MPBind: A Meta-Motif Based Statistical Framework and Pipeline to Predict Binding Potential of SELEX-derived Aptamers

Peng Jiang <PJiang@morgridge.org>
Morgridge Institute for Research,
Madison, WI 53707, USA

Contents

Introduction

Aptamers are chemically synthesized short single-stranded DNA or RNA oligonucleotides, which have the ability to bind to a variety of targets, such as small molecules (Yang and Bowser, 2013), proteins (Ng, et al., 2006) and the surface of cells (Cerchia, et al., 2005; Sefah, et al., 2010). It is regarded as a promising way to generate large-scale affinity reagents, which can be an alternative of antibody. To generate high affinity aptamers, it starts with a random oligonucleotide pools. Then those oligonucleotides are evolved through a process called systematic evolution of ligands by exponential enrichment (SELEX), which involves several rounds of selection. Despite of its widespread applications, SELEX derived aptamers are suffering from high false positive rates (Cho, et al., 2010).

MPBind is a meta-motif based statistical framework and pipeline to predict SELEX derived binding aptamers. Briefly, MPBind calculates four kinds of p-values (1-sided) for each motif, representing different features. The p-values are then transformed to Z-scores (Z1, Z4, Z3 and Z4) via $Z = \Phi^{-1}(1-p)$, where Φ is the standard normal cumulative distribution function. For each motif, MPBind used Stouffer's method to combine those four Z-scores into one combined Z-score. For any given aptamer sequence, MPBind uses an n-mer window to scan it. The binding potential is inferred from the combinations of those motifs and estimated by Meta-Z-Score. MPBind provides several options for users, such as whether to use unique reads or redundant reads to train parameters, motif length etc. It also provides data preprocess functions to users, e.g., transform FASTQ/FASTA to plain text format, primer trimming and transform antisense sequences to sense sequences based on primer sequence matching.

Methods

Given a motif length, MPBind will enumerate all possible n-mers (e.g., 4096 motifs for 6-mers) and calculate the frequency of each motif in the random sequence region from each round of SELEX-Seq. Then it will calculate four kinds of statistical tests:

Statistical Test 1:
We assume that high binding motifs should be enriched in the final SELEX round when compared to the control round. The control round can be either the initial library sequencing (R0) or sequencing rounds controlled by PCR cycles without target selection. MPBind will calculate the total number of occurrences of each motif (e.g., TGAGTT) in the final round as well as in the control round. A one-sided Fisher's exact test (right tail) is calculated for each motif. For example, a motif has 100 total occurrences in final round of SELEX-Seq and 50 occurrences in control round. Assuming the total number of scanned unique positions in SELEX-Seq and Control-Seq are 1000 and 800, respectively, a 1-sided P-value is calculated for this motif based on a two by two table ([100, 1000-100] Vs. [50, 800-50]).

Statistical Test 2:
We assume that in the final round of SELEX-Seq, the percentage of reads which contain high binding n-mers should be enriched when compared to the control round. Thus for each motif, we calculate the number of motifs containing reads as well as the number of reads which do not contain this motif in the final SELEX-Seq and in the control round. A similar one-sided Fisher's exact test (right tail) is calculated.

Statistical Test 3:
We assume that the relative frequency of binding motifs should increase with each SELEX round. The relative frequency of each motif is defined as the total number of occurrences of motifs divided by the total number of unique motif positions within all random sequence regions for a given round. A one-sided Spearman correlation is calculated for each motif by the relative motif frequency against the SELEX round numbers.

Statistical Test 4:
We assume that the percentage of reads, which contain binding motifs, should increase with each SELEX round. A one-sided Spearman correlation is calculated for each motif based on the percentage of reads containing this motif against the SELEX round numbers.

For each p-value, we transform it to Z-Score:

$$Z = \Phi^{-1}(1-p) \qquad (1)$$

where Φ is the standard normal cumulative distribution function.

