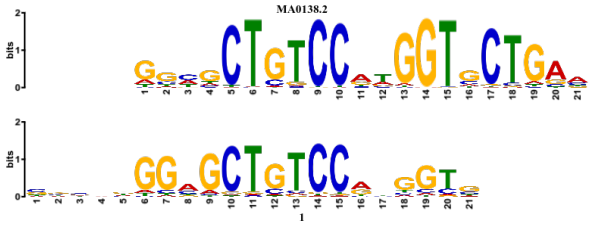
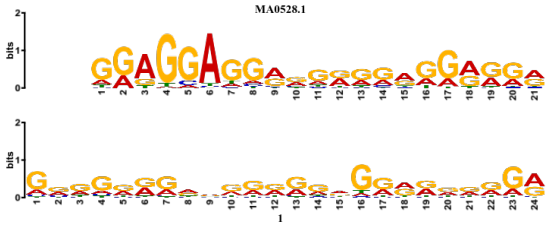
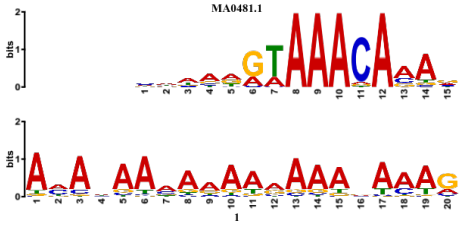
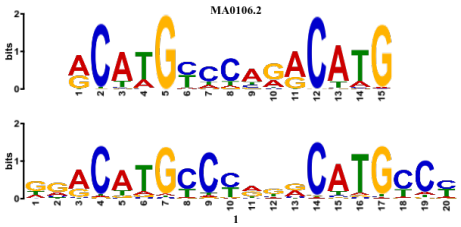
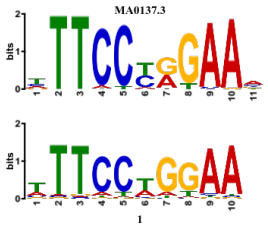
CTCF	CTCF	8.57e-18	
E2F4	EWSR1-FLI1	1.66e-04	
FOXA1	FOXA1	1.79e-09	
FOXA2	FOXA2	8.59e-17	
NRSF	REST	9.22e-08	
NTERA	ZNF263	3.55e-15	
OCT4	FOXP1	3.19e-05	

Table 4.6 continued

P53	TP53	4.64e-10	
STAT1	STAT1	2.49e-06	

The results show that 6 of 10 of the datasets could match correctly to the motifs annotated in JASPAR database. They are CTCF, FOXA1, FOXA2, NRSF, P53, and STAT1. However, the datasets CREB, E2F4, NTERA, and OCT4 matched to different TF motifs from JASPAR database. The best motifs from MEME-ChIP, ChIPMunk, and RSAT peak-motifs do not match correctly to motifs in JASPAR. A reason is the datasets contain motifs that are not annotated in the JASPAR database.

Next, paired sample t-test was used as the significance test on the means of the best AUCs. Table 4.7 shows the p-values of the t-test.

Table 4.7: P-values of paired sample t-test on the comparison of the best AUCs of the ENSPART and other algorithms.

Comparison	p-value
ENSPART – MEME-ChIP (Whole)	0.0474
ENSPART – MEME-ChIP (30%)	0.0002
ENSPART – ChIPMunk	0.4975
ENSPART – RSAT peak-motifs	0.0037

The p-value results show that, best AUC scores according to Match of ENSPART are statistical significantly better than MEME-ChIP on the whole datasets, MEME-ChIP on 30% of datasets, and RSAT peak-motifs, because the p-values are less than α (< 0.05). On the other hand, mean of best AUC of ENSPART is less than mean of best AUC of ChIPMunk. This indicates that ChIPMunk has better performance than ENSPART. However, the p-value of ENSPART and ChIPMunk comparison is not statistical significant at 0.05. Thus, there is not enough evidence to claim that accuracy of ChIPMunk is significantly better than ENSPART. The findings of this experiment can be summarised as:

i. ENSPART has better accuracy performance than MEME-ChIP on whole dataset in terms of AUC.

ii. ENSPART has better accuracy performance than MEME-ChIP on 30% of dataset in terms of AUC.

iii. ENSPART has better accuracy performance than RSAT peak-motifs in terms of AUC.

Moreover, the experiment also accepts the hypothesis that, by employing novel partitioning and merging techniques, ENSPART has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

4.3 Comparison using unpartitioned datasets

Datasets 2 as described in Section 3.4.2 is used for the comparison of ENSPART and the other individual motif discovery tools. Table 4.8 shows the information of the sampled datasets from Datasets 2.

Table 4.8: The average sequence length, total bp count, number of sequences, and the percentages of each nucleotides of the sampled datasets.

TF	Average length	Total bp count	Number of sequences	% of A	% of C	% of G	% of T
CEBPA	486.3	243164	500	29.4	20.5	20.6	29.5
CREB1	1174.7	587370	500	27	22.8	22.9	27.3
CTCF	187.8	93897	500	29.1	21.2	21.4	28.3
E2F4	860.0	430003	500	27.5	22.4	22.9	27.2
FOXA1	267.7	133828	500	30	20.2	20.4	29.5
KLF4	760.3	380143	500	28	21.9	22	28.1
MYCN	372.3	186155	500	19.7	30.3	30.1	19.9
NRSF	449.5	224739	500	22.9	27.3	27	22.8
P53	492.0	245981	500	30.2	20	20.2	29.6
STAT1	838.6	419275	500	28.6	21.5	21.2	28.7

4.3.1 Motif discovery tools

Similar to the previous experiment, the tools that were used for motif discovery were ENSPART, ChIPMunk, MEME-ChIP online, and RSAT peak-motifs. However, more intermediate results were collected for the evaluation purpose. These include the results from AMD, BioProspector, MEME-ChIP of ENSPART, MDscan, Weeder2, and MotifSampler of the ENSPART's individual classifiers. MEME-ChIP online is different from "MEME-ChIP from ENSPART" in terms of parameters used. MEME-ChIP online refers web service with default parameters, while "MEME-ChIP from ENSPART" refers to MEME-ChIP individual classifier from ENSPART that was being invoked three times with different parameters.

Besides that, AlignACE and W-AlignACE were dropped from ENSPART because both algorithms were extremely slow comparing to other individual algorithms. However, MotifSampler was added to ENSPART as a new individual classifier. The parameters for the individual algorithms within the ENSPART were same as the previous experiment.

ENSPART involves merging of the data *n* times, where the *n* was set to three (3). In this experiment, the results of each merging were also collected. This is to evaluate the performance of the multiple times of merging. Comparing to the previous experiment, this experiment uses the whole sample (size of 500 sequences) for all motif discovery tools.

Similar to the previous experiment, this experiment reponds to the hypothesis:

i. The proposed ensemble framework that employs novel merging technique has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

4.3.2 Findings

Table 4.9 shows the comparison of the best AUCs of the discovered motifs by ENSPART and the individual tools used by ENSPART.

Table 4.9: Comparison of best AUCs of the discovered motifs by ENSPART and individual tools. The AUCs of AMD, BioPropsector, MEME-ChIP, MDscan, MotifSampler, and Weeder2 are the best AUCs from the three (3) runs of each tool.

Tools	CEBPA	CREB1	CTCF	E2F4	FOXA1	KLF4	MYCN	NRSF	P53	STAT1	Average
ENSPART G1	**0.6108**	**0.7399**	0.4093	0.6906	**0.6759**	0.6552	0.8122	**0.6846**	**0.6300**	0.7001	**0.6609**
ENSPART G2	0.6078	0.7390	0.4093	0.6827	0.6754	**0.6609**	**0.8143**	**0.6846**	0.6292	**0.7003**	0.6603
ENSPART G3	0.6079	0.7360	**0.4129**	0.6827	0.6722	0.6524	**0.8143**	0.6841	0.6173	0.6776	0.6557
AMD	0.6082	0.7265	0.3890	0.6801	0.6657	0.6453	0.8122	0.6814	0.6065	0.6793	0.6494
BioProspector	0.5493	0.7334	0.3842	0.6824	0.6663	0.6418	0.7474	0.6217	0.5355	0.6755	0.6238
MEME-ChIP	0.5993	0.7289	0.4110	**0.6921**	**0.6759**	0.6490	0.7909	0.6623	0.6075	0.6930	0.6510
MDscan	0.5691	0.7098	0.3747	0.5766	0.5208	0.5602	0.7642	0.3731	0.5768	0.5745	0.5600
MotifSampler	0.5958	0.7282	0.4083	0.6693	0.6605	0.6492	0.8087	0.6759	0.6183	0.6854	0.6500
Weeder2	0.5856	0.7263	0.4020	0.6561	0.6302	0.6217	0.7790	0.5552	**0.6300**	0.6561	0.6242

ENSPART Gn denotes n times of merging. Therefore, ENSPART G1 indicates the candidate motifs were merged once based on the KFV similarity, G2 indicates the motifs were merged twice, and G3 indicates the motifs were merged three times. There are some AUCs showing the same value. For instance CTCF of ENSPART G1 and MEME-ChIP are both 0.4110. This is because the candidate motif discovered by MEME-ChIP was not merged with any other candidate and it produces the best AUC value. As being stated by Hu et al. (2006), the ensemble approach always performs better than individual motif discovery tools or at least had the same level of performance as the individual tools.

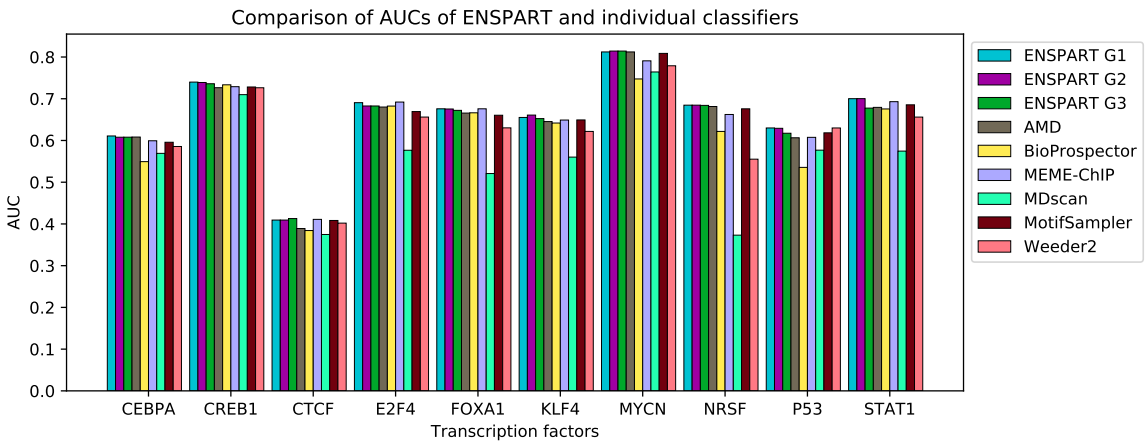


Figure 4.13: Comparison of AUCs from ENSPART and individual tools.

Figure 4.13 shows that ENSPART has better AUC scores comparing to other individual tools. Besides that, most motifs can achieve the AUC scores above 0.5, except CTCF. CTCF AUC score is possible affected by the characteristic of the dataset, which has the shortest average sequence length comparing to the other datasets. This causes the ROC based on Match algorithm does not get good score. However, ENSPART G3 is showing the highest score among the other tools. In addition, MDscan does not produce high AUC scores on E2F4,

FOXA1, KLF4, NRSF, and STAT1,

Next, paired sample t-test is conducted as the significance test on the average of the AUCs. Table 4.10 shows the p-values of the t-tests.

Table 4.10: Comparison of paired sample t-test p-values of the best AUCs from ENSPART and individual classifiers used by ENSPART.

	ENSPART G1	ENSPART G2	ENSPART G3
ENSPART G1		0.3207	0.0319
ENSPART G2	0.3207		0.0471
ENSPART G3	0.0319	0.0471	
AMD	0.0008	0.0017	0.0124
BioProspector	0.0022	0.0026	0.0058
MEME-ChIP	0.0050	0.0127	0.1275
MDscan	0.0024	0.0025	0.0032
MotifSampler	0.0002	0.0000	0.0105
Weeder2	0.0052	0.0060	0.0131

The first part of the table shows the p-values by comparing the AUCs of ENSPART G1, G2, and G3. Table 4.9 shows that the average values are decreasing when the number of merging is increased, but the p-values show that there is no significant decreasing of the average by comparing ENSPART G1 and G2. This is because the p-values are greater than 0.05.

The second part of the table shows the comparison of ENSPART G1, G2, and G3 to the individual classifiers: AMD, BioPropspector, MEME-ChIP MDscan, MotifSampler, and Weeder2. The table shows that, ENSPART G1, G2, and G3 are significantly better than the individual classifiers, because all the p-values are less than 0.05.

Besides that, ENSPART was also compared to contemporary ChIP-seq motif discovery tools as in previous experiment. They are ChIPMunk, MEME-ChIP (online), and RSAT peak-motifs. Table 4.11 shows the AUCs of the best candidate motifs discovered by the tools.

Table 4.11: Comparison of AUCs of the discovered motifs by ENSPART, ChIPMunk, MEME-ChIP (online), and RSAT peak-motifs.

Tools	CEBPA	CREB1	CTCF	E2F4	FOXA1	KLF4	MYCN	NRSF	P53	STAT1	Average
ENSPART G1	0.6108	**0.7399**	0.4093	**0.6906**	**0.6759**	0.6552	0.8122	0.6846	**0.6300**	0.7001	**0.6609**
ENSPART G2	0.6078	0.7390	0.4093	0.6827	0.6754	**0.6609**	**0.8143**	0.6846	0.6292	**0.7003**	0.6603
ENSPART G3	0.6079	0.7360	0.4129	0.6827	0.6722	0.6524	**0.8143**	0.6841	0.6173	0.6776	0.6557
ChIPMunk	**0.6153**	0.7237	**0.4226**	0.6716	0.5850	0.6563	0.7320	0.6194	0.6091	0.6988	0.6334
MEME-ChIP	0.5340	0.7289	0.3535	0.6734	**0.6759**	0.6436	0.7889	0.5767	0.5171	0.6557	0.6148
RSAT peak-motifs	0.5548	0.6828	0.3892	0.6646	0.6451	0.6184	0.7856	**0.6860**	0.6081	0.6347	0.6269

The results show that ChIPMunk has better AUCs on CEBPA and CTCF transcription factors, and RSAT peak-motifs has better AUC on NRSF. Notably, MEME-ChIP web service which uses default parameter has same AUC as ENSPART G1 and G2 on FOXA1 transaction factor. Since ENSPART employs MEME-ChIP as one of the classifier, hence both MEME-ChIP web service and ENSPART are able to discover the same candidate motifs.

