A Peak-Finder Meta Server for ChIP-Seq Analysis

Husen Umer
Abstract

A Peak-Finder Meta Server for ChIP-Seq Analysis

Husen Umer

Chromatin immunoprecipitation (ChIP) coupled with ultra high-throughput parallel sequencing (ChIP-seq) is widely used to study transcriptional regulation on a genome-wide scale. Numerous computational tools have been developed to identify transcription factor (protein) binding sites from large ChIP-seq datasets. The diversity of the datasets and the algorithm dependencies make it hard to get a satisfactory result.

Many studies have compared the performance and accuracy of the algorithms using empirical datasets. It is shown that selecting the best algorithm to analyze a ChIP-seq dataset for detecting binding sites of a specific transcription factor depends on the dataset conditions. A systematic solution to compare the results of multiple algorithms to produce the best putative binding sites is still lacking.

In this thesis project a new software package was introduced to provide a single interface for several state-of-the-art algorithms. A voting mechanism and a scoring mechanism were implemented to identify a set of the best predicted transcription factor binding sites (peaks) by normalizing and comparing the predicted peaks of the selected algorithms. The methods were applied on some publicly available datasets and the results were validated by comparing them to the results of the selected algorithms and their corresponding binding motifs. The discovered motifs showed a very high similarity to the consensus motifs of the selected transcription factors.
Acknowledgements

I would like to express my sincerest gratitude towards Prof. Jan Komorowski, Marcin Kruczyk, Susanne Bornelv and the staff at The Linnaeus Centre at Uppsala University for their support. Many thanks for giving me such useful advices and for sharing brilliant ideas and interesting discussions. Thanks Hugh Salter for your valuable reviews. I would also like to acknowledge Claes Wadelius and Ola Wallerman at Rudbeck laboratory for their help in confirming the biological results. My deepest thanks go to my family and friends whom provided me with continuous support and encouragement.
# Contents

1 Background  
1.1 Introduction ................................................. 1  
1.2 Next Generation Sequencing (NGS) ............................ 2  
1.2.1 ChIP-Sequencing ........................................ 2  
1.3 Peak Detection ................................................ 4  
1.4 Relevant Biological Concepts .................................. 4  
1.5 Aims of the Project ........................................... 6  
1.6 Expected Readers .............................................. 6  
1.7 Structure of The Thesis ........................................ 6  

2 Peak Detection Algorithms ...................................... 7  
2.1 Characteristics ................................................ 8  
2.2 The Selected Peak Finders .................................... 9  

3 Methodology ........................................................ 15  
3.1 Functions of the PFMS ......................................... 15  
3.2 Peak Detection .................................................. 16  
3.2.1 Tag Handling ............................................... 16  
3.2.2 Peak Identification ......................................... 16  
3.2.3 Output Format Conversion ................................. 17  
3.3 Peak Selection Methods .......................................... 17  
3.3.1 In-degree Centrality Voting Mechanism .................. 18  
3.3.2 Scoring Mechanism ....................................... 19  

4 Results and Evaluation .......................................... 21  
4.1 Results of The Project ....................................... 21  
4.2 Evaluation of PFMS ........................................... 21  
4.2.1 Analysis Methods ......................................... 22  
4.2.2 Results and Discussion ................................... 23  
4.2.3 Simulation of Physical Genomic Locations ............ 25  
4.2.4 External Validations Using High Scoring Binding Motifs 25
5 Conclusion and Future Work
  5.1 Conclusion ............................................. 29
  5.2 Future Work ........................................... 29

A PFMS User Manual ...................................... 31
List of Abbreviations

PFMS: PFMS Peak Finder Meta Server
NGS: Next Generation Sequencing
ChIP: Chromatin Immunoprecipitation
DNA: Deoxyribonucleic Acid
RNA: Ribonucleic Acid
FDR: False Discovery Rate
TF: Transcription Factor
TFBS: Transcription Factor Binding Site
HMM: Hidden Markov Model
WIG: Wiggle Data Format
BED: BED Data Format
MACS: Model-based analysis of ChIP-Seq
SISSRs: Site Identification from Short Sequence Reads
STAT1: Signal Transducers and Activators of Transcription
NRSF: Neuron-Restrictive Silence Factor
GABP: Growth Associated Binding Protein
Chapter 1

Background

1.1 Introduction

The development of new technologies is revolutionizing genome-wide analysis and scientists’ abilities to have a better understanding of the biological meanings behind the long DNA sequences. In contrast, demand for analyzing very large datasets is ever increasing, especially with the introduction of ChIP-sequencing which is a recent method of Next Generation Sequencing (NGS) used to analyze protein interactions with DNA.

ChIP-seq combines chromatin immunoprecipitation (ChIP) with massively parallel DNA sequencing to identify the cistrome of DNA-associated proteins; i.e the DNA binding sites of a transcription factor. The sites are usually represented in the form of peaks. Large quantities of data generated from ChIP-seq experiments require effective computational analysis for uncovering biological mechanisms. To fulfill this need numerous algorithms have been designed. Meanwhile, the abundance of these algorithms and technical variability on which they identify protein binding sites make an assessment of the current methods timely \cite{1} and hard to define the best algorithm for a specific ChIP-seq dataset. Many studies have been conducted to evaluate performance of these algorithms by measuring their sensitivity, accuracy and usability \cite{1} or by emphasizing their biological conclusions \cite{2}. Another study also surveys current peak detection algorithms and reviews common methods for verification of peak-detection results \cite{3}. Unfortunately a systematic solution to identifying significant peaks from a dataset without depending on the dataset conditions and algorithm variations is still lacking. To solve this problem, we introduce a software framework called Peak Finder MetaServer (PFMS) to identify, potentially, the best peaks by performing a systematic comparison of the results of multiple algorithms according to user criteria.
1.2 Next Generation Sequencing (NGS)

In the past few years, NGS has markedly accelerated multiple areas of genomics research, enabled experiments that were not previously feasible or affordable technically [4]. One of the main applications of NGS is mapping of DNA-binding proteins and chromatin analysis using ChIP-seq techniques.

1.2.1 ChIP-Sequencing

Chromatin immunoprecipitation (ChIP) followed by massively parallel sequencing (ChIP-seq) has become an important approach for genome-wide study of in vitro protein-DNA interactions and gene regulation [5]. It can be used to analyze many important DNA-interacting proteins including RNA polymerases, transcription factors, transcriptional co-factors, and histone proteins [6]. These genome-wide ChIP analysis approaches have led to many important discoveries related to transcriptional regulation, epigenetic regulation through histone modification, nucleosome organization, and interindividual variation in protein-DNA interactions [7].

Work Flow

In this approach, proteins in contact with genomic DNA are chemically cross-linked to their binding sites typically with mild formaldehydone (Figure 1.1). The cells are lysed and the DNA is randomly cut into small fragments by sonication or digestion with micrococcal nuclease. The proteins cross-linked with the DNA are immunoprecipitated with antibodies specific for the proteins of interest; i.e the DNA fragments attached with target proteins are isolated from the rest of the chromatin. Reverse cross-linking, which is breaking the bonds between the proteins and DNA, is followed by purification of the DNA fragments. The contents of the samples are size-selected such that length of the fragments is 200 to 300 base pairs. Adaptors attached to the selected fragments and amplification takes place in majority of the current massively sequencing technologies.