Thus for each motif, we have 4 kinds of Z-scores (Z1, Z2, Z3 and Z4). We further use the Stouffer's method to combine those 4 Z-Scores into one Z-score:

$$Z= \frac{\sum_{i=1}^{k} z_i}{\sqrt{k}} \qquad (2)$$

For any given aptamer sequence, we use an n-mer window to scan it with all potential n-mer motifs. The Meta-Z-Score is calculated as the aggregate of motif level Z-scores for all potential motifs across the aptamer using Stouffer's method (formula (2)). An overview of MPBind method is shown in Figure 1.

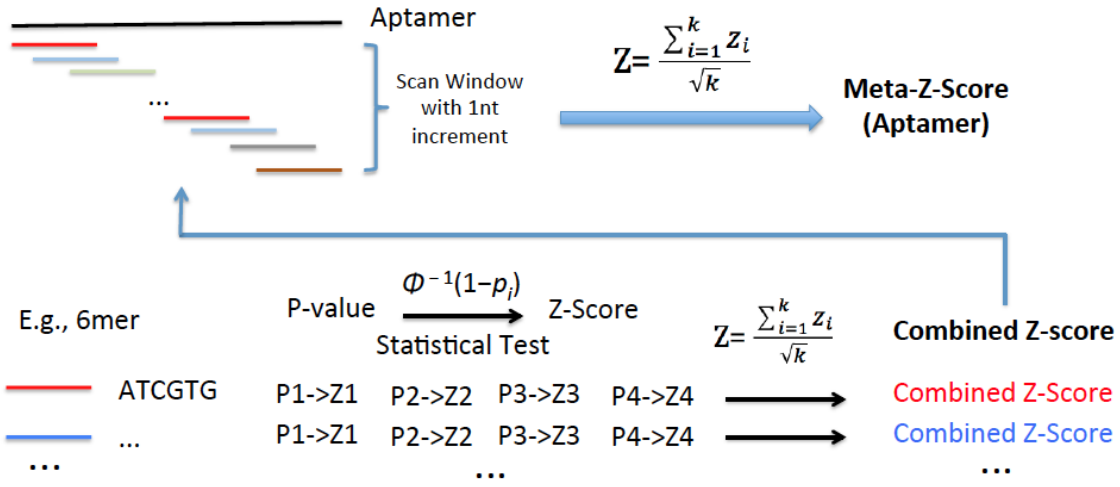

Figure 1. Overview of the MPBind method. A combined Z-Score is determined for each motif, then each aptamer is scanned with each motif-level Z-Score to arrive at a Meta-Z-Score for each entire aptamer.

Prerequisites

Python (version >=2.4.3) and R (version >=2.13.0) are required to be installed.

Installation

- Download MPBind (Linux or MacOS)
- tar –xzf
- Add MPBind directory to the $PATH environment variable (Optional) or you need to type the absolute path of MPBind directory before you run this program.

New Features (v2.1):

(1) This version allows less stringent input. For example, it allows rounds to be defined with commas with or without spaces in between.
(2) This version allows users to run MPBind in any folder (the previous version assumed that user should run program within the data folder).
(3) It gives more information for the screen output (e.g., status checking, running time, and summary of output).

Usage:

Step 1: Preprocess SELEX-Seq reads (Optional)

MPBind requires input files should be in plain text format with each row only contains sense aptamer sequences. To this end, MPBind provides MPBind_ Preprocess.py script to transform raw sequencing reads formats (FASTQ or FASTA) to plain text format. It will also automatically transform antisense reads to sense reads based on matching primer sequences.