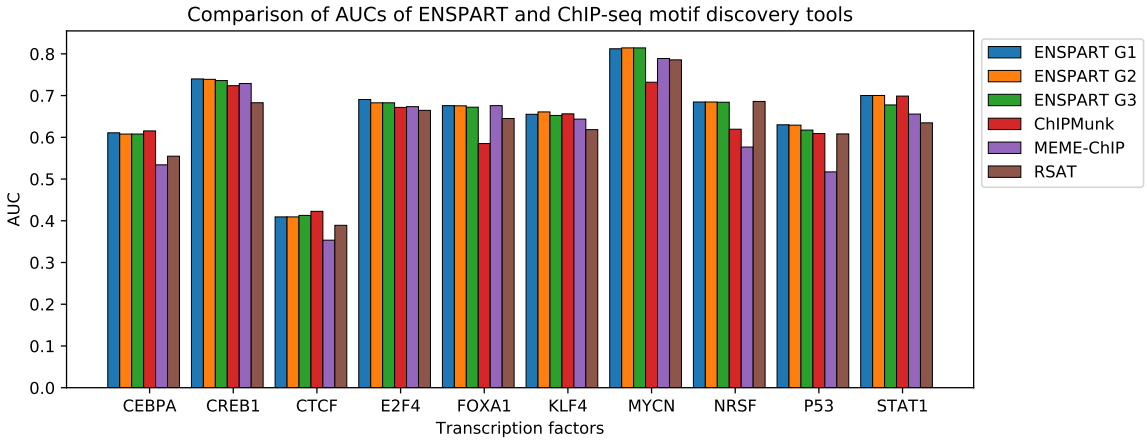


Figure 4.14: Comparison of AUCs from ENSPART, ChIPMunk, MEME-ChIP, and RSAT peak-motifs.

Figure 4.14 shows that ENSPART has higher AUC scores comparing to ChIPMunk, MEME-ChIP, and RSAT peak-motifs in most datasets, except CEBPA, CTCF, and NRSF. Notably, ChIPMunk does not score high AUCs on FOXA1, MYCN, and NRSF. On the other hand, RSAT peak-motifs scores high AUC on NRSF. The figure also shows that ENSPART results are generally higher than the ChIPMunk, MEME-ChIP, and RSAT peak-motifs.

T-test is performed on the means of ENSPART and the ChIP-seq motif discovery tools. Table 4.12 shows the t-test result.

Table 4.12: Comparison of paired sample t-test p-values of the best AUCs from ENSPART, ChIPMunk, MEME-ChIP (online), and RSAT peak-motifs.

	ChIPMunk	MEME-ChIP	RSAT
ENSPART G1	0.0229	0.0031	0.0003
ENSPART G2	0.0257	0.0033	0.0003
ENSPART G3	0.0562	0.0058	0.0003

T-test results show that all the p-values are less than α, 0.05 value. This indicates that, ENSPART G1, G2, and G3 are all significantly better than ChIPMunk, MEME-ChIP online, and RSAT peak-motifs in terms of means of best candidate motifs AUCs.

Consequently, Table 4.10 summarises that, there is enough evidence to support the claim that ENSPART has better accuracy performance than individual tools that were employed by ENSPART. Furthermore, Table 4.12 summarises that, there is enough evidence to support the claim ENSPART has better accuracy performance than ChIPMunk, MEME-ChIP, and RSAT peak-motifs. Moreover, p-value of ENSPART G1 and G3 does not provide enough evidence to support the claim that merging candidate motifs three times can improve the AUCs. However, p-value of ENSPART G1 and G2 indicates that there is enough evidence to support the claim that merging candidate motifs twice can improve the AUCs.

In summary, the t-tests of the experiment on the datasets without partitioning show that:

i. ENSPART has better accuracy performance than each individual classifier in terms of AUC.

ii. ENSPART has better accuracy performance than other contemporary ChIP-seq algorithms, namely ChIPMunk, MEME-ChIP, and RSAT peak-motifs, in terms of AUC.

Hence, the experiment accepts the hypothesis that, by employing novel merging technique, ENPART has better accuracy performance than the contemporary ChIP-seq motif discovery tools.

4.4 Comparison using simulated datasets

In this experiment, ENSPART was compared to GimmeMotifs and MotifVoter using Datasets 3 as described in Section 3.4.3. Table 4.13 shows a comparison of tools used by different ensemble algorithms. In the ensemble approaches, while the individual tools played a key role in identifying true motifs in an input dataset, merging the results obtained by different tools would be far more important. All the three ensemble techniques: ENSPART, GimmeMotifs, and MotifVoter employed a combination of tools from different categories: probabilistic, enumerative, and heuristic. While it is arguable that the individual set of tools used by those three ensemble methods are different, the aim of this study is to increase the chances to predict true motifs by merging the results from different tools, rather than using voting or averaging the results from individual tools. Hence, the final results does not depend on how many tools are used, but on how these tools come to a consensus on the motifs they produced. Having said that, it is essential to ensure diversification of the different categories of tools, rather than the number of tools used.

The parameters of ENSPART were identical to the parameters in the previous experiment on "datasets without partitioning". The individual tools used by GimmeMotifs were run with default parameters of each tool (van Heeringen & Veenstra, 2011). Similarly, MotifVoter run the individual tools with default parameters of each tool, except MDscan and MEME. In MotifVoter, MDscan's motif width parameter was set to 15, and MEME was set to assume that the motifs may appear more than once in a sequence (Wijaya et al., 2008).

Table 4.13: Comparison of individual tools used in ensemble algorithms. The three ensemble-based motif discovery tools use both probabilistic and enumerative individual tools. MDmodule is a modified version of MDscan. MotifVoter uses ANNSpec which is not categorised as probabilistic or enumerative.

Approach	ENSPART	GimmeMotifs	MotifVoter
Probabilistic	-	-	AlignACE
	BioProspector	BioProspector	BioProspector
	-	ChIPMunk	-
	-	Improbizer	Improbizer
	MEME-ChIP	MEME	MEME
	MotifSampler	MotifSampler	MotifSampler
Enumerative	AMD	AMD	-
	MDscan	MDmodule	MDscan
	-	-	Mitra
	-	Posmo	-
	-	-	SPACE
	-	-	Trawler
	Weeder2	Weeder	Weeder
ANN	-	-	ANNSpec

Table 4.13 shows the comparison of the individual tools of the ensemble approach in this experiment. ENSPART uses six (6) tools, GimmeMotifs uses nine (9) tools, and MotifVoter uses eleven (11) tools. ChIPMunk used by GimmeMotifs is one of tools that is compared in previous experiments, which shows high AUC scores in several comparisons.

4.4.1 Evaluation metric

Precision and recall rates were employed to compare the performance of ENSPART, GimmeMotifs, and MotifVoter. The motif profiles predicted by the three ensemble-based motif discovery tools were used to scan for binding sites using different matching threshold values. FIMO (Grant et al., 2011) from MEME Suite was used to detect the location of the candidate binding sites in the input sequences using the PWMs obtained from the tools. FIMO

calculates log-likelihood ratio score for each candidate site in the sequence according to the given PWM. The higher score indicates lower p-value for the likelihood between the located binding site and discovered motif. In the experiment, the negative strands were ignored, because rMotifGen only generates the sequences based on positive strands. The example of the FIMO output is shown as below,

sequence_name	start	stop	strand	score	p-value	q-value	matched_sequence
rMotifGen_RandSeq_1	62	73	+	13.0854	2.6E-05	0.0126	CTATTAATTAAA

The "start" and "stop" columns denote the start and end positions of the sequence that matches the motif's PWM. Score is the log-likelihood ratio score. P-value and q-value are calculated corresponding to the score. The "matched_sequence" is the k-mer of the sequence that is matched to the PWM. Therefore, with the "start" and "stop", location is able to identify whether the matched location is a hit to the known binding site locations.

In the evaluation, we also generated artificial sequences that do not contain any true sites with the same size of the input sequences, that is, 100 sequences each dataset (Table 3.10). These sequences are generated by using "fasta-dinucleotide-shuffle" tool available as part of MEME Suite. Combining the input and artificial sequences for FIMO site detection scrutinise the discrimination ability of motifs discovered by the evaluated tools. The true positive (TP), false positive (FP), and false negative (FN) are defined as below:

i. TP - The detected site is overlapped with at least 1 bp with a true site.

ii. FP - Detected sites that are not true sites.

iii. FN - The true sites that are failed to be detected.

Five (5) different p-value thresholds were used on FIMO, 0.0001, 0.005, 0.01, 0.02, and 0.05. The threshold parameter controls FIMO to display only the output that the p-value is less than the threshold parameter. Then the precision and recall rates were calculated according to Equation 4.27 and Equation 4.28 respectively,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Equation 4.27}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \qquad \text{Equation 4.28}$$

In addition, F1 score was also calculated according to Equation 4.29,

$$\text{F1 score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \qquad \text{Equation 4.29}$$

This experiment responds to the hypothesis:

i. The proposed ensemble framework is able to perform significantly better than several contemporary ensemble-based motif discovery tools.

4.4.2 Findings

Table 4.14: Sequence logos of the discovered motifs by ENSPART, GimmeMotifs, and MotifVoter. Expected sequence logos are the true motifs based on positions of the binding sites generated by rMotifGen.

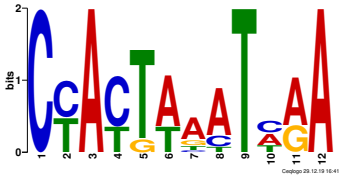
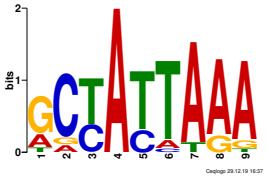
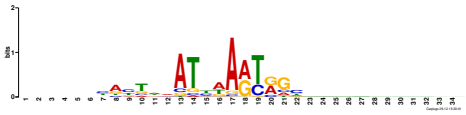
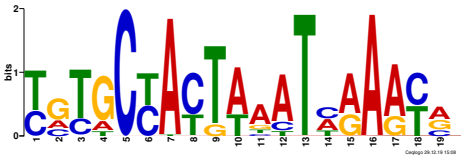
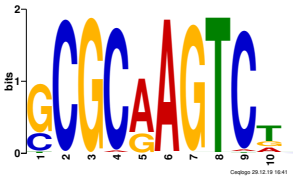
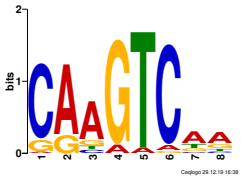
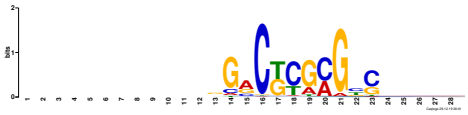
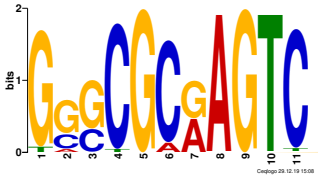
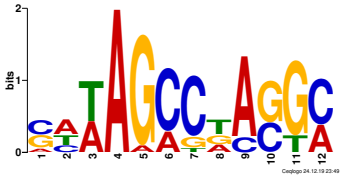
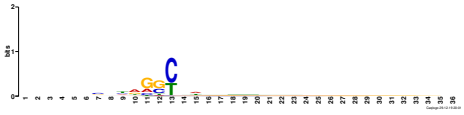
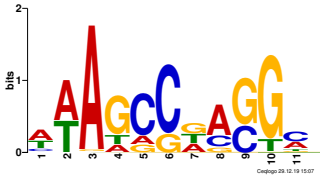
Name	Sequence logos	
CTCF	 ENPSART	 GimmeMotifs
	 MotifVoter	 Expected
E2F4	 ENPSART	 GimmeMotifs
	 MotifVoter	 Expected
FOXA1	 ENPSART	GimmeMotifs
	 MotifVoter	 Expected

Table 4.14 continued

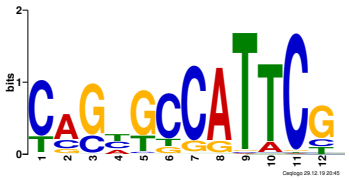
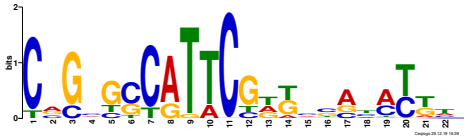
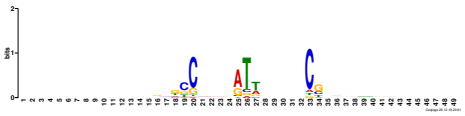
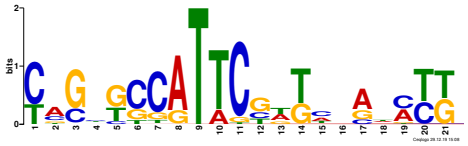
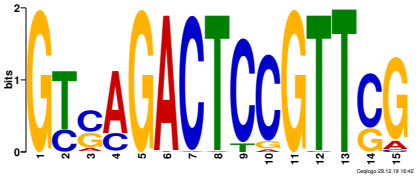
NRSF	 ENPSART	 GimmeMotifs
	 MotifVoter	 Expected
P53	 ENPSART	GimmeMotifs
	MotifVoter	Expected

Table 4.14 shows the sequence logos of the best motifs discovered by ENSPART, GimmeMotifs, and MotifVoter according to the precision and recall rates. The expected sequence logos are the simulated motifs by using rMotifGen tool. The sequence logos help the readers to understand the conservation characteristics of the discovered motifs of the tools. The sequence logos produced by MotifVoter are mostly having long motif length (CTCF, E2F4, FOXA1, and NRSF). This is because the discovered candidate motifs of the individual tools are aligned using MUSCLE which allows gap.

Table 4.15 shows the results of the precision and recall rates of the best motifs in terms of highest F1 score discovered by ENSPART, GimmeMotifs, and MotifVoter. It is noted that,

when the threshold increases from 0.0005 to 0.005, ENSPART maintains level of precision and recall rates. For examples, for CTCF, E2F4, and P53 datasets, their precision and recall rates at (0.990, 0.990), (0.942, 0.990), (1.000, 0.981) respectively. This indicates that the discovered motif is discriminative and high-quality. On the NRSF dataset, when the threshold value increases, the precision and recall rates of ENSPART decrease but remain better than GimmeMotifs and MotifVoter.

Table 4.15: Comparison of precision and recall rates of the best motifs between ENSPART, GimmeMotifs, and MotifVoter. The column "Prec." is precision rate.

		Enspart G1		Enspart G2		Enspart G3		GimmeMotifs		MotifVoter	
TF	Threshold	Prec.	Recall	Prec.	Recall	Prec.	Recall	Prec.	Recall	Prec.	Recall
CTCF	0.0001	0.989	1.000	0.989	0.989	1.000	0.989	0.921	0.972	0.977	0.955
	0.0005	0.990	0.990	0.971	0.990	0.925	0.934	0.875	0.894	0.944	0.883
	0.001	0.990	0.990	0.971	0.990	0.868	0.911	0.800	0.838	0.875	0.802
	0.002	0.990	0.990	0.971	0.990	0.847	0.870	0.732	0.721	0.748	0.688
	0.005	0.990	0.990	0.971	0.990	0.847	0.870	0.602	0.513	0.643	0.549
E2F4	0.0001	0.978	1.000	0.978	0.978	1.000	0.978	1.000	1.000	0.967	0.908
	0.0005	0.942	0.990	0.942	0.942	0.962	0.917	0.857	0.868	0.862	0.757
	0.001	0.942	0.990	0.942	0.942	0.962	0.917	0.792	0.764	0.746	0.683
	0.002	0.942	0.990	0.942	0.942	0.962	0.917	0.746	0.594	0.661	0.584
	0.005	0.942	0.990	0.942	0.942	0.962	0.917	0.542	0.455	0.547	0.431
FOXA1	0.0001	0.946	0.875	0.946	0.875	0.946	0.875	0.962	0.833	0.571	0.364
	0.0005	0.867	0.852	0.867	0.852	0.867	0.852	0.797	0.770	0.571	0.410
	0.001	0.756	0.756	0.756	0.756	0.756	0.756	0.634	0.634	0.500	0.394
	0.002	0.639	0.678	0.673	0.632	0.673	0.632	0.561	0.561	0.437	0.352
	0.005	0.619	0.647	0.595	0.657	0.595	0.657	0.409	0.367	0.395	0.320
NRSF	0.0001	0.984	0.968	0.952	1.000	0.945	0.963	0.964	0.964	0.600	0.750
	0.0005	0.873	0.904	0.873	0.873	0.833	0.885	0.909	0.885	0.423	0.393
	0.001	0.784	0.852	0.768	0.784	0.783	0.758	0.824	0.800	0.418	0.397
	0.002	0.661	0.750	0.663	0.661	0.676	0.649	0.733	0.696	0.447	0.362
	0.005	0.688	0.669	0.643	0.688	0.658	0.637	0.733	0.696	0.443	0.317
P53	0.0001	1.000	0.980	0.990	1.000	0.978	1.000	0.990	0.981	-	0.000
	0.0005	1.000	0.981	0.981	1.000	0.981	0.981	0.954	0.928	0.167	0.125
	0.001	1.000	0.981	0.981	1.000	0.981	0.981	0.954	0.928	0.167	0.125
	0.002	1.000	0.981	0.981	1.000	0.981	0.981	0.954	0.928	0.167	0.125
	0.005	1.000	0.981	0.981	1.000	0.981	0.981	0.954	0.928	0.167	0.125
Average		0.917	0.899	0.913	0.888	0.879	0.872	0.808	0.781	0.560	0.472

GimmeMotifs precision and recall rates are better than ENSPART at the threshold 0.005 on the NRSF dataset. The average of the precision and recall rates of the ENSPART are above 0.9 whereas GimmeMotifs and MotifVoter achieved (0.808, 0.781) and (0.560, 0.472) respectively. Hence, the table summarises that, on average ENSPART has better precision and recall performance than GimmeMotifs and MotifVoter.

