Journal of Bioinformatics and Computational Biology **2** (2004): 289-307

Journal of Bioinformatics and Computational Biology

BINDING MATRIX: A NOVEL APPROACH FOR BINDING SITE RECOGNITION

Jan T. Kim

Jan E. Gewehr*

Thomas Martinetz

Institute for Neuro- and Bioinformatics, University of Lübeck
Seelandstraße 1a, D-23569 Lübeck, Germany
email: kim@inb.uni-luebeck.de

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

Recognition of protein-DNA binding sites in genomic sequences is a crucial step for discovering biological functions of genomic sequences. Explosive growth in availability of sequence information has resulted in a demand for binding site detection methods with high specificity. The motivation of the work presented here is to address this demand by a systematic approach based on Maximum Likelihood Estimation.

A general framework is developed in which a large class of binding site detection methods can be described in a uniform and consistent way. Protein-DNA binding is determined by binding energy, which is an approximately linear function within the space of sequence words. All matrix based binding word detectors can be regarded as different linear classifiers which attempt to estimate the linear separation implied by the binding energy function. The standard approaches of consensus sequences and profile matrices are described using this framework.

A maximum likelihood approach for determining this linear separation leads to a novel matrix type, called the binding matrix. The binding matrix is the most specific matrix based classifier which is consistent with the input set of known binding words. It achieves significant improvements in specificity compared to other matrices. This is demonstrated using 95 sets of experimentally determined binding words provided by the TRANSFAC database.

Keywords: transcription factor; binding site; weight matrix; maximum likelihood

1. Introduction

All processes which implement biological functions based on genomic sequence information require that DNA-binding proteins execute certain functions on the genome at specific locations, called *binding sites*. A DNA-binding protein has to bind at its

*present address: Ludwig-Maximilians-Universität München, Institut für Informatik, Lehr- und Forschungseinheit für Bioinformatik, Amalienstraße 17, D–80333 München, Germany

binding sites with a sufficient strength. On a genome, there also exist sites where the protein must not bind, because execution of its function on such positions would be detrimental.

The DNA-binding domain of the protein makes direct contact to a segment of DNA. This local sequence at a contact site, which is typically between 10 and 20 base pairs long, is called a *sequence word*. The sequence word at site i largely determines whether i is a binding site for the protein or not.

The bioinformatic challenge is to determine which sites on a genomic sequence are binding sites. As a data base, usually only a small set of experimentally verified binding sites is available. Negative examples, i.e. non-binding sites, are usually not available. From the few known binding words, a word model, such as a consensus sequence or a profile matrix[1,2] has to be constructed. With this model, unknown genomic sequences are then scanned, and each site is classified as either a binding site or a non-binding site.

Specificity of such classifiers becomes increasingly important as the amount of genomic sequence which is electronically available for bioinformatic analysis continues to grow rapidly. The binding matrix, which we present in this contribution, results from a maximum likelihood approach. It provides the highest specificity which, given a set of known binding words, can be achieved with a matrix based classifier, and it constitutes a significant improvement with respect to other matrix based classifiers for binding words.

2. Methods

2.1. *Orthogonal Coding of Sequences*

Let $\mathcal{A} = \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$ denote the alphabet of base pairs. Orthogonal coding is a mapping from $\mathcal{A}^L$ to $\mathbb{R}^{4L}$. A sequence of L nucleotide symbols is represented by a vector $\mathbf{w} = (w_{\mathrm{A},1}, w_{\mathrm{C},1}, w_{\mathrm{G},1}, w_{\mathrm{T},1}, w_{\mathrm{A},2}, \ldots, w_{\mathrm{G},L}, w_{\mathrm{T},L}) \in \mathbb{R}^{4L}$, where $w_{b,l} = 1$ if b is the l-th symbol in the sequence and $w_{b,l} = 0$ otherwise. Thus, each symbol is represented by a quartet of components of $\mathbf{w}$. Within each quartet, exactly one component is 1. By this construction, quartets representing different symbols are orthogonal, hence the name "orthogonal coding". As an example, the orthogonal coding of the sequence `GAT` is $(0,0,1,0,1,0,0,0,0,0,0,1)$. The set of all words of length L is denoted by $\mathcal{W} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_K\}$. The cardinality of $\mathcal{W}$ amounts to $K = 4^L$.

2.2. *The Structure of Word Space*

The points representing orthogonal codings of sequence words are arranged in a specific structure within $\mathbb{R}^{4L}$. Firstly, all words lie on the surface of a $4L$-dimensional hypersphere, evidenced by $||\mathbf{w}||^2 = L$. Secondly, all words are located within a $3L$-dimensional linear subspace of $\mathbb{R}^{4L}$, called the *continuous sequence space*[3], since $\forall\ l, 1 \leq l \leq L : \sum_{b \in \mathcal{A}} w_{b,l} = 1$ is valid. The intersection of the $4L$-dimensional

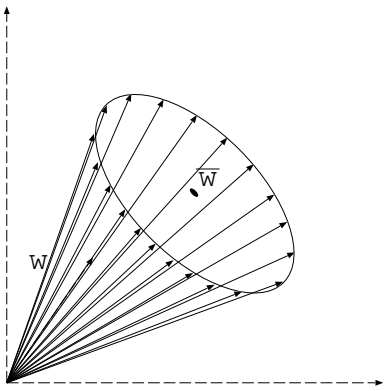


Fig. 1. A low-dimensional sketch of the word space structure. Dashed lines show Euclidean basis vectors of a three-dimensional space. Individual words $\mathbf{w}$ are depicted by solid arrows. The center of the hypersphere outlined by the words, $\overline{\mathbf{w}}$, is depicted by a dot.

hypersphere and the continuous sequence space is a $3L$-dimensional hypersphere. Its center is given by $\overline{\mathbf{w}} = (1/4, 1/4, \ldots, 1/4)$, as shown by observing that $||\mathbf{w} - \overline{\mathbf{w}}||^2 = 3L/4$ is valid for all $\mathbf{w} \in \mathcal{W}$. For symmetry reasons, the words are homogeneously distributed on the surface of this $3L$-dimensional hypersphere. Thirdly, $\overline{\mathbf{w}}^T(\mathbf{w}_1 - \mathbf{w}_2) = 0$ holds for all $\mathbf{w}_1, \mathbf{w}_2 \in \mathcal{W}$, which means that the continuous sequence space is orthogonal to $\overline{\mathbf{w}}$. A sketch of this structure is displayed in Fig. 1.