Command:

python MPBind_ Preprocess.py < Parameters>

Required Parameters:
-Infile: Input file name
-t: input file format (FASTA or FASTQ)
-Forward_primer: Forward primer sequence
-Reverse_primer: Reverse primer sequence
-primer_max_mismatch: The maximal mismatches allowed to match primers
-Outfile: Output file name

Command Example

```
python MPBind_ Preprocess.py -Infile  Test.fastq –t  FASTQ -Forward_primer
AGCAGCACAGAGGTCAGATG -Reverse_primer TTCACGGTAGCACGCATAGG -primer_max_mismatch 1
–Outfile Test_sequence.txt
```

Step 2: MPBind (training)

MPBind requires the input sequences should be in plain text format

Input file Example (plain text):

```
CTTTGCCACCGGGTTGTAGTTACGGCTGA
CTTTGCCACCGGGTTGTAGTTACGGCTGA
TTATGTTTTTTTTTTTTTTTAATGCCCTG
GTTTTCAAAGAGGCTCGACCTGACTTCTA
GGTTTGCTGAGGTGGGCTCTGTTTAACCT
GCAGGTGTGGTTTGCTGAGGTGGGCCCTG
TTCCCCAATAACATCGTATACCCGCGCCC
```

Command:

Python MPBind_Train.py <Parameters>

Required Parameters:
-R0: Initial library file [plain text format]
-RS: SELEX round files (e.g., R1,R2,R3, …) [Plain text format, separated by comma]
-RC: Control Seq round (No target and just control PCR amplification) <Optional, default=NULL>

-mer: Motif length (e.g., 5,6,7) <default=6>
-U: <1: Unique reads only; 2: Redundant reads only; 3 Both> (default=1)
  # 1: Unique reads only: merged duplicates to one read
  # 2: Redundant reads only: Using all reads
  # 3: Both: MPBind will generate two sub-folders for 'Unique reads only' and 'Redundant reads only', respectively.
-Out: Output file folder <Optional, default=MPBind_Out>

Command Example

```
Python MPBind_Train.py -R0 R0.txt -RS R1.txt,R2.txt,R3.txt,R4.txt,R5.txt,R6.txt,R7.txt -RC Control.txt -nmer 6 -U 3 -Out MPBind_Out_R01234567_Unique_and_Redundant
```

Output files:
It will generate *.train.nmer (e.g., Test.train.6mer) files under the output file folder.

Step 3: MPBind (Prediction)
Command:

```
python MPBind_Predict.py  <Parameters>

Required Parameters:
-Train: *.train.nmer (e.g., Test.train.6mer) files generated by MPBind_Train.py
-Aptamer: Aptamer sequences to be predicted [Plain text format]
-Sort: <TRUE or FALSE> Sort Aptamer sequences based on combined meta-Z-score < default=FALSE>
-Out: Output file
```

Command Example

```
python MPBind_Predict.py -Train Test.train.6mer -Aptamer To_be_predicted.txt -Sort TRUE -Out Predicted_Aptamers.txt
```

Output file (columns):

- Aptamer.Seq: Aptamer sequences (e.g., TTTTGTTTTTTGTTTTCTTTTCCCCCCTC)
- Z1.Scan: Z1-Scores for each scanned position using n-mer window
- Z1.MetaScore: Combined Z-Score using Z1 only
- Z2.Scan: Z2-Score for each scanned position using n-mer window
- Z2.MetaScore: Combined Z-Score using Z2 only
- Z3.Scan: Z3-Score for each scanned position using n-mer window
- Z3.MetaScore: Combined Z-Score using Z3 only
- Z4.Scan: Z4-Score for each scanned position using n-mer window
- Z4.MetaScore: Combined Z-Score using Z4 only
- Z_Combined.Scan: Combined Z-Score for each scanned position using n-mer window (e.g., 8.0,7.4,3.4 …)
- **Z_Combined.MetaScore: Meta-Combined Z-Score**

Citation:

Jiang P., Meyer S., Hou Z., Nicholas E. Propson, Soh H.T., Thomson J.A., Stewart R., MPBind: A Meta-Motif Based Statistical Framework and Pipeline to Predict Binding Potential of SELEX-derived Aptamers. (2014), *Bioinformatics* 30 (18): 2665-2667.