SUPPLEMENTARY DATA

TFBSshape: an expanded motif database for DNA shape features of transcription factor binding sites

Tsu-Pei Chiu, Beibei Xin, Nicholas Markarian, Yingfei Wang, and Remo Rohs*

Quantitative and Computational Biology, Departments of Biological Sciences, Chemistry, Physics & Astronomy, and Computer Science, University of Southern California, Los Angeles, CA 90089, USA

*To whom correspondence should be addressed. Tel: +1 213 740 0552; Fax: +1 213 821 4257; Email: rohs@usc.edu

Present address: Beibei Xin, State Key Laboratory of Agrobiotechnology and National Maize Improvement Center, Department of Plant Genetics and Breeding, China Agricultural University, Beijing 100193, China

SUPPLEMENTARY METHODS

Procedure for the comparison of unmethylated and methylated DNA sequences

Step 1: Prepare the data for comparison. Paired unmethylated and methylated DNA sequences are required for this comparison. The EpiSELEX-seq dataset (1) provides unmethylated and methylated transcription factor binding site (TFBS) sequences, while the JASPAR (2) and UniPROBE (3) databases include only unmethylated TFBS sequences. Therefore, *in silico* methylation, which turns all CpG dinucleotides to MpG dinucleotides (with 5-methylcytosine bases in both strands) for each TFBS sequence, is performed on the unmethylated TFBS sequences obtained from JASPAR and UniPROBE in order to create paired unmethylated and methylated TFBS sequences.

Step 2: Predict DNA shape features on paired TFBS sequences. For given paired unmethylated sequences S_u and methylated sequences S_m, DNA shape features for sequences S_u and S_m including helix twist (HelT), minor groove width (MGW), propeller twist (ProT), and Roll are calculated using DNAshapeR (4) (an R/Bioconductor package available at https://bioconductor.org/packages/release/bioc/html/DNAshapeR.html).

This prediction results in four pairs of matrices including $\left(M_u^{HelT}, M_m^{HelT}\right)$, (M_u^{MGW}, M_m^{MGW}), $(M_u^{ProT}, M_m^{ProT}), and\ \left(M_u^{Roll}, M_m^{Roll}\right)$. The value changes between each pair of matrices are calculated and denoted as $M_{\Delta HelT}$, $M_{\Delta MGW}$, $M_{\Delta ProT}$, and $M_{\Delta Roll}$, where positive values represent the effect of increasing minor groove width or shape angles, while negative values indicate the effect of decreasing minor groove width or shape angles. Zeros imply no effect on shape upon methylation.

Step 3: Visualize the effect of CpG methylation on DNA shape. This study adopts a box plot representation where the five-number summary, including minimum, first quartile, median, third quartile, and maximum, is derived (as shown in Figure 2D and Supplementary Figure S1B).

Step 4: Perform statistical t-test. The one-sample t-test hypothesis testing with *mean*=0 determines whether there is a significant difference between unmethylated and methylated TFBS sequences at a particular position of TFBSs pertaining to DNA shape. We assume the existence of a difference in terms of $\Delta shape$ as an alternative hypothesis. *P* values calculated for the hypothesis test represent the probability of mistakenly rejecting the null hypothesis when the null hypothesis is true. The significance level is denoted with (*) for $P \leqslant 0.1$, (**) for $P \leqslant 0.05$, (***) for $P \leqslant 0.01$, and (****) for $P \leqslant 0.001$. (n.s.) means non-significant, and (NA) means not applicable. Notably, for the $\Delta shape$ at a particular position that only has zero values or an average equal to zero for interquartile range (IQR), we consider it to be not applicable.

Step 5: Compare CpG-only and MpG-only TFBS sequences. In some cases, a CpG-containing TFBS is not the optimal TFBS for a TF; therefore, oftentimes, the effect of methylation is concealed. In order to solve this issue, TFBS sequences with CpG or MpG dinucleotides are extracted and subjected to the comparison. The comparison steps 2–5 will be repeated for this case and an additional box plot will be generated (as shown in Figure 2D and Supplementary Figure S1B).

