GREEDY MIXTURE LEARNING FOR MULTIPLE MOTIF DISCOVERY IN BIOLOGICAL SEQUENCES

K. Blekas, D.I. Fotiadis and A. Likas

1 – 2002

Preprint, no 1 – 02 / 2002

Department of Computer Science
University of Ioannina
45110 Ioannina, Greece

Greedy mixture learning for multiple motif discovery in biological sequences

Konstantinos Blekas, Dimitrios I. Fotiadis and Aristidis Likas

Department of Computer Science and
Biomedical Research Institute, Foundation of Research and Technology, Hellas
University of Ioannina
45110 Ioannina
Greece

Abstract

This paper studies the problem of discovering subsequences, known as *motifs*, that are common to a given collection of related biosequences, by proposing a greedy algorithm for learning a mixture of motifs model through likelihood maximization. The approach adds sequentially a new motif to a mixture model by performing a combined scheme of global and local search for appropriately initializing its parameters. In addition, a hierarchical partitioning scheme based on kd-trees is presented for partitioning the input dataset in order to speed-up the global searching procedure. The proposed method compares favorably over the well-known MEME approach and treats successfully several drawbacks of MEME. Experimental results indicate that the algorithm is advantageous in identifying motifs with significant conservation and leads to the development of larger protein fingerprints.

1 Introduction

In protein sequence analysis motif identification is one of the most important problems covering many application areas. It concerns the discovery of portions of protein strands of major biological interest with important structural and functional features. For example, conserved blocks within groups of related sequences (*families*) can often highlight features which are responsible for structural similarity between proteins and can be used to predict the three dimensional structure of a protein. Consequently, the motifs are biologically informative in the sense of modeling efficiently sequences and holding useful information about biological families. Therefore, the proteins belonging to a family can be considered as sequences of motifs separated by an arbitrary number of randomly selected characters which indicate the *background* information. The last observation is also associated with the problem of multiple alignment of sequences where motif occurrences represent the alignment regions that can be visualized more easily compared to the background information. A detailed discussion about motif discovery applications can be found in [17].

Instead of using them for extracting conservative information and identifying structurally/ functionally important residues, the notion of motifs can also be used for characterizing biological families and searching for new family members [6]. Motifs may enclose powerful diagnostic features, generate rules for determining whether or not an unknown (not characterized) sequence belongs to a family and thus define a characteristic function for that family. This leads to the development of diagnostic signatures (*fingerprints*) that contain groups of conserved motifs used to characterize a family. The PRINTS (or PRINT-S) database [1] is an example of protein fingerprints database containing ungapped motifs that will be used in our experiments.

Usually, patterns or motifs can be distinguished into two general classes: deterministic and probabilistic [6, 7]. A *deterministic* motif encloses grammatical inference properties in order to describe syntactically a conserved region of related sequences using an appropriate scoring function based on matching criteria. Special symbols, such as arbitrary characters, wild-cards and gaps of variable length can be further used to extend the expressive power of deterministic patterns allowing a certain number of mismatches. The PROSITE database [10] consists of a large collection of such patterns used to identify protein families. On the other hand, a *probabilistic* motif is described by a probabilistic model that assigns a probability to the match between the motif and a sequence. The *position weight matrix* (PWM) provides a simplified model of probabilistic *ungapped* motifs representing the relative frequency of each character at each motif position. The ungapped mode suggests that the motif contains a sequence of statistically significant characters (*contiguous motif*) and corresponds to local regions of biological interest. Examples of more complicated probabilistic motifs (allowing gaps, insertions and/or deletions) are profiles and Hidden Markov models [9].

Many computational approaches have been introduced for the problem of motif identification in a set of biological sequences which differ according to the type of motifs discovered. In the literature there exist excellent surveys [6, 17, 7] on topics related to motif discovery techniques. The SAM approach [11], Gibbs sampling [12], MEME [3] and probabilistic suffix trees [5] represent probabilistic methods for finding multiple shared motifs within a set of unaligned biological sequences. Among those, the MEME algorithm fits a two-component finite mixture model to a set of sequences using the *Expectation Maximization* (EM) algorithm [8], where one component describes the motif (ungapped substrings) and the other describes the background (other positions in the sequences). Multiple motifs are discovered by sequentially applying a new mixture model with two components to the sequences remaining after erasing the occurrences of the already identified motifs.