2.3. *Scoring of Words and Linear Classification*

Given a *scoring vector* $\mathbf{m} \in \mathbb{R}^{4L}$, a real-valued score $S_{\mathbf{m}}(\mathbf{w})$ can be assigned to each word of length L by computing the inner product

$$S_{\mathbf{m}}(\mathbf{w}) := \mathbf{m}^T\mathbf{w} = \sum_{b\in\mathcal{A}}\sum_{l=1}^{L} m_{b,l}w_{b,l}. \tag{1}$$

The components of a scoring vector can also be arranged in a $4 \times L$ table which is frequently called a "matrix" in the literature. It should be noted that such scoring tables are not used as matrices in a mathematical sense. In fact, scoring a word with a "matrix" is the same as scoring it with (1). Throughout this paper, we use the term "matrix" solely for consistency with the literature.

In conjunction with a threshold value Θ, scoring can be used to define a hyperplane in $\mathbb{R}^{4L}$. Words which satisfy the inequality

$$\mathbf{m}^T\mathbf{w} - \Theta \geq 0 \tag{2}$$

lie on one side of the hyperplane (equality indicating points exactly in the plane), while words which do not meet this criterion are located on the other side. Geometrically, $\mathbf{m}$ is a normal vector to the hyperplane and the expression $\mathbf{m}^T\mathbf{w} - \Theta$

measures the distance of $\mathbf{w}$ from the plane in multiples of $||\mathbf{m}||$. Thus, a linear classifier in $\mathcal{W}$ is parameterized by $\mathbf{m}$ and Θ. The number of words above the hyperplane specified by $\mathbf{m}$ and Θ, i.e. those which satisfy (2), is denoted by $k_{\mathbf{m},\Theta}$.

2.4. *Biological Function and Binding Site Recognition by a Transcription Factor*

For many DNA binding proteins, it has been shown experimentally that the free energy of a transcription factor binding to a word $\mathbf{w}$ can be approximated by a sum of independent contributions provided by the individual base pairs in the word[4,5,6,7,8,9]. Let $e_{b,l}$ denote the amount of binding energy contributed by base pair b when present at position l. These components can be aggregated into a $4L$-dimensional vector $\mathbf{e}$, which we will refer to as the energy vector or the energy matrix. Under the assumption of additivity, binding free energy for a word $\mathbf{w}$ can be calculated by

$$E(\mathbf{w}) = \mathbf{e}^T\mathbf{w}. \tag{3}$$

Transcription factors regulate the rate of transcription initiation, which is a complex process that involves a large number of proteins. As a result of this, there exists a binding energy threshold E_*. If $E(\mathbf{w}) < E_*$, the availability of the transcription factor at the binding site determines the rate of transcription initiation. Thus, if the factor activates transcription and $E(\mathbf{w}) \ll E_*$, hardly any transcription takes place; the gene is "switched off". If $E(\mathbf{w}) \geq E_*$, availability of the transcription factor at the site is high, and the rate of transcription initiation is largely determined by other steps in the complex initiation process. All words meeting this criterion are called *binding words*, denoted by $\mathcal{V} = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{k_*}\} = \{\mathbf{w} \in \mathcal{W} : E(\mathbf{w}) \geq E_*\}$. The cardinality of the set of binding words is denoted by k_*. The inequality

$$\mathbf{e}^T\mathbf{w} - E_* \geq 0 \tag{4}$$

describes a linear separation of $\mathcal{W}$. Words above the hyperplane $(\mathbf{e}, E_*)$ are able to implement a functional binding site and hence are called *binding words*, while the other words are called *non-binding words*.

2.5. *The Machine Learning Task*

From a machine learning perspective, the experimentally determined binding words $\hat{\mathcal{V}} = \{\hat{\mathbf{v}}_1, \hat{\mathbf{v}}_2, \ldots, \hat{\mathbf{v}}_n\} \subseteq \mathcal{V}$ represent a set of training data. The problem of binding site detection is to estimate the energy vector $\mathbf{e}$ and the energy threshold E_*, and thus to deduce a binding word model from the training data. The set of training data is very small, i.e. $n \ll k_*$. Therefore, complex classification approaches in which models have a high-dimensional parameter space are difficult to parameterize, as overfitting can easily occur. Negative examples, i.e. experimentally determined non-binding words, are not available, necessitating a one-class classification approach.

The binding matrix, introduced in Section 3 below, is a linear classifier and thus its complexity is appropriately restricted. Considering the biological function of a transcription factor within the word space structure generated by orthogonal coding leads to a one-class maximum likelihood approach which systematically maximizes specificity.

2.6. *Profile Matrix*

The profile matrix[1,2,10,11,12] is defined as a $4 \times L$ table with components $p_{b,l}$ containing the occurrence frequency of base b at position l within the set of words experimentally found at binding sites. In orthogonal coding, the presence of b at position l in a binding word $\hat{\mathbf{v}}$ is represented by $v_{b,l} = 1$, and $v_{b',l} = 0$ denotes that $b' \neq b$. Thus, the scoring vector equivalent to the profile matrix is

$$\mathbf{p} = \frac{1}{n} \sum_{i=1}^{n} \hat{\mathbf{v}}_i. \tag{5}$$

$\mathbf{p}$ is an element of the continuous sequence space. The profile matrix represents the arithmetic mean of the experimentally determined binding words. Scoring a word with $\mathbf{p}$ according to Eq. (1) is equal to computing the score with a profile matrix, which is traditonally described as adding up the values which the individual symbols in the word "select" from the matrix[2].

2.7. *Logarithmic Profile Matrix*

The logarithmic profile matrix, denoted by $\mathbf{g}$, consists of the logarithmized occurrence frequencies. According to an analysis by Berg und von Hippel[13], based on statistical mechanics, the logarithms of the base frequencies should be proportional to the binding energy contributions of the bases[2].

Differently from the plain profile matrix, it is not practical to use the frequencies from the experimentally determined binding words directly because typically, several of these frequencies amount to 0 due to a small sample size. Therefore, a small sample correction is required[13], according to which the components of $\mathbf{g}$ are computed as

$$g_{b,l} = \log\left(\frac{n \cdot p_{b,l} + 1}{n + 4}\right). \tag{6}$$

2.8. *Consensus Sequence*

The consensus sequence[1,2], historically the earliest binding word model, results from finding a word with maximal similarity to the known binding words. The score of a word is the number of positions occupied by matching characters in the consensus, and binding sites are predicted where this score exceeds some threshold. More than

one acceptable symbol may be specified per position. The consensus sequence can be represented by a scoring vector $\mathbf{c}$ with components defined by

$$c_{b,l} := \begin{cases} 1 & \text{if } p_{b,l} \geq t_l \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The components $p_{b,l}$ are taken from the profile matrix (Eq. 5). By appropriately setting the t_l values, all variants of the consensus sequence concept can be obtained. In this contribution, we use $t_l = \max_b p_{b,l}$, which yields the "strict" consensus sequence.