Procedure for shape alignment

Step 1: Define alignment basis. The user can choose one or multiple DNA shape features for shape alignment through the TFBSshape user-interface. Assuming that the user selects shape features $X \subseteq \{Buckle, EP, HelT, MGW, Opening, ProT, Rise, Roll, Shear, Shift, Slide, Stagger, Stretch, Tilt\}$ and the size of X is n, the corresponding shape feature matrices for TF1 and TF2 are calculated using DNAshapeR (4) and represented by $2n$ matrices, denoted as $Buckle_{TF1}, Buckle_{TF2}, \ldots, Tilt_{TF1}, and\ Tilt_{TF2}$.

Step 2: Prepare comparison matrices. Each shape matrix, including $Buckle_{TF1}, Buckle_{TF2}, \ldots, Tilt_{TF1}, and\ Tilt_{TF2}$, is averaged in a column and results in a row vector, denoted as $V_{TF1}^{Buckle}, V_{TF2}^{Buckle}, \ldots, V_{TF1}^{Tilt}, and\ V_{TF2}^{Tilt}$. Each row vector pertaining to the type of shape feature is normalized using min-max normalization with the global minimum and maximum values retrieved from the DNAshape pentamer query table (see https://rohslab.usc.edu/tools.html for data). The resulting row vectors are then merged into two matrices, M_{TF1} and M_{TF2}, for TF1 and TF2 respectively. The dimension of each matrix is $n \times m$, where n is the size of X and m is the width of the shape feature matrix. The length of the vectors might be different due to the nature of intra- and inter-base pair parameters. We add zeros to the end of the vectors to keep them the same size.

Step 3: Determine the best alignment through cross comparison of M_{TF1} and M_{TF2}. The comparison starts with an initial number $i = 6$ which stands for the minimum length of continuous base pairs within two TFBSs that are selected for comparison. This setting can avoid an invalid measurement of high similarity resulting from extremely short sequences. In each iteration of comparison, all possible sub-matrices are extracted from M_{TF1} and M_{TF2}. The dimension of the compared sub-matrices is $n \times i$. For example, the comparison starts with $M_{TF1}[0:6]$ against $M_{TF2}[0:6]$, $M_{TF1}[0:6]$ against $M_{TF2}[1:7]$, …, $M_{TF1}[1:7]$ against $M_{TF2}[0:6]$, and all the way to $M_{TF1}[m-7:m-1]$ against $M_2[m-7:m-1]$. The similarities between two sub-matrices are compared with Pearson correlation coefficient (PCC). For the next iteration, the number i is increased by one and the comparison procedure is repeated. For instance, the first comparison in this iteration is based on $M_{TF1}[0:7]$ against $M_{TF2}[0:7]$. The comparison results are saved, and the system will suggest the best alignment with the highest PCC. In some cases, the alignment results in lower PCC but larger ED. We will then choose the alignment with higher PCC.

Procedure for mutation design function

Step 1: Generate all possible mutations. For any given wild-type (WT) DNA sequence s of length L, the user can specify l bases in lower case which are intended to be mutated. If at most k bases are expected to be mutated, there are in total $\binom{l}{k}(4^k - 1)$ possible mutations.

Step 2: Calculate DNA sequence and shape distance.

1) DNA sequence distance L_{seq}

Levenshtein distance measures the dissimilarity between two strings of characters. Here we use Levenshtein distance to quantify the “sequence distance” between two DNA sequences,

which are referred to as WT sequence s and each candidate mutated sequence s'. The distance is the sum of the number of deletions, insertions, or substitutions, required to transform s into s'. For example:

• If s= ACCTGTA and s'= ACCTGTA, then L_{seq}= 0, because no transformations are needed;

• If s= ACCTGTA and s'= ACCTCTA, then L_{seq}= 1, because 1 substitution is needed (G to C at position 5);

• If s= ACCTGTA and s' = AACCTGT, then L_{seq}= 2, because 1 deletion (delete A at the end of s) and 1 insertion (insert A before position 1 of s) are needed.

Levenshtein distance definitions were implemented with a Python module named *editdistance*, which is available at https://github.com/aflc/editdistance.

