IniMotif Manual

Lu Cheng

October 29th, 2009

Abstract

IniMotif devotes to discover motifs from large amounts of sequencing data. All input sequences should be with **the same length l**. We also assume the existence of a **consensus sequence** (width w) in the data.

The software is designed based on 1-base mutation Model. At first the substring with the highest count will be chosen as the consensus sequence, then all substrings of a sequence will be compared to the consensus sequence. The substring with the lowest hamming distance is regarded as the binding site of this sequence. Binding sites which are 1-base mutation of the consensus sequence are used to construct the motif.

For the second motif discovery mode, the software removes l-width sequences which contains the w-width consensus sequence of the first run and its 1-base mutations from the input sequence pool. Then it starts motif discovery in the same manner as that in the first motif discovery, from the rest of the l-width sequences.

1 Algorithm Description

This section gives a simple description of the algorithm used by IniMotif. Here we focus on how we calculate the motif.

Algorithm 1 IniMotif Algorithm

```
Initialize $l$, $w$, read input sequence file.
Count all $l$-width input sequences, so that we get $n$ unique $l$-width sequences $(s_1, s_2 \dots s_n)$ with counts $(N_{s_1}, N_{s_2} \dots N_{s_n})$.
Break $(s_1, s_2 \dots s_n)$ into $w$-width sbustrings (overlaping), count all the $w$-width sbustrings, so that we get $(w_1, w_2 \dots w_m)$, with counts $(N_{w_1}, N_{w_2} \dots N_{w_n})$. {Identical substrings of $s_i$ are only counted once.}
Select the $w$-width substring $w_{consensus}$ with the highest count in $(w_1, w_2 \dots w_m)$ as the consensus sequence.
Set the mapping from binding sites to their counts $B = \varnothing$
for $s_i \in (s_1, s_2 \dots s_n)$ do
    if $s_i \ni (w_1, w_2 \dots w_m)$, $where\ d(w_i, w_{consensus}) \leq 1$ then
        add $(w_1, w_2 \dots w_m)$ to $B$, where $B(w_i) = B(w_i) + N_{s_i}/m$. {$w_i$ are all substrings with the least Hamming distance of $s_i$}
    end if
end for
Set the $4 \times w$ position frequency matrix $M = \varnothing$
for $b \in A, C, G, T$ do
    for $j \in 1, 2 \dots w$ do
        Set $M_{bj} = B(w_{consensus}(b, j))$, where $w_{consensus}(b, j)$ is a $w$-width substring by specifying the $j$th base of $w_{consensus}$ as $b$.
    end for
end for
Output $M$
```

2 Instructions for running the program

The IniMotif software consists of four modules: The first java module calculates motifs and produces all necessary data; the second matlab module draws figures from the data produced by the first module; the third perl module draws logos for all produced motifs; the fourth perl module organizes all information in the form of HTML pages. We strongly recommend you to read Section 6 if you are not clear about the predefined specifics.

2.1 The java module

- How to run the module?
  cd IniMotifDirectory
  make compile
  java -jar InitialMotif.jar -MODE [-skip]
  # In case of low memory, please use "java -Xms4g -Xmx4g -jar InitialMotif.jar -MODE" instead
- Different Modes
  the "-MODE" can be replaced by the following parameters:
  - -uniform: the input directory contains all barcode files of all cycles and all batchs in one folder, only the primary motifs will be calculated
  - -secondary: the input directory contains all barcode files of all cycles and all batchs in one folder, both the primary and secondary motifs will be calculated
- "-skip" option
  the "-skip" option enables one to skip processing data that has already been processed, it is useful when new sequence files are added to the input directory. For a sequence file, IniMotif checks whether the necessary data files are available in the corresponding output directory. If all data files are available, then this sequence file will not be processed.
- Parameter setting (in paras/para.txt)
  - line 1: Description of the parameters
  - line 2: the input directory (includes sequence files of all batches and all cycles)
  - line 3: the output directory
  - line 4: the location of the background distribution file
  - line 5: the location of the barcode-TfName table file (`fullbarcode \t TfName`)
  - line 6: the minimum motif width
  - line 7: the maximum motif width
  - line 8: the length of the input sequences
- Other settings

  Background base distribution Background base distribution is set in BackDis.txt, in the order of A, C, G, T. It is suggested to be estimated from the original batch (batch '0').