3. The Binding Matrix

3.1. *Maximum Likelihood Approach*

Following the preceding introduction, a binding site has to bind to the transcription factor with a binding energy of at least E_*, and consequently, the major evolutionary constraint on a binding site is that it must contain a binding word. If we disregard all additional evolutionary constraints and assume that additivity holds (see Section 2.4), the probability of encountering word $\mathbf{w}$ at a binding site is

$$P(\mathbf{w}|\mathbf{e}, E_*) = \begin{cases} \frac{1}{|\mathcal{V}|} = \frac{1}{k_*} & \text{if } \mathbf{e}^T\mathbf{w} \geq E_* \\ 0 & \text{if } \mathbf{e}^T\mathbf{w} < E_*. \end{cases} \tag{8}$$

Based on this distribution, the probability to observe $\hat{\mathcal{V}}$ as the set of empirically determined binding words, given a transcription factor described by $(\mathbf{e}, E_*)$, amounts to

$$P(\hat{\mathcal{V}}|\mathbf{e}, E_*) = \prod_{i=1}^{n} P(\hat{\mathbf{v}}_i|\mathbf{e}, E_*). \tag{9}$$

Assuming that $\hat{\mathcal{V}}$ does not contain any non-binding words (e.g. as a result of experimental classification errors), we obtain

$$P(\hat{\mathcal{V}}|\mathbf{e}, E_*) = \frac{1}{k_*^n}. \tag{10}$$

This enables a maximum likelihood approach in which the parameters $(\mathbf{e}, E_*)$ are estimated by those values $(\mathbf{q}, \Theta)$ for which (9) becomes maximal. $\mathbf{q}$ is called the *binding matrix*. An algorithm for computing $\mathbf{q}$ is described in the next section.

3.2. *Computation of the Binding Matrix*

As motivated above, the binding matrix $\mathbf{q}$ is defined by

$$\begin{aligned} (\mathbf{q}, \Theta_{\mathrm{BM}}) &= \arg\max_{(\mathbf{m},\Theta)} P(\hat{\mathcal{V}}|\mathbf{m}, \Theta) \\ &= \arg\max_{(\mathbf{m},\Theta)} \frac{1}{k_{\mathbf{m},\Theta}^n} \end{aligned} \tag{11}$$

$k_{\mathbf{m},\Theta}$ decreases monotonically as Θ grows, therefore, Θ should be maximized in order to minimize the denominator in (11). However, (10) is valid only if all exper-

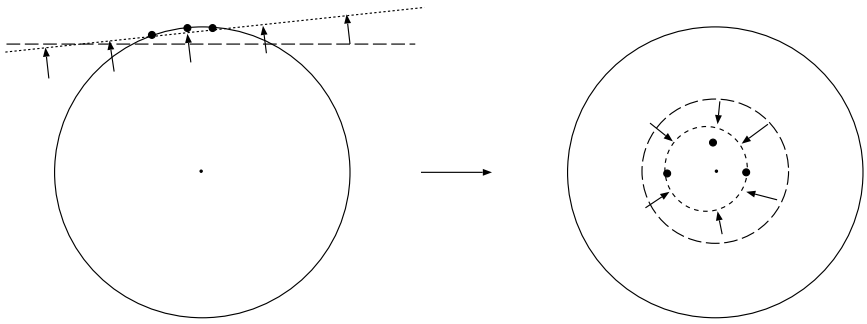


Fig. 2. The maximum likelihood estimation of the spherical binding word distribution looks for the sphere with the smallest radius which still contains all data points. In our case this is equivalent to looking for the plane which "cuts off" the smallest segment while keeping all data points on this segment.

imentally determined binding words score at least Θ. Otherwise, $P(\hat{\mathcal{V}}|\mathbf{m},\Theta) = 0$ is valid, as at least one of the factors in the product in Eq. (9) becomes zero. This implies a set of n constraints. $(\mathbf{q}, \Theta_{\mathrm{BM}})$ specifies the hyperplane with maximal Θ, i.e. which has the maximal distance from $\overline{\mathbf{w}}$ under the condition that all experimentally determined binding words are on or above the hyperplane. This is illustrated by Fig. 2. Based on these considerations, the binding matrix can be obtained by solving the problem of maximizing Θ under the constraints

$$||\mathbf{q}||^2 = 1 \tag{12}$$

$$\forall \hat{\mathbf{v}} \in \hat{\mathcal{V}} : \mathbf{q}^T \hat{\mathbf{v}} \geq \Theta \tag{13}$$

The quadratic constraint (12) ensures that $\mathbf{q}$ is a normal unit vector of the hyperplane, preventing a trivial and undesired maximization of Θ by minimization of $||\mathbf{q}||$. The linear constraints (13) ensure correct classification of the known binding words.

The optimization approach for the binding matrix can figuratively be described as "pushing the hyperplane of separation as far away from the sphere center as possible". The binding matrix approach is similar to the one-class approaches with support vector machines[14]. Differently from the support vector machine, the binding matrix approach does not involve a projection of the data into a high-dimensonal feature space. The specific choice of the continuous sequence space results from the objective of estimating the binding energy matrix, as described above.

3.3. *Implementation*

The binding matrix algorithm has been implemented as a C++ function. This function invokes the program `ampl`[15] for solving the constrained optimization problem, using `loqo`[16] as the solver. In addition, functions for computing profile matrices, logarithmic profile matrices and consensus sequence matrices have been developed

in C++. Programs for conducting the analyses described below are based on this library. A web-based interface for computing binding matrices from input binding word set is available at `http://www.inb.uni-luebeck.de/bmatrix/`.

3.4. *The Binding Matrix is the Consistent Matrix with Maximal Specificity*

Wolff *et.al.*[17] have introduced the criterion of consistency, i.e. the property of a (matrix based) classifier to correctly classify all known binding words. Consistency can always be achieved by choosing a sufficiently low threshold setting. However, such a threshold setting may result in a classifier with unacceptably low specificity.

The linear constraints which are applied in computing the binding matrix $\mathbf{q}$ ensure consistency while maximization of Θ effectively minimizes the number of words which are classified as binding words. Thus, the binding matrix provides the matrix based classifier which, under the constraint of being consistent with all known binding words, results in the minimal number of words classified as binding words. All other matrices classify a larger set of words as binding words when employed with the maximal threshold setting compatible with consistency.

The binding matrix is, by construction, the consistent matrix (*sensu* Wolff *et.al.*) with maximal specificity. If we assume that binding energy can be reasonably approximated by a sum of additive energy contributions of individual base pairs (see Section 2.4), this means that there exists an approximately spherical area on the $3L$-dimensional sphere in which words with high binding energies are concentrated. On average, this area will be included in the set of binding words predicted by the binding matrix. All other consistent matrices will recognize additional words which contain an elevated fraction of false positives with these matrices.

In summary, the binding matrix approximates the maximally specific consistent matrix if the binding energy distribution can be approximated by an additive model. If additivity does not apply, matrix based binding word classifiers in general, including (of course) the binding matrix, are not suitable.