There were perfect precision and recall rates (1.0 and 1.0 respectively) discovered by ENSPART for the FOXA1, which are not shown in the table. This is because the motif will hit TP when there is at least 1 bp overlaps on a true site, this implies that the best motif which has perfect precision and recall rates does not necessary perfectly resemble the expected motifs. As a result, these motifs are excluded from the report.

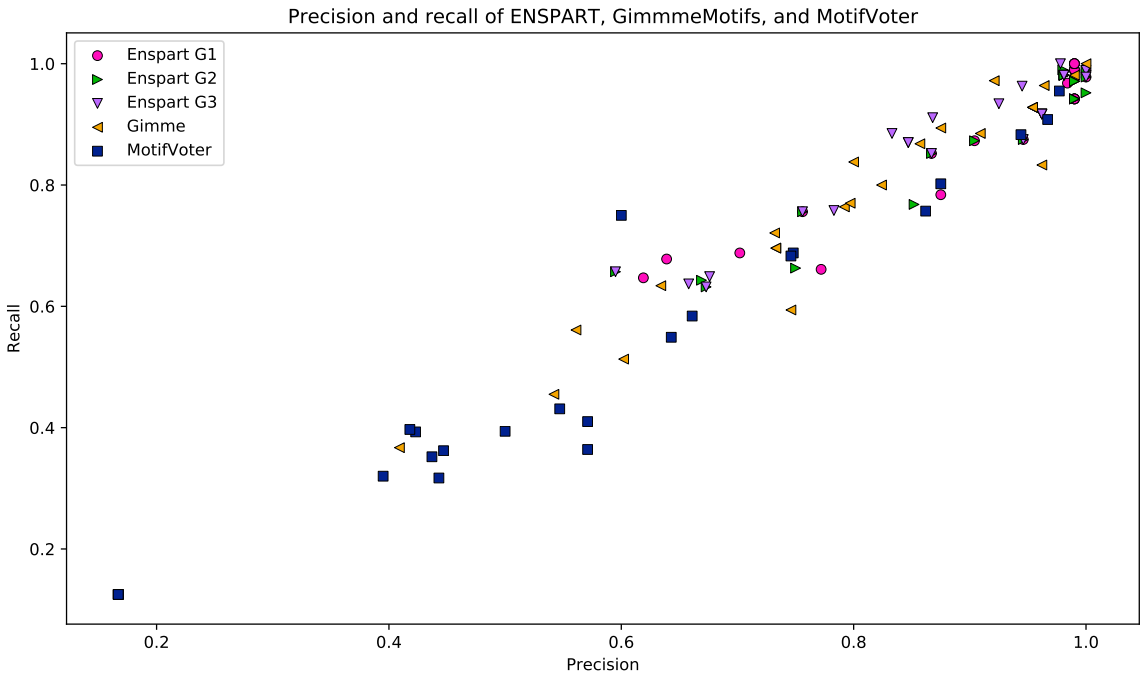


Figure 4.15: Scatter plot of the precision and recall rates for the three tools: ENSPART, GimmeMotifs, and MotifVoter. The points represent the precision and recall rate produced by FIMO at a certain threshold value.

Figure 4.15 shows the scatter plot of the precision and recall rates of ENSPART, GimmeMotifs, and MotifVoter. The plot shows that ENSPART results are more concentrated at the upper right. ENSPART results are generally above (0.6, 0.6) of the precision and recall. On the other hand, GimmeMotifs also produces result at the upper right of the chart. However, GimmeMotifs' results are more dispersed comparing to ENSPART. MotifVoter is also dispersed across the chart. Besides that, one of the MotifVoter results is having a very low precision and recall rates.

Table 4.16 shows the F1 score of each best motifs according to the FIMO threshold parameters. The table is corresponding to Table 4.15. Both tables show that the precision rates, recall rates, and F1 scores of ENSPART are decreased slightly when more merging are performed. This implies that, more merging might have negative effect on the quality of the motifs, and this is conform to the AUC results in the previous experiments. Nevertheless, the merging has greatly reduced the number of redundant motifs in the output. This table summarises that, the average of the F1 scores of ENSPART are higher than GimmeMotifs and MotifVoter. This also means that, ENSPART has better performance in motif discovery in terms of F1 score comparing to GimmeMotifs and MotifVoter.

Table 4.16: Comparison of F1 scores of the best motifs between ENSPART, GimmeMotifs, and MotifVoter.

TF	Threshold	Enspart G1	Enspart G2	Enspart G3	GimmeMotifs	MotifVoter
CTCF	0.0001	0.9947	0.9947	0.9947	0.9459	0.9655
	0.0005	0.9901	0.9806	0.9296	0.8842	0.9128
	0.001	0.9901	0.9806	0.8889	0.8186	0.8370
	0.002	0.9901	0.9806	0.8584	0.7266	0.7170
	0.005	0.9901	0.9806	0.8584	0.5536	0.5924
E2F4	0.0001	0.9889	0.9889	0.9890	1.0000	0.9365
	0.0005	0.9655	0.9655	0.9390	0.8627	0.8060
	0.001	0.9655	0.9655	0.9390	0.7778	0.7132
	0.002	0.9655	0.9655	0.9390	0.6615	0.6201
	0.005	0.9655	0.9655	0.9390	0.4944	0.4819
FOXA1	0.0001	0.9091	0.9091	0.9091	0.8929	0.4444
	0.0005	0.8595	0.8595	0.8595	0.7833	0.4776
	0.001	0.7561	0.7561	0.7561	0.6341	0.4407
	0.002	0.6582	0.6520	0.6520	0.5614	0.3897
	0.005	0.6324	0.6244	0.6244	0.3868	0.3532
NRSF	0.0001	0.9760	0.9756	0.9541	0.9643	0.6667
	0.0005	0.8879	0.8879	0.8586	0.8969	0.4074
	0.001	0.8268	0.8074	0.7705	0.8120	0.4071
	0.002	0.7121	0.7038	0.6621	0.7143	0.4000
	0.005	0.6947	0.6556	0.6472	0.7143	0.3696
P53	0.0001	0.9951	0.9849	0.9886	0.9854	-
	0.0005	0.9951	0.9806	0.9806	0.9406	0.1429
	0.001	0.9951	0.9806	0.9806	0.9406	0.1429
	0.002	0.9951	0.9806	0.9806	0.9406	0.1429
	0.005	0.9951	0.9806	0.9806	0.9406	0.1429
Average		0.9078	0.9003	0.8752	0.7933	0.5213

Table 4.17: Comparison of paired sample t-test p-values of precisions from ENSPART, GimmeMotifs, and MotifVoter.

	ENSPART G1	ENSPART G2	ENSPART G3
ENSPART G1		0.0684	0.0002
ENSPART G2	0.0684		0.0005
ENSPART G3	0.0002	0.0005	
GimmeMotifs	0.0001	0.0002	0.0024
MotifVoter	0.0000	0.0000	0.0000

Table 4.18: Comparison of paired sample t-test p-values of recalls from ENSPART, GimmeMotifs, and MotifVoter.

	ENSPART G1	ENSPART G2	ENSPART G3
ENSPART G1		0.0004	0.0005
ENSPART G2	0.0004		0.0081
ENSPART G3	0.0005	0.0081	
GimmeMotifs	0.0002	0.0007	0.0010
MotifVoter	0.0000	0.0000	0.0000

Table 4.19: Comparison of paired sample t-test p-values of F1 scores from ENSPART, GimmeMotifs, and MotifVoter.

	ENSPART G1	ENSPART G2	ENSPART G3
ENSPART G1		0.0002	0.0001
ENSPART G2	0.0002		0.0011
ENSPART G3	0.0001	0.0011	
GimmeMotifs	0.0001	0.0004	0.0014
MotifVoter	0.0000	0.0000	0.0000

Table 4.17 to Table 4.19 show the p-values of the paired sample t-tests from precision, recall, and F1 scores from the ENSPART, GimmeMotifs, and MotifVoter. The tables show that, ENSPART algorithm is significantly better than GimmeMotifs and MotifVoter in terms of

precision, recall, and F1 score (p-value < 0.05). Hence, the t-test results show that, there is enough evidence to support the claim that ENSPART achieved higher precision and recall than GimmeMotifs and MotifVoter using the five datasets.

On the other hand, when ENSPART merges the candidate motifs for n times, where $n = \{1, 2, 3\}$, the tables show that merging three times is significantly decreasing the results of precision rate, recall rate, and F1 score. The p-values of G1 and G3 from precision rate, recall rate, and F1 score are less than 0.05.

The results of Table 4.17 also shows that, this experiment accepts the hypothesis ENSPART is able to perform significantly better than some contemporary ensemble-based motif discovery tools, for instance GimmeMotifs and MotifVoter.

4.4.3 Coverage metric

Coverage metric (Schapire & Singer, 2000) was used to evaluate the performance of the three ensemble-based motif discovery tools: ENSPART, GimmeMotifs, and MotifVoter. The purpose of coverage is to assess the performance of the motif discovery tools for all possible labels of sequences, not only the top-ranked labels as ranked by FIMO. Coverage is defined as the average distance to cover all the possible labels assigned to a sample (T. Li, Zhang, & Zhu, 2006). It evaluates how many steps are needed, on average, to go down the list of labels in order to cover all the proper labels of the sequence (Schapire & Singer, 2000; M. L. Zhang & Zhou, 2007; M. L. Zhang, Peña, & Robles, 2009). It is a lowest-best metric that if the algorithm does not make any classification error, the coverage value will be zero (Schapire & Singer, 2000).

Let f as a function produced by a learning system, which given an instance x_i and its associated label set Y_i, a successful learning system will tend to produce larger values for labels in Y_i than those not in Y_i, that is, $f(x_i, y_1) > f(x_i, y_2)$ for any $y_1 \in Y_i$ and $y_2 \notin Y_i$. In motif discovery, it means that a motif m discovered by motif discovery tool, is expected to produce larger values in positive dataset that contains the motif than background dataset that does not contain the motif. This can be perfomed by FIMO, because FIMO computes the score of the motif matched in a given sequence, where the larger score means better match.

Hence, coverage can be denoted as Equation 4.30

$$coverage(f) = \frac{1}{n}\sum_{i=1}^{n} \max_{y \in Y_i} rank_f(x_i, y) - 1 \qquad \text{Equation 4.30}$$

where $rank_f$ is a ranking function. The purpose of the ranking function is to order the labels that the top-most labels are new instance. This implies that, $y_2 \notin Y_i$ is new instance and should be ranked higher than $y_1 \in Y_i$. This also means, if $f(x_i, y_i) > f(x_i, y_2)$, then $rank_f(x_i, y_1) < rank_f(x_i, y_2)$. Therefore, a successful motif m should produce low coverage score.

A modified coverage metric based on D. Wang and Lee (2009) was used. The coverage score of a motif m is denoted as Equation 4.31

$$coverage(m) = \frac{1}{n}\sum_{i=1}^{n} rank(x_i) \qquad \text{Equation 4.31}$$

where n is the number of motif instances, x_i is an instance of m, and $rank(x_i)$ is the number of random k-mers having the score higher than score of x_i. Hence, a random background

dataset based on *Homo sapiens* third order Markov model was generated by using RSAT web tool. The background dataset contains 2500 sequences, each sequence is 1000 bp long. The positive datasets are generated by rMotifGen in which the position of the binding sites are known. FIMO was used to calculate the score of every sequence in the datasets by using p-value threhold 0.0001.

Table 4.20: Coverage scores of the best motifs discovered by ENSPART, GimmeMotifs, and MotifVoter.

TF	Tool	Coverage
CTCF	ENSPART	612.0438
	GimmeMotifs	1205.2376
	MotifVoter	1302.4882
E2F4	ENSPART	506.0537
	GimmeMotifs	1225.9708
	MotifVoter	589.0833
FOXA1	ENSPART	909.5909
	GimmeMotifs	1467.6525
	MotifVoter	2764.5747
NRSF	ENSPART	1111.3540
	GimmeMotifs	749.0844
	MotifVoter	1531.3830
P53	ENSPART	546.0559
	GimmeMotifs	473.5612
	MotifVoter	1671.6216

Table 4.20 shows the coverage scores of ENSPART, GimmeMotifs, and MotifVoter according to each TF. The motif with the highest F1 score from each tool was selected for the evaluation. The result shows that 3 out of 5 datasets (CTCF, E2F4, and FOXA1), ENSPART produces lowest coverage scores. On the other hand, GimmeMotifs has lowest coverage scores for both NRSF and P53. Moreover, MotifVoter has lower score comparing to GimmeMotifs on E2F4 dataset. As a conclusion, ENPART has better performance

comparing to GimmeMotifs and MotifVoter.

4.5 Discussion

There were three experiments being conducted: (i) experiment on the partitioned datasets, (ii) experiment on the non-partitioned datasets, and (iii) experiment on the simulated datasets. The first experiment showed that ENSPART achieved 0.766 on average AUC using 30% of the datasets, which was significantly higher than MEME-ChIP with whole datasets (0.745), MEME-ChIP with 30% of the datasets (0.612) and RSAT peak-motifs with 30% of the datasets (0.713). The second experiment demonstrated that ENSPART achieved the highest average AUC comparing to individual tools. ENSPART produced average AUC 0.661, followed by MEME-ChIP (0.651), MotifSampler (0.650), AMD (0.649), ChIPMunk (0.633), RSAT peak-motifs (0.627), Weeder2 (0.624), BioProspector (0.624), MEME-ChIP online (0.615), and MDscan (0.560). The third experiment showed that ENSPART achieved the highest average precision rates, recall rates, and F1 scores comparing to GimmeMotifs and MotifVoter with (0.917, 0.899, 0.908), (0.808, 0.781, 0.793), and (0.560, 0.472, 0.521) respectively.