  SELEX experiment Setting The Well_Plate_Barcode.txt files contains the mappings between (well, plate) and barcode. The elements of each line are seperated by tab, i.e. `\t`.
- Function explaination
  After producing necessary data, the java module will call Matlab module to produce the data, then call the perl module for drawing logos, finally it calls the perl module to produce HTML files. The commands will be shown in the STDOUT. The matlab module will produce ".png" figures by default. **Note**: the sequence data of the background batch (batch '0') should always be included in the input directory.

2.2 The matlab module

- How to run the module?
  `cd IniMotifDirectory/CreateFigure`
  `matlab &` (graphical mode) OR `matlab -nodisplay` (non-display mode)
  `drawAllBatches(root_dir, widths, MODE, FIGURE_TYPE)`

  `root_dir` : The root directory is the directory which contains data for all batches, i.e. 'IniMotifOutput' or 'SecondMotif'.

  `widths` : Prespecified motif widths, [6:10] means width 6 to 10, [8:8] means width 8

  `MODE` : `'normal'` for producing new figures, `'skip'` for skipping already existed figures

  `FIGURE_TYPE` : Figure format of matlab output figures, 'png' or 'ps'. By default it is set as 'png'.

  An example: `drawAllbatches('selex/IniMotifOutput', [6:10], 'skip', 'png')`

- Useful function-1: `drawOneCycle(CYCLE_DIR, WIDTH)`
  This function draw SubStrCount-HamDisFig and PosDisFig under one cycle directory. `CYCLE_DIR` is the directory of one cycle, which includes processed data for all different widthes, `WIDTH` specifies the widths under this cycle directory.

- Useful function-2: `drawTrendFig(batch_dir,back_dir,WIDTH,out_dir)`
  This function draws the TrendFig for all barcodes under one batch directory. The batch directory contains different cycles `batch_dir` is the directory of the batch for drawing the Trend figure. `back_dir` is the directory of the initial cycle, i.e. `'IniMotifOuput/0/0'`. `WIDTH` should be an integer here, `out_dir` specifies the output directory for the trend figures.

2.3 The perl module

This module contains two parts: the first part for drawing logos and the second part for producing HTML pages.

- **Logo construction**
  `cd IniMotifDirectory/CreateLogo`
  `perl drawAllLogo.pl -NORMAL . IniMotifOutputDir [skip]`
  This command produces logos for all motifs files in the output, where
  `IniMotifOutuputDir` is the output directory of IniMotif, i.e. directory contains all outputs;
  `skip` will not process a motif file if a logo already exists

- **HTML construction**
  `cd IniMotifDirectory/HTML`
  `perl genPages.pl .  IniMotifOutputDir minWidth maxWidth`
  This command creates all HTML pages for organizing the data, where
  `IniMotifOutuputDir` is the output directory of IniMotif, i.e. directory contains all outputs;
  `minWidth, maxWidth` mean the minimum and maximum motif widths, respectively.

- **A useful script**
  Sometimes we want to view the results in another computer, but the results are quite huge. Thus it is useful if we just extract the figures and HTML pages.
  `cd IniMotifDirectory/HTML`
  `perl downloadPages.pl IniMotifOutputDir downloadDir`
  The above commands download all necessary figures and HTML files for visualization, where
  `IniMotifOutuputDir` is the output directory of IniMotif, i.e. directory contains all outputs;
  `downloadDir` is the directory for storing the figures and HTML files.

3 Result Description

This section explains all the files under the WIDTH directories, which are the most basic unit folders in the IniMotif output directory. Our default directory are these WIDTH directories. Here we use `fullbarcode` to represent any legal full barcode, and `barcode` to represent the central part of a `fullbarcode`.

3.1 Motif & logo

The `fullbarcode`.pwm (in motif folder) files show the motif information. The first line gives `fullbarcode` and its corresponding TF name. The following two lines are the consensus sequence and its reverse complement, followed by the count of the consensus sequence. Then the Alignment matrix, Frequency matrix, Information content-based weight matrix (for calculating free binding energy) are given. Finally the logo matrix for drawing the logo and the background base distribution are provided.

The `fullbarcode`.png files (in logo fodler) show the logos drawn from the logo matrixes in the corresponding `fullbarcode`.pwm motif files. The title of logo consists of the `fullbarcode` and its corresponding TF name.

