Daianna Gonzalez Padilla
February 9, 2022
Motif search

Evidence for non-randomness of TFBS

Abstract

This work uses the *E.coli* K12 transcription factor TrpR and obtains the **Position Site Specific Matrix PSSM** of the TFBS to accomplish pattern matching based on that matrix. Through the generation of two negative controls lacking biological meaning, here we demonstrate the specificity of the TFBS in the gene upstream regions of *E.coli* genome. The number of matches obtained under these controls hover around 100, but none of those matches correspond to an annotated TrpR regulated gene promoter.

Introduction

Transcription factors TF are regulatory proteins that affect the transcription of a certain set of genes. In prokarya they usually bind to specific sites of gene promoters, the upstream regions of operons. These sites are called **transcription factor binding sites TFBS** and one of the major concerns around them is their efficient and reliable identification so that regulatory networks can be created and analyzed. To accomplish that, matching between known TFBS patterns and biological sequences under study, is based on matrices that take into account each nucleotide probability.

Methodology

First, TFBS of TrpR in *E.coli* K12 and the PSSM are obtained in `RegulonDB`. Secondly, the PSSM is used in `RSAT matrix-scan` tool to find and count the number of matches in the sites of all the promoters throughout *E.coli* genome obtained with `retrieve sequences`, taking 1.00E-04 as the *p-value* threshold used for scanning. Based on the real data, we expect to obtain ~10 sites with at least one match embracing 12 genes and 5 operons.

Then, two not biologically relevant scenarios are created to count the number of matches expected by chance. The first negative control generated with the `random sequences` tool, creates random sequences that mimic the hexanucleotide frequencies of the *E. coli* upstream sequences and with the same length as them. These sequences are then scanned with the PSSM to calculate the number of matches using the previous *p-value* threshold.

For the second negative control `permute-matrix` is used to take the original PSSM in *transfac* format and create two permuted matrices of it in *tab* format, conserving each nucleotide frequency but changing its position by interchanging columns. These new matrices are used to scan all *E. coli* upstream sequences. The same *p-value* threshold is used.

Results

With the real biological sequences of *E. coli* and the original PSSM, there were 98 total matches in 83 genes, including the 100% of the TrpR operons annotated in RegulonDB.

In the first negative control, there were 68 matches within 62 of the 4497 total random sequences.

In the second negative control, with the first permuted matrix there were 130 matches in 126 gene promoters and with the second 88 matches in 81 gene promoters. However, in both cases none of these promoters corresponded to a TrpR regulated gene promoter so there were 0% of RegulonDB operons included.

Though not shown in methodology, when the *p-value* threshold is 1.00E-03, more matches are obtained, around 1000 and with 1.00E-05 there are much fewer matches, around 10.

Discussion

Since the first and second negative controls conserve nucleotide frequencies and the length of sequences and PSSM width, respectively, the changes in the motif predictive power of the models are due to their own features and not because of the data used. In that sense, the capability to predict correct motifs lies in the sequences and the PSSM created with TFBS. That means that TrpR, as well as every TF has specific binding sites in non-coding DNA and thus is possible to predict putative TFBS through the known ones. These results are supported by the fact that TFs interact with DNA through certain aminoacid residues that have particular affinities for some nucleotides or combinations of them, so they do not bind in random sequences of the genome.

Finally, depending on what is more important for each particular study, sensitivity or specificity, the threshold must be defined. In this case, since the number of false positives is high, it is more convenient to set a lower *p-value* threshold in order to make the model more strict.