Subsequently, both ends or one end of the fragments in the generated sample are sequenced through high-throughput sequencing. Finally, the obtained short reads are aligned onto a reference genome of interest based on one of the alignment algorithms to generate a genome-wide protein-binding map [4][8]. Practically, the alignment step remains one of the most computationally expensive part of the entire process [9].

The result of the alignment is a ChIP-seq sample that contains the enriched sites of the protein of interest across the genome with the exception of
Some unmapped reads. The mapped sites are called tags and they are used as input to peak detection algorithms. Meanwhile the same process can be repeated without the immunoprecipitation step to obtain a control sample in which both enriched and non-enriched regions are included. This sample can be used as a statistical base for improving peak detection process in the downstream analysis.
1.3 Peak Detection

Peak detection, in ChIP-seq analysis, is the process of analyzing the aligned short reads of sequenced ChIP-seq samples to identify protein binding sites and assign significance to them by comparing the detected peaks with a control sample or a statistical approaches. Usually detecting binding sites is seen as peak detection problem with emphasizing the biological or statistical aspects [10].

The algorithms that are designed to identify peaks (or transcription factor binding sites in this context) are called peak finders, peak detectors, or simply peak detection algorithms. In general, the binding site detection algorithms can be divided into window-based and overlapped-based approaches. The main distinction between the algorithms is the definition of enrichment measures. Another key difference is the methodology of handling control data and counting False Discovery Rate (FDR). All the included algorithms in PFMS are made to work with ChIP-seq alone as well as with control data.

1.4 Relevant Biological Concepts

Deoxyribonucleic Acid (DNA)

DNA is a double-stranded molecule in a helix shape. Each strand of DNA is a linear unbranched polymer in which the monomeric subunits are four chemically distinct nucleotides or bases: adenine (A), guanine (G), cytosine (C) and thymine (T). These nucleotides can be linked together in any order in chains hundreds, thousands or even millions of units in length [11]. The two strands run in opposite direction of each other and they are connected through chemical bonds between the bases in which ‘A’ always pairs with ‘T’ and ‘G’ always pairs with ‘C’. The complete DNA of an organism is called its genome. Furthermore, DNA is organized into long structures called chromosomes. These chromosomes duplicate before cell division in a process called DNA replication. The genome is divided into short segments that carry genetic information which are called genes and other segments that are involved in cellular processes, have structural purposes or involved in genetic regulation.

DNA has many important roles in all the known living organisms. One of its main roles is carrying hereditary information from parents along to their offspring. The information is encoded in the bases of the DNA and reproduced in all the cells of an organism. It is DNA that directs the development of biochemical, anatomical, physiological, and, to some extent, behavioral traits [12]. One of the most important functions of DNA is protein creation.
through the processes of transcription and translation. Genome sequences provide clues to important biological questions such as: identifying protein-coding genes in DNA sequences, determining gene functions, studying expression of interacting groups of genes and comparing genomes of different species to determine relationships between them and accelerating genome mapping of closely related species [12].

**Proteins**

Proteins are polymers with specific sequences of amino acid monomers called polypeptide, a single protein may consist of one or more polypeptides. Proteins are created from DNA through the processes of transcription and translation. In the first stage of protein synthesis, transcription, DNA is transcribed to RNA to protect DNA and its genetic information and to make the process faster by copying one gene to many RNA transcripts. RNA is very similar to DNA with some chemical differences in its nucleotide structure with carrying out different biological roles. The type of RNA that carries a genetic message from the DNA to protein-synthesizing machinery of the cell is called mRNA. It works as a bridge between DNA and protein synthesis. Next, in the translation phase the base sequences of a mRNA molecule are translated into orderly linking sequences of amino acids to form polypeptide in ribosomes [12]. Each triplet of RNA bases is translated into one amino acid.

Proteins have major functions in the cells. Some proteins give cells their shape and structure, while others facilitate cell processes such as proliferation and apoptosis. Using different combinations of the bases (A, C, T, G), DNA encodes all the different proteins, (i.e. genes make proteins) and to regulate the process master genes turn other genes on and off to make sure that the right proteins are made at the right time in the right cells.

**Transcription factor binding sites**

As we described in the previous section, proteins are created through the processes of transcription and translation. The transcription of genetic information from DNA to mRNA in living cells is regulated by some special proteins known as transcription factors (TF). These transcription factors are produced by genes. The genomic positions (loci) that transcription factors bind to are called transcription factor binding sites (TFBSs).

In a broader context, Genes are the cells operating system. Genes are performing cellular processes through protein synthesis. Initializing protein synthesis is due to TFs. Since TFs are released by their counter part binding
sites in DNA; i.e TFBSs, operating system of the cells can be understood by identifying and characterizing TFBSs.

**Sequence Motifs**

The recurring patterns in DNA that are conjectured to have a biological function are called motifs. Often they indicate protein binding sites such as nucleases or transcription factors or they are involved in important processes such as transcription termination [13]. There are numerous motif finding algorithms which can be used to find motifs from the binding sites predicted by peak detection algorithms. In this project we have validated our methodology by finding motifs from the results of the meta server. The identified highest ranked motifs are compared to consensus motifs of the selected transcription factors.

### 1.5 Aims of the Project

Given the variations of peak finders and the lack of a golden standard it is difficult, if not impossible, to decide which peak finder produces the most accurate peaks for a ChIP-seq dataset. Our aims were to: (i) provide a flexible interface for analyzing huge ChIP-seq datasets using several peak finders.(ii) produce the best putative peaks (binding sites) according to user-defined criteria by providing systematic comparison approaches.

### 1.6 Expected Readers

The report is mainly addressing the readers that have a basic background in computer science and bioinformatics. The results section may be of interest to the researchers who are conducting studies in the field of molecular biology, more specifically genetics.

### 1.7 Structure of The Thesis

This report is organized as follows. Chapter 2 introduces the characteristics of different peak detection algorithms and describes the main properties of the peak finders included in PFMS. Chapter 3 describes the main functions of PFMS and highlights the used methodologies. Chapter 4 explains the evaluation and validation methodologies and here we discuss and summarize the results. Finally, the conclusions are presented in Chapter 5.
Chapter 2

Peak Detection Algorithms

The aim of peak finding in ChIP-seq analysis context is to identify genomic regions with large densities of mapped sequence tags relative to measured or estimated background. A simple approach for achieving this goal is to take a sequence of mapped tags along the genome and to allow every contiguous sequence of base pairs with more than a predefined threshold number of tags covering them to be selected as an enriched binding site. However, the experimental noise and inherent complexities of the tags increase the requirement for more sophisticated algorithms. Numerous algorithms are designed based on different statistical models and enrichment measures. Differences in the characteristics of the algorithms make them identify different set of peaks for the same dataset. Results of the analysis in chapter 4 show how different peak finders generate different results for the same dataset. However, the existence of these different algorithms gives users the option to analyze their datasets under different conditions but on the other hand, choosing an appropriate tool becomes more difficult. In general the algorithms can be conceptually characterized based on the following basic attributes: (i) Building a signal profile, (ii) Building a background distribution model, (iii) Peak calling criteria, (iv) Post-filtering peaks, (v) Significance ranking [9].