In this paper we present an innovative approach for discovering significant motifs in a set

of sequences based on recently developed incremental schemes for Gaussian mixture learning [13, 22]. Our method learns a mixture of motifs model in a greedy fashion by incrementally adding components (motifs) to the mixture until reaching some stopping criteria or up to a desired number of motifs. Starting with one component that models the background, at each step a new component is added which corresponds to a candidate motif. The algorithm tries to identify a good initialization for the parameters of the new motif by performing *global* search over the input substrings together with *local* search based on partial EM steps for fine tuning of the parameters of the new component. In addition, a hierarchical clustering procedure is proposed based on kd-tree techniques [4, 20, 21] for partitioning the input dataset of substrings, which can reduce the time complexity for global searching and therefore accelerate the initialization procedure.

In analogy to the MEME approach, our technique discovers motifs when neither the number of motifs nor the number of occurrences of each motif in each sequence is known. However, as it will be experimentally shown and discussed in more detail later, the main difference with MEME technique is the way that the mixture models are applied. Although both methods treat the multiple motif identification problem through mixture learning using the EM algorithm, our approach is able to effectively fit multiple-component mixture models. This is achieved through a combined scheme of global and local search, which overcomes the problem of poor initialization of EM that frequently gets stuck on local maxima of the likelihood function. This results in exploring the input dataset efficiently and the discovery of greater number of motifs. The experiments with the PRINTS database verify this feature of our method which is of biological importance since it may lead to the discovery of larger protein fingerprints.

On the other hand, the inability of efficiently handling a mixture model with g components ($g > 2$) causes the MEME algorithm to reduce the multiple-component problem to the iterative application of a two-component mixture model. Each time a new motif is discovered the occurrences of this motif are erased, pruning in such way the input dataset. Therefore the MEME approach does not allow the parameters of this motif to be reestimated in future steps, and thus future discovered motifs cannot contribute to possible re-allocation of the letter distribution in the motif positions. This drawback becomes significant in the case where they exist motifs that partially match, since these motifs are recognized by the MEME algorithm as one "composite" motif that cannot be further analyzed due to the removal of the motif occurrences.

The outline of this paper has as follows. In section 2 the proposed greedy mixture learning approach for motif discovery in a set of sequences is presented, together with a novel technique

for partitioning the data space in order to reduce the time complexity of global searching. Section 3 presents experimental results considering both artificial and real biological datasets to evaluate the performance of our method in motif discovery problems. The comparative results indicate the superiority of the proposed greedy EM approach and establish its ability to generate more powerful diagnostic signatures. Finally, section 4 summarizes the proposed method and addresses directions for future research work.

2 Greedy EM algorithm for motif discovery

2.1 The mixture of motifs model

Consider a finite set of characters $\Sigma = \{\alpha_1, \ldots, \alpha_\Omega\}$ where $\Omega = |\Sigma|$. Any sequence $S = a_1 a_2 \ldots a_L$, such that $L \geq 1$ and $a_i \in \Sigma$, is called a *string* (or *sequence*) over the character set Σ. The sequence S starts from position 1 and ends at position $L = |S|$. The consecutive characters $a_i \ldots a_{i+W-1}$ form a *substring* x_i of S length W, identified by the starting position i over the string S. There are $n = L - W + 1$ such possible substrings of length W generated from sequence S.

We assume a set of N unaligned sequences $S = \{S_1, \ldots S_N\}$ of length $L_1, \ldots, L_N$, respectively. In order to deal with the problem of motif discovery of length W we construct a new dataset containing all substrings of length W in S. Since for each original sequence S_s (of length L_s) there are $m_s = L_s - W + 1$ possible substrings of length W, we obtain a training dataset $X = \{x_1, \ldots, x_n\}$ of n substrings ($n = \sum_{s=1}^{N} m_s$) for the learning problem.