3.2 SubStrDis

`fullbarcode`.cnt files in this folder provide the counts of the each w-width substring (overlapping) in the original sequence files. The .cnt files are used for drawing the substring changing trend figures. The first column are the indexes of each w-width substring; The second column are the counts of each w-width substring. Note that the count of each w-width substring (exclude palindrome) is the sum of this substring and its reverse complement. If the count of a w-width substring is less than 3 or 1/10 of the expected number of a random sequence, it will not be recorded in the .cnt file

3.3 SubStrDis-HamDis & SubStrDis-HamDis-Figure

The `fullbarcode`.dat files (in SubStrDis-HamDis folder) provide data for drawing the `fullbarcode`.png files in the SubStrDis-HamDis-Figure files.

- Structure of `fullbarcode`.dat
  The first row shows the fullbarcode and TF name. Each of the following rows represents a w-width substring, which is counted from each w-width substring with the least Hamming distance to the consensus sequence on each l-width sequence in the input sequencing file.
  - The first column labels a substring. A subtring with higher Hamming distance than its reverse complement with labeled as 1; otherwise it is labeled as 0.
  - The second column is the count of this w-width substring, note that it is not the sum of both strands this time
  - The third column shows the Hamming distance to the consensus sequence
  - The fourth column is the index of the substring
- Explanation of the SubStrCount-HamDis figure
  As shown in Figure 1, the title shows the fullbarcode of TF `Fli1` is `A_AAGCG_2`, and the total number of dots in this figure is 5861. Each dot in the figure is a 8-width 'binding site', i.e. the 8-width substring with the least Hamming distance to the consensus sequence among all 8-width substrings of a 14-width sequence. The X value of a dot is the Hamming distance to the consensus sequence; the Y value of the dot is the count of the 'binding site' in all the 14-width sequences of the input sequence file. Note that some variations of the position have been added to the dots for better visualization. The sequences with Hamming distance 0 should be the consensus sequences, one of which is the reverse complementary sequence of the other. The text on the up-right corner shows the 14-width sequence with the highest count, followed by the portion of unique 14-width sequences. There are 62274 sequences (14bp) in the input file, 5598 of which contain a binding site.

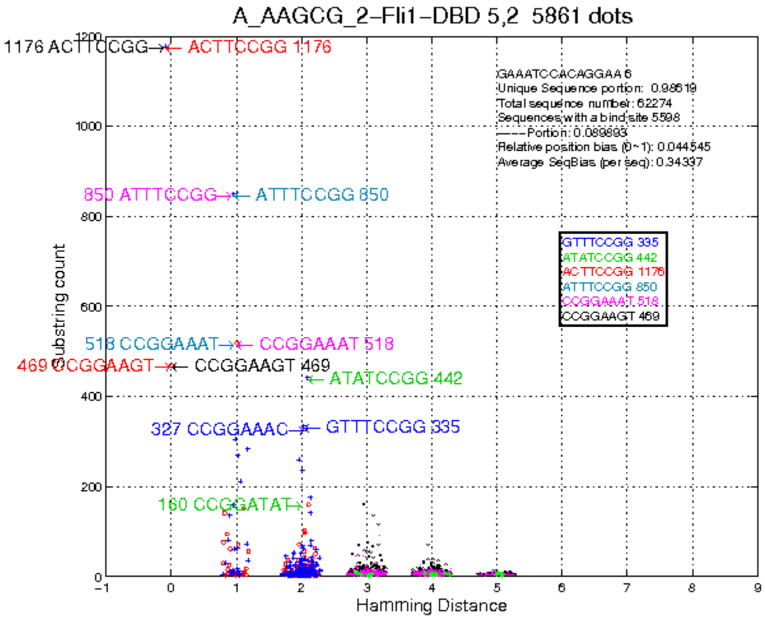

Figure 1: SubStrDis-HamDis-Figure

The relative position bias shows the bias for location of the 8-width binding sites on the 14-width sequences. The 'Average SeqBias' shows the average sequencing bias (See Section 3.5).

3.4 BindSitePosDis & PosDis_Figure

The `fullbarcode`.dis files (in BindSitePosDis folder) contain other information besides data for drawing `fullbarcode`.png files under PosDis_Figure.