There are some preprocessing steps which may be necessary to get accurate binding sites. Among those is accounting for distribution of tags in forward and reverse strands of the ChIP-seq sample. Since sequencing technologies usually sequence one end of the reads (either 5’ or 3’), the mapped tags of each strand are slightly shifted from each other. An easy approach to handle this issue is through computing an average distance \( d \) between the corresponding tags of each strand followed by shifting the tags of each strand towards its direction by \( d/2 \) and then taking an overlap of the two distributions. Another issue is considering the differences in the length of the fragments or sites and estimating an average fragment length.
2.1 Characteristics

Building a signal profile
Smoothing of tag counts allows reliable binding site identification and enhances peak resolution. A signal profile can be built based on the ChIP-seq sample by selecting the high-density sites along the genome. Mainly, there are two approaches to identify these sites. In the window-based (illustrated in figure 2.1A), as applied in CisGenome [14], a window of a fixed width is slided along the genome and centered at each site. The number of tags within it is counted and consecutive windows exceeding a threshold value are merged. The second approach (illustrated in figure 2.1B), as applied in FindPeaks [15], is to extend the ChIP-seq tags along their strand direction and to count overlaps (peak height) above a threshold value as peak regions. In case of performing tag shifting, the signal profile is modeled using the modified signal values obtained after shifting the tags (Figure 2.2A illustrates tag shifting).

Figure 2.1: Building signal profile. (A) Window based: A window of size 4 is slided along the genome and sites that have more than 4 tags at their centers are included in the signal profile, (B) Overlap count: The regions that have more than 4 overlapped tags are included in the signal profile
Building a background distribution model
A background model is used to filter out certain types of false positives in the treatment data. In the absence of experimental control data, the background distribution is modeled based on Poisson distribution, negative binomial distribution, Monte Carlo simulation or other statistical methods. When available, control data may be used to determine parameters for these distributions. Usually the signal includes some extra regions or noise due to the errors made during the sample preparation. Signal processing techniques such as background subtraction can be applied to remove those extra regions. Alternatively the signal can be thresholded by its enrichment ratio relative to the control. The control data can also be used to define enrichment measures in the peak identification process and to assign significance measures such as FDR to the identified sites (Figure 2.2B).

Peak calling criteria
The sites generated in the signal profile that satisfy certain enrichment or quality measures are the selected putative peaks. The quality criteria is either based on a minimum enrichment relative to the background model or a predefined threshold (Figure 2.2C).

Post-filtering peaks
The putative peaks can further be improved by eliminating artifacts. Eliminating sites that have unequal distribution densities between the two DNA strands and sites with more than one peak (duplicate hits) are the most popular ways to remove artifacts. These types of artifacts, optionally, can be filtered by some of the current peak detection algorithms (Figure 2.2D).

Significance ranking
The identified peaks are usually ranked based on one or more of the quantitative measures that represent the significance of each peak. The ranking measures include: peak height, fold enrichment, p-value, q-value, FDR or other similar measures (Figure 2.2E).

2.2 The Selected Peak Finders
Numerous peak finders are implemented with variations in the characteristics described above. The peak finders included in the meta server are described below.
Figure 2.2: The basic binding site detection components (Adapted from: [9])

(A) Upper left, plot ‘+’ strand reads in blue, ‘-’ strand reads in red, combined distribution after shifting the ‘+’ and ‘-’ reads toward the center by the read shift value in purple and subsequently a signal profile is formed. (B) In the presence of control data (brown), the same process as in A is repeated to form a background profile; otherwise, a random genomic background may be assumed. (C) The signal and background profiles are compared in order to define regions of enrichment. (D) The enriched regions are filtered. (E) The identified regions are ranked according to the relative strength or statistical significance (P(s): probability of observing a location with s reads covering it).
Model-based analysis of ChIP-Seq (MACS)

The tag shifting step is performed through shifting the treatment tags to their midpoint. MACS [16] takes the advantage of observed bimodal enrichment patterns of binding sites to model the shift size. It removes the redundant tags. The Poisson distribution is used to effectively capture local biases in the genome sequence and model the background. With the presence of control data FDR is estimated by dividing number of ChIP peaks over the control sample by the number of control peaks over the ChIP sample. MACS slides windows across the genome and candidate peaks with p-value below a predefined threshold are captured. The obtained peaks are measured by p-value, fold enrichment and FDR.

CisGenome

The window-based approach is used to scan the genome to identify the sites with read counts greater than a user-chosen cutoff. CisGenome [14] builds the background model using negative binomial distribution. However, with the presence of control data, CisGenome normalizes the difference between the treatment and control samples statistically then it uses conditional binomial distribution to model the signal profile by selecting ChIP reads that are significantly enriched relative to the control reads. Signals passing a user-chosen cutoff are used to generate predicted binding regions. As a post-processing step, the predicted sites are shifted. CisGenome also provides boundary refinement and single-strand filtering options for further refinement. The identified peaks are ranked by FDR.

FindPeaks

FindPeaks [15] estimates DNA fragment length distributions using several modes and extends each tag directionally. Additionally, tag shifting is provided as an option. FindPeaks builds a signal profile on the basis of overlapping tags (peak height) along the genome. A background is modeled by estimating the FDR using a Monte Carlo simulation to determine the likelihood of observing a peak of a given height. It uses control data, when available, to assign a probability to each peak of its likelihood of being part of the signal. Candidate peaks that satisfy a minimum FRD threshold are identified. The identified peaks can be further refined using filters such as: eliminating single site duplicates, enable directional peak detection, peak trimming and sub-peak identification. The peaks are ranked by their height [15].
Site Identification from Short Sequence Reads (SISSRs)

An average of the DNA fragments is estimated based on the ChIP-seq reads or a user-chosen value. SISSRs [17] uses strand-specific window scan with consecutive windows overlapping by half of the window size. For each window a net tag count is computed by subtracting the number of antisense tags located in the current window from the number of sense tags located in the same window. Every time the net tag count changes from positive to negative, the transition points are the selected start and end of a candidate binding site. That is if the tag counts in each strand of the inferred binding site passes a user-chosen cutoff. A Poisson background or a negative control sample, if available, is used to estimate the FDR, which is computed as the ratio of the number of peaks indicated by the background model to the number of peaks identified in the real data. Each identified peak is weighted by the number of directed reads supporting the binding site (tag density).

HPeak

The tags are directionally extended from their start positions. Next, the entire genome is partitioned into small bins of fixed length. Subsequently the fragments in each bin are counted to obtain a genome wide ChIP DNA coverage profile. Unlike the other peak finders, HPeak [18] applies a two state Hidden Markov Model (HMM) on the coverage profile to identify the candidate peaks in which the bins are classified into either ChIP-enriched (peaks) or non-enriched (background). The significance of enrichment of the peaks is adjusted and the maximum coverage among all the bins in each site is provided.