A mixture of motifs model f for an arbitrary substring x_i assuming g components can be written as:

$$f(x_i; \Psi_g) = \sum_{j=1}^{g} \pi_j \phi_j(x_i; \theta_j), \tag{1}$$

where Ψ_g is the vector of all unknown parameters in the mixture model of g components, i.e. $\Psi_g = [\pi_1, \ldots, \pi_{g-1}, \theta_1, \ldots, \theta_g]$. The mixing proportion π_j ($\pi_j \geq 0$, $\forall j = 1, \ldots, g$) can be viewed as the prior probability that data x_i has been generated by the jth component of the mixture and they satisfy $\sum_{j=1}^{g} \pi_j = 1$.

Each one of the g components corresponds to either a motif or the background. A motif j can be modeled by a position weight matrix $\text{PWM}_j = [p^j_{l,k}]$ of size $[\Omega \times W_j]$, where each value $p^j_{l,k}$ denotes the probability that the letter α_l is located in motif position k. Although the general model considers motifs of variable length W_j, in the sequel we assume motifs of constant length W. On the other hand, a background component j is represented using a probability vector BPM_j (of length Ω), where each parameter value ϱ^j_l denotes the probability of letter α_l to occur at an arbitrary position. The probability that a substring $x_i = a_{i1} \ldots a_{iW}$,

where $a_{ik} \in \Sigma$ $(k = 1, \ldots, W)$ has been generated by the component j is

$$\phi_j(x_i; \theta_j) = \begin{cases} \prod_{k=1}^{W} p^j_{a_{ik},k} & \text{if } j \text{ is motif} \\ \prod_{k=1}^{W} q^j_{a_{ik}} & \text{if } j \text{ is background} \end{cases}, \tag{2}$$

where the probability matrix PWM_j (or BPM_j) corresponds to the parameter vector θ_j.

The log-likelihood of the observed dataset X corresponding to the above model is

$$\mathcal{L}(\Psi_g) = \sum_{i=1}^{n} \log f(x_i; \Psi_g). \tag{3}$$

Formulating the problem as an incomplete-data problem [15], each substring x_i can be considered as having arisen from one of the g components of the mixture model of Equation 1. We can define the parameters $z_{ij} = 1$ or 0 (missing parameters) indicating whether x_i has been generated by the j-th component of the mixture $(i = 1, \ldots, n\ ;\ j = 1, \ldots, g)$. Then, the *complete*-data log-likelihood $\mathcal{L}^c$ is given by

$$\mathcal{L}^c(\Psi_g) = \sum_{i=1}^{n} \sum_{j=1}^{g} z_{ij} \{\log \pi_j + \log \phi_j(x_i; \theta_j)\}. \tag{4}$$

The EM algorithm can be applied for the log-likelihood maximization problem by treating the z_{ij} as missing data. The following update equations are obtained for each component j [16, 2, 3]

$$z_{ij}^{(t+1)} = Pr(z_{ij} = 1 | x_i, \Psi_g^{(t)}) = \frac{\pi_j^{(t)} \phi_j(x_i; \theta_j^{(t)})}{f(x_i; \Psi_g^{(t)})}, \tag{5}$$

$$\pi_j^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} z_{ij}^{(t+1)}, \tag{6}$$

$$\theta_j^{(t+1)} = \begin{cases} \hat{p}^j_{l,k} = \frac{\hat{c}^j_{l,k}}{\sum_{l=1}^{\Omega} \hat{c}^j_{l,k}} & \text{if } j \text{ is motif} \\ \hat{q}^j_l = \frac{\hat{c}^j_l}{\sum_{l=1}^{\Omega} \hat{c}^j_l} & \text{if } j \text{ is background} \end{cases}, \tag{7}$$

where the elements $\hat{c}^j_{l,k}$ $(\hat{c}^j_l)$ correspond to the observed frequency of letter α_l at position k of motif j occurrences (at background j arbitrary positions) and can be formally expressed as

$$\begin{array}{ll} \hat{c}^j_{l,k} = \sum_{i=1}^{n} z_{ij}^{(t+1)} \mathbf{I}(a_{ik}, l) & \text{if } j \text{ denotes motif} \\ \hat{c}^j_l = \sum_{i=1}^{n} z_{ij}^{(t+1)} \sum_{k=1}^{W} \mathbf{I}(a_{ik}, l) & \text{if } j \text{ denotes background} \end{array}.$$