- The `fullbarcode`.dis file structure
  - Row 1: the l-width sequence with the highest count in the input sequencing file.
  - Row 2: the count of the highest l-width sequence
  - Row 3: the proportion of unique sequences in the input sequencing file
  - Row 4: the total number of l-width sequences in the input sequencing file
  - Row 5: binding site position distribution on the forward strand
  - Row 6: binding site position distribution on the reverse strand
- Explanation to binding site position distribution figures
  As shown in Figure 2, the blue and red bars shows the position distribution of the binding sites. The distribution is normalized over all postions on both forward and reverse strands. The dashed line means the uniform distribution.

3.5 SeqBias

- Definition of SeqBias
  The `fullbarcode`.bias files under the SeqBias provide information of the sequencing bias. Since DNA is a double strand, we have equal chance to sequence both strands. This equality is supposed to be shown in our data. But in real cases, the numbers of substrings on one strand differ a lot from the other. We use seqbias to indicate the difference. Here we denote all w-width substrings

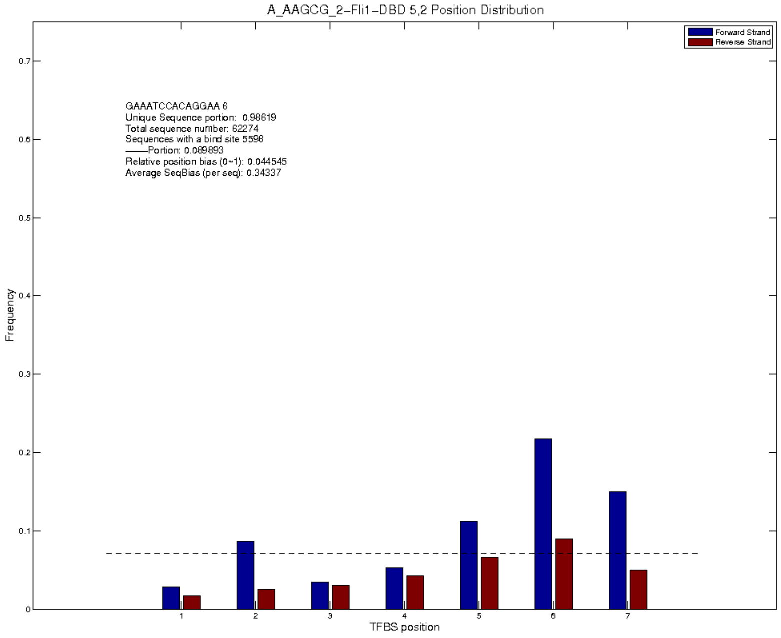

Figure 2: SubStrDis-HamDis-Figure

as A, and the count of a random substring s in A as Count[s]. The following formula calculates the average sequencing bias.

$$seqbias(s) = \frac{\sum_{s \in A} |Count[s] - Count[RevCom(s)]|}{\sum_{s \in A} |Count[s] + Count[RevCom(s)]|} \approx \frac{\bar{p} - \frac{1}{2}}{\frac{1}{2}} \tag{1}$$

where the $\bar{p}$ is the average proportion of a substring in the data. This formula tells how biased the sequencing is from equal equal sequencing state. If both strands are equally sequenced in the data, then there is no bias, which means the value of the formula is 0. If only one strand is sequenced, the value of this formula is 1, which means maximum bias. By experience, if the sequencing bias exceeds 0.5, we think there exists large sequencing bias in the data and needs further investigation.

- Structure of `fullbarcode`.bias file
  The first row is the `fullbarcode` and TF name; The following rows show counts of each w-width substrings and its reverse complement in the input sequencing files. Note that if both counts of a w-width substrings and its reverse complement are smaller than 5 or 1/10 of the expected number of a random w-width substring, the w-width substring will not be recorded.

4 TrendFig

This section gives a simple introduction to the substring trend changing figures.
We count all the 7-width substrings (overlapping) of the input 14-width sequences. Identical repetitive 7-width substrings within one 14-width sequences are excluded. Then the substring counts are normalized to the total substring count. The portion of each 7-width substring is represented as $f = SubStrCount/TotalCount$, as shown in the lower left part of the figure. For better illustration, we also plot the relative portion of each 7-width substring, as shown in the upper part of the figure. Each line in the above figure represents a substring. Top 100 substrings (in different colors) with the highest average counts over all selex rounds, and 200 substrings (grey dashed line) randomly selected