E-RANGE

E-RANGE [19] provides options to find the expected fragment length and shifts the tags accordingly. The signal profile is modeled based on tag aggregation. Poisson distribution is used to determine background model and with the presence of control data, fold enrichment and optionally p-values are determined. The control data is also used to estimate FDR by dividing the number of control tags over the number of ChIP tags. The candidate peaks are identified based on the background model and each peak is weighted using p-value.
SeqSite

SeqSite \cite{20} uses a tag clustering approach for modeling the signal profile in which the tags within a pre-defined distance belong to an individual cluster and the clusters with tag count more than a minimum threshold are selected as candidate peaks. Single-strand tag clusters can optionally be filtered. Subsequently, the candidate peaks are identified based on p-values or FDR. The identified peaks are refined by smoothing the tags with a user-chosen kernel density.
Chapter 3

Methodology

The implemented meta server analyzes a given data set first by extracting the tags of a specified chromosome. If it is necessary, it converts the extracted sample in order to prepare the required format for each of the included peak finders. The selected peak finders are executed in parallel or sequentially to identify a list of putative peaks (TFBSs). The obtained results are normalized and converted to a unified format. Two peak selection methods are developed, a voting mechanism and a scoring mechanism, to select a significant integrated list of peaks from the results of the selected peak finders. The results obtained with PFMS can be visualized using the current genome browsers. These steps are explained further in this chapter.

3.1 Functions of the PFMS

The implemented software can be used for performing the following tasks:

- Identifying peaks using different peak finders through a single interface and a uniform data format with the ability of customizing each of the peak finders from one interface.

- Integrating peaks from different peak finders and producing a list of the best putative peaks by providing two peak selection methods.

- Producing the obtained peaks in output formats that are compatible with the current genome browsers.

- Analyzing individual chromosomes or a list of chromosomes based on user-chosen parameters.
3.2 Peak Detection

The peak finders mentioned in the second chapter can be combined to analyze a given data set. A control dataset can optionally be used to improve the peak detection process. The following steps are taken to identify peaks with each peak finder and to uniform the peaks in an appropriate format for peak aggregation and selection.

3.2.1 Tag Handling

Accepted Data Format: PFMS accepts BED format, in which each tag of a given dataset should has the following four fields separated by tab, optional fields can exist as well:

<table>
<thead>
<tr>
<th>Chromosome-number</th>
<th>Start-position</th>
<th>Stop-position</th>
<th>Strand</th>
</tr>
</thead>
</table>

Chromosome-number: the name of the chromosome (e.g. chr1, chrX).  
Start-position: the starting position of the tag in the chromosome.  
Stop-position: the ending position of the tag in the chromosome.  
Strand: the strand sign of the tag (+ or -)

Data Preprocessing: prior to peak detection using multiple peak finders, the following steps are applied for a given dataset:

- Chromosome filtering: As the meta server is designed to process a single chromosome, the given sample that may contain all the chromosomes of the genome of interest is split into single chromosome dataset(s). And each obtained dataset is processed individually. By default the chromosome number of the first tag of the data set is selected for the whole process but users can specify the chromosome of interest. In addition, it is also possible to use the meta server to analyze all the chromosomes of a given dataset sequentially.

- Format conversion: Since some of the peak finders are requiring data formats other than the BED format used by PFMS (defined above), the given BED sample, if necessary, is converted to the specific format of the peak finer.

3.2.2 Peak Identification

The given ChIP and control datasets, after conversion if required, is analyzed to obtain a list of peaks from each of the included peak finders. Each
peak finder uses a copy of the given dataset(s) and starts the peak detection process. The settings of each peak finder can be customized based on user’s requirements.

### 3.2.3 Output Format Conversion

The obtained lists of peaks from each peak finder is converted into one of the following formats (Table 3.1) based on the selected peak selection method:

- **BED format**: each peak is defined by its chromosome number, start position and end positions.

- **Wiggle variable step**: it starts with a track definition line and a fixed step size followed by start positions of each peak and the score of enrichment. The step size is chosen based on minimum step size of the obtained results.

```
track name="BED sample format"
chrN start stop
chr1 842525 842750
chr1 858351 858875
chr1 875351 875775
chr1 894491 894946
chr1 906251 906475
chr1 907276 907500
```

```
track name="WIG sample format" description=""
variableStep chrom=chr1 span=25
start score
842525 12
852550 14
852575 10
852600 11
852550 14
```

Table 3.1: (a) The table on the left shows the BED format used by PFMS. The BED dataset starts with a definition line and followed by lines containing chromosome number, start and stop positions of the peaks. (b) The table to the right shows the WIG format used by PFMS. The WIG dataset starts with a definition line, the second line contains format type, chromosome number and span size for the entire dataset. After these lines Start positions with scores of each peak are listed

### 3.3 Peak Selection Methods

The peaks detected by several peak finders are unified into one of the formats defined above and consequently they are integrated. The regions with a certain frequency or score are selected using one of the methods presented in this section. The design of the peak selection methods is due to study of [21]:

17
3.3.1 In-degree Centrality Voting Mechanism

A directed edge is made between each peak (node) and the peak finder (voter) that has detected (elected) it. Since every node in the list of candidate nodes has at least one directed edge, all nodes are considered. All the nodes are combined and the overlapped regions of each node are aggregated. The vote count \( k \) of each overlapping region is increased based on the number of directed edges to it. The regions that have votes \( k \) more or equal than predefined threshold \( \text{min\_rank} \) are the selected \( \text{min\_rank} \) significant peaks. Subsequently the regions that do not have a sufficient number of directed edges are removed. When the threshold value of \( \text{min\_rank} \) is not set, it is computed as:

\[
\text{minimum\_rank} = \left\lfloor \frac{\text{number of voters}}{2.0} - 0.1 \right\rfloor + 1
\]

(3.1)

This method is used when the detected peaks are given in the BED format.

A case study: Assume 6 peak finders (PF1, PF2,...etc) detected a set of peaks from a given dataset. The results were analyzed using the voting mechanism. The aim was to observe how the selected peaks are changing when different values of \( \text{min\_rank} \) is used (Figure 3.1). The peaks that were identified by each peak finder were selected when \( \text{min\_rank}=1 \) (First row in the figure), the peaks that were identified by at least 2 peak finders were selected when \( \text{min\_rank}=2 \) and so on. Using this feature makes it possible to set the degree of selectiveness. In order to select the minimum number of peaks the value of \( \text{min\_rank} \) should be set to the number of peak finders. It means only the peaks that identified by all the peak finders are selected (Fifth row of the figure).

Figure 3.1: Voting mechanism for peak selection. Each of the top six rows shows the selected regions by voting mechanism when the \( \text{min\_rank} \) value of 1 to 6 was chosen. The 6 rows at the bottom are the identified peaks by six peak finders, respectively.
3.3.2 Scoring Mechanism

When the WIG format is used, individual peaks are weighted based on their statistical significance. In order to get advantage of this ranking a scoring mechanism is provided. In this method, the peaks identified by all the peak finders are selected and the overlapping regions are integrated. The integrated regions are weighted based on the score of the overlapped peaks. Optionally, low weighted peaks can be eliminated from the selected list of peaks. This method is used when the detected peaks are given in the WIG format.