ENSPART algorithm demonstrated significantly better performance than using single motif discovery tools and existing state-of-the-art ensemble approaches. Though the process is exhaustive by running multiple motif discovery tools, by using the proposed merging method, it can improve the quality of the motif models produced. This was shown in the findings by using AUC evaluation in the experiment of using partitioned datasets. The advantage of the proposed merging method solves the problem with the motif length differences. It uses KFV to gather the similar motifs with different lengths. Then, it uses the sum of square error (SSE) to align the best position to merge the motifs. As a result, it is possible to produce longer

motifs.

ENSPART is able to perform better than MEME-ChIP, ChIPMunk, and RSAT peak-motifs because ensemble approach involves multiple motif discovery tools, and each motif discovery tool has its strength. It can discover some motifs which MEME-ChIP, ChIPMunk, and RSAT peak-motifs are not able to discover, since the three motif discovery tools above are using single algorithm. On the other hand, the strength of each individual motif discovery tool in ENSPART is complementary. This allows ENSPART to produce better motifs compared to MEME-ChIP, ChIPMunk, and RSAT-peak motifs as being shown in the findings.

By using the partitioning technique, ENSPART is able to reduce the search space so that the individual motif discovery tools are able to search through the partitioned datasets. This is a useful characteristic, because large datasets such as ChIP-seq datasets require different algorithm in order to perform motif discovery, for example ChIPMunk was developed for ChIP-seq datasets. However, by adopting ensemble approach, partitioning of the datasets and running multiple classifiers allow ENSPART to solve the large search space problem in the ChIP-seq datasets. This has been proved in the experiment with paritioned datasets, and it fulfils the objective of proposing a novel method that can reduce the sequence search space for motif discovery.

Besides that, another advantage of using KFV for motif similarity comparison is the feasibility of the output format. This is because PWM or PFM can be used directly for similarity comparison. Most of the *de novo* motif discovery tools are able to generate discovered motifs using matrix model such as PWM. Therefore, comparing ENSPART to

MotifVoter, ENSPART does not require to align the discovered binding sites in order to produce the new motifs. MotifVoter requires alignment of the discovered binding sites using MUSCLE (Edgar, 2004). This means that, the algorithm of MotifVoter is more exhaustive comparing to ENSPART, because it requires different interpretation of the outputs for each classifier, since the presentation of the discovered binding sites are varied across the motif discovery tools. The merging of the motifs using KFV is being implemented in the three experiments. The first and second experiments show that the discovered motifs through the merging can produce better performance in terms of AUCs. The results fulfil the objective of this study, that is, a novel merging technique being introduced in ENSPART that uses KFVs, without affecting the quality of the produced motifs.

GimmeMotifs uses a clustering approach very similar to ENSPART. GimmeMotifs uses WIC score (van Heeringen & Veenstra, 2011) to evaluate the similarity of the motifs in matrix form, then merge the similar motifs by averaging. However, the experiment of simulated data shows that ENSPART can discover the motifs that are significantly better than GimmeMotifs. It is possible that the individual classifiers being used and the merging the motifs according to the KFV in ENSPART are contributing to the precision and recall.

Furthermore, in the experiment of using partitioned datasets, ChIPMunk shows higher AUC scores, though it is not significantly better than ENSPART according to the paired sample t-test. There is a room for improvement for optimising ENSPART algorithm. Nevertheless, in the experiment of using non-partitioned datasets, ENSPART shows the AUC scores that are statistically significantly better than ChIPMunk. Moreover, in the experiment of simulated datasets, ENSPART also demonstrated the capability of discovered the best motif which has higher precision and recall comparing to GimmeMotifs, which employs

ChIPMunk as one of the individual classifiers. This shows the potentiality of ENSPART over ChIPMunk. Yet, further experiment may need to be conducted to identify the significance of the results.

In the experiments of using simulated datasets, the results show that the merging discovered motifs n times (as in ENSPART G2 and ENSPART G3), F1 scores are being reduced. This is possibly because the averaging of the matrix during merging will smooth the motif. As a result, the precision and recall of the best motif is slightly reduced. Nonetheless, in the experiment of non-partitioned datasets, the AUC scores of ENSPART G2 (KLF4, MYCN, and STAT1) and ENSPART G3 (CTCF, MYCN) are higher than ENSPART G1. This is because after the merging operation is performed, the motifs are smoothed and are better to categorise the foreground and background data sequences using Match algorithm.

In addition, one of the interesting findings from the experiment of partitioned datasets is that the ENSPART is able to discover the motifs globally with the incomplete information. This is corresponding to the characteristic of ensemble approach that it is able to solve the problem with insufficient amount of data (Polikar, 2006). This is a useful advantage because this implies that the traditional motif discovery tools which are not designed for large-scale datasets are able to discover the motifs from ChIP-seq datasets through partitioning.

ENSPART partitioning and merging method is similar to divide-and-conquer algorithm. Large-scale dataset has complex search space problem. Hence, by using partitioning, the search space is reduced. However, because of the dataset is divided, the discovered motifs are not definitely represent the global optimum. This explains the reason why using three sets of 10% dataset does not produce a better prediction performance comparing to ChIPMunk

which uses one set of 30% dataset. The advantage of using smaller sample size is that, the data sequences can be scanned by the traditional motif discovery tools. Nevertheless, this must involves sampling of the data with random selection. Therefore, the motifs or the patterns that have high frequency of occurrences are appearing in each sample. Non-overlapping partitioning is performed, this is to guarantee that 30% of the datasets are truly scanned. Each individual tool scans for each subset with different parameters is to make sure that different parameters are able to discover more potential motifs. Similar to GimmeMotifs, this may result same motifs discovered multiple times. Nonetheless, ENSPART uses a novel grouping algorithm. Each candidate motif is labelled to a group with the highest similarity based on the empirical threshold value 0.27 of the KFV comparison. Consequently, the motifs with higher number of occurrences will only affect to the other motifs that are within the same group. When merging the candidate motifs by averaging, it is possible the PWM is being smoothed. This causes the F1 scores being reduced when the number of merging increased, as shown in the experiment with simulated data. Therefore, over merging should be avoided.

This study shows that ENSPART has better performance than MEME-ChIP and RSAT peak-motifs by using partitioning method for ChIP-seq datasets. Moreover, the experiment also shows that ENSPART can discover the motifs which are significantly better than the individual classifiers that are being used. When ENSPART uses partitioning on the datasets, ChIPMunk shows better AUC comparing to ENSPART. However, ENSPART shows significant improvement comparing to ChIPMunk if partitioning is not being used. Lastly, ENSPART also demonstrates statistical significant better precision and recall rates than GimmeMotifs and MotifVoter. Morevoer, the coverage metric also shows that ENSPART has better performance comparing to GimmeMotifs and MotifVoter. The results of the experiment fulfil the objective of the study that, ENSPART is significantly better

than contemporary ensemble-based motif discovery tools, for instance, GimmeMotifs and MotifVoter.

The experiment results also summarise that the following hypotheses are accepted,

i. ENSPART has better accuracy performance than MEME-ChIP on whole dataset in terms of AUC.

ii. ENSPART has better accuracy performance than MEME-ChIP on 30% of dataset in terms of AUC.

iii. ENSPART has better accuracy performance than each individual classifier in terms of AUC.

iv. Without partitioning of the datasets, ENSPART has better accuracy performance than other contemporary ChIP-seq algorithms, namely ChIPMunk, MEME-ChIP, and RSAT peak-motifs, in terms of AUC.

v. ENSPART has better precision and recall rates than contemporary ensemble motif discovery algorithms, namely GimmeMotifs and MotifVoter.

4.6 Conclusion

In this study, ENSPART uses data partitioning method to reduce the search space for a traditional motif discovery tool. Instead of using all the samples from the whole dataset, only 30% of the dataset is actually used. Through the merging by using the KFV to measure the similarity, the results is able to produce accuracy higher than MEME-ChIP that scans

through the whole dataset. This is the advantage of ensemble learning that with the limited training data, it is still able to learn for prediction.

The experiment on the partitioned datasets showed that motif discovery on 30% datasets by ENSPART is able to achieve higher average AUC than MEME-ChIP with 100% of datasets. Besides that, ENSPART has better performance than ChIPMunk and RSAT peak-motifs in terms of AUC. The experiment on the non-partitioned datasets showed ENSPART is also performed better than individual motif discovery tools such as MotifSampler, AMD, ChIPMunk, RSAT peak-motifs, Weeder2, BioProspector, MEME-ChIP online, and MDscan. As the MotifSampler, AMD, Weeder2, BioProspector, MEME-ChIP, and MDscan are also used as the individual tools of ENSPART's ensemble approach, the experiment results demonstrated that the quality of the discovered motif is not solely contributed by a single individual tool. Lastly, the experiment on the simulated datasets demonstrated ENPSART is better than GimmeMotifs and MotifVoter in terms of precision rate, recall rate, and F1 score.

The findings of the study imply several important points:

i. Using ENSPART, it is able to discover the motifs from the partial datasets, that have the performance similar to the motifs discovered by single algorithm, namely MEME-ChIP, on the whole datasets.

ii. By using the proposed merging method, it is able to improve the accuracy of the motifs in represents as PWM in terms of AUCs.

iii. Over merging of the similar candidate motifs will reduce the F1 scores.

CHAPTER 5

CONCLUSION

5.1 Conclusion

i. Data partitioning is a technically feasible approach for employing multiple pre-ChIP era motif discovery tools for large-scale motif analysis task. The size of the partitions should be a parameter in the algorithm. While in the current study, a partition size of 10% is employed, the size can be adjusted according to the actual dataset size, i.e. smaller percentages for larger dataset sizes, so that it can be tackled by the classic motif discovery tools. The smaller size of the partition from the whole dataset indicates the smaller search space. By using this smaller partition or subset, the traditional motif discovery tools are able to scan and discover the candidate motifs. The results of the algorithm has been demonstrated in the experiment that uses partitioned datasets. Though the experiment does not measure the search space of the datasets directly, using the partitioned dataset is definitely smaller than the size of the whole datasets. Finally, the experiment results showed that ENSPART is able to perform better than MEME-ChIP, ChIPMunk, and RSAT peak-motifs.

ii. The effectiveness of ensemble method lies in the design of the merging algorithm of multiple redundant motifs. Hundreds or thousands of candidate motifs can possibly be returned when multiple motif discovery tools with multiple runs are employed. We found the alignment free algorithm for motif profiles comparison is effective and computationally efficient. While in this study only KFV method is used to represent the motifs, other alignment free methods would be likewise useful. The KFV with the $k = 2$ is used in this study, as opposed to the higher values such as $k = 3$ or $k = 4$.

Though $k = 4$ was reported to achieve the maximal overall accuracy for similarity comparison (M. Xu & Su, 2010), it is not used in this study as it is 22 times slower than $k = 2$. Moreover, a threshold value is introduced in ENSPART for the merging. This implies that, using higher value of k requires re-defining the threshold value, which is possibly producing similar result after the merging as using $k = 2$. Though higher k can achieve accuracy for similarity comparison, there is no guarantee that higher k will improve the performance of ENSPART. The hypothesis of higher k that can produce better result in ENSPART requires further empirical study.

iii. AUC is one of the measurement metrics of this study. AUC is commonly used in various bioinformatic studies, such as GAPWM (L. Li et al., 2007), GimmeMotifs (van Heeringen & Veenstra, 2011), MotifLab (Klepper & Drabløs, 2013), DeMo Dashboard (Lanchantin et al., 2016), EMQIT (Smolinska & Pacholczyk, 2017), and deep learning in bioinformatics (Min et al., 2016). This study adopts the AUC scoring function from GAPWM which uses Match (Kel et al., 2003) algorithm. Match algorithm calculates the information content of the discovered motifs against foreground and background data. Hence, TPR and FPR can be computed. Cut-off values are used to plot the ROC curve. A good motif is able to classify the true positive and true negative from the given data. Therefore, a good motif is expected to have large AUC. By using ROC generation tool from GAPWM, namely "rocpwm", the discovered motifs are able to be represented as ROCs, and AUCs are computed. The higher AUC score indicates that the information content of the discovered motif in PWM model has better accuracy. Besides that, precision rate and recall rate are also used in this study for the evaluation on motif discovered from the simulated datasets.

iv. The performance of ensemble versus genome-scale motif discovery tools are

comparable or even better reflected in our comprehensive evaluation results. Genome-scale motif discovery tools are usually heuristic in nature for speed-up purpose and that would miss out many true binding sites. On the contrary, the multiple runs of various individual motif discovery tools in ensemble can increase the chances of obtaining more true binding sites. Hence, it is an advantage to employ ensemble rather than just a single genome-scale motif discovery tool.

v. While ENSPART performances rely heavily on the reliability of individual motif discovery tools, true positive and false positive can be distinguished through the over-representation evaluation of the merged motifs in every iteration. Hence, false positives would be filtered out. Furthermore, different tools are applying motif search strategy that are complementary. That increases the likelihood of searching motifs of different characteristics. For instance, words-based tools are effective at searching for motifs which are long and conserved motifs, while EM and Gibbs sampling-based tools are better for less conserved motifs.

vi. In terms of performance, motifs produced by ENSPART show significant improvement on their discriminating property (i.e. AUC) than by its individual classifiers. That result is consistent with existing results reported in the existing ensemble algorithms. However, as ensemble algorithm runs multiple tools, the total time taken if run them sequentially would be longer in comparison to the genome-scale tools. The classical motif discovery tools such as AlignACE and MotifSampler are poorly scale with increases of dataset size. Therefore, using them to search for motifs from a subset of whole input dataset would be time-consuming as well. In the ENSPART design, we have not considered the efficiency of the merging algorithm, therefore its speed is still considerably slow.

As a conclusion, the research study fulfils the objectives:

i. A novel ensemble approach framework, namely ENSPART, is developed and it can reduce the search space by using partitioning method.

ii. By using a novel merging technique, ENSPART is able to show significant improvement of the accuracy than ChIP-seq motif discovery tools such as MEME-ChIP, ChIPMunk, and RSAT peak-motifs.

iii. ENSPART shows significant improvement of the precision and recall comparing to the contemporary ensemble-based motif discovery tools such as GimmeMotifs and MotifVoter.

5.2 Limitations

i. The proposed ensemble approach with partitioning produces false positives based on the Match (Kel et al., 2003) when plotting the ROCs. The possible reason is caused by the motifs appeared in the partitions when scanned by the individual classifiers using ensemble approach, yet these candidate motifs were not appeared in the whole datasets when evaluated by Match for ROC. This limitation can be improved by pruning the candidate motifs after calculating Match score from the whole datasets. The pruning of these false positive motifs is impossible to work without the whole dataset information.

ii. There are various evaluation metrics of motifs, such as the ROC AUC (van Heeringen & Veenstra, 2011; L. Li et al., 2007) and the F-score (Z. Wei & Jensen, 2006). However, the discovered motif's instances require further verification if they are indeed functioning. The lack of lab verified binding sites hinders the more objective evaluation of a novel algorithm. Tompa et al. (2005) datasets which were frequently

used for benchmark are non-NGS datasets. Thus, they are not used in this research study.

iii. The merging method in ENSPART can be further improved it terms of speed and accuracy. The current implementation perform pair-wise alignment of matched motif profiles, which is heuristic. Multiple aligning of all matched motifs can be difficult to implement as it is a NP-hard problem. Changing the order in which how the matched motifs is merged can produce a different final motif. While in our evaluation, the effect of that "ordering" issue seems not concerning, further study is needed to ensure that it does not affect negatively on the quality of the final motif.

iv. Due to the limitation of existing ensemble tools, only GimmeMotifs and MotifVoter are selected for comparison. Most existing tools can only accept very small input dataset sizes which is not feasible for our datasets used in this study. From our comprehensive literature search, GimmeMotifs is the most recent and best ensemble tool for DNA motif discovery, therefore, it is safe to conclude that ENSPART performed better than the existing state-of-the-art ensemble approach.