4. Comparative Assessment of Specificity

The performance of the binding matrix was tested by comparison with the alternative methods described above, the profile matrix, the logarithmic profile matrix, and the consensus sequence. The TRANSFAC database[18], version 5.2.1 was used as a database for the performance tests. TRANSFAC provides matrices, associated with sets of sequence fragments containing the words from which the matrices were compiled. These fragments were used to assemble sets of known binding words. TRANSFAC provides the position of the binding word within each fragment. However, in some instances, a fragment only contains a part of the binding word. We excluded such partial binding words in all our analyses. There are 95 matrices for which a set of $n \geq 5$ complete binding words is provided. These 95 binding word

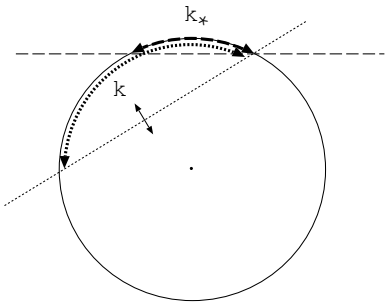


Fig. 3. The hyperplane defined by $(\mathbf{e}, E_*)$ cuts off a segment containing k_* binding words. The hyperplane defined by a classifier $(\mathbf{m}, \Theta)$ can be shifted along $\mathbf{m}$ by changing the threshold value Θ. The threshold value at which the hyperplane cuts off the smallest segment including all k_* words cut off by $(\mathbf{e}, E_*)$ is given by $\Theta_{\min}$. This is the highest threshold value which achieves maximal sensitivity. The lower the number $k_{\mathbf{m},\Theta}$ of words which are recognized as binding words the better the estimation of the true hyperplane.

sets are used as a basis for the analyses presented in Section 5. For a subset of 13 matrices, $n \geq 30$ binding words are available from the database.

The performance of a binding word classifier can be characterized by sensitivity and specificity. For a classifier based on a scoring vector $\mathbf{m}$, sensitivity and specificity can be adjusted by varying the threshold Θ. There always exists a setting $\Theta_{\min}(\mathbf{m}) = \min\{S_{\mathbf{m}}(\mathbf{w}) : \mathbf{w} \in \mathcal{V}\}$ such that sensitivity amounts to 1, however, at the cost of a specificity less than 1. This is geometrically illustrated in Fig. 3.

Let $k_{\min}(\mathbf{m})$ denote the number of words which satisfy $S_{\mathbf{m}}(\mathbf{w}) \geq \Theta_{\min}(\mathbf{m})$. By choosing $\Theta_{\min}(\mathbf{m})$ as the threshold, maximal sensitivity is achieved and specificity amounts to $\mathrm{spec}(\mathbf{m}) = 1 - \frac{k_{\min}(\mathbf{m}) - k_*}{K - k_*}$. If $\mathbf{m}$ is colinear to the energy vector $\mathbf{e}$ (see Section 2.4), $k_{\min}(\mathbf{m}) = k_*$ and specificity consequently amounts to 1, which means that perfect classification is achieved.

Given only the subset $\hat{\mathcal{V}} \subset \mathcal{V}$, the value of $k_{\min}(\mathbf{m})$ remains unknown, but $\hat{\Theta}_{\min}(\mathbf{m}) = \min\{S_{\mathbf{m}}(\mathbf{w}) : \mathbf{w} \in \hat{\mathcal{V}}\}$ and $\hat{k}_{\min}(\mathbf{m})$, the number of words which satisfy $S(\mathbf{w}) \geq \hat{\Theta}_{\min}(\mathbf{m})$, can be determined. $\hat{k}_{\min}(\mathbf{m})$ evidently underestimates $k_{\min}(\mathbf{m})$. As a general approach to compensate for this underestimation, the scoring vector $\mathbf{m}$ may be computed based on a subset of the known words, and the remaining words may be used to adjust the threshold to some value $\Theta < \hat{\Theta}_{\min}(\mathbf{m})$, which provides a more realistic impression of $k_{\min}(\mathbf{m})$ than $\hat{\Theta}_{\min}(\mathbf{m})$. This concept is the basis of the assays which are described in Sections 4.1 and 4.2.

For two scoring vectors $\mathbf{m}_1$ and $\mathbf{m}_2$, observing $\hat{k}_{\min}(\mathbf{m}_1)/K < \hat{k}_{\min}(\mathbf{m}_2)/K$ implies that $\mathrm{spec}(\mathbf{m}_1) \geq \mathrm{spec}(\mathbf{m}_2)$. This is strictly true if the words recognized by $\mathbf{m}_1$ are a subset of those recognized by $\mathbf{m}_2$. Otherwise, exceptions are possible, especially if $\hat{\mathcal{V}}$ is a strongly biased subset of $\mathcal{V}$, but on average, $\hat{k}_{\min}(\mathbf{m}_1)/K < \hat{k}_{\min}(\mathbf{m}_2)/K$ indicates that $\mathbf{m}_1$ approximates $\mathbf{e}$ better than $\mathbf{m}_2$. This inequality is most useful if $\hat{k}_{\min} > k_*$ is satisfied, i.e. if the classifier recognizes many false

positives. In this case, the approximation $\mathrm{spec}(\mathbf{m}) \approx 1 - \hat{k}_{\min}(\mathbf{m})/K$ becomes valid, and lowering $\hat{k}_{\min}$ is likely to indicate an improvement in specificity.

A biologically plausible range for the value of k_*/K can be derived from sequence information content analysis, even though the absolute value of k_* is not known for any transcription factor. Sequence information content[19] can, according to[20], be calculated by $R_{\mathrm{seq}} = -\log_2(k_*/K)$. Furthermore, R_{seq} cannot strongly deviate from $R_{\mathrm{freq}} = -\log_2(f)$, where f denotes the density of binding sites on the genome[13,19,20,21,22]. If R_{seq} is too low, the density of sites that are recognized by the transcription factor reaches a level at which spurious binding interferes with many genetic processes and becomes detrimental. Based on Schneider *et.al.*[19], one may reasonably expect that binding site density should not exceed two, or at most four per 1000 base pairs, corresponding to $R_{\mathrm{freq}} > 8$. If $k_{\min}(\mathbf{m})/K \geq 10^{-2}$, this indicates that $k_{\min}(\mathbf{m}) \gg k_*$, and that therefore specificity can be characterized based on $k_{\min}(\mathbf{m})/K$, as described above, in good approximation.

Technically, the computation of $k_{\mathbf{m},\Theta}(\mathbf{m})$ requires enumerating all words of length L and checking for each whether it satisfies $\mathbf{m}^T\mathbf{w} \geq \Theta$. This is prohibitively time-consuming for word lengths substantially greater than 10. Therefore, $k_{\mathbf{m},\Theta}/K$ was estimated by randomly sampling 100 000 words.

