Supplementary Methods

Method comparison

The performance of BCRANK was compared to DRIM and MDscan on the USF1 data. To evaluate the predictions we considered how frequent the E-box (CACGTG), the previously established USF1 binding sequence, occurred in the predicted sites. We also investigated how many of the bindings were in the 2518 stringent and 3771 relaxed USF1 regions as defined in our previous study.

The results are summarized in supplementary Table S1. BCRANK found CACGTGAC as top motif regardless of the length parameter, and that sequence was found at 1757 sites. Of these, 1194 (68%) and 1529 (87%) were included in the stringent and relaxed sets respectively. For motif length 6, MDscan predicted 7724 sites, all of them identical to the E-box. 54% and 78% were in the stringent and relaxed set, indicating a higher false positive rate than for BCRANK. For length 8 and 10, MDscan predicted 5618 and 2270 sites but only about 25% of them contained the E-box. DRIM detected ACGTGA as top consensus for length 6 and 81% of the matching sites had the E-box. The results for length 8 were identical to what we found by BCRANK. For length 10, ACGNGAC was reported, with 65% of the sites matching the E-box.

The BCRANK algorithm

BCRANK is a method that takes a ranked list of genomic regions as input and outputs short DNA sequences that are overrepresented in some part of the list. The algorithm was developed for detecting transcription factor (TF) binding sites in a large number of enriched regions from high-throughput ChIP-chip or ChIP-seq experiments, but it can be applied to any ranked list of DNA sequences. The BCRANK algorithm is available from Bioconductor (www.bioconductor.org), where illustrative examples can be found.

BCRANK uses a heuristic search strategy. First a score is computed for an initial short consensus sequence, typically selected at random. The score takes into account both the number of consensus occurrences and the rank of the genomic regions. Then all consensus sequences in a neighborhood of the start guess are evaluated and the one with highest score is kept as the starting point for the next iteration. Once the algorithm finds a local optimum, the algorithm is terminated and the locally optimal consensus is reported as a result. In order to increase the chance of detecting the globally optimal solution, the algorithm may be restarted several times using different random starting points. Alternatively, BCRANK can be used for assigning scores to previously established consensus sequences. The sections below describe in more detail how the neighborhood, scoring function, start guess and penalties are implemented. In this document, the word 'neighborhood' is used instead of 'similar consensus sequences' that was used in the main text.

Neighborhood

In BCRANK, all consensus sequences are represented by IUPAC nucleotide symbols. The neighborhood of a consensus sequence S consists of all consensuses that can be generated from S by first adding one IUPAC letter N (representing any nucleotide) to either side of S and then flipping any base to any other IUPAC symbol. Since there are 15 symbols in total, a sequence of length l will have $14 \cdot (l+2)$ neighbors. After each search step any flanking Ns are removed from the highest scoring sequence in the neighborhood. The removal and additions of flanking Ns allows the algorithm to shorten and extend the predicted binding sites.

Scoring function

The score tells whether a given consensus sequence is overrepresented in some part of the ranked list or not. Starting from N ranked regions and a consensus sequence c, a binary vector of size N is created, with 1 at position i if c is occurring in sequence number i, and 0 if not. The reverse complement of c is also allowed to match. Then the cumulative sum of the match vector is computed and stored in a vector called A. The A-vector tells where in the ranked list most occurrences are located.

To compute a score, A is compared to what it would look like if the genomic regions were randomly ordered. Therefore a large number R of random orderings of the input regions are generated, and a corresponding vector A_j is computed for each re-ordering $1 \leq j \leq R$ as above. For each j, the difference D_j between A_j and A is estimated by the area between the lines. When calculating D_j, the A and A_j vectors are first scaled so they range between 0 and 1.

D_j will be close to zero when the consensus occurrences are distributed as expected by random sampling. If on the other hand all D_j are far off from zero, c is biased towards some part of the list. Therefore the score is calculated as the t-statistic T for the D_j being drawn from a distribution centered around zero. Consensus sequences that are biased towards some part of the list will thus get high scores whereas consensuses with no bias will get low scores. Moreover, consensuses that are matching just a few regions will not get a high T even if it is matching only among the top ranked regions. This is because there will be a high variation within the D_j values which will result in a low T.

Penalties

The t-statistic gives consensus sequences that are biased towards some part of the list. But there may be other issues to take into account if the aim is to detect TFBS from ChIP-chip or ChIP-seq data. Therefore, BCRANK implements two optional penalties, *P1* and *P2*, with values between 0 and 1. The final scoring function is defined as: $score = T \cdot P1 \cdot P2$. If a penalty is not used it will be set to 1.

- *P1* - Penalty on non-specific bases. Let l be the length of the consensus sequence and b the total number of fixed bases (A, C, G, T) in the sequence. If there are no fixed bases, b is set to 0.5. The penalty is then defined as $P1 = b / l$.

- *P2* - Penalty on repetitive motifs. Let r_n, $n \in \{1,2\}$ be the number of input DNA regions that contain at least n occurrences of the consensus. Then $P2 = 1 - (r_2 / r_1)$.

Start guess

In case the algorithm is used for *ab inito* search, the initial guess is a randomly generated consensus of a specified length, with 10 bases as default. Multiple restarts with different random start guesses are usually required to increase the chance of finding the globally optimal solution.

Additional information

- The algorithm randomly re-orders the data when the score is calculated. This implies that the same consensus sequence will get slightly different BCRANK scores in the same data when run with different re-orderings.

- The algorithm performs a breadth-first search, meaning that the highest scoring neighbor in the neighborhood is selected in each search step.

- The algorithm keeps track of all consensus sequences that have already been tested so the same sequence is not visited twice when performing a search.