5.3 Future works

Some possible future works from this study were discussed.

In this study, ENSPART applies the merging process three (3) times to reduce redundantly discovered motifs. Future research can study the results of the motifs from different number of merging. For example, the merging process can be done iteratively by using a validation set so that best result can be discovered. The algorithm can also be integrated with simulated annealing technique to avoid being trapped in local optimum. While the current threshold

value 0.27 used for merging work reasonably well, it should be further investigated how to properly choose this value as guideline to users.

In addition, it will be interesting to use parallel computing technique to bring ENSPART to another level to bring the whole learning process into the pipeline. For example, GimmeMotifs employs inter process communication (IPC) in the ensemble learning and CUDA-MEME uses CUDA to accelerate the motif discovery process. CUDA is a parallel computing platform introduced by NVidia. Deep learning frameworks use CUDA-enabled GPU to speed up the learning process. These are all based on parallel computing. This will reduce the computation time, which also implies that the larger partitions or more partitions can be experimented.

Several motif discovery tools were implemented with web interface to ease the operations from the biologists, such as WebMOTIFS (Romer et al., 2007), RSAT (Thomas-Chollier et al., 2008), MotifVoter (Wijaya et al., 2008), and MEME Suite (Bailey et al., 2009). ENSPART can also be implemented with the web interface in the future, this allows users to study the discovered motifs in a more intuitive way. Since all the individual motif discovery tools employed by ENSPART are command-lines, it is possible to represent the output with more visual information for the users. This will help the biologists to analyse the discovered motifs from ENSPART.

ENSPART is able to use any motif discovery tool as long as it supports command-line interface. There are more motif discovery tools not being employed by ENSPART, such Trawler, GADEM, Improbizer, and Homer. It will be interesting to study the effect of different motif discovery tools on the result. We may discover the correlation of the

individual motif discovery tools and the final output.

Other than changing the individual motif discovery tools of ENSPART, we can also study the optimal parameters of ENSPART by using Genetic Algorithm (GA) or Particle Swarm Optimisation (PSO). The parameters such as number of partitions, size of partition, number of runs, and similarity threshold value are possible to be optimised. This may improve the overall quality of the discovered motifs.

Genomic analysis is a broad area with numerous features can be explored and studied. Though many algorithms show high accuracy of prediction, there is still room for improvement to study especially human genome. This is because the function of nearly three billion bases in human genome is unknown (Consortium, 2012). Various modern technologies such as deep learning can be applied together with ensemble approach in bioinformatics research.

REFERENCES

Alipanahi, B., Delong, A., Weirauch, M. T., & Frey, B. J. (2015). Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology*, *33*(8), 831–838. Retrieved from http://www.nature.com/doifinder/10.1038/nbt.3300 doi: 10.1038/nbt.3300

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, *25*(17), 3389–3402. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/9254694

Andersson, R., Gebhard, C., Miguel-Escalada, I., Hoof, I., Bornholdt, J., Boyd, M., … Sandelin, A. (2014). An atlas of active enhancers across human cell types and tissues. *Nature*, *507*(7493), 455–461. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/24670763 doi: 10.1038/nature12787

Ao, W., Gaudet, J., Kent, W. J., Muttumu, S., & Mango, S. E. (2004). Environmentally induced foregut remodeling by PHA-4/FoxA and DAF-12/NHR. *Science*, *305*(5691), 1743–1746. Retrieved from http://www.sciencemag.org/content/305/5691/1743.abstract doi: 10.1126/science.1102216

Bailey, T. L. (2011). DREME: motif discovery in transcription factor ChIP-seq data. *Bioinformatics*, *27*(12), 1653–1659.

Bailey, T. L., Boden, M., Buske, F. A., Frith, M., Grant, C. E., Clementi, L., … Noble, W. S. (2009). MEME SUITE: tools for motif discovery and searching. *Nucleic Acids Research*, *37*(Web Server issue), W202–W208. Retrieved from http://nar.oxfordjournals.org/content/early/2009/05/20/nar.gkp335.full doi: 10.1093/nar/gkp335

Bailey, T. L., & Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in bipolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology* (pp. 28–36). Menlo Park, California: AAAI Press. Retrieved from http://www.cs.toronto.edu/$\sim$brudno/csc2417_15/10.1.1.121.7056.pdf

Bailey, T. L., & Elkan, C. (1995). The value of prior knowledge in discovering motifs with MEME. In *Proceedings of the Third Intelligent Systems for Molecular Biology* (Vol. 3, pp. 21–29). Retrieved from http://www.aaai.org/Papers/ISMB/1995/ISMB95-003.pdf

Bailey, T. L., & Gribskov, M. (1998). Combining evidence using p-values: application to sequence homology searches. *Bioinformatics*, *14*(1), 48–54. Retrieved from http://bioinformatics.oxfordjournals.org/content/14/1/48.short doi: 10.1093/bioinformatics/14.1.48

Bailey, T. L., Williams, N., Misleh, C., & Li, W. W. (2006). MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Research*, *34*(Web Server issue), W369–W373. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1538909 doi: 10.1093/nar/gkl198

Barash, Y., Bejerano, G., & Friedman, N. (2001). Algorithms in Bioinformatics. In O. Gascuel & B. M. E. Moret (Eds.), *Algorithms in Bioinformatics* (Vol. 2149). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://link.springer.com/10.1007/3-540-44696-6 doi: 10.1007/3-540-44696-6

Barrett, T., Troup, D. B., Wilhite, S. E., Ledoux, P., Rudnev, D., Evangelista, C., … Edgar, R. (2009). NCBI GEO: archive for high-throughput functional genomic data. *Nucleic Acids Research*, *37*(Database issue), D885–D890. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18940857 doi: 10.1093/nar/gkn764

Ben-Hur, A., Ong, C. S., Sonnenburg, S., Schölkopf, B., & Rätsch, G. (2008). Support vector machines and kernels for computational biology. *PLoS Computational Biology*, *4*(10), e1000173. Retrieved from http://dx.plos.org/10.1371/journal.pcbi.1000173 doi: 10 .1371/journal.pcbi.1000173

Bi, C., & Rogan, P. K. (2004). Bipartite pattern discovery by entropy minimization-based multiple local alignment. *Nucleic Acids Research*, *32*(17), 4979–4991. Retrieved from http://nar.oxfordjournals.org/content/32/17/4979.short doi: 10.1093/nar/gkh825

Boyer, L. A., Lee, T. I., Cole, M. F., Johnstone, S. E., Levine, S. S., Zucker, J. P., … Young, R. A. (2005). Core transcriptional regulatory circuitry in human embryonic stem cells. *Cell*, *122*(6), 947–956. Retrieved from http://www.sciencedirect.com/science/article/pii/S0092867405008251 doi: 10.1016/j.cell.2005.08.020

Brazma, A., Jonassen, I., Eidhammer, I., & Gilbert, D. (1998). Approaches to the automatic discovery of patterns in biosequences. *Journal of Computational Biology*, *5*(2), 279–305. Retrieved from http://online.liebertpub.com/doi/abs/10.1089/cmb.1998.5 .279

Brejová, B., DiMarco, C., Vinar, T., Hidalgo, S., Holguin, G., & Patten, C. (2000). *Finding patterns in biological sequences* (Tech. Rep.). University of Waterloo. Retrieved from http://compbio.fmph.uniba.sk/$\sim$bbrejova/papers/data/2000motiftr.pdf

Buhler, J., & Tompa, M. (2002). Finding motifs using random projections. *Journal of Computational Biology*, *9*(2), 225–242. Retrieved from http://online.liebertpub.com/doi/abs/10.1089/10665270252935430 doi: 10.1089/10665270252935430

Carlson, J. M., Chakravarty, A., DeZiel, C. E., & Gross, R. H. (2007). SCOPE: a web server for practical de novo motif discovery. *Nucleic Acids Research*, *35*(Web Server), W259–W264. Retrieved from https://academic.oup.com/nar/article-lookup/

doi/10.1093/nar/gkm310 doi: 10.1093/nar/gkm310

Carvalho, A., Freitas, A. T., Oliveira, A. L., & Sagot, M. F. (2005). A highly scalable algorithm for the extraction of cis-regulatory regions. *Proceedings of 3rd Asia-Pacific Bioinformatics Conference (APBC'05)*, 273–282. Retrieved from http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.102.9439

Chan, T. M., Leung, K. S., & Lee, K. H. (2007). TFBS identification by position- and consensus-led genetic algorithm with local filtering. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation - GECCO '07* (p. 377). New York, New York, USA: ACM Press. Retrieved from http://portal.acm.org/citation.cfm?doid=1276958.1277037 doi: 10.1145/1276958.1277037

Chan, T. M., Leung, K. S., & Lee, K. H. (2008). TFBS identification based on genetic algorithm with combined representations and adaptive post-processing. *Bioinformatics*, *24*(3), 341–349. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18065426 doi: 10.1093/bioinformatics/btm606

Chang, C. C., & Lin, C. J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, *2*(3), 1–27. Retrieved from http://dl.acm.org/citation.cfm?doid=1961189.1961199 doi: 10.1145/1961189.1961199

Chawla, N., Eschrich, S., & Hall, L. (2001). Creating ensembles of classifiers. In *Proceedings 2001 IEEE International Conference on Data Mining* (pp. 580–581). IEEE Computer Society. Retrieved from http://ieeexplore.ieee.org/document/989568/ doi: 10.1109/ICDM.2001.989568

Che, D., Jensen, S., Cai, L., & Liu, J. S. (2005). BEST: Binding-site Estimation Suite of Tools. *Bioinformatics*, *21*(12), 2909–2911. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bti425 doi: 10.1093/

bioinformatics/bti425

Chen, C. Y., Tsai, H. K., Hsu, C. M., May Chen, M. J., Hung, H. G., Huang, G. T. W., & Li, W. H. (2008). Discovering gapped binding sites of yeast transcription factors. In *Proceedings of the National Academy of Sciences of the United States of America* (Vol. 105, pp. 2527–2532). Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2268170 doi: 10.1073/pnas.0712188105

Chen, X., Guo, L., Fan, Z., & Jiang, T. (2008). W-AlignACE: an improved Gibbs sampling algorithm based on more accurate position weight matrices learned from sequence and gene expression/ChIP-chip data. *Bioinformatics*, *24*(9), 1121–1128. Retrieved from http://bioinformatics.oxfordjournals.org/content/24/9/1121.long doi: 10.1093/bioinformatics/btn088

Chen, X., Xu, H., Yuan, P., Fang, F., Huss, M., Vega, V. B., … Ng, H. H. (2008). Integration of external signaling pathways with the core transcriptional network in embryonic stem cells. *Cell*, *133*(6), 1106–1117. Retrieved from http://www.sciencedirect.com/science/article/pii/S009286740800617X doi: 10.1016/j.cell.2008.04.043

Choong, A. C. H., & Lee, N. K. (2017). Evaluation of convolutionary neural networks modeling of DNA sequences using ordinal versus one-hot encoding method. *Computer and Drone Applications (IConDA), 2017 International Conference*, 60–65. Retrieved from https://www.biorxiv.org/content/early/2017/09/10/186965 doi: 10.1101/186965

Chou, K. (2001). Prediction of protein cellular attributes using pseudo-amino acid composition. *Proteins: Structure, Function, and Bioinformatics*, *43*(3), 246–255. Retrieved from http://onlinelibrary.wiley.com/doi/10.1002/prot.1035/full doi: 10.1002/PROT.1035

Consortium, E. P. (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature*, *489*(7414), 57–74. Retrieved from http://www.nature.com/nature/journal/v489/n7414/abs/nature11247.html

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. Retrieved from https://doi.org/10.1007/BF00994018 doi: 10.1007/BF00994018

Das, M. K., & Dai, H. K. (2007). A survey of DNA motif finding algorithms. *BMC Bioinformatics*, *8 Suppl 7*, S21. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2099490 doi: 10.1186/1471-2105-8-S7-S21

Defrance, M., Janky, R., Sand, O., & van Helden, J. (2008). Using RSAT oligo-analysis and dyad-analysis tools to discover regulatory signals in nucleic sequences. *Nature Protocols*, *3*(10), 1589–1603. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18802440 doi: 10.1038/nprot.2008.98

D'haeseleer, P. (2006). How does DNA sequence motif discovery work? *Nature Biotechnology*, *24*(8), 959–961. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/16900144 doi: 10.1038/nbt0806-959

Edgar, R. C. (2004). MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, *32*(5), 1792–1797. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/15034147http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC390337https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkh340 doi: 10.1093/nar/gkh340

Erwin, G. D., Oksenberg, N., Truty, R. M., Kostka, D., Murphy, K. K., Ahituv, N., … Capra, J. A. (2014). Integrating diverse datasets improves developmental enhancer prediction. *PLoS Computational Biology*, *10*(6), e1003677. Retrieved from

http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003677 doi: 10.1371/journal.pcbi.1003677

Eser, U., & Churchman, L. S. (2016). FIDDLE: An integrative deep learning framework for functional genomic data inference. *bioRxiv*. Retrieved from http://biorxiv.org/content/early/2016/10/17/081380

Eskin, E., & Pevzner, P. A. (2002). Finding composite regulatory patterns in DNA sequences. *Bioinformatics*, *18*(Suppl 1), S354–S363. Retrieved from http://bioinformatics.oxfordjournals.org/content/18/suppl_1/S354.short doi: 10.1093/bioinformatics/18.suppl_1.S354

Ettwiller, L., Paten, B., Ramialison, M., Birney, E., & Wittbrodt, J. (2007). Trawler: de novo regulatory motif discovery pipeline for chromatin immunoprecipitation. *Nature Methods*, *4*(7), 563–565. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/17589518 doi: 10.1038/nmeth1061

Fawcett, T. (2006). *An introduction to ROC analysis* (Vol. 27) (No. 8). Retrieved from http://www.sciencedirect.com/science/article/pii/S016786550500303X doi: 10.1016/j.patrec.2005.10.010

Firpi, H. A., Ucar, D., & Tan, K. (2010). Discover regulatory DNA elements using chromatin signatures and artificial neural network. *Bioinformatics*, *26*(13), 1579–1586. Retrieved from http://bioinformatics.oxfordjournals.org/content/26/13/1579.full doi: 10.1093/bioinformatics/btq248

Friberg, M., von Rohr, P., & Gonnet, G. (2005). Scoring functions for transcription factor binding site prediction. *BMC Bioinformatics*, *6*, 84. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1140076 doi: 10.1186/1471-2105-6-84

Frith, M. C., Saunders, N. F. W., Kobe, B., & Bailey, T. L. (2008). Discovering sequence motifs with arbitrary insertions and deletions. *PLoS Computational Biology*, *4*(4), e1000071. Retrieved from http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000071 doi: 10.1371/journal.pcbi.1000071

Furey, T. S., Cristianini, N., Duffy, N., Bednarski, D. W., Schummer, M., & Haussler, D. (2000). Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, *16*(10), 906–914. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/16.10.906 doi: 10.1093/bioinformatics/16.10.906

Gao, Z., Liu, L., & Ruan, J. (2017). Logo2PWM: a tool to convert sequence logo to position weight matrix. *BMC Genomics*, *18*(S6), 709. Retrieved from http://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-017-4023-9 doi: 10.1186/s12864-017-4023-9

Gelfand, M. S., Koonin, E. V., & Mironov, A. A. (2000). Prediction of transcription regulatory sites in Archaea by a comparative genomic approach. *Nucleic Acids Research*, *28*(3), 695–705. Retrieved from https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/28.3.695 doi: 10.1093/nar/28.3.695

Ghandi, M., Lee, D., Mohammad-Noori, M., & Beer, M. A. (2014). Enhanced regulatory sequence prediction using gapped k-mer features. *PLoS Computational Biology*, *10*(7), e1003711. Retrieved from http://dx.plos.org/10.1371/journal.pcbi.1003711 doi: 10.1371/journal.pcbi.1003711

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. *Reading: Addison-Wesley*.