The indicator $\mathbf{I}(a_{ik}, l)$ denotes a binary function which takes value 1 if the substring x_i contains letter α_l at position k and 0 otherwise, i.e.

$$\mathbf{I}(a_{ik}, l) = \begin{cases} 1 & \text{if } a_{ik} \equiv \alpha_l \\ 0 & \text{otherwise} \end{cases}.$$

Equations 5-7 can be used to estimate the parameter values Ψ_g of the g-component mixture model which maximize the log-likelihood function (Equation 4). Since it has been shown that the application of the EM algorithm to mixture problems monotonically increases the likelihood function [15], these EM steps ensure the convergence of the algorithm to a local maximum of the likelihood function. However, its great dependence on parameter initialization and its local nature (it gets stuck in local maxima of the likelihood function) do not allow us to directly apply the EM algorithm to a g component mixture of motifs model.

To overcome this problem of poor initialization of the model parameters several techniques have been introduced [15]. The MEME approach for example uses a dynamic programming algorithm which estimates the goodness of many possible starting points based on the likelihood measurement of the model after one iteration of EM [2, 3]. Our method provides a more efficient combined scheme by applying global search over appropriate defined candidate motifs, followed by a local search for fine tuning the parameters of a new motif. In the following subsection we describe a procedure for adding a new motif that ensures the proper initialization of its parameters and it is shown how the monotone increase of the likelihood can be guaranteed.

2.2 Greedy mixture learning

Assume that a new component $\phi_{g+1}(x_i;\theta_{g+1})$ is added to a g-component mixture model $f(x_i;\Psi_g)$. The new component corresponds to a motif modeled by the position weight matrix PWM_{g+1} denoted by the parameter vector θ_{g+1}. Then the resulting mixture has the following form

$$f(x_i;\Psi_{g+1}) = (1-a)f(x_i;\Psi_g) + a\phi_{g+1}(x_i;\theta_{g+1}), \tag{8}$$

with $a \in (0,1)$. The vector Ψ_{g+1} specifies the new parameter vector and consists of the parameter vector Ψ_g of the g-component mixture, the weight a and the parameter vector θ_{g+1}. Then, the log-likelihood for Ψ_{g+1} is given by

$$\mathcal{L}(\Psi_{g+1}) = \sum_{i=1}^{n} \log f(x_i;\Psi_{g+1}) = \sum_{i=1}^{n} \log\{(1-a)f(x_i;\Psi_g) + a\phi_{g+1}(x_i;\theta_{g+1})\}. \tag{9}$$

The above formulation proposes a two-component likelihood maximization problem, where the first component is described by the old mixture $f(x_i;\Psi_g)$ and the second one is the motif component $\phi_{g+1}(x_i;\theta_{g+1})$ with $\theta_{g+1} = [p_{l,k}^{g+1}]$ $(l = 1,\ldots,\Omega;\ k = 1,\ldots,W)$ describing the position weight matrix PWM_{g+1}. If we consider that the parameters Ψ_g of $f(x_i;\Psi_g)$ remain fixed during maximization of $\mathcal{L}(\Psi_{g+1})$, the problem can be treated by applying searching techniques to optimally specify the parameters a and θ_{g+1} which maximize $\mathcal{L}(\Psi_{g+1})$.

An efficient technique for the specification of θ_{g+1} is presented in [22] which carries out a combination of local and global searching. In particular, an EM algorithm performs local search for the maxima of likelihood with respect to a and θ_{g+1}, where the learning procedure is applied only to the mixing weight a and the probabilistic quantities $p_{l,k}^{g+1}$ of the newly inserted component (motif-model). Following Equations 5-7 and assuming that the new component models is a motif, the following update procedures can be derived

$$z_{i,g+1}^{(t+1)} = Pr(z_{i,g+1} = 1 | x_i, \theta_{g+1}^{(t)}, a^{(t)}) = \frac{a^{(t)} \phi_{g+1}(x_i; \theta_{g+1}^{(t)})}{(1 - a^{(t)}) f(x_i; \Psi_g) + a^{(t)} \phi_{g+1}(x_i; \theta_{g+1}^{(t)})}, \tag{10}$$