Normalization

The peaks detected by the individual peak finders are usually scored by different enrichment measures. Due to different ranges of the scores, weighting of the selected peaks may be biased to the peaks ranked with large scale scores. To overcome this problem the obtained peaks from each peak finder are normalized.

After sorting the lists of the peaks detected by the peak finders, the scores of a set of the highest ranked peaks from one of the list $PF1$ are summed and an average score is computed $\text{avg\_score}$ (Equation 3.2) to obtain an average score value for the selected list of peaks. The same process is repeated for all the other lists. One of the computed $\text{score\_avg}$ is selected as $\text{normalized\_score}$. A normalization ratio is computed for each list by dividing $\text{score\_avg}$ of the list over the normalized score. Finally, scores of the detected peaks $p$ are multiplied by the normalization ratio computed for the selected list to obtain normalized score value for all the peaks (Equation 3.3). Assume $n$ is the number of the highest ranked peaks to be considered, the scores of each peak finder $PFs$ are normalized as follows:

$$
\text{avg\_score}(PF1) = \frac{\sum_{p=0}^{n} \text{score}(p)}{n}
$$

Assume: $\text{normalized\_score} = \text{avg\_score}(PF1)$

$$
\text{normalization\_ratio}(PF2) = \frac{\text{avg\_score}(PF2)}{\text{normalized\_score}}
$$

$$
\text{new\_score\_of\_PF2}(p) = \text{score}(p) \times \text{normalization\_ratio}(PF2)
$$

Peak Selection

The goal of this method is to select the peaks that are detected by all the peak finders and emphasizing on the regions shared by more than one peak.
finder. The overlapped regions of the peaks within a selected step size are aggregated and weighted by summing the scores of the overlapped regions. A cutoff value can be set to obtain \( n \) percentage of the highest ranked of the selected peaks.

**A case study:** Three sets of peaks were detected for a single dataset using three different peak finders. A single set out of them was selected using the scoring mechanism (Figure 3.2). The first graph shows how the scoring mechanism consolidated all the detected peaks into a single set with increasing significance of the overlapped regions.

![Figure 3.2: Scoring mechanism for peak selection.](image)

The bottom three graphs indicate set of peaks obtained by the different peak finders. The top graph shows the results of scoring mechanism approach which combines the peaks detected by the included peak finders. The summit points indicate that these regions are detected by more than one peak finder.
Chapter 4

Results and Evaluation

4.1 Results of The Project

The PFMS software framework, which is the outcome of this project, identifies 'computationally accurate' binding sites among thousands to millions of putative binding sites generated from different peak detection algorithms. As shown in the next section the peaks which are selected by PFMS appear to be statistically more accurate than the peaks identified by a single peak finder. The features implemented in PFMS are intended to facilitate the analysis of large ChIP-seq datasets (Details about the software can be found in the appendix of this thesis). Another aim was to analyze some 'well known' ChIP-seq datasets to evaluate the PFMS. The results of the analyses could be studied further with the possibility of biological findings.

4.2 Evaluation of PFMS

Evaluating the results of PFMS (selected binding sites) is a challenging task due to the lack of a widely accepted golden standard. Although there is no comprehensive list of all genomic locations bound by a target transcription factor under experimental conditions (true positives), the ChIP-seq datasets of several transcription factors have been analyzed in numerous studies. Many binding sites have been identified with their motifs and in some cases the results are validate using qPCR experiments. The availability of these results gives the opportunity to evaluate reproducibility of PFMS across biological replicates.

Four of those transcription factor ChIP-seq datasets with their corresponding controls that are publicly available were selected to evaluate performance of PFMS across diverse studies: FoxA1 (hepatocyte nuclear factor
3a) transcription factor in MCF7 cells was obtained from the study of [16]. NRSF (neuron-restrictive silence factor) and GABP (growth associated binding protein) in Jukarta cells were published in the study of [5] and STAT1 (Signal Transducers and Activators of Transcription) with its matching control sample were generated in interferon-stimulated HeLa S3 cells from the study of [22]. All the selected datasets have a well defined canonical motif which can be used to assess quality and confidence of the detected peaks (See section 4.2.4). A short summary of the selected datasets is given in table 4.1).

4.2.1 Analysis Methods

In order to evaluate the comparison approaches under different experimental conditions, four different experiments were made for each of the selected transcription factor datasets. In two experiments, the voting mechanism was used to compare the predicted binding sites obtained with each peak finder. The experiments were made using the individual ChIP samples alone (without control data) as well as with using corresponding control samples. While in the other two experiments, the scoring mechanism was used to analyze the ChIP samples alone as well as with their corresponding control data. That is beside two other comparative experiments with different sets of peak finders for evaluating the peak finders performance in order to show the combination sets of peak finders that work best in certain circumstances. A short summary of each experiment is shown in Table 4.2 with providing an abbreviation for each of them.

All the included peak finders in PFMS were used in the voting comparison (BED format), while only four (MACS, CisGenome, FindPeaks and HPeak) of them were used in the score comparison (WIG format) due to the fact that Erange, SISSRs and SeqSite are not generating the compatible wiggle format. All the peak finders were run with their default parameters which are likely to be the choice of an average user. Further, in all the experiments that included control data HPeak was excluded since it was not applicable.

In total, sixteen experiments were made for all the datasets using both comparison approaches and various sets of peak finders. From these experiments twenty four results were obtained for each transcription factor dataset (That is including the results of each peak finder which were considered for comparison purposes). In this thesis report, only the results of FoxA1 are shown, while the results of the other experiments are provided in supplementary materials. In the following, the results including the control data identified by adding an addition C letter to the name of the experiment along with B for BED or W for WIG to specify the used comparison method.
4.2.2 Results and Discussion

As expected, characteristics of the selected peaks by PFMS in each experiment differed based on the algorithms selection, the used comparison method and the inclusion of control data. Obviously, the peaks that were obtained using the voting mechanism approach were fewer and have a wider range than those obtained using the scoring mechanism approach since WIG format is a more dense format. For this reason we compare the experiments that were made using BED format (namely B, BC, B-F and BC-F) separately from the experiments that were made using WIG format (W and WC).

In experiment B, it was possible to use results of all the seven peak finders to evaluate PFMS, while in experiment B-C, HPeak had to be excluded since it was not working with control data. Further, to make the results more comparable FindPeaks was excluded in two other experiments (B-F and BC-F), since it dramatically increases the number of peaks and decreases their width which has a ‘computationally’ huge effect on PFMS’s result.

A striking observation was reduction of the number of detected peaks in experiment BC compare to B while their average width were slightly reduced. Improvements in the detected peaks by the inclusion of control data distinguished MACS, CisGenome and Finpeaks from the other peak finders. Consequently, the results of PFMS were improved. Even though no improvements were noticed with the results of Erange and SeqSite and the sites obtained with SISSRs were slightly improved.