4.1. *Leave-One-Out Assay*

The size of the data sets provided by TRANSFAC, i.e. the number of binding words known for a transcription factor, ranges from 1 to 73. For many matrices, the set of known binding words is too small to be reasonably split into a training and a test set. Therefore, we have used leave-one-out tests for comparative specificity analysis. In these tests, one word of the set of known binding words, denoted by $\hat{\mathbf{u}}$, is left out for testing, the rest is used for training.

For an individual test, the scoring matrices were computed based on the training set. The test word $\hat{\mathbf{u}}$ was then employed to adapt the threshold to $\Theta = \min\{\hat{\Theta}_{\min}(\mathbf{m}), S_{\mathbf{m}}(\hat{\mathbf{u}})\}$. The threshold was thus set to obtain matrices which are consistent with all known binding words, including the one which has not been used for computing the matrix. $k_{\mathbf{m},\Theta}/K$ was then estimated as described above to assess the specificity at this sensitivity level. For each set of binding words, multiple tests were performed such that each word was left out once as the test word.

4.2. *Training-And-Test Assay*

Additional performance tests based on more than one test word separate from training data were carried out for large sets of known binding words with $n \geq 30$. For this type of test, the input set was split into a training set, containing 2/3 of the words selected at random, and a test set consisting of the remaining words, denoted by $\{\hat{\mathbf{u}}_1, \hat{\mathbf{u}}_2, \ldots, \hat{\mathbf{u}}_j\}$. The matrices were computed based on the training set, and, analogously to the procedure used for the leave-one-out test, the threshold

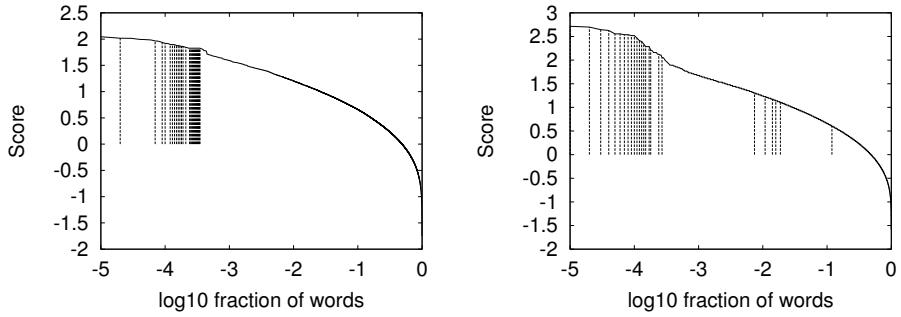


Fig. 4. Sorted score plots for the binding matrix (left) and the profile matrix (right), computed for the 26 binding words known for the human SRF. The rank is displayed logarithmically on the horizontal axis and the score is shown on the vertical axis. The dotted lines indicate the scores of the known binding words. With the profile matrix, about 10% of the words have scores higher than one of the experimentally verified binding words. With the binding matrix, however, less than 0.1% of the words have scores exceeding $\hat{\Theta}_{\min}$.

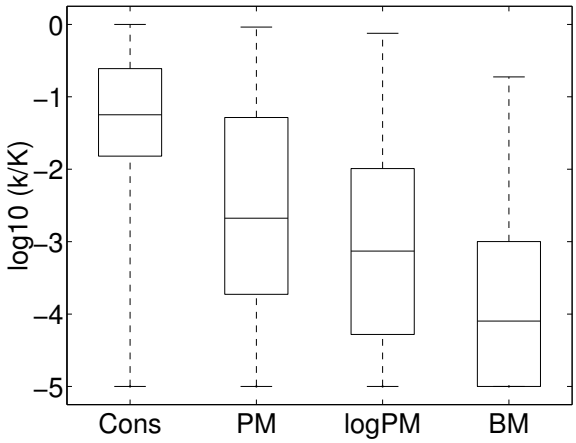


Fig. 5. Box plot showing the $\hat{k}_{\min}(\mathbf{m})/K$ values for the consensus sequence (Cons), the profile matrix (PM), the logarithmic profile matrix (logPM) and the binding matrix (BM), on a logarithmic scale. The threshold was set to $\hat{\Theta}_{\min}(\mathbf{m})$ for all matrices. Boxes encompass the middle (i.e. the second and third) quartiles, the horizontal line in the box shows the median. The bars extend to the minimal and the maximal value, respectively. The floor at 10^{-5} is due to the estimation of $k_{\mathbf{m},\Theta}/K$ based on 100 000 random samples.

was set to $\Theta = \min\{\hat{\Theta}_{\min}(\mathbf{m}), S_{\mathbf{m}}(\hat{\mathbf{u}}_1), S_{\mathbf{m}}(\hat{\mathbf{u}}_2), \ldots, S_{\mathbf{m}}(\hat{\mathbf{u}}_j)\}$. Tests were repeated 10 000 times with randomly generated training set selections.

5. Results and Discussion

Generally, the binding matrix computed from a binding word set is similar, but not identical to the profile matrix and the logarithmic profile matrix. The significance of the differences between the profile and the binding matrix is revealed by the sorted score plots shown in Fig. 4. For these plots, 100 000 words were randomly drawn in addition to the $n = 26$ known binding words. The scores for all words were calculated with the profile and the binding matrix. The scores were sorted in descending order, thus determining the rank for each word. The results show that $\hat{k}_{\min}(\mathbf{p})/K \approx 10^{-1}$, while $\hat{k}_{\min}(\mathbf{q})/K \approx 10^{-3.5}$.

From the information theoretic perspective, $\hat{k}_{\min}(\mathbf{p})/K \approx 10^{-1}$ implies that $R_{\text{seq}} \leq 3.2$, which is definitely too low to be realistic. The value of $\hat{k}_{\min}(\mathbf{q})$, on the other hand, leads to $R_{\text{seq}} \leq 11.6$, which appears reasonably compatible with the estimations explained in Section 4.

Fig. 5 shows aggregated results obtained for the 95 matrices for which at least 5 complete binding words were available. The minimum value, obtained with sets containing very similar binding words, is $\hat{k}_{\min}(\mathbf{m})/K = 10^{-5}$ due to the estimation procedure based on 100 000 random samples.

For the consensus sequence, the profile matrix and the logarithmic profile matrix, there are sets for which this ratio is close to 1, which means that the most specific, consistent classifiers based on these matrices classify almost no words as non-binding words. With the binding matrix, however, the maximal value of $\hat{k}_{\min}(\mathbf{q})/K$ is about 0.25. In other words, for all binding word sets from TRANSFAC, the binding matrix computes a consistent classifier according to which at least 75% of the words are classified as non-binding words. The corresponding estimate $R_{\text{seq}} \geq 2$ is obviously far from being biologically plausible.