$$a^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} z_{i,g+1}^{(t+1)}, \tag{11}$$

$$\theta_{g+1}^{(t+1)} = [\hat{p}_{l,k}^{g+1}], \text{ where } \hat{p}_{l,k}^{g+1} = \frac{\hat{c}_{l,k}^{g+1}}{\sum_{l=1}^{\Omega} \hat{c}_{l,k}^{g+1}}, \tag{12}$$

where

$$\hat{c}_{l,k}^{g+1} = \sum_{i=1}^{n} z_{i,g+1}^{(t+1)} \mathbf{I}(a_{ik}, l).$$

The above *partial* EM steps constitute a simple and fast method for local searching the maxima of $\mathcal{L}(\Psi_{g+1})$. However, the problem of poor initialization still remains since this scheme is very sensitive to the proper initialization of the two parameters a and θ_{g+1}. For this reason a global search strategy has been developed [22] which facilitates the global search over the parameter space. In particular, by substituting the log-likelihood function, Equation 9, and using a Taylor approximation about a point $a = a_0$, we can use the resulting estimate to search for the optimal θ_{g+1} value. Therefore we expand $\mathcal{L}(\Psi_{g+1})$ by second order Taylor expansion about $a_0 = 0.5$ and then the resulting quadratic function is maximized with respect to a. This results into the following approximation

$$\hat{\mathcal{L}}(\Psi_{g+1}) = \mathcal{L}(\Psi_{g+1} | a_0) - \frac{[\dot{\mathcal{L}}(\Psi_{g+1} | a_0)]^2}{2\ddot{\mathcal{L}}(\Psi_{g+1} | a_0)}, \tag{13}$$

where $\dot{\mathcal{L}}(\Psi_{g+1})$ and $\ddot{\mathcal{L}}(\Psi_{g+1})$ are the first and second derivatives of $\mathcal{L}(\Psi_{g+1})$ with respect to a. It can be shown [22] that, for a given parameter vector θ_τ, a local maximum of $\mathcal{L}(\Psi_{g+1})$ near $a_0 = 0.5$ is given by

$$\hat{\mathcal{L}}(\theta_\tau) = \sum_{i=1}^{n} \log \frac{f(x_i; \Psi_g) + \phi_{g+1}(x_i; \theta_\tau)}{2} + \frac{1}{2} \frac{[\sum_{i=1}^{n} \delta(x_i, \theta_\tau)]^2}{\sum_{i=1}^{n} \delta^2(x_i, \theta_\tau)}, \tag{14}$$

and is obtained for

$$\hat{a} = \frac{1}{2} - \frac{1}{2} \frac{\sum_{i=1}^{n} \delta(x_i; \theta_\tau)}{\sum_{i=1}^{n} \delta^2(x_i; \theta_\tau)}, \tag{15}$$

where

$$\delta(x_i, \theta_\tau) = \frac{f(x_i; \Psi_g) - \phi_{g+1}(x_i; \theta_\tau)}{f(x_i; \Psi_g) + \phi_{g+1}(x_i; \theta_\tau)}. \tag{16}$$

If the estimated value of a falls outside the range $(0, 1)$ then we can initialize the partial EM with the approximation $\hat{a} = 0.5$ for $g = 1$ and $\hat{a} = 2/(g+1)$ for $g \geq 2$, according to [13].

The above methodology has the benefit of modifying the problem of maximizing the likelihood function (Equation 9) to become independent on the selection of initial value for the mixing weight a. In addition, this procedure reduces the parameter search space and restricts global searching on finding good initial values θ_τ of the probability matrix θ_{g+1} (probabilities $p_{l,k}^{g+1}$ composing the position weight matrix PWM_{g+1}) characterizing the new component (motif-model). The last observation is made clearer from Equation 14 where $\hat{\mathcal{L}}(\theta_\tau)$ depends only on $\phi_{g+1}(x_i; \theta_\tau)$, while $f(x; \Psi_g)$ remains fixed during optimization. The only problem is now the identification of a proper initial value θ_τ so as to conduct partial EM steps. Therefore we need to find *candidates* for the initialization of the motif parameters.