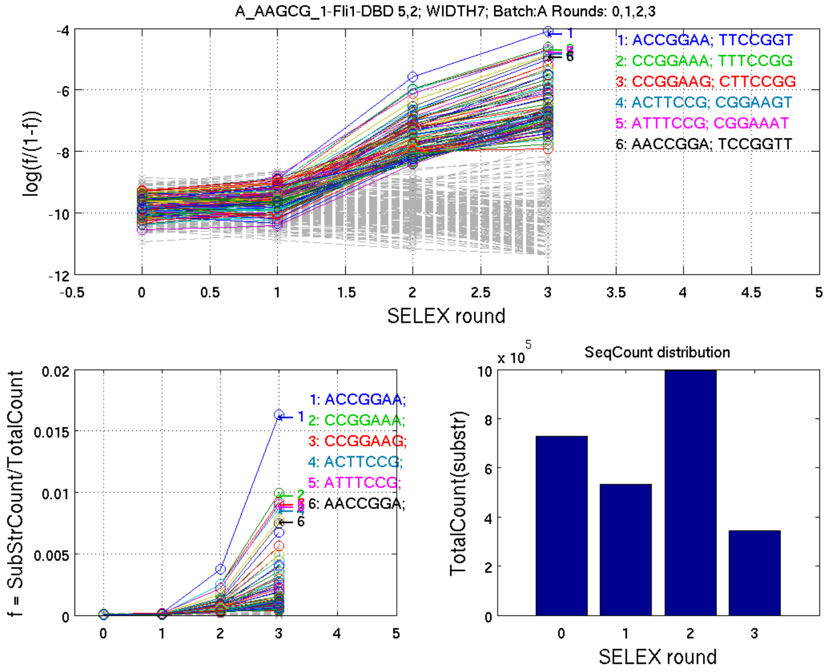

Figure 3: Substring trends

from all substrings are shown in Figure 3. The title includes TF names, barcode, width, and cycles used to construct this figure. The bars show the total count of substrings in each round. To produce the trend figure, the substring distribution of the initial pool (cycle 0) needs to be specified.

5 HTML pages

The HTML pages are quite easy to read. The only difficult part is the `CONSENSUS_1` and `CONSENSUS_2` in the index page. `CONSENSUS_1` means the 8-width consensus sequence (SELEX cycle 3) in the first run of motif discovery. `CONSENSUS_2` means the 8-width consensus sequence (SELEX cycle 3) in the second run of motif discovery. When the 8-width consensus sequence of cycle 3 is not avaiable, `'NNNNNNNN'` will be used. The size of images can be set in `barcode.css` file. Also `Ctrl++` and `Ctrl--` are used to zoom out/in for viewing the HTML pages.

6 Technical specifics

This section introduces the specifics of the IniMotif software.

6.1 Full barcode structure

The `fullbarcode` looks like: `0_ACGTC_0, A_ACGTC_3` ...
The first part shows the sequencing batch; the central part is the barcode; and the last part is the selex cycle. Here we use the regular expression to represent the full barcode: `[0A-Z][A-Z]*_[A-Z]+_[0-9]+`.

6.2 Hierarchy of Input and output

Figure 4 shows the hierarchy of input and output.