The sets for which these large $\hat{k}_{\min}(\mathbf{m})/K$ values are observed contain words which have little similarity to the others. As a possible explanation, the binding word sets that give extremely high $\hat{k}_{\min}(\mathbf{q})/K$ values could contain words which have falsely been annotated as binding words. Alternatively, transcription factors may exist which have more than one binding domain. In such a case, description of the factor's binding properties with two independent matrices would be adequate. However, further investigation regarding these issues is beyond the scope of the work reported here.

The majority of results exhibits a clear trend, evidenced by the median and its surrounding quartiles. Compared to $\hat{k}_{\min}(\mathbf{c})/K$, the value obtained with the consensus sequence, the ratio obtained with the profile matrix, $\hat{k}_{\min}(\mathbf{p})/K$, is lower by more than one decimal order of magnitude, and the logarithmic profile matrix achieves a median $\hat{k}_{\min}(\mathbf{g})/K$ about two orders of magnitude below that which is obtained with the consensus sequence. This is improved by yet another order of magnitude by $\hat{k}_{\min}(\mathbf{q})/K$, the binding matrix.

It should be noted that $\hat{k}_{\min}(\mathbf{m})/K$ may underestimate k_*/K. Therefore, R_{seq} values estimated on the basis of $\hat{k}_{\min}(\mathbf{m})/K$ may be larger than the true R_{seq}

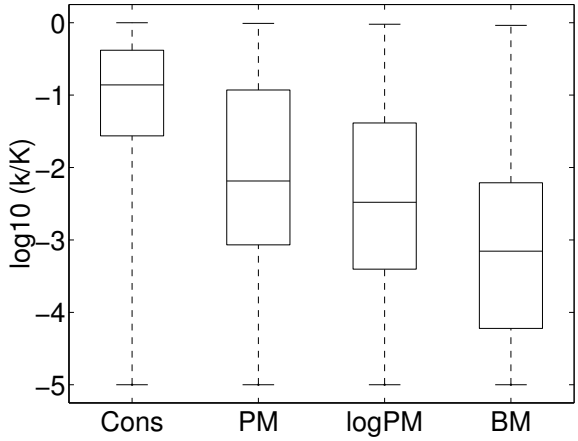


Fig. 6. Results of the leave-one-out assays for sets containing at least 5 experimentally verified binding words. Plots were generated and labelled as described in the legend of Fig. 5. Compared to the profile matrix, the binding matrix achieves an increase in recognition performance of about an order of magnitude, about the same increase the profile matrix could achieve compared to the simple consensus sequence.

value. The median around 10^{-4} (corresponding to $R_{\text{seq}} \approx 13.3$), found with the binding matrix, is therefore not indicative of an inconsistency with the empirically observed range for R_{seq}. Rather, it should be attributed to the small size of most binding word sets. The assays discussed below use test data to compensate for this underestimation of k_*/K, and thus complement and extend the performance analysis of the various matrix based classifiers presented above.

5.1. *Leave-one-out assays*

The aggregated results for word sets with $n \geq 5$ of the leave-one-out assays is shown in Fig. 6. The median $k_{\mathbf{m},\Theta}/K$ values are larger than the corresponding ones displayed in Fig. 5. This is not surprising because for all matrix types, the matrix will tend to diverge from a word that is left out, resulting in lower threshold values and correspondingly larger $k_{\mathbf{m},\Theta}/K$ ratios.

The binding matrix produces the lowest $k_{\mathbf{m},\Theta}/K$ values, followed by the logarithmic profile matrix, the profile matrix and the consensus sequence. Thus, the threshold value at which the binding word which was left out is correctly classified is, on average, significantly higher for the binding matrix than for the other matrix types. It therefore can be expected that binding words which are not known at binding matrix computation will typically receive relatively high scores while with other matrix types, the score of unknown binding words will typically be less strongly elevated. This result indicates that, on average, the binding matrix allows a higher specificity at the 100% sensitivity level on training data, and it may indicate that

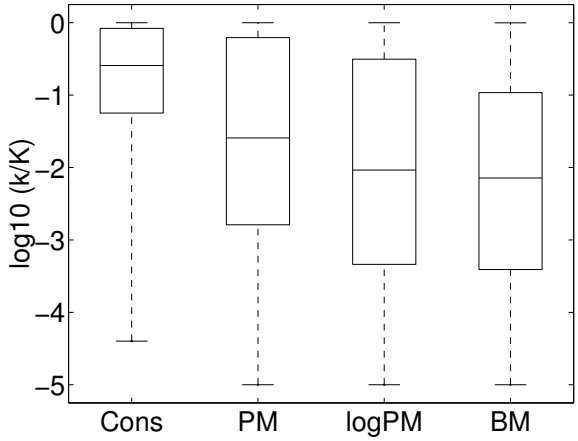


Fig. 7. Results of the training-and-test assays for 13 transcription factors with $n \geq 30$ experimentally verified binding words. Plots are generated and labelled as described in the legend of Fig. 5. The lowest median $k_{\mathbf{m},\Theta}$ value is achieved with the binding matrix.

the binding matrix is a better estimator for the binding energy matrix $\mathbf{e}$ than the other matrix types.

5.2. *Training-and-test assays*

Training-and-test assays were conducted for the 13 binding word sets with $n \geq 30$. Fig. 7 shows the aggregated results. Qualitatively, the trend which emerged from the previous results is confirmed; on average, the binding matrix attains the lowest $k_{\mathbf{m},\Theta}$ values, indicating the highest potential for specificity. However, this trend is less pronounced than in the leave-one-out tests. The improvement is about half an order of magnitude compared to the profile matrix, and somewhat less than a quarter of an order of magnitude compared to the logarithmic profile matrix.

The main problem, which affects all matrix types, is revealed by the high absolute value of the $k_{\mathbf{m},\Theta}/K$ ratio, which is not significantly lower than 10^{-2} even for the binding matrix. This observation indicates that with the word sets provided by TRANSFAC, one has to expect that all matrix types assign low scores to some unknown binding words, and as a consequence, these words will be undetectable in practice. This means that the sets of experimentally verified binding words do not permit any of the four classifiers tested here to approach the ideal of perfect specificity and perfect sensitivity at the same time. This is, at least to a substantial extent, due to the small size of the binding word sets.

6. Conclusion and Outlook

6.1. *Matrix Estimation by Maximum Likelihood*

Matrices and thresholds for binding word detection constitute linear classifiers in the space of orthogonally encoded words. The binding behaviour of the transcription factor itself can approximately be described by such a classifier. The binding matrix results from estimating the binding behaviour of the transcription factor using a maximum likelihood approach.