2) DNA shape distance L_{shape}

DNA shape features, either the four shape features HelT, MGW, ProT, and Roll or a user-selected set of shape features, for s and s' are first calculated and normalized between 0 and 1 by using DNAshapeR (4). $L_{shape}$is defined as the Euclidean distance between $shape_s$ and $shape_{s'}$ as follows:

$$shape_s = (HelT_s^3, \dots, HelT_s^{L-2}, MGW_s^3, \dots, MGW_s^{L-2}, ProT_s^2, \dots, ProT_s^{L-2}, Roll_s^2, \dots, Roll_s^{L-2})$$

$$shape_{s'} = (HelT_{s'}^3, \dots, HelT_{s'}^{L-2}, MGW_{s'}^3, \dots, MGW_{s'}^{L-2}, ProT_{s'}^2, \dots, ProT_{s'}^{L-2}, Roll_{s'}^2, \dots, Roll_{s'}^{L-2})$$

$$L_{shape} = \sqrt{(HelT_s^3 - HelT_{s'}^3)^2 + \cdots + (Roll_s^{L-2} - Roll_{s'}^{L-2})^2}$$

Note that shape features in $shape_s$ and $shape_{s'}$ are one or multiple of HelT, MGW, ProT, and Roll defined by the user. In this version of the TFBSshape database, we assume that the user considers mutations with only base substitutions rather than base deletions and insertions. The reason is that once deletions and insertions happen, the shape profiles for WT and mutated sequence are not aligned, thus resulting in incorrect calculation of L_{shape}. We decide to use current L_{shape} calculation because the current version of the TFBSshape database collects one mode of aligned TFBSs for each TF only and no alternative binding mode with deletions/insertions. Considering that there are certain cases where TFs bind to TFBSs with different lengths, i.e. TFBSs with insertions/deletions, an improved calculation of L_{shape} will be implemented in a future TFBSshape update.

Step 3: Rank candidate mutations by DNA sequence and shape distance. For each candidate mutation, we obtained its sequence distance L_{seq} and shape distance L_{shape} to the WT sequence as mentioned above. Then we rank the list of all mutations according to L_{seq} and L_{shape} separately and calculate their percentiles. Therefore, each mutation has a $(L_{seq}, Percent_{seq})$ pair and a $(L_{shape}, Percent_{shape})$ pair. A larger percentile corresponds to a larger distance. For example, if a particular candidate mutated sequence has a percentile of 80% in shape distance and 30% in sequence distance, it means that this candidate has a sequence distance that ranks the top 20% largest among all candidates, and a shape distance

bigger than 30% of all candidates. This is implemented through the *scipy.stats.percentileofscore* module in python.

Step 4: Select mutations. The user usually has a target as either 'Keep sequence, change shape' or 'Keep shape, change sequence'. 'Keep sequence, change shape' means selecting a mutation that has a small DNA sequence distance but has a large shape distance. When the user specifies a 'Sequence Threshold' T_{seq} and 'Shape Threshold' T_{shape}, retrieved mutations will satisfy $Percent_{seq} < T_{seq}$ & $Percent_{\text{shape}} > T_{shape}$. Accordingly, 'Keep shape, change sequence' will suggest mutations that preserve DNA shape features and satisfy $Percent_{seq} > T_{seq}$ & $Percent_{\text{shape}} < T_{shape}$.

Step 5: Indicate the existence of retrieved mutations. Many determinants affect TF–DNA binding both *in vitro* and *in vivo*. The mutation that is most likely to bind a TF because it preserves most base or shape readout features might not be the best experimental design. We can further check if a selected mutation was detected in previously detected TFBSs. A mutation that was found in an existing binding assay will be labeled with 'yes' in the 'Is in the Pool' column of the final result table.

Given WT sequence s= GTGAgCACGTGgTT, $L = 14, l = 2, k = 2$, 'Keep shape, change sequence', and $Percent_{seq} = 60$, $Percent_{\text{shape}} = 80$,

Supplementary Figure S2 shows an example for step 1 to 5. Here the selected binding site profile is the JASPAR database (2) with TF ID MA0058.1 (MAX), the same example shown in Figure 4 in the main manuscript.