Grant, C. E., Bailey, T. L., & Noble, W. S. (2011). FIMO: scanning for occurrences of a

given motif. *Bioinformatics*, *27*(7), 1017–1018. Retrieved from http://bioinformatics.oxfordjournals.org/content/27/7/1017.long doi: 10.1093/bioinformatics/btr064

Gribskov, M., McLachlan, A. D., & Eisenberg, D. (1987). Profile analysis: detection of distantly related proteins. In *Proceedings of the National Academy of Sciences of the United States of America* (Vol. 84, pp. 4355–4358). doi: 10.1073/pnas.84.13.4355

GuhaThakurta, D., & Stormo, G. D. (2001). Identifying target sites for cooperatively binding factors. *Bioinformatics*, *17*(7), 608–621. Retrieved from http://bioinformatics.oxfordjournals.org/content/17/7/608.short doi: 10.1093/bioinformatics/17.7.608

Gupta, S., Stamatoyannopoulos, J. A., Bailey, T. L., & Noble, W. S. (2007). Quantifying similarity between motifs. *Genome Biology*, *8*(2), R24. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1852410 doi: 10.1186/gb-2007-8-2-r24

Harbison, C. T., Gordon, D. B., Lee, T. I., Rinaldi, N. J., Macisaac, K. D., Danford, T. W., … Young, R. A. (2004). Transcriptional regulatory code of a eukaryotic genome. *Nature*, *431*(7004), 99–104. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/15343339 doi: 10.1038/nature02800

Hashim, F. A., Mabrouk, M. S., & Al-Atabany, W. (2019). Review of different sequence motif finding algorithms. *Avicenna Journal of Medical Biotechnology*, *11*(2), 130–148. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/31057715

Haudry, Y., Ramialison, M., Paten, B., Wittbrodt, J., & Ettwiller, L. (2010). Using Trawler_standalone to discover overrepresented motifs in DNA and RNA sequences derived from various experiments including chromatin immunoprecipitation. *Nature Protocols*, *5*(2), 323–334. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/20134431 doi: 10.1038/nprot.2009.158

Hertz, G. Z., & Stormo, G. D. (1999). Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, *15*(7), 563–577. Retrieved from http://bioinformatics.oxfordjournals.org/content/15/7/563 doi: 10.1093/bioinformatics/15.7.563

Hirose, S., Shimizu, K., Kanai, S., Kuroda, Y., & Noguchi, T. (2007). POODLE-L: a two-level SVM prediction system for reliably predicting long disordered regions. *Bioinformatics*, *23*(16), 2046–2053. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btm302 doi: 10.1093/bioinformatics/btm302

Hon, L. S., & Jain, A. N. (2006). A deterministic motif finding algorithm with application to the human genome. *Bioinformatics*, *22*(9), 1047–1054. Retrieved from http://bioinformatics.oxfordjournals.org/content/22/9/1047 doi: 10.1093/bioinformatics/btl037

Hu, J., Li, B., & Kihara, D. (2005). Limitations and potentials of current motif discovery algorithms. *Nucleic Acids Research*, *33*(15), 4899–4913.

Hu, J., Yang, Y. D., & Kihara, D. (2006). EMD: an ensemble algorithm for discovering regulatory motifs in DNA sequences. *BMC Bioinformatics*, *7*(1), 342. Retrieved from http://www.biomedcentral.com/1471-2105/7/342/abstract doi: 10.1186/1471-2105-7-342

Hughes, J. D., Estep, P. W., Tavazoie, S., & Church, G. M. (2000). Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. *Journal of Molecular Biology*, *296*(5), 1205–1214. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/10698627 doi: 10.1006/jmbi.2000.3519

Ichinose, N., Yada, T., & Gotoh, O. (2012). Large-scale motif discovery using DNA gray code and equiprobable oligomers. *Bioinformatics*, *28*(1), 25–31. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3244767 doi: 10.1093/bioinformatics/btr606

Jensen, S. T., & Liu, J. S. (2004). BioOptimizer: a Bayesian scoring function approach to motif discovery. *Bioinformatics*, *20*(10), 1557–1564. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bth127 doi: 10.1093/bioinformatics/bth127

Ji, H., & Wong, W. (2006). Computational biology: toward deciphering gene regulatory information in mammalian genomes. *Biometrics*, *62*(3), 645–663.

Jia, C., & He, W. (2016). EnhancerPred: a predictor for discovering enhancers based on the combination and selection of multiple features. *Scientific Reports*, *6*(1), 38741. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/27941893 doi: 10.1038/srep38741

Jin, V. X., Apostolos, J., Nagisetty, N. S. V. R., & Farnham, P. J. (2009). W-ChIPMotifs: a web application tool for de novo motif discovery from ChIP-based high-throughput data. *Bioinformatics*, *25*(23), 3191–3193. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2778340 doi: 10.1093/bioinformatics/btp570

Jin, V. X., O'Geen, H., Iyengar, S., Green, R., & Farnham, P. J. (2007). Identification of an OCT4 and SRY regulatory module using integrated computational and experimental genomics approaches. *Genome Research*, *17*(6), 807–817. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1891340/ doi: 10.1101/gr.6006107

Johnson, D. S., Mortazavi, A., Myers, R. M., & Wold, B. (2007). Genome-wide mapping of in vivo protein-DNA interactions. *Science*, *316*(5830), 1497–1502. Retrieved from

http://www.ncbi.nlm.nih.gov/pubmed/17540862 doi: 10.1126/science.1141319

Jothi, R., Cuddapah, S., Barski, A., Cui, K., & Zhao, K. (2008). Genome-wide identification of in vivo protein–DNA binding sites from ChIP-Seq data. *Nucleic Acids Research*, *36*(16), 5221–5231. Retrieved from http://nar.oxfordjournals.org/content/36/16/5221 doi: 10.1093/nar/gkn488

Kel, A., Gößling, E., Reuter, I., Cheremushkin, E., Kel-Margoulis, O. V., & Wingender, E. (2003). MATCH: a tool for searching transcription factor binding sites in DNA sequences. *Nucleic Acids Research*, *31*(13), 3576–3579. Retrieved from http://nar.oxfordjournals.org/content/31/13/3576.full doi: 10.1093/nar/gkg585

Kelley, D. R., Snoek, J., & Rinn, J. (2015). *Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks.* (Tech. Rep.). Retrieved from http://biorxiv.org/content/early/2016/02/18/028399.abstract doi: 10.1101/028399

Kellis, M., Patterson, N., Birren, B., Berger, B., & Lander, E. S. (2004). Methods in comparative genomics: genome correspondence, gene identification and regulatory motif discovery. *Journal of Computational Biology*, *11*(2-3), 319–355. Retrieved from http://online.liebertpub.com/doi/abs/10.1089/1066527041410319 doi: 10.1089/1066527041410319

Kibet, C. K., & Machanick, P. (2016). Transcription factor motif quality assessment requires systematic comparative analysis. *F1000Research*, *4*. doi: 10.12688/f1000research.7408.2

Kim, T. H., Abdullaev, Z. K., Smith, A. D., Ching, K. A., Loukinov, D. I., Green, R. D., … Ren, B. (2007). Analysis of the vertebrate insulator protein CTCF-binding sites in the human genome. *Cell*, *128*(6), 1231–1245. Retrieved from http://www.sciencedirect

.com/science/article/pii/S009286740700205X doi: 10.1016/j.cell.2006.12.048

Kleftogiannis, D., Kalnis, P., & Bajic, V. B. (2015). DEEP: a general computational framework for predicting enhancers. *Nucleic Acids Research*, *43*(1), e6. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/25378307 doi: 10.1093/nar/gku1058

Klepper, K., & Drabløs, F. (2013). MotifLab: a tools and data integration workbench for motif discovery and regulatory sequence analysis. *BMC Bioinformatics*, *14*(1). Retrieved from https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-14-9

Klepper, K., Sandve, G. K., Abul, O., Johansen, J., & Drablos, F. (2008). Assessment of composite motif discovery methods. *BMC Bioinformatics*, *9*, 123. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2311304 doi: 10.1186/1471-2105-9-123

Kloft, M., Brefeld, U., Sonnenburg, S., & Zien, A. (2011). lp-norm multiple kernel learning. *Journal of Machine Learning Research*, *12*(Mar), 953–997. Retrieved from http://www.jmlr.org/papers/v12/kloft11a.html

Kohonen, T. (1998). The self-organizing map. *Neurocomputing*, *21*(1-3), 1–6. Retrieved from https://www.sciencedirect.com/science/article/pii/S0925231298000307 doi: 10.1016/S0925-2312(98)00030-7

Kuksa, P. P., & Pavlovic, V. (2010). Efficient motif finding algorithms for large-alphabet inputs. *BMC Bioinformatics*, *11 Suppl 8*(Suppl 8), S1. Retrieved from http://www.biomedcentral.com/1471-2105/11/S8/S1 doi: 10.1186/1471-2105-11-S8-S1

Kulakovskiy, I. V., Boeva, V. A., Favorov, A. V., & Makeev, V. J. (2010). Deep and wide digging for binding motifs in ChIP-Seq data. *Bioinformatics*, *26*(20), 2622–2623. doi: 10.1093/bioinformatics/btq488

Kuttippurathu, L., Hsing, M., Liu, Y., Schmidt, B., Maskell, D., Lee, K., … Kong, S. (2011). CompleteMOTIFs: DNA motif discovery platform for transcription factor binding experiments. *Bioinformatics*, *27*(5), 715–717. Retrieved from https://academic.oup.com/bioinformatics/article/27/5/715/263043

Lanchantin, J., Singh, R., Wang, B., & Qi, Y. (2016). Deep motif dashboard: visualizing and understanding genomic sequences using deep neural networks. *CoRR*. Retrieved from http://arxiv.org/abs/1608.03644

Lawrence, C., Altschul, S., Boguski, M., Liu, J., Neuwald, A., & Wootton, J. (1993). Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, *262*(5131), 208–214. Retrieved from http://www.sciencemag.org/content/262/5131/208.short doi: 10.1126/science.8211139

Lawrence, C. E., & Reilly, A. A. (1990). An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins: Structure, Function, and Genetics*, *7*(1), 41–51. Retrieved from http://doi.wiley.com/10.1002/prot.340070105 doi: 10.1002/prot.340070105

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. Retrieved from https://www.nature.com/nature/journal/v521/n7553/abs/nature14539.html

Lee, D. (2016). LS-GKM: a new gkm-SVM for large-scale datasets. *Bioinformatics*, *32*(14), 2196–2198. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btw142 doi: 10.1093/bioinformatics/btw142

Lee, D., Gorkin, D. U., Baker, M., Strober, B. J., Asoni, A. L., McCallion, A. S., & Beer, M. A. (2015). A method to predict the impact of regulatory variants from DNA sequence. *Nature Genetics*, *47*(8), 955–61. Retrieved

from http://www.ncbi.nlm.nih.gov/pubmed/26075791http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4520745 doi: 10.1038/ng.3331

Lee, D., Karchin, R., & Beer, M. A. (2011). Discriminative prediction of mammalian enhancers from DNA sequence. *Genome Research*, *21*(12), 2167–2180. Retrieved from http://genome.cshlp.org/content/21/12/2167.short doi: 10.1101/gr.121905.111

Lee, N. K., & Choong, A. C. H. (2013). Filtering of background DNA sequences improves DNA motif prediction using clustering techniques. *Procedia - Social and Behavioral Sciences*, *97*, 602–611. Retrieved from http://www.sciencedirect.com/science/article/pii/S1877042813037245 doi: 10.1016/j.sbspro.2013.10.279

Lee, N. K., Choong, A. C. H., & Omar, N. (2016). ENSPART: An ensemble framework based on data partitioning for DNA motif analysis. *2016 IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE)*, 87–94. Retrieved from http://ieeexplore.ieee.org/document/7789964/ doi: 10.1109/BIBE.2016.68

Lee, N. K., Li, X., & Wang, D. (2018). A comprehensive survey on genetic algorithms for DNA motif prediction. *Information Sciences*, *466*, 25–43. Retrieved from https://www.sciencedirect.com/science/article/pii/S0020025518305206 doi: 10.1016/J.INS.2018.07.004

Lee, N. K., & Oon, Y. B. (2013). Potential perils of biological sequence visualization using sequence logo. In *2013 10th International Conference Computer Graphics, Imaging and Visualization* (pp. 106–111). IEEE. Retrieved from http://ieeexplore.ieee.org/document/6658172/ doi: 10.1109/CGIV.2013.26

Lee, N. K., & Wang, D. (2011). SOMEA: self-organizing map based extraction algorithm for DNA motif identification with heterogeneous model. *BMC Bioinformatics*, *12 Suppl 1*(Suppl 1), S16. Retrieved from http://www.biomedcentral.com/1471-2105/12/

S1/S16 doi: 10.1186/1471-2105-12-S1-S16

Leslie, C. S., Eskin, E., & Noble, W. S. (2002). The spectrum kernel: A string kernel for SVM protein classification. In *Pacific Symposium on Biocomputing* (Vol. 7, pp. 566–575).

Levitsky, V. G., Ignatieva, E. V., Ananko, E. A., Turnaev, I. I., Merkulova, T. I., Kolchanov, N. A., & Hodgman, T. C. (2007). Effective transcription factor binding site prediction using a combination of optimization, a genetic algorithm and discriminant analysis to capture distant interactions. *BMC Bioinformatics*, *8*(1), 481. Retrieved from http://www.biomedcentral.com/1471-2105/8/481 doi: 10.1186/1471-2105-8-481

Li, H., Rhodius, V., Gross, C., & Siggia, E. D. (2002). Identification of the binding sites of regulatory proteins in bacterial genomes. *Proceedings of the National Academy of Sciences of the United States of America*, *99*(18), 11772–11777. Retrieved from http://www.pnas.org/content/99/18/11772.short doi: 10.1073/pnas.112341999

Li, L. (2009). GADEM: a genetic algorithm guided formation of spaced dyads coupled with an EM algorithm for motif discovery. *Journal of Computational Biology*, *16*(2), 317–329.

Li, L., Liang, Y., & Bass, R. L. (2007). GAPWM: a genetic algorithm method for optimizing a position weight matrix. *Bioinformatics*, *23*(10), 1188–1194. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/17341493 doi: 10.1093/bioinformatics/btm080

Li, T., Zhang, C., & Zhu, S. (2006). Empirical Studies on Multi-label Classification. In *2006 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06)* (pp. 86–92). Retrieved from https://ieeexplore.ieee.org/abstract/document/4031884 doi: 10.1109/ICTAI.2006.55

Lihu, A., & Holban, S. (2015). A review of ensemble methods for de novo motif discovery

in ChIP-Seq data. *Briefings in Bioinformatics*, *16*(6), 964–973. Retrieved from https://academic.oup.com/bib/article-lookup/doi/10.1093/bib/bbv022 doi: 10.1093/bib/bbv022

Linhart, C., Halperin, Y., & Shamir, R. (2008). Transcription factor and microRNA motif discovery: the Amadeus platform and a compendium of metazoan target sets. *Genome Research*, *18*(7), 1180–1189.