The binding matrix is based on a uniform binding word probability distribution on the binding sites. Other distributions result in other maximum likelihood estimators (Gewehr *et al.*, in preparation). However, the uniform distribution is consistent with maximum entropy, which should be assumed if no other, more specific information regarding the distribution is available. While the distribution of binding energies has been studied in considerable detail[7,9,13,23], the probability distribution for words on binding sites can not yet be reliably estimated from empirical data. Therefore, the binding matrix is an adequate choice because it is the maximum likelihood estimate based on the maximum entropy distribution.

It is interesting to note that there are analogies between the maximization approach employed in the calculation of the binding matrix and probabilistic approaches to motif finding. For example, Gibbs sampling[24] has been used to parameterize a binding word model and a background sequence model such that the probability ratio between both models is maximal. Differently from motif finding algorithms, which also include those based on Expectation Maximization[25], the objective function which is maximized for computing the binding matrix does not have multiple, local optima.

6.2. *The Binding Matrix as the Maximally Specific Matrix*

Using 95 binding word sets provided by the TRANSFAC database, we could show that potential specificity, as quantified by $k_{\mathbf{m},\Theta}/K$, can be improved by about an order of magnitude compared to the logarithmic profile matrix and the profile matrix. From this perspective, the binding matrix may constitute a step which is comparable to the replacement of consensus sequences with profile matrices, which allowed for a similar improvement.

According to analyses based on information theory[19,20], the $k_{\mathbf{m},\Theta}/K$ values provide a good estimate of the density of predicted binding sites on a genome. Preliminary studies with the genome sequence of *Escherichia coli* K 12 (GenBank accession `NC_000913`) have confirmed that the density of predicted binding words is indeed very accurately estimated by the random sequence model implied by $k_{\mathbf{m},\Theta}/K$ analysis. Significant deviations between $k_{\mathbf{m},\Theta}/K$ are expected with genomes in which the frequencies of bases (or short words) are skewed. In such cases, the binding matrix could be extended to account for such effects by using corresponding priors.

The binding word sequence information R_{seq} provides an indication for biologi-

cal plausibility of a binding word classifier. For the binding matrix, the R_{seq} values implied by the $k_{\mathbf{m},\Theta}/K$ ratios are within the order of magnitude which is predicted by information theoretic analysis. However, for the other matrices, requiring consistency with all known binding words implies $k_{\mathbf{m},\Theta}/K$ values which are at least one order of magnitude lower than expected, indicating that a substantial amount of words (in the range of 90%) classified as binding words are false positives.

Leave-one-out assays revealed that the chance of unknown binding words to receive high scores is better with the binding matrix than with the other matrix types. Further assays, in which more than one word is set aside for evaluating the classifiers, give the same results, but the differences between the binding matrix, the logarithmic profile matrix and the profile matrix become less pronounced. This is to be attributed to the scarcity of data, at least to a substantial extent. One may hope for the situation to improve as larger binding word sets become available in the future. Analyses with further data sets are currently underway (Gewehr *et.al.*, in preparation). Preliminary simulation based studies which we have conducted indicate that the advantage of the binding matrix, quantified by the accuracy with which the energy matrix $\mathbf{e}$ is estimated, over the other matrix types indeed increases with the number of available binding words.

The optimization approach underlying the binding matrix is to maximize the threshold score under the constraint that all experimentally verified binding words are correctly classified. Maximization of the threshold results in optimization of specificity while the constraints ensure consistency with the set of known binding words in the sense of Wolff *et.al.*[17]. Thus, the binding matrix is the consistent matrix with the highest possible specificity.

6.3. *Further Improvements and Perspectives*

For some binding word sets, it is impossible to obtain a matrix based classifier which correctly recognizes all binding words in the set and at the same time implies a biologically plausible value for R_{seq}. Some of these cases may be due to the presence of non-binding words in a binding word set, e.g. due to annotation errors. The binding matrix provides a minimal $k_{\mathbf{m},\Theta}$ value even in such cases. This is achieved by the response of the binding matrix upon the addition of a new word to the training set. If the new word is not recognized by the binding matrix computed from the previously known binding words, the new binding matrix is moved more strongly towards the new word than the other matrices. This, in turn, allows for a higher threshold value without misclassification of known binding words.

This responsiveness is, on average, advantageous if new binding words are added to the training data. However, if a non-binding word is added, the binding matrix may more easily be moved further away from the energy vector than the other matrices. Therefore, the binding matrix depends on careful assembly of binding word sets more critically than the other matrix types, particularly if it is to be used as an estimator of the energy matrix. Regarding classifier quality, the presence of

non-binding words in the training set incurs large penalties in specificity when a threshold which provides a consistent classifier is chosen. It is worthwhile to observe that even in this case, the binding matrix provides the most specific consistent matrix.

As a general approach to obtain consistent matrices with high specificity, Wolff *et.al.*[17] suggest removal of outlying binding words from the training set. Outliers are detected on the basis of scores. In particular, if only one word attains the score $\Theta_{\min}(\mathbf{m})$, removal of this word is guaranteed to yield a reduced training set for which $\hat{k}_{\min}(\mathbf{m})$ will be smaller than for the original set. However, with the binding matrix, many words in the training set receive a score of Θ_{BM}. This is due to the systematic maximization of the threshold upon which computation of the binding matrix is based. Therefore, outliers can not be detected solely on the basis of scores, but additional criteria, e.g. dissimilarity to the other words with a score of Θ_{BM}, can be applied. As a simple and obvious heuristic, one could also use the profile matrix as a means for detecting such outliers.

The words thus removed from the training data may have been put into the data set due to misannotation, and in this case, their removal is desirable. It is, however, also possible that no annotation errors are involved. To account for such cases, a new word set could be assembled from the outliers that were removed, and a binding matrix could then be computed for this set of outliers. This approach would be suitable for transcription factors that possess multiple binding domains. In principle, unsupervised clustering of the binding word set might be applied to detect such conditions, but the data sets which are currently available are too small for this approach.

In another perspective, the superior specificity achieved by binding matrix may obviate the need for removal of words from the training data set altogether. In this view, the binding matrix may provide an alternative to specificity improvement through training set reduction. Of course, both approaches can also be applied in combination.

Matrix based classifiers are not suitable for detecting binding motifs that have a variable length. Obviously, this limitation also applies to the binding matrix; more advanced classifiers are needed for detecting such sites. However, if parts of constant length can be identified within a variable-length motif, matrices for detecting these motif parts are suitable components for constructing such an advanced classifier. It appears that in the past, such approaches have been hampered by limitations in matrix specificity. Therefore, the binding matrix may be particularly suitable to serve as such a building block.