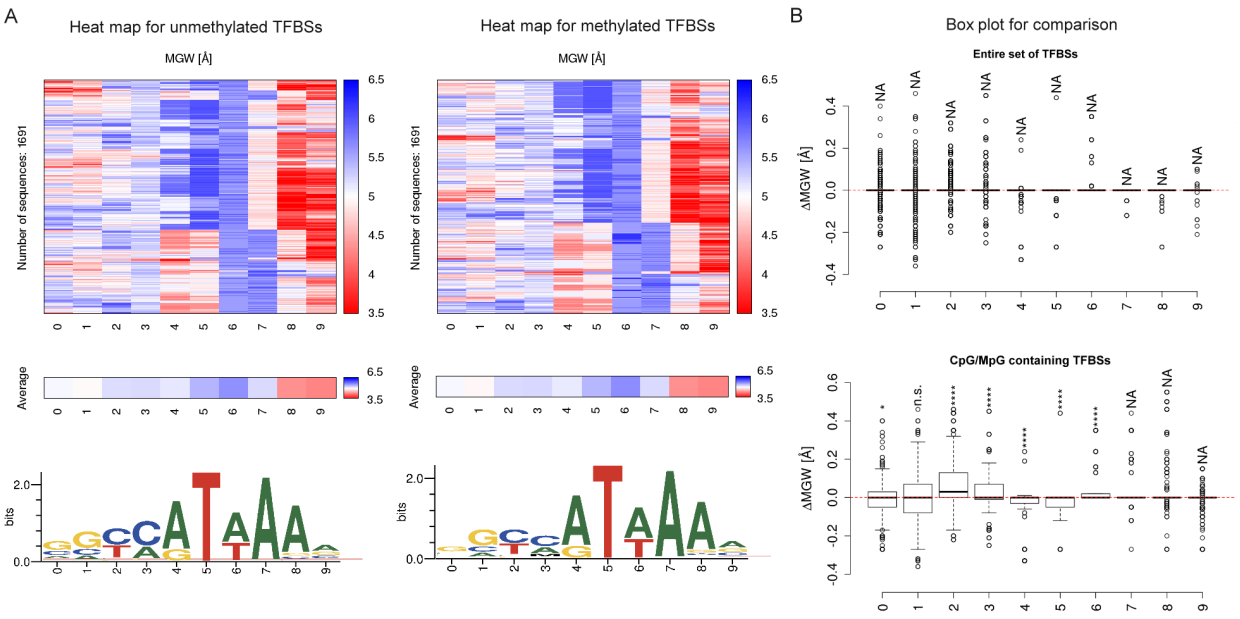
AUTHOR CONTRIBUTIONS

T.P.C. and B.X. conceived the architecture and components of this new TFBSshape release (https://tfbsshape.usc.edu). T.P.C. conceived and implemented the Model-View-Controller architecture. B.X. with the help of T.P.C. collected the data and designed the Model component for the MySQL database. T.P.C. designed the Model for TFBSs comparison and shape alignment, implemented the workflow and controller components, and established the MySQL database server. B.X. designed the Model component for mutation design with the help of Y.W., and implemented the Core table of the database and evaluated the dataset statistics. N.M. with the help of T.P.C. implemented the function for calculating shape features. N.M. with the help of T.P.C. implemented the Viewer components and user interface. T.P.C. wrote the manuscript with the help of B.X. and contributions from all authors. R.R. conceived the original idea of TFBSshape and directed the project, including the writing of the manuscript. Contributions to the original version of TFBSshape and contributions not rising to the authorship level are detailed in the main manuscript's Acknowledgement section.