In order to show the differences among the results of PFMS with the results obtained from each peak finder, the number of peaks detected by each peak finder and PFMS were counted for all the experiments and the results are shown in table 4.3. Further, to give a better insight into the detected peaks an average width for the list of peaks of each algorithm and PFMS was computed. In the case of BED format, the average value for each obtained list of peaks was calculated by summing the base distance between the start and end positions of each peak and dividing the sum by the total number of peaks. While in WIG format the span size of the tags were considered, the results are given in table 4.4.

Significantly more peaks were obtained when WIG format was used. Since the number of detected peaks by the used peak finders was considerably increased (MACS: from 26K to 27M, CisGenome 7.5K to 11.5M, HPeak: 75K to 1M, FindPeaks 37M to 74M). As noticed from the experiment WC FindPeaks detected fewer peaks when control data was included and the span size of the peaks detected with CisGenome was increased from 25 to 50.
### Table 4.1: ChIP-seq samples used for evaluating PFMS.

<table>
<thead>
<tr>
<th>Sample</th>
<th>Cell type</th>
<th>Reads (million)</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>FoxA1</td>
<td>MCF7</td>
<td>3.901</td>
<td>[16]</td>
</tr>
<tr>
<td>Control</td>
<td>MCF7</td>
<td>5.221</td>
<td>[16]</td>
</tr>
<tr>
<td>NRSF</td>
<td>Jurkat</td>
<td>8.813</td>
<td>[5]</td>
</tr>
<tr>
<td>GABP</td>
<td>Jurkat</td>
<td>7.862</td>
<td>[5]</td>
</tr>
<tr>
<td>Control</td>
<td>Jurkat</td>
<td>17.404</td>
<td>[5]</td>
</tr>
<tr>
<td>STAT1</td>
<td>HeLa S3</td>
<td>26.731</td>
<td>[22]</td>
</tr>
<tr>
<td>Control</td>
<td>HeLa S3</td>
<td>23.435</td>
<td>[22]</td>
</tr>
</tbody>
</table>

### Table 4.2: Experiments made for FoxA1 ChIP data

<table>
<thead>
<tr>
<th>Exp.</th>
<th>MACS</th>
<th>CisGenome</th>
<th>FindPeaks</th>
<th>SeqSite</th>
<th>Erange</th>
<th>SISSRs</th>
<th>HPeak</th>
<th>PFMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>26,326</td>
<td>7,443</td>
<td>37,698 M</td>
<td>2,996</td>
<td>12,310</td>
<td>11,109</td>
<td>75,207</td>
<td>433,731</td>
</tr>
<tr>
<td>B-F</td>
<td>26,326</td>
<td>7,443</td>
<td>N/A</td>
<td>2,996</td>
<td>12,310</td>
<td>11,109</td>
<td>75,207</td>
<td>12,259</td>
</tr>
<tr>
<td>BC</td>
<td>28,942</td>
<td>21,647</td>
<td>3.728 M</td>
<td>2,996</td>
<td>12,083</td>
<td>9,275</td>
<td>N/A</td>
<td>261,448</td>
</tr>
<tr>
<td>BC-F</td>
<td>28,942</td>
<td>21,647</td>
<td>N/A</td>
<td>2,996</td>
<td>12,083</td>
<td>9,280</td>
<td>N/A</td>
<td>9,273</td>
</tr>
<tr>
<td>W</td>
<td>27,124 M</td>
<td>11,552 M</td>
<td>74,679 M</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>990,693</td>
</tr>
<tr>
<td>WC</td>
<td>27,124 M</td>
<td>14,680 M</td>
<td>2.325 M</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>298,208 M</td>
</tr>
</tbody>
</table>

Table 4.3: Number of Peaks identified by the peak finders and PFMS.

B-F and BC-F indicate experiments B and BC, respectively, but without using FindPeaks.

<table>
<thead>
<tr>
<th>Exp.</th>
<th>MACS</th>
<th>CisGenome</th>
<th>FindPeaks</th>
<th>SeqSite</th>
<th>Erange</th>
<th>SISSRs</th>
<th>HPeak</th>
<th>PFMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>286</td>
<td>221</td>
<td>19</td>
<td>180</td>
<td>151</td>
<td>116</td>
<td>304</td>
<td>5</td>
</tr>
<tr>
<td>B-F</td>
<td>286</td>
<td>221</td>
<td>N/A</td>
<td>180</td>
<td>151</td>
<td>116</td>
<td>304</td>
<td>185</td>
</tr>
<tr>
<td>BC</td>
<td>280</td>
<td>294</td>
<td>6</td>
<td>180</td>
<td>152</td>
<td>116</td>
<td>N/A</td>
<td>5</td>
</tr>
<tr>
<td>BC-F</td>
<td>280</td>
<td>294</td>
<td>N/A</td>
<td>180</td>
<td>152</td>
<td>116</td>
<td>N/A</td>
<td>179</td>
</tr>
<tr>
<td>W</td>
<td>10</td>
<td>25</td>
<td>10</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>25</td>
<td>10</td>
</tr>
<tr>
<td>WC</td>
<td>10</td>
<td>50</td>
<td>10</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 4.4: Average peak width obtained with the peak finders and PFMS

24
4.2.3 Simulation of Physical Genomic Locations

The peaks detected in chromosome 4 (chr4) of transcription factor FoxA1 ChIP-seq dataset from the previous experiments were simulated in UCSC genome browser [23] based on version h18 of the human assembly. In experiment B, PFMS found all the peaks that are detected by at least four of the included peak finders since the minimum rank parameter was set to 4 (Figure 4.1). A comparison between the peaks identified by PFMS in the experiments B and BC was made to illustrate the impact of using control data in regions of chromosome 21 (Figure 4.2). Further to assess PFMS’s accuracy a random location of the genome were zoomed to show that only the desired peaks are selected (Figure 4.3).

The peaks obtained with FindePeaks, CisGenome, MACS, HPeak and PFMS in the experiment W are shown in Figure 4.4. Again, the inclusion of control data increased the performance of the used peak finder as well as PFMS. Noticeably CisGenome was the only peak finder that detected negative peaks based on the background control data. This had impacts on the results of PFMS as well. A portion of Chromosome 4 is shown in Figure 4.5 to illustrate the impacts.

4.2.4 External Validations Using High Scoring Binding Motifs

There are very few methods that can be used to evaluate detected binding sites of a specific transcription factor, one method is to find the highest motifs from the sequences of the putative sites and to compare them against consensus motifs of a given transcription factor. This method can only be applied for the transcription factors that have already been analyzed. However, due to inaccuracy in the analyses there may appear other undiscovered motifs.

In the case of transcription factor FoxA1, sequences of the binding sites from experiment B-F were obtained using version h18 of the human assembly from UCSC Table browser [24]. The obtained sequences were examined using BioProspector [25] to find the highest ranked motifs. The top five motifs of length 10 were considered. Finally, The selected motifs were compared against JASPAR database [26] and a very high similarity to the consensus motifs of FoxA1 was observed (Figure 4.6).
Figure 4.1: Detected binding sites of FoxA1 TF in Chromosome 4 using BED format (Results of experiment B). The top row shows results of PFMS. The bottom row shows the ChIP-seq treatment data. And the remained rows illustrate the results of the used peak finders.