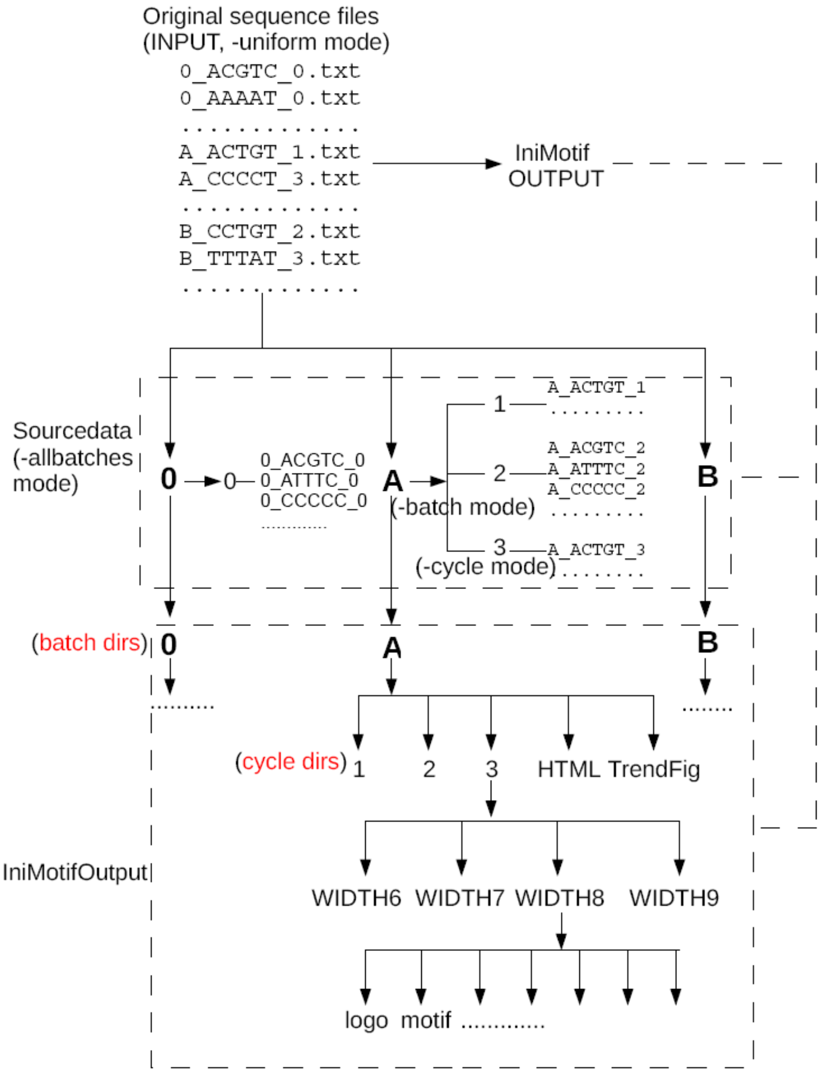

Figure 4: Hierachy of IniMotif Output

We assume all input sequences files are placed in one folder. Then the sequences files are separated into different subfolders according to their batches and cycles, as shown in the 'sourcedata' folder.

The output of IniMotif is organized the same way as the input. The top level is the batch folders, then there are cycle folders under the batch folders, under the cycle folders, the data are organized according to motif widths. For the contents in the width directories, please refer to Section 3.

6.3 Organization of the background distribution file `BackDis.txt`

The background distribution file `BackDis.txt` tells the frequencies of `'ACGT'` in the initial SELEX pool. It should be one line file which consists of 4 numbers between 0 and 1, whose summation should be 1. The numbers are tab separated.

6.4 Organization of the `Barcode-TfName.txt` file

This file constructs the mapping from fullbarcode to TF names. This file contains two columns. The first column should contain the fullbarcode, the second column are their corresponding TF names. The columns are tab separated.

6.5 Organization of the `Well_Plate_Barcode.txt` file

This file constructs the mapping from barcode to (plate, well) of SELEX experiments. Note that we use barcode here, which is the central part of the full barcode. This file contains three columns. The first column shows the well of a SELEX experiment; the second column is the plate; the third column is the barcode. The columns are tab separated.

7 Implementation details

There exist many implementation details, this section lists the most importants ones.

- When calculating the consensus sequence, identical substrings of a l-width sequence will only be counted once
- When calculating the consensus sequence, both a w-width sequence and its reverse complements will be counted (except palindrome)
- When searching w-width binding sites on a l-width sequence, first all substrings with the least Hamming distance are picked; then if the least Hamming distance is less than 2, the picked substrings are chosen as binding sites; finally the count of the l-width sequence is averaged over all chosen binding sites, as the count of each binding site. Thus you may see 0.5 sequences in the final position frequency matrix.
- When producing data for SubStrCount-HamDis figure, substrings with counts less than a threshold (here we set as 3) are not included.
- When producing data for TrendFig, if both counts of a substring and its reverse complent are less than a threshold (here we set as 3), the substring will be discarded.

8 Dependencies of IniMotif

- Operation System: Linux
- Java, JDK 1.5
- Perl 5.0
- Perl, TFBS module (http://tfbs.genereg.net/)
- Matlab 7.0

9 Limitations

This section gives a simple description of the limitations of IniMotif.

All sequences used in this program should be shorter than 31bp.

For a width 12-16 motif discovery task from $2,000 \sim 50,000$ 20bp sequences, apporximatedly 4g mememory is needed for the java module, 2g memeory is needed for the matlab module, and 1g is needed for the logo and HTML pages construction.

Please contact `lu.cheng.cs@gmail.com` if there is any problem.