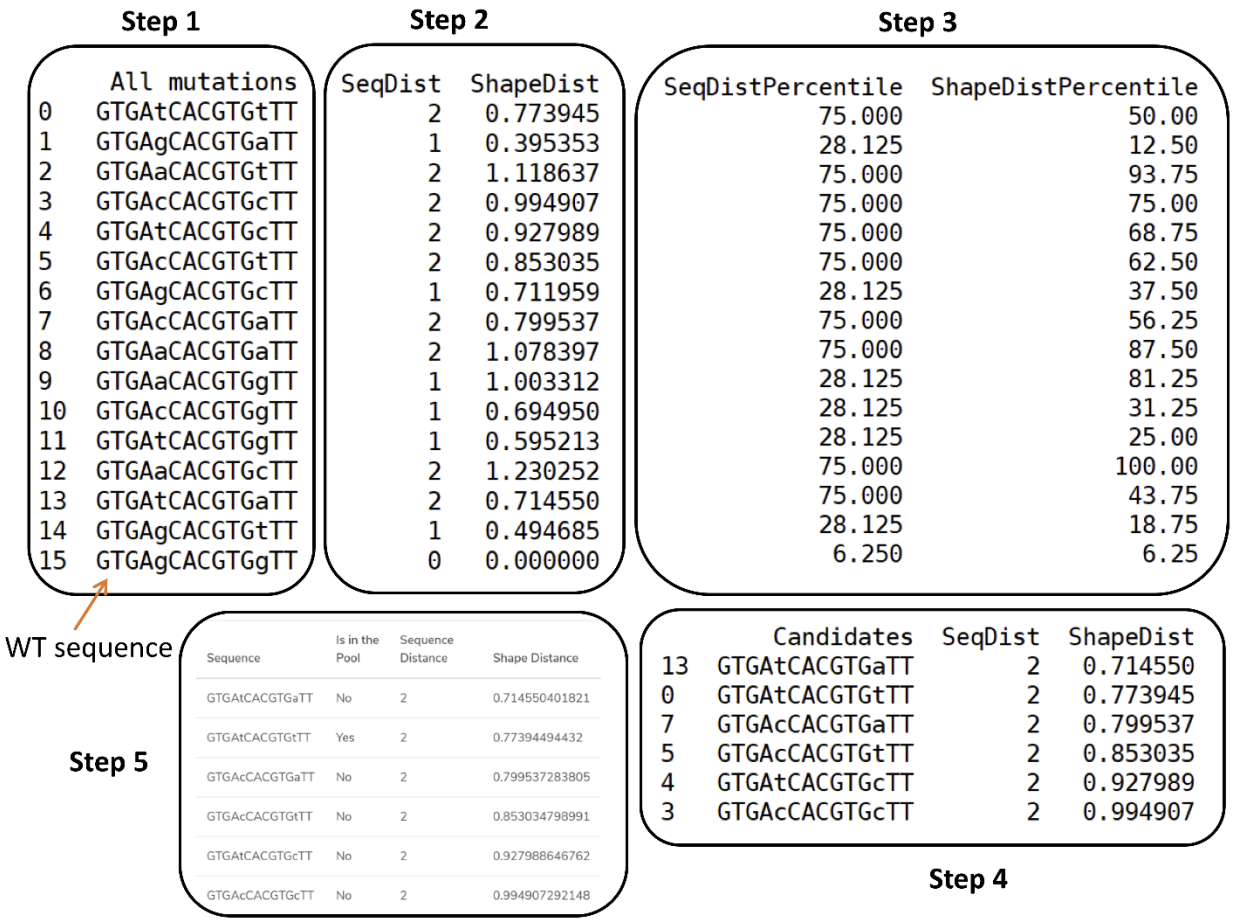
SUPPLEMENTARY REFERENCES

1. Kribelbauer, J.F., Laptenko, O., Chen, S., Martini, G.D., Freed-Pastor, W.A., Prives, C., Mann, R.S. and Bussemaker, H.J. (2017) Quantitative Analysis of the DNA Methylation Sensitivity of Transcription Factor Complexes. *Cell Rep.*, **19**, 2383-2395.
2. Khan, A., Fornes, O., Stigliani, A., Gheorghe, M., Castro-Mondragon, J.A., van der Lee, R., Bessy, A., Cheneby, J., Kulkarni, S.R., Tan, G. *et al.* (2018) JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic Acids Res.*, **46**, D260-D266.
3. Hume, M.A., Barrera, L.A., Gisselbrecht, S.S. and Bulyk, M.L. (2015) UniPROBE, update 2015: new tools and content for the online database of protein-binding microarray data on protein-DNA interactions. *Nucleic Acids Res.*, **43**, D117-D122.
4. Chiu, T.P., Comoglio, F., Zhou, T., Yang, L., Paro, R. and Rohs, R. (2016) DNAshapeR: an R/Bioconductor package for DNA shape prediction and feature encoding. *Bioinformatics*, **32**, 1211-1213.

SUPPLEMENTARY FIGURES



Supplementary Figure S1. An example of the structural profile derived from the TFBS datasets where the majority of the TFBS sequences does not contain a CpG dinucleotide. (A) The DNA logos for HOXA1 TFBS sequences derived from UniPROBE (UP00264) show the CpG-containing TFBS is not the optimal binding site, and thus, (B) the box plot in the top panel shows a concealed methylation effect when considering the entire set of sequences where the majority of them does not contain a CpG dinucleotide. When comparing the TFBS sequences containing only CpG and MpG dinucleotides, the significant difference between unmethylated and methylated TFBSs in terms of shape can be observed from the box plot in the bottom panel.



Supplementary Figure S2. An example for the mutation design process for WT sequence s= GTGAgCACGTGgTT.