Liu, B., Fang, L., Long, R., Lan, X., & Chou, K.-C. (2016). iEnhancer-2L: a two-layer predictor for identifying enhancers and their strength by pseudo k-tuple nucleotide composition. *Bioinformatics*, *32*(3), 362–369. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btv604 doi: 10.1093/bioinformatics/btv604

Liu, F. F. M., Tsai, J. J. P., Chen, R. M., Chen, S. N., & Shih, S. H. (2004). FMGA: Finding motifs by genetic algorithm. In *Proceedings - Fourth IEEE Symposium on Bioinformatics and Bioengineering, BIBE 2004* (pp. 459–466). doi: 10.1109/BIBE.2004.1317378

Liu, X., Brutlag, D., & Liu, J. (2001). BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pacific Symposium on Biocomputing*, *138*, 127–138. Retrieved from http://psb.stanford.edu/psb-online/proceedings/psb01/liu.pdf

Liu, X. S., Brutlag, D. L., & Liu, J. S. (2002). An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments. *Nature Biotechnology*, *20*(8), 835–839. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/12101404 doi: 10.1038/nbt717

Lomvardas, S., Barnea, G., Pisapia, D. J., Mendelsohn, M., Kirkland, J., & Axel,

R. (2006). Interchromosomal interactions and olfactory receptor choice. *Cell*, *126*(2), 403–413. Retrieved from http://www.sciencedirect.com/science/article/pii/S0092867406008555 doi: 10.1016/J.CELL.2006.06.035

Ma, X., Kulkarni, A., Zhang, Z., Xuan, Z., Serfling, R., & Zhang, M. Q. (2012). A highly efficient and effective motif discovery method for ChIP-seq/ChIP-chip data using positional information. *Nucleic Acids Research*, *40*(7), e50. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3326300 doi: 10.1093/nar/gkr1135

Machanick, P., & Bailey, T. L. (2011). MEME-ChIP: motif analysis of large DNA datasets. *Bioinformatics*, *27*(12), 1696–1697. Retrieved from http://bioinformatics.oxfordjournals.org/content/27/12/1696.full doi: 10.1093/bioinformatics/btr189

MacIsaac, K. D., Gordon, D. B., Nekludova, L., Odom, D. T., Schreiber, J., Gifford, D. K., … Fraenkel, E. (2006). A hypothesis-based approach for identifying the binding specificity of regulatory proteins from chromatin immunoprecipitation data. *Bioinformatics*, *22*(4), 423–429. Retrieved from https://academic.oup.com/bioinformatics/article/22/4/423/183598 doi: 10.1093/bioinformatics/bti815

Mahony, S., Benos, P., Smith, T., & Golden, A. (2006). Self-organizing neural networks to support the discovery of DNA-binding motifs. *Neural Networks*, *19*(6-7), 950–962.

Mahony, S., & Benos, P. V. (2007). STAMP: a web tool for exploring DNA-binding motif similarities. *Nucleic Acids Research*, *35*(Web Server issue), W253–W258. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1933206 doi: 10.1093/nar/gkm272

Marinari, E., & Parisi, G. (1992). Simulated tempering: a new monte carlo scheme. *Europhysics Letters (EPL)*, *19*(6), 451–458. Retrieved from http://stacks.iop.org/

0295-5075/19/i=6/a=002 doi: 10.1209/0295-5075/19/6/002

Maston, G. a., Evans, S. K., & Green, M. R. (2006). Transcriptional regulatory elements in the human genome. *Annual Review of Genomics and Human Genetics*, *7*(1), 29–59. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/16719718 doi: 10.1146/annurev.genom.7.080505.115623

McLeay, R. C., & Bailey, T. L. (2010). Motif Enrichment Analysis: a unified framework and an evaluation on ChIP data. *BMC Bioinformatics*, *11*(1), 165. Retrieved from https://doi.org/10.1186/1471-2105-11-165 doi: 10.1186/1471-2105-11-165

Min, S., Lee, B., & Yoon, S. (2016). Deep learning in bioinformatics. *Briefings in Bioinformatics*, *18*(5), bbw068. Retrieved from https://academic.oup.com/bib/article-lookup/doi/10.1093/bib/bbw068 doi: 10.1093/bib/bbw068

Nishida, K., Frith, M. C., & Nakai, K. (2009). Pseudocounts for transcription factor binding sites. *Nucleic Acids Research*, *37*(3), 939–944. Retrieved from http://nar.oxfordjournals.org/content/37/3/939 doi: 10.1093/nar/gkn1019

Noonan, J., & McCallion, A. (2010). Genomics of long-range regulatory elements. *Annual Review of Genomics and Human Genetics*, *11*, 1–23.

Opitz, D., & Maclin, R. (1999). Popular ensemble methods: an empirical study. *Journal of Artificial Intelligence Research*, *11*, 169–198. Retrieved from http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.105.506

Orenstein, Y., & Shamir, R. (2014). A comparative analysis of transcription factor binding models learned from PBM, HT-SELEX and ChIP data. *Nucleic Acids Research*, *42*(8), e63–e63. Retrieved from https://academic.oup.com/nar/article/42/8/e63/1067315 doi: 10.1093/nar/gku117

Pavesi, G., Mauri, G., & Pesole, G. (2001). An algorithm for finding

signals of unknown length in DNA sequences. *Bioinformatics*, *17*(Suppl 1), S207–S214. Retrieved from http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/17.suppl_1.S207 doi: 10.1093/bioinformatics/17.suppl_1.S207

Pavesi, G., Mauri, G., & Pesole, G. (2004). In silico representation and discovery of transcription factor binding sites. *Briefings in Bioinformatics*, *5*(3), 217–236. Retrieved from http://bib.oxfordjournals.org/content/5/3/217.short

Pavesi, G., Mereghetti, P., Mauri, G., & Pesole, G. (2004). Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Research*, *32*(suppl 2), W199–W203. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/15215380 doi: 10.1093/nar/gkh465

Pesce, M., & Schöler, H. R. (2001). Oct-4: gatekeeper in the beginnings of mammalian development. *Stem Cells*, *19*(4), 271–278. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/11463946 doi: 10.1634/stemcells.19-4-271

Pevzner, P. A., & Sze, S.-H. (2000). Combinatorial approaches to finding subtle signals in DNA sequences. In *Proceedings of the Eighth Intelligent Systems for Molecular Biology* (pp. 269–278). Retrieved from http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.91.5339

Pisanti, N., Carvalho, A. M., Marsan, L., & Sagot, M.-F. (2006). RISOTTO: Fast extraction of motifs with mismatches. In *LATIN 2006: Theoretical Informatics* (pp. 757–768). Retrieved from http://link.springer.com/chapter/10.1007/11682462_69 doi: 10.1007/11682462_69

Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, *6*(3), 21–45. Retrieved from http://ieeexplore.ieee.org/document/1688199/ doi: 10.1109/MCAS.2006.1688199

Qin, Q., & Feng, J. (2017). Imputation for transcription factor binding predictions based on deep learning. *PLoS Computational Biology*, *13*(2), e1005403. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/28234893 doi: 10.1371/journal.pcbi.1005403

Quinlan, A. R., & Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, *26*(6), 841–842. Retrieved from http://bioinformatics.oxfordjournals.org/content/26/6/841 doi: 10.1093/bioinformatics/btq033

Rigoutsos, I., & Floratos, A. (1998). Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm. *Bioinformatics*, *14*(1), 55–67. Retrieved from http://bioinformatics.oxfordjournals.org/content/14/1/55.short doi: 10.1093/bioinformatics/14.1.55

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, *33*(1-2), 1–39. Retrieved from http://link.springer.com/10.1007/s10462-009-9124-7 doi: 10.1007/s10462-009-9124-7

Rombauts, S., Florquin, K., Lescot, M., & Van de Peer, Y. (2003). Computational approaches to identify promoters and cis-regulatory elements in plant genomes. *Plant Physiology*, *132*(July), 1162–1176. Retrieved from http://www.plantphysiol.org/content/132/3/1162.short doi: 10.1104/pp.102.017715.nomes

Romer, K. a., Kayombya, G.-R., & Fraenkel, E. (2007). WebMOTIFS: automated discovery, filtering and scoring of DNA sequence motifs using multiple programs and Bayesian approaches. *Nucleic Acids Research*, *35*(Web Server issue), W217–W220. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1933171 doi: 10.1093/nar/gkm376

Roth, F. P., Hughes, J. D., Estep, P. W., & Church, G. M. (1998). Finding DNA

regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nature Biotechnology*, *16*(10), 939–945. Retrieved from http://dx.doi.org/10.1038/nbt1098-939 doi: 10.1038/nbt1098-939

Rouchka, E. C., & Hardin, C. T. (2007). rMotifGen: random motif generator for DNA and protein sequences. *BMC Bioinformatics*, *8*, 292. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/17683637 doi: 10.1186/1471-2105-8-292

Salekin, S., Zhang, J. M., & Huang, Y. (2017). A deep learning model for predicting transcription factor binding location at single nucleotide resolution. In *2017 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)* (pp. 57–60). IEEE. Retrieved from http://ieeexplore.ieee.org/document/7897204/ doi: 10.1109/BHI.2017.7897204

Sandelin, A., Alkema, W., Engström, P., Wasserman, W. W., & Lenhard, B. (2004). JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*, *32*(90001), 91D–94D. Retrieved from http://nar.oxfordjournals.org/content/32/suppl_1/D91 doi: 10.1093/nar/gkh012

Sandve, G. K., & Drabløs, F. (2006). A survey of motif discovery methods in an integrated framework. *Biology Direct*, *1*, 11. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1479319 doi: 10.1186/1745-6150-1-11

Satya, R. V., & Mukherjee, A. (2004). New algorithms for finding monad patterns in DNA sequences. In A. Apostolico & M. Melucci (Eds.), *String Processing and Information Retrieval* (Vol. 3246, pp. 273–285). Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved from http://www.springerlink.com/index/10.1007/b100941 doi: 10.1007/b100941

Schapire, R. E., & Singer, Y. (2000). BoosTexter: a boosting-based system for text

categorization. *Machine Learning*, *39*(2), 135–168. doi: 10.1023/A:1007649029923

Schneider, T. D. (2002). Consensus Sequence Zen. *Applied Bioinformatics*, *1*(3), 111–119. Retrieved from http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1852464/

Schneider, T. D., & Stephens, R. (1990). Sequence logos: a new way to display consensus sequences. *Nucleic Acids Research*, *18*(20), 6097–6100. Retrieved from https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/18.20.6097 doi: 10.1093/nar/18.20.6097

Shi, J., Yang, W., Chen, M., Du, Y., Zhang, J., & Wang, K. (2011). AMD, an Automated Motif Discovery Tool Using Stepwise Refinement of Gapped Consensuses. *PLoS ONE*, *6*(9), e24576.

Shida, K. (2006). GibbsST: a Gibbs sampling method for motif discovery with enhanced resistance to local optima. *BMC Bioinformatics*, *7*(1), 486. Retrieved from http://www.biomedcentral.com/1471-2105/7/486 doi: 10.1186/1471-2105-7-486

Shlyueva, D., Stampfel, G., & Stark, A. (2014). Transcriptional enhancers: from properties to genome-wide predictions. *Nature Reviews Genetics*, *15*(4), 272–286. Retrieved from http://dx.doi.org/10.1038/nrg3682 doi: 10.1038/nrg3682

Sinha, S. (2006). On counting position weight matrix matches in a sequence, with application to discriminative motif finding. *Bioinformatics*, *22*(14), e454–e463. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/16873507 doi: 10.1093/bioinformatics/btl227

Sinha, S., & Tompa, M. (2000). A Statistical Method for Finding Transcription Factor Binding Sites. In *Proceedings of the Eighth Intelligent Systems for Molecular Biology.* Retrieved from www.aaai.org

Sinha, S., & Tompa, M. (2003). YMF: a program for discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Research*, *31*(13),

3586–3588. Retrieved from http://nar.oxfordjournals.org/content/31/13/3586.short doi: 10.1093/nar/gkg618

Smith, T., & Waterman, M. (1981). Identification of common molecular subsequences. *Journal of Molecular Biology*, *147*(1), 195–197. Retrieved from http://www.sciencedirect.com/science/article/pii/0022283681900875 doi: 10.1016/0022-2836(81)90087-5

Smolinska, K., & Pacholczyk, M. (2017). EMQIT: a machine learning approach for energy based PWM matrix quality improvement. *Biology Direct*, *12*(1), 17. Retrieved from http://biologydirect.biomedcentral.com/articles/10.1186/s13062-017-0189-y doi: 10.1186/s13062-017-0189-y

Sonego, P., Kocsor, A., & Pongor, S. (2008). ROC analysis: applications to the classification of biological sequences and 3D structures. *Briefings in Bioinformatics*, *9*(3), 198–209. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18192302 doi: 10.1093/bib/bbm064

Stormo, G. D. (2000). DNA binding sites: representation and discovery. *Bioinformatics*, *16*(1), 16–23. Retrieved from http://bioinformatics.oxfordjournals.org/content/16/1/16 doi: 10.1093/bioinformatics/16.1.16

Su, J., Teichmann, S. A., & Down, T. A. (2010). Assessing computational methods of cis-regulatory module prediction. *PLoS Computational Biology*, *6*(12), e1001020. Retrieved from http://dx.doi.org/10.1371/journal.pcbi.1001020 doi: 10.1371/journal.pcbi.1001020

Sumazin, P., Chen, G., Hata, N., Smith, A. D., Zhang, T., & Zhang, M. Q. (2005). DWE: discriminating word enumerator. *Bioinformatics*, *21*(1), 31–8. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/15333453 doi: 10.1093/bioinformatics/bth471

Takusagawa, K. T., & Gifford, D. K. (2004). Negative information for motif discovery. *Pacific Symposium on Biocomputing*, 360–371. doi: 10.1142/9789812704856_0034

Thijs, G., Lescot, M., Marchal, K., Rombauts, S., de Moor, B., Rouze, P., & Moreau, Y. (2001). A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, *17*(12), 1113.