2.3 Candidate selection for initializing new model parameters

A reasonable approach of initializing motif parameters θ_{g+1} is to search for candidates directly over the total dataset of substrings $X = \{x_\tau\}$, $(\tau = 1, \ldots, n)$, where $x_\tau = a_{\tau 1} \ldots a_{\tau W}$. For this reason we associate with each substring x_τ a position weight matrix θ_τ constructed as follows

$$\theta_\tau = [p_{l,k}^\tau], \text{ where } \quad p_{l,k}^\tau = \begin{cases} \lambda & \text{if } a_{\tau,k} = \alpha_l \\ \frac{1-\lambda}{\Omega - 1} & \text{otherwise} \end{cases}. \tag{17}$$

The parameter λ has a fixed value in the range $(0, 1)$, where its value depends on the Σ alphabet size (Ω) and must satisfy $\lambda \geq 1/\Omega$ (e.g. $\lambda \gg 0.05$ for protein sequence data where $\Omega = 20$). Therefore, the (local) log-likelihood $\hat{\mathcal{L}}(\theta_\tau)$ is determined by selecting among the θ_τ matrices $(\tau = 1, \ldots, n)$ the one which maximizes the right hand size of Equation 14, i.e.

$$\hat{\theta}_{g+1} = \arg\max_{\theta_\tau} \hat{\mathcal{L}}(\theta_\tau).$$

In order to accelerate the above searching procedure the following quantities $\xi_{\tau,i}$ for each substring $x_i = a_{i1} \ldots a_{iW}$ can be computed

$$\xi_{\tau,i} (= \phi_{g+1}(x_i; \theta_\tau)) = \prod_{k=1}^{W} p_{a_{ik},k}^\tau, \tag{18}$$

which substitute for the $\phi_{g+1}(x_i; \theta_\tau)$ in Equations 14 and 16. Following this observation, the searching is made over these quantities $\xi_{\tau,i}$ which maximize Equation 14. The constructed matrix Ξ with elements $\xi_{\tau,i}$ can be calculated once during the initialization phase of the learning algorithm and will be applied each time a new component is added to a g-component

mixture, in order to identify the initial probabilistic matrix for the new applied motif. Similar techniques that use the same approach for searching for global solutions over the parameter space have been proposed in [19, 22] (for Gaussian mixture models).

The drawback of searching for candidates over all substrings n of the dataset is the increasing time complexity ($\mathcal{O}(n^2)$) of the search procedure. Indeed, $\mathcal{O}(n^2)$ computations are needed since the likelihood of every substring under every such candidate parameterized model must be evaluated. In order to reduce the complexity, we perform a *hierarchical clustering* pre-processing phase based on the notion of kd-trees. Original kd-trees [4] were proposed in an attempt to speed-up the execution of nearest neighbor queries by defining a recursive binary partitioning of a k-dimensional dataset, where the root node contains all data. Most such techniques partition the data at each tree level using an appropriate hyperplane perpendicular to the direction which presents major variance of the data [20, 21]. Efficient kd-tree techniques for the specification of candidate components have been presented in [14, 21] for the case of Gaussian mixture models.

In our approach we propose a modified approach in order to deal with sequential data. Starting with a root node that contains the total set of substrings X, at each step we partition the set of substrings at each node using an appropriate criterion based on maximum variance. In particular, after calculating the relative frequency values $f_{l,k}$ of each character α_l at each substring position k (of length W) in the subset of substrings containing that node, we identify the position q that exhibits the greatest variance over the alphabet Σ. The above procedure can be formally described as

$$q = \arg \max_{k=1,\ldots,W} \{ \sum_{\substack{\alpha_l \in \Sigma \\ f_{l,k}>0}} (\max_{\alpha_l \in \Sigma}(f_{l,k}) - f_{l,k})^2 \},$$

where $\max_{\alpha_l \in \Sigma}(f_{l,k})$ indicates the maximum relative frequency value among characters in position k. After identifying the position q with the maximum variance, the partitioning procedure is implemented by initially sorting the characters α_l in that position among the substrings according to their relative frequency values $f_{l,q}$ and then labeling them as *odd* or *even*. Finally, the set of substrings in the node can be partitioned into two subsets (left and right) which are successively filled with the substrings that contain the odd (left) and the even (right) characters in the position q. An example is shown in Figure 1 where the third position that has the greatest character variance is selected for partitioning.