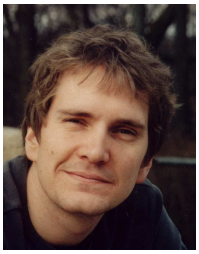
In a longer perspective, it will be particularly interesting to extend this approach for modelling higher order motifs such as composite regulatory elements. Improving specificity of recognition for each component may, under favourable conditions, strongly boost the specificity of recognition of entire composite elements. Thus, the binding matrix may serve as a component for methods for extracting regulatory information from genomic sequences.

References

1. Kornelie Frech, Kerstin Quandt, and Thomas Werner. Finding protein-binding sites in DNA sequences: The next generation. *TIBS*, 22:103–104, 1997.
2. Gary D. Stormo. DNA binding sites: Representation and discovery. *Bioinformatics*, 16:16–23, 2000.
3. Martin Vingron and Peter R. Sibbald. Weighting in sequence space: A comparison of methods in terms of generalized sequences. *Proc. Natl. Acad. Sci. USA*, 90:8777–8781, 1993.
4. M. Mulligan, D. Hawley, R. Entriken, and W. McClure. *Escherichia coli* promoter sequences predict *in vitro* RNA polymerase selectivity. *Nucl. Acids Res.*, 12:789–800, 1984.
5. A. Sarai and Y. Takeda. Lambda repressor recognizes the approximately 2-fold symmetric half-operator sequences asymmetrically. *Proc. Natl. Acad. Sci. USA*, 86:6513–6517, 1989.
6. Y. Takeda, A. Sarai, and V. Rivera. Analysis of the sequence-specific interactions between *Cro* repressor and operator DNA by systematic base substitution experiments. *Proc. Natl. Acad. Sci. USA*, 86:439–443, 1989.
7. G.D. Stormo, S. Strobl, M. Yoshioka, and J.S. Lee. Specificity of the *Mnt* protein. independent effects of mutations at different positions in the operator. *J. Mol. Biol.*, 229:821–826, 1993.
8. Gary D. Stormo and Dana S. Fields. Specificity, free energy and information content in protein-DNA-interactions. *Trends in Biochemical Sciences*, 23:109–113, 1998.
9. Panayiotis V. Benos, Martha L. Bulyk, and Gary D. Stormo. Additivity in protein-DNA interactions: How good an approximation is it? *Nucleic Acids Research*, 30:4442–4451, 2002.
10. G.D. Stormo, T.D. Schneider, L. Gold, and A. Ehrenfeucht. Use of the perceptron algorithm to distinguish translational initiation sites in *E. coli*. *Nucl. Acids Res.*, 10:2997–3011, 1982.
11. R. Harr, M. Haggstrom, and P. Gustafsson. Search algorithm for pattern match analysis of nucleic acid sequences. *Nucl. Acids Res.*, 11:2943–2957, 1983.
12. Roger Staden. Computer methods to locate signals in nucleic acid sequences. *Nucl. Acids Res.*, 12:505–519, 1984.
13. Otto G. Berg and Peter H. von Hippel. Selection of DNA binding sites by regulatory proteins. statistical-mechanical theory and application to operators and promoters. *J. Mol. Biol.*, 193:723–750, 1987.
14. David M. J. Tax and Robert P. W. Duin. Data domain description using support vectors. In *European Symposium on Artificial Neural Networks (ESANN 1999)*, pages 251–256, Brussels, 1999. D-Facto Publications.
15. Robert Fourer, David M. Gay, and Brian W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, Belmont, CA, USA, 2002.
16. R.J. Vanderbei. LOQO: An interior point code for quadratic programming. *Optimization Methods and Software*, 11:451–484, 1999.
17. Horst Wolff, Ruth Brack-Werner, Markus Neumann, Thomas Werner, and Ralf Schneider. Integrated functional and bioinformatics approach for the identification and experimental verification of RNA signals: Application to HIV-1 INS. *Nucleic Acids Research*, 31:2839–2851, 2003.
18. E. Wingender, X. Chen, R. Hehl, H. Karas, I. Liebich, V. Matys, T. Meinhardt, M. Prüß, I. Reuter, and F. Schacherer. TRANSFAC: An integrated system for gene expression regulation. *Nucl. Acids Res.*, 28:316–319, 2000.
19. Thomas D. Schneider, Gary D. Stormo, and Larry Gold. Information content of bind-

ing sites on nucleotide sequences. *J.Mol.Biol.*, 188:415–431, 1986.
20. Jan T. Kim, Thomas Martinetz, and Daniel Polani. Bioinformatic principles underlying the information content of transcription factor binding sites. *Journal of Theoretical Biology*, 220:529–544, 2003.
21. Otto G. Berg and Peter H. von Hippel. Selection of DNA binding sites by regulatory proteins. II. the binding specificity of cyclic AMP receptor protein to recognition sites. *J. Mol. Biol.*, 200:709–723, 1988.
22. Thomas D. Schneider. Evolution of biological information. *Nucleic Acids Research*, 28:2794–2799, 2000.
23. Martha L. Bulyk, Philip L.F. Johnson, and George M. Church. Nucleotides of transcription factor binding sites exert interdependen effects on the binding affinities of transcription factors. *Nucleic Acids Research*, 30:1255–1261, 2002.
24. Charles E. Lawrence, Stephen F. Altschul, Mark S. Boguski, Jun S. Liu, Andrew F. Neuwald, and John C. Wootton. Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.
25. Timothy L. Bailey and Charles Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In Russ B. Altman, Douglas L. Brutlag, Peter D. Karp, Richard H. Lathrop, and David B. Searls, editors, *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 28–36, Menlo Park, CA, 1994. AAAI Press.

Jan T. Kim received his diploma degree in biology in 1991 and his doctoral degree in 1996, both from the University of Cologne. He was at the Max-Planck-Institute for Plant Breeding in cologne from 1990 to 2000. During this time, he investigated evolution of regulatory networks and morphogenesis with a special focus on the evolution of flowering plants. He also developed and implemented biological databases and other bioinformatics systems. Since 2000, he is with the Institute for Neuro- and Bioinformatics at the University of Lübeck, where he works on transcription factors, regulatory networks, and evolutionary developmental bioinformatics.



Jan E. Gewehr received his diploma degree in computer science from the University of Lübeck in 2003. Since then, he is with the bioinformatics research group, headed by Ralf Zimmer, at the University of Munich. His current research focus is on protein structure analysis.

Thomas Martinetz studied Physics and Mathematics in Munich and Cologne. From 1988 to 1991 he was with the Theoretical Biophysics Group at the Beckman Institute of the University of Illinois at Urbana-Champaign, focussing on research on self-organising neural networks. In 1991 he joined the Neuroinformatics Research Center of Siemens AG in Munich. In 1996 he became head of the Complex Systems Group at the Institute for Neuroinformatics of the University of Bochum. Since 1999 he is Director of the Institute for Neuro- and Bioinformatics at the University of Lübeck. The main lines of research at his institute are in biological information processing, in particular in neural vision, pattern recognition, and learning.