Figure 4.2: A comparison between a set of peaks identified by PFMS for FoxA1 TF in chromosome 21 using BED format. First row which is labeled PFMS, shows the results of PFMS in experiment B (using ChIP data alone). The second row which is labeled PFMSC shows the results of PFMS when control data was included in experiment BC.

Figure 4.3: Enlarged view of the detected binding sites of FoxA1 TF in Chromosome 4 using BED format with including control data (Results of experiment BC). The bottom row shows results of PFMS. And the remained rows illustrate the results of the used peak finders, the labels indicate the corresponding used peak finder.
Figure 4.4: Detected binding sites of FoxA1 TF in Chromosome 4 using WIG format (Results of experiment W). The result of PFMS is shown in the top row followed by the results of HPeak, CisGenome, MACS and FindPeaks.

Figure 4.5: Enlarged view of the detected binding sites of FoxA1 TF in Chromosome 4 using WIG format with including control data (Results of experiment WC). The result of PFMS, which includes the sites identified by all the peak finders, is shown in the top row followed by the results of CisGenome, MACS and FindPeaks. (Note the negative signals produced with CisGenome)
Figure 4.6: Results of de novo motif search (using BioProspector) versus consensus motifs of FoxA1 in JASPAR Core database.
The right column represents top five predicted motifs from the binding sites detected by PFMS in experiment ‘B-F’ using BioProspector and the right column represents consensus motifs of FoxA1 in JASPAR database.
Chapter 5

Conclusion and Future Work

5.1 Conclusion

The amount of data generated from next generation sequencing technologies is ever increasing, in contrast the demands for new computation tools for analyzing purposes is getting increased as well. Numerous algorithms have been designed to complement the goal of identifying cistrome DNA-protein interactions from the data obtained using chromatin immunoprecipitation (ChIP) with massively parallel DNA sequencing. Providing a framework to combine those algorithms and more importantly analyzing their results further makes the analysis process much easier and faster.

In this thesis project, a MetaServer was developed to provide a single interface for seven up-to-date peak finders with the ability of extending it to add new peak finders. Most importantly, the detected peaks with a set of peak finders are compared to obtain a list of the best putative peaks according to user-defined criteria. The users have ability to choose the deepness of the enriched peaks by setting the comparison parameters. Four widely studied transcription factor datasets are used to evaluate the developed software and the results are simulated and then validate using the highest predicted ranked motifs of the selected transcription factors.

5.2 Future Work

The following outlines are some of the functions that are expected to improve the meta server for making the analysis process faster and obtaining more accurate results:

- Extending the meta server by including new peak finders. In order to obtain higher resolution peaks, it is recommended to include peak
finders that are designed based on different algorithmic approaches than the methods discussed in the second chapter.

- Applying the developed comparison approaches on a full genome scale to make the analysis process faster.

- Improving BED comparison approach by considering statistical significance of the enriched peaks with each peak finder and ranking the meta server results accordingly (Using a weighted voting mechanism).

- The control data can be used to get more accurate peaks from the meta server. Accounting for negative peaks would cover more sites in the genome.

- Currently the input data set format is restricted to BED, supporting other widely used data set formats makes the meta server more accessible.
Appendix A

PFMS User Manual

1.1 Overview

PFMS is a software application to identify genome wide transcription factor binding sites in CHIP-Seq data. PFMS provides a flexible interface for integrating several peak finders and produces the best putative peaks (binding sites) from the peaks identified by the included peak finders according to user-defined criteria. The peaks are selected based on one of the two developed systematic comparison approaches: voting mechanism or scoring mechanism.

1.2 Download PFMS

PFMS is a free software implemented in python and intended to be used for the purpose of academic research in the Bioinformatics and Genomic areas.

The latest version of PFMS is freely available under GNU GENERAL PUBLIC LICENSE and it can be obtained via: http://www.lcb.uu.se/. It is users responsibility to consider licenses of the included peak finders.

1.3 Supplementary Material

Users are expected to have their own datasets and/or background data. However, the datasets that have been used to evaluate PFMS can be obtained from these resources:
FoxA1 [10]: http://liulab.dfci.harvard.edu/MACS/Sample.html
NRSF, GABP \[5\] and STAT1 \[16\]:
http://bioinfo.au.tsinghua.edu.cn/seqsite/

The results obtained with PFMS for the datasets above are provided as supplementary material on: \text{http://www.lcb.uu.se/}

1.4 Installation

PFMS 1.0 has been implemented and tested on Unix-like systems. Windows users are encouraged to try it using \text{Cygwin}.

Please make sure the following software are installed:

- Python 2.6 or higher (required to use PFMS)
- GCC or C compiler (Some of the peak-finders are implemented in C)
- Perl (Required to use SISSr peak-finder)
- JRE 1.6 (Required to use FindPeaks peak-finder)

Note:
\text{a)On most of the UNIX-like systems (including Mac OS X and Linux) Python, C compiler and Perl are installed by default.}
\text{b)Without Perl and/or JRE1.6, PFMS would still work but you will not be able to use SISSRs and/or FindPeaks.}

After downloading the compressed source distribution version of PFMS users are expected to extract it by using one of the available archive tools:

\text{gunzip -c PFMS-1.0.tar.gz | tar xf-}

Navigate to the extracted directory: \text{cd directory_path/PFMS-1.0}

Installing PFMS with root access

The following command installs the python modules to python’s standard location and the supported peak finders to python’s prefix directory (hint: root access is needed to perform the installation)

\text{sudo python setup.py install}
Installing PFMS by Normal users

If root access was not obtainable to perform the previous installation, PFMS can still be installed as standalone and the only drawback is that experiments can be made only in the extracted directory (aka. PFMS-1.0) in order to run the meta-server since everything will be installed there.

```python
python setup.py install -normal
```

In order to remove PFMS, navigate into the extracted directory and type:

```bash
sudo python setup.py remove
```

### 1.5 PFMS Usage

In order to execute PFMS with it’s default settings, use one of the following commands based on the installation type

```bash
PFMetaserver -i <input_file.bed> <-o output_label> [Options]
```

Note: If PFMS is used by a normal user (without system installation) then ‘PFMetaserver’ needs to be replaced with ‘python PFMetaserver.py’:

```bash
python PFMetaserver.py -i <input_file.bed> -o <output_label> [Options]
```

#### 1.5.1 Default Settings

**Data Set Handling:** by default, reads that are from the same chromosome as the first read of the given input file are handled while `-chr` option forces PFMS to process the reads from the specified chromosome. Additionally, in order to handle a whole dataset use `-all_chr` which tells PFMS to split the given dataset to individual chromosomes [using FindPeaks split tool [15]] and process each chromosome using desired comparison approach. After identifying peaks for each chromosome individually it combines all the results to a single output file (results of each chromosome can be kept, optionally, using `-store_results` option).