Thomas-Chollier, M., Herrmann, C., Defrance, M., Sand, O., Thieffry, D., & van Helden, J. (2012). RSAT peak-motifs: motif analysis in full-size ChIP-seq datasets. *Nucleic Acids Research*, *40*(4), e31–e31. Retrieved from https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkr1104 doi: 10.1093/nar/gkr1104

Thomas-Chollier, M., Sand, O., Turatsinze, J.-V., Janky, R., Defrance, M., Vervisch, E., … van Helden, J. (2008). RSAT: regulatory sequence analysis tools. *Nucleic Acids Research*, *36*(Web Server issue), W119–W127. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18495751 doi: 10.1093/nar/gkn304

Tompa, M., Li, N., Bailey, T. L., Church, G. M., De Moor, B., Eskin, E., … Zhu, Z. (2005). Assessing computational tools for the discovery of transcription factor binding sites. *Nature Biotechnology*, *23*(1), 137–144. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/15637633 doi: 10.1038/nbt1053

Tran, N. T. L., & Huang, C.-H. (2014). A survey of motif finding web tools for detecting binding site motifs in ChIP-Seq data. *Biology Direct*, *9*(1), 4. Retrieved from http://biologydirect.biomedcentral.com/articles/10.1186/1745-6150-9-4 doi: 10.1186/1745-6150-9-4

Tuteja, G., White, P., Schug, J., & Kaestner, K. H. (2009). Extracting transcription factor targets from ChIP-Seq data. *Nucleic Acids Research*, *37*(17), e113–e113. Retrieved from http://nar.oxfordjournals.org/content/37/17/e113 doi: 10.1093/nar/gkp536

Valen, E., Sandelin, A., Winther, O., & Krogh, A. (2009). Discovery of regulatory elements is improved by a discriminatory approach. *PLoS Computational Biology*, *5*(11), e1000562. Retrieved from http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1000562 doi: 10.1371/journal.pcbi.1000562

Valouev, A., Johnson, D. S., Sundquist, A., Medina, C., Anton, E., Batzoglou, S., … Sidow, A. (2008). Genome-wide analysis of transcription factor binding sites based on ChIP-Seq data. *Nature Methods*, *5*(9), 829–834. Retrieved from http://www.nature.com/articles/nmeth.1246 doi: 10.1038/nmeth.1246

van Heeringen, S. J., & Veenstra, G. J. C. (2011). GimmeMotifs: a de novo motif prediction pipeline for ChIP-sequencing experiments. *Bioinformatics*, *27*(2), 270–271. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3018809 doi: 10.1093/bioinformatics/btq636

van Helden, J., Rios, A., & Collado-Vides, J. (2000). Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic Acids Research*, *28*(8), 1808–1818.

Visel, A., Minovitsky, S., Dubchak, I., & Pennacchio, L. A. (2007). VISTA Enhancer Browser–a database of tissue-specific human enhancers. *Nucleic Acids Research*, *35*(Database), D88–D92. Retrieved from https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkl822 doi: 10.1093/nar/gkl822

Wang, C., Zhang, M. Q., & Zhang, Z. (2013). Computational identification of active enhancers in model organisms. *Genomics, Proteomics & Bioinformatics*, *11*(3), 142–150. Retrieved from http://www.sciencedirect.com/science/article/pii/S1672022913000478 doi: 10.1016/j.gpb.2013.04.002

Wang, D., & Do, H. T. (2012). Computational localization of transcription factor

binding sites using extreme learning machines. *Soft Computing*, *16*(9), 1595–1606. Retrieved from http://link.springer.com/article/10.1007/s00500-012-0820-x doi: 10 .1007/s00500-012-0820-x

Wang, D., & Lee, N. K. (2009). MISCORE: Mismatch-based matrix similarity scores for dna motif detection. In *International Conference on Neural Information Processing* (pp. 478–485). Springer. doi: 10.1007/978-3-642-02490-0_59

Wasserman, W., & Sandelin, A. (2004). Applied bioinformatics for the identification of regulatory elements. *Nature Reviews Genetics*, *5*(4), 276–287.

Wei, C. L., Wu, Q., Vega, V. B., Chiu, K. P., Ng, P., Zhang, T., … Ruan, Y. (2006). A global map of p53 transcription-factor binding sites in the human genome. *Cell*, *124*(1), 207–219. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/16413492 doi: 10 .1016/j.cell.2005.10.043

Wei, Z., & Jensen, S. (2006). GAME: Detecting cis-regulatory elements using a genetic algorithm. *Bioinformatics*, *22*(13), 1577–1584.

Weirauch, M. T., Cote, A., Norel, R., Annala, M., Zhao, Y., Riley, T. R., … Hughes, T. R. (2013). Evaluation of methods for modeling transcription factor sequence specificity. *Nature Biotechnology*, *31*(2), 126–134. Retrieved from http://www.nature .com/articles/nbt.2486 doi: 10.1038/nbt.2486

Whitaker, J. W., Nguyen, T. T., Zhu, Y., Wildberg, A., & Wang, W. (2015). Computational schemes for the prediction and annotation of enhancers from epigenomic assays. *Methods*, *72*, 86–94. Retrieved from http://www.sciencedirect.com/science/article/pii/ S1046202314003259 doi: 10.1016/j.ymeth.2014.10.008

Wijaya, E., Rajaraman, K., Yiu, S. M., & Sung, W. K. (2007). Detection of generic spaced motifs using submotif pattern mining. *Bioinformatics*, *23*(12), 1476–1485.

Retrieved from http://bioinformatics.oxfordjournals.org/content/23/12/1476 doi: 10 .1093/bioinformatics/btm118

Wijaya, E., Yiu, S. M., Son, N. T., Kanagasabai, R., & Sung, W. K. (2008). MotifVoter: a novel ensemble method for fine-grained integration of generic motif finders. *Bioinformatics*, *24*(20), 2288–2295. Retrieved from http://bioinformatics .oxfordjournals.org/content/24/20/2288 doi: 10.1093/bioinformatics/btn420

Wingender, E. (2008). The TRANSFAC project as an example of framework technology that supports the analysis of genomic regulation. *Briefings in Bioinformatics*, *9*(4), 326–332. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/18436575 doi: 10 .1093/bib/bbn016

Won, K. J., Chepelev, I., Ren, B., & Wang, W. (2008). Prediction of regulatory elements in mammalian genomes using chromatin signatures. *BMC Bioinformatics*, *9*(1), 547. Retrieved from https://doi.org/10.1186/1471-2105-9-547 doi: 10.1186/1471-2105-9 -547

Workman, C. T., & Stormo, G. D. (1999). ANN-SPEC: A method for discovering transcription factor binding sites with improved specificit. In *Biocomputing 2000* (Vol. 5, pp. 467–478). World Scientific. Retrieved from http://www.worldscientific .com/doi/abs/10.1142/9789814447331_0044 doi: 10.1142/9789814447331_0044

Xu, M., & Su, Z. (2010). A novel alignment-free method for comparing transcription factor binding site motifs. *PLoS ONE*, *5*(1), e8797. Retrieved from http://dx.doi.org/10.1371/ journal.pone.0008797 doi: 10.1371/journal.pone.0008797

Xu, X., Bieda, M., & Jin, V. (2007). A comprehensive ChIP–chip analysis of E2F1, E2F4, and E2F6 in normal and tumor cells reveals interchangeable roles of E2F family members. *Genome Research*, *17*(11), 1550–1561. Retrieved from http://

genome.cshlp.org/content/17/11/1550.short doi: 10.1101/gr.6783507.porter

Yang, J. H., Li, J. H., Jiang, S., Zhou, H., & Qu, L. H. (2013). ChIPBase: a database for decoding the transcriptional regulation of long non-coding RNA and microRNA genes from ChIP-Seq data. *Nucleic Acids Research*, *41*(D1), D177–D187. Retrieved from http://academic.oup.com/nar/article/41/D1/D177/1054545/ChIPBase-a-database-for-decoding-the doi: 10.1093/nar/gks1060

Yanover, C., Singh, M., & Zaslavsky, E. (2009). M are better than one: an ensemble-based motif finder and its application to regulatory element prediction. *Bioinformatics*, *25*(7), 868–874. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2660878 doi: 10.1093/bioinformatics/btp090

Yao, Z., MacQuarrie, K. L., Fong, A. P., Tapscott, S. J., Ruzzo, W. L., & Gentleman, R. C. (2014). Discriminative motif analysis of high-throughput dataset. *Bioinformatics*, *30*(6), 775–783. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btt615 doi: 10.1093/bioinformatics/btt615

Zambelli, F., & Pavesi, G. (2011). A faster algorithm for motif finding in sequences from ChIP-Seq data. In *International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics* (pp. 201–212). Springer, Berlin, Heidelberg. doi: 10.1007/978-3-642-35686-5_17

Zambelli, F., Pesole, G., & Pavesi, G. (2013). Motif discovery and transcription factor binding sites before and after the next-generation sequencing era. *Briefings in Bioinformatics*, *14*(2), 225–237. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3603212 doi: 10.1093/bib/bbs016

Zambelli, F., Ré, M., & Pavesi, G. (2009). *The Beacon Tools for the analysis of gene expression and its regulation.* Retrieved from http://www.ecti-thailand.org/assets/

papers/29_pub_1.pdf

Zaslavsky, E., & Singh, M. (2006). A combinatorial optimization approach for diverse motif finding applications. *Algorithms for Molecular Biology*, *1*, 13. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1570465&tool=pmcentrez&rendertype=abstract doi: 10.1186/1748-7188-1-13

Zeng, H., Edwards, M. D., Liu, G., & Gifford, D. K. (2016). Convolutional neural network architectures for predicting DNA–protein binding. *Bioinformatics*, *32*(12), i121–i127. Retrieved from http://bioinformatics.oxfordjournals.org/lookup/doi/10.1093/bioinformatics/btw255 doi: 10.1093/bioinformatics/btw255

Zhang, M. L., Peña, J. M., & Robles, V. (2009). Feature selection for multi-label naive Bayes classification. *Information Sciences*, *179*(19), 3218–3229. doi: 10.1016/j.ins.2009.06.010

Zhang, M. L., & Zhou, Z. H. (2007). ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, *40*(7), 2038–2048. doi: 10.1016/j.patcog.2006.12.019

Zhang, X., Odom, D. T., Koo, S.-H., Conkright, M. D., Canettieri, G., Best, J., … Montminy, M. (2005). Genome-wide analysis of cAMP-response element binding protein occupancy, phosphorylation, and target gene activation in human tissues. *Proceedings of the National Academy of Sciences of the United States of America*, *102*(12), 4459–4464. Retrieved from http://www.pnas.org/content/102/12/4459 doi: 10.1073/pnas.0501076102

Zhang, Y., Liu, T., Meyer, C. A., Eeckhoute, J., Johnson, D. S., Bernstein, B. E., … Liu, X. S. (2008). Model-based analysis of ChIP-Seq (MACS). *Genome Biology*, *9*(9), R137. Retrieved from http://genomebiology.com/2008/9/9/R137 doi: 10.1186/gb-2008-9-9-r137

Zhang, Y., Yang, Y., Zhang, H., Jiang, X., Xu, B., Xue, Y., … Shi, Q. (2011). Prediction of novel pre-microRNAs with high accuracy through boosting and SVM. *Bioinformatics*, *27*(10), 1436–1437. Retrieved from https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btr148 doi: 10.1093/bioinformatics/btr148

Zhou, J., & Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning-based sequence model. *Nature Methods*, *12*(10), 931–934. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/26301843 doi: 10.1038/nmeth.3547

Zhou, K. R., Liu, S., Sun, W. J., Zheng, L. L., Zhou, H., Yang, J. H., & Qu, L. H. (2017). ChIPBase v2.0: decoding transcriptional regulatory networks of non-coding RNAs and protein-coding genes from ChIP-seq data. *Nucleic Acids Research*, *45*(D1), D43–D50. Retrieved from https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkw965 doi: 10.1093/nar/gkw965

Zhu, L., Zhang, H.-B., & Huang, D.-S. (2017). Direct AUC optimization of regulatory motifs. *Bioinformatics*, *33*(14), i243–i251. Retrieved from http://fdslive.oup.com/www.oup.com/pdf/production_in_progress.pdf doi: 10.1093/bioinformatics/btx255

Zia, A., & Moses, A. M. (2012). Towards a theoretical understanding of false positives in DNA motif finding. *BMC Bioinformatics*, *13*(1), 151. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3436861&tool=pmcentrez&rendertype=abstract doi: 10.1186/1471-2105-13-151

Zielezinski, A., Vinga, S., Almeida, J., & Karlowski, W. M. (2017). Alignment-free sequence comparison: benefits, applications, and tools. *Genome Biology*, *18*(1), 186. Retrieved from http://genomebiology.biomedcentral.com/articles/10.1186/s13059-017-1319-7 doi: 10.1186/s13059-017-1319-7

APPENDICES

Appendix 1: Individual classifiers used by ENSPART

Table A1 shows the invocation of different tools with different parameters used by ENSPART.

Table A1: Command invocation on the partitioned datasets.

Name	Command
MDscan	MDscan -i file -o mdscan1-1/out.txt MDscan -n 12 -w 12 -s 40 -i file -o mdscan2-1/out.txt MDscan -n 16 -w 12 -s 60 -t 10 -i file -o mdscan3-1/out.txt
BioProspector	BioProspector -i file -o biopro1-1/result1.txt -W 12 BioProspector -i file -o biopro2-1/result1.txt -W 12 -g 1 BioProspector -i file -o biopro3-1/result1.txt -W 12 -g 1 -h 1
MEME-ChIP	meme-meme-chip file -oc memechip1/ -meme-nmotifs 5 -meme-maxsize 300000 meme-meme-chip file -oc memechip2/ -meme-minw 10 -seed 2 -meme-maxw 20 -meme-nmotifs 5 -meme-maxsize 300000 meme-meme-chip file -oc memechip3/ -meme-minw 10 -seed 3 -ccut 95 -meme-maxw 20 -meme-nmotifs 5 -meme-maxsize 300000
Weeder 2	weeder2 -f file -O HS -chipseq weeder2 -f file -O HS -chipseq -em 2 weeder2 -f file -O HS -chipseq -sim 0.8
AlignACE	AlignACE -i file -o align1-1/result.txt AlignACE -i file -o align2-1/result.txt -p 0.9 -r 10 AlignACE -i file -o align3-1/result.txt -p 0.7 -r 12
W-AlignACE	W-AlignACE.exe -i file > walignace1-1/result.txt W-AlignACE.exe -i file -minpass 180 > walignace2-1/result.txt W-AlignACE.exe -i file -gcback 0.40 > walignace3-1/result.txt
AMD	AMD -F file -B background_sequence.txt AMD -F file -B background_sequence.txt -CO 0.8 -FC 1.0 AMD -F file -B background_sequence.txt -CO 0.5 -FC 1.4
MotifSampler	MotifSampler -f fasta_file -b inclusive_bg -o output.txt -m output.matrix -n 5 MotifSampler -f fasta_file -b inclusive_bg -o output.txt -m output.matrix -n 5 -w 10 -p 0.4 MotifSampler -f fasta_file -b inclusive_bg -o output.txt -m output.matrix -n 5 -w 12 -p 0.6

Appendix 2: Invocation of rocpwm

The following is the invocation of rocpwm,

```
rocpwm foreground_fasta background_fasta gapwm_output roc_output
```

where the foreground_fasta is the FastA format foreground, which were the datasets collected in FastA format; background_fasta is negative sequences shuffled by using foreground; gapwm_output is the output generated from GAPWM; and roc_output is the target output file which is written in the plain text form.

Appendix 3: GAPWM output

PWM file generated by GAPWM is described as following,

```
4	6
0	7	4	8	9	2
2	4	1	3	2	5
1	2	9	8	6	1
0	8	6	0	4	9
this line is anything, but not read
```

The first line is the dimension of the following matrix. The first line and the matrix are separated by tab. Then after the matrix, there are text which is not used. The PWM is either normalized or a PFM.

Therefore, each motif discovered were converted to the above format. Then, each of them is

used to calculate the ROC. The following is the output format of ROC from rocpwm,

```
0.9609  0.0234  0.0001
0.9607  0.0234  0.0001
0.9605  0.0234  0.0001
0.9603  0.0234  0.0001
0.9601  0.0234  0.0001
...
...
0.0002  1.0000  0.9999
0.0000  1.0000  1.0000
```

There are three columns in the ROC output. The first column is the cut-off value of the ROC. It is started from the highest background score to the lowest background score decreasing linearly. The score value is calculated based on Match (Kel et al., 2003). The second column is the TPR and the third column is the FPR. As a result, using the TPR and FPR, the ROC can be plotted.