The above recursive procedure builds a tree with several nodes and the partitioning for a node is terminated (*leaf node*) when the number of included substrings is lower than a fixed value T. We refer to this as T-*size* kd-tree scheme. Every node of the tree contains a subset (cluster) of the original set of substrings and each such cluster is characterized by its *centroid*

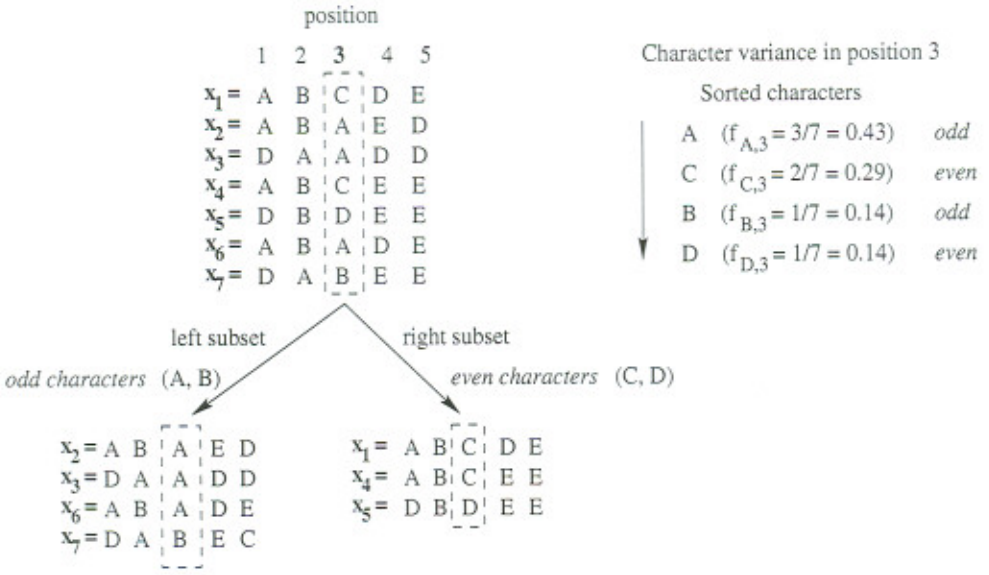

Figure 1: Partitioning occurs in the *third* position that presents the maximum character variance

(*consensus substring*). Therefore, the total set of leaf nodes consists of C centroids and their corresponding position weight matrices (obtained by Equation 17) constitute the candidate motif parameters used in global searching. Experiments have shown that this partitioning technique drastically accelerates the global search procedure without affecting significantly its performance.

Except for the T-size, another scheme for terminating the partitioning can be derived by considering a distance threshold among the substrings included in the subset of a node. Using as distance $d(x_i, x_j)$ between two substrings x_i, x_j the *Hamming distance* (degree of dissimilarity),

$$d(x_i, x_j) = \frac{1}{W} \sum_{k=1}^{W} (1 - \mathbf{I}(a_{ik}, a_{jk})),$$

where

$$\mathbf{I}(a_{ik}, a_{jk}) = \begin{cases} 1 & \text{if } a_{ik} \equiv a_{jk} \\ 0 & \text{otherwise} \end{cases},$$

we find the substring x_t which has the lowest average distance value among all T substrings included in a node, i.e.

$$t = \arg \min_{j=1,\ldots,T} \{ \frac{1}{T} \sum_{i=1}^{T} d(x_i, x_j) \}.$$

By specifying a distance threshold value η $(0 \leq \eta \leq 1)$ we decide whether to partition that node according to whether all distance values $d(x_i, x_t)$ (or the average value) are lower than the threshold η. In positive case this node is considered as a *leaf node*, otherwise it is further