**Comparison Approach:** BED format is used when `-wig` is not given in the list of command line parameters and a peak is considered to be significant when it’s at least identified or overlapped with more than half of the used peak finders. And in the case of WIG format, all the peaks are selected since `-percentage 100` is used. But they can be filtered by minimizing the percentage ratio.
**Peak Finders:** In the case of BED format MACS, CisGenome, SISSRs, Erange and SeqSite are used. While only MACS, CisGenome and FindPeaks are used with WIG format.

**Execution Mode:** When more than two processors are available on the target machine, PFMS makes a process pool to execute each peak finder in a single process and combines the results. The maximum number of used processors can be restricted with -max_cpu option. PFMS can be forced to run in sequential mode by using -sequential option (not recommended).

### 1.5.2 Command line options

The following is a list of the available features and options.

Hence: The parameters enclosed between square brackets are optional.

- `-i input_file.bed`: Input data file path (currently, only the standard 6-column BED format is accepted).

- `-o output_label`: Used to label output directory and file names.

- `[-control control_file.bed]`: Background data file path (currently, only the standard 6-column BED format is accepted).

- `[-chr chromosome]`: Forces PFMS to process the specified chromosome instead of chromosome number of the first read of the input file.

- `[-min_rank number]`: The minimum threshold value to select a peak (it should be in range of the quantity of used peak finders).

- `[-wig]`: Uses WIG format instead of BED. (please note this feature can only be used with MACS, CisGenome, FindPeaks and HPeak).

- `[-percentage number]`: Specifies the percentage of the identified peaks to be obtained, to be used only with -wig option (default is 100).

- `[-parallel]`: Forces PFMS to execute the peak-finders in parallel (it’s the default if more than two processors are available).

- `[-sequential]`: Forces PFMS to execute the peak-finders Sequentially (it’s the default mode when fewer than two processors are available or the Python 2.6 or higher is not available).

- `[-max_cpu_use number]`: Sets the maximum number of processors to be used by PFMS (default is 6).  

34
[**-min_cpu number**]: PFMS is running in parallel mode if minimum number of processors (CPU) was available on the system (default is 2).

[**-store_results**]: Keeps the original files generated by the peak-finders (plus results of the spitted chromosomes when -all_chr is used).

[**-min_size number**]: Minimum file size (in KB) of a peak-finder result in order to be included in the comparison (default is 1).

[**-all_chr**]: Executes PFMS for each chromosome in a given dataset and combines the results (Hint: PFMS is essentially intended to be used for one chromosome).

[**-cisgenome**]: Detects the binding sites using CisGenome [14]

[**-macs**]: Detects the binding sites using MACS [16]

[**-findpeaks**]: Detects the binding sites using Findpeaks [15]

[**-hpeak**]: Detects the binding sites using HPeak [18] (can be used with no presence of control data)

[**-erange**]: Detects the binding sites using Erange [19] (can be used only with BED comparison approach)

[**-sissr**]: Detects the binding sites using SISSRs [17] (can be used with BED comparison approach)

[**-seqsite**]: Detects the binding sites using SeqSite [20] (can be used only with BED comparison approach)

[**-help**]: Prints a usage message with a list of the implemented options.

### 1.5.3 Output Visualization

The identified transcription factor binding sites (peaks) can be visualized using UCSC genome browser, integrated genome browser (IGB) or any other browser that supports either BED or WIG format.

The results can be found under: output_label_Results directory. The identified with PFMS are stored either in: output_label_Results.bed/wig or output_label_Results.bed/wig (when -all_chr was used). That is beside the result of each peak finder for each chromosome that will be stored separately when -store_results is used.
1.5.4 A Usage Example

Assume, the ChIP-seq data file is named ‘Treat.bed’ and the control data is named ‘Input.bed’, both are located under the current working directory. The experiment requirement goal is to find all the TFBSs in chromosome four that are identified by, at least, four peak finders out of six with using BED comparison approach and label the results with ‘FoxA1.peaks’. In addition, keep results of all the peak finders.

PFMetaserver -i Treat.bed -control Input.bed -o FoxA1_peaks -macs -sissr -seqsite -cisgenome -erange -hpeak -min\_rank 4 -store\_results

1.6 Included Peak Finders

A list of the peak-finders included in the current version of PFMS is given in the following table. It’s worth mentioning that some of the peak-finders probably have other useful features beside binding site detection from ChIP-seq samples, for instance RNA-seq and downstream analysis. But in the current version the main focus is on ChIP-seq. For more details users are recommended to consult the peak-finders’ manual page.

<table>
<thead>
<tr>
<th>Source</th>
<th>Ref.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MACS v1.3.7</td>
<td>[16]</td>
</tr>
<tr>
<td>CisGenome v2.0</td>
<td>[14]</td>
</tr>
<tr>
<td>Findpeaks v3.1.9.2</td>
<td>[15]</td>
</tr>
<tr>
<td>Hpeak v1.1</td>
<td>[18]</td>
</tr>
<tr>
<td>E-range v.2.1</td>
<td>[19]</td>
</tr>
<tr>
<td>SeqSite v1.0</td>
<td>[20]</td>
</tr>
<tr>
<td>SISSRs v1.4</td>
<td>[17]</td>
</tr>
</tbody>
</table>

Table A.1: Peak Finders included in PFMS

1.6.1 Customizing Peak Finder’s Parameters

PFMS comes with a configuration file which is used to customize the optional parameters of each peak-finder.

If you have installed PFMS on the system directory (the first installation type), you should be able to locate pfms.conf file in a directory called PeakFinders in one of the following places:

```
/usr/local/  -- Unix-like systems with standard python installation
```
/usr/   – Unix-like systems with non-standard python installation  
C:\Python – Windows systems

But PFMS is used under the original source directory, then the pfms.conf file should exist in PFMS-1.0/PeakFinders directory.

Configuration File Style

The configuration file is divided into two sections

1. Peak-finder parameters: List of optional parameters for each peak-finder can be stated in a single line followed by the peak-finder’s name and a colon (please consider the peak-finder’s usage options)

2. Peak-finders paths related to the PeakFinders/ directory: This is particularly useful to upgrade a peak-finder to a newer version (as far as the new version has the same directory structure and input format as it’s current version) or to force PFMS to look for a specific peak finder in a different location.

Below is the default content of pfms.conf:
MACS: gsize=1000000000
SISSR: -s 3080000000 -F 50 -L 100 -w 50
HPEAK:
ERANGE:
CISGENOME:
SEQSITE: -F
FINDPEAKS: -dist_type 1 -wig_step_size 10
#Peakfinder’s path related to PeakFinders/ directory which is parent directory of this file and the included peak-finders
SISSR-PATH:/sissrs_v1.4/sissrs.pl
CISGENOME-PATH:/cisGenome-2.0/
FINDPEAKS-PATH:/findpeaks/
HPEAK-PATH:/HPeak/HPeak-1.1/HPeak.pl
SEQSITE-PATH:/SeqSite1.0/
ERANGE-PATH:/Erange/commoncode/
#If you already have installed macs on your system then change the line below with
MACS-PATH:macs
MACS-PATH:/MACS-1.3.7.1/lib-modified/macs
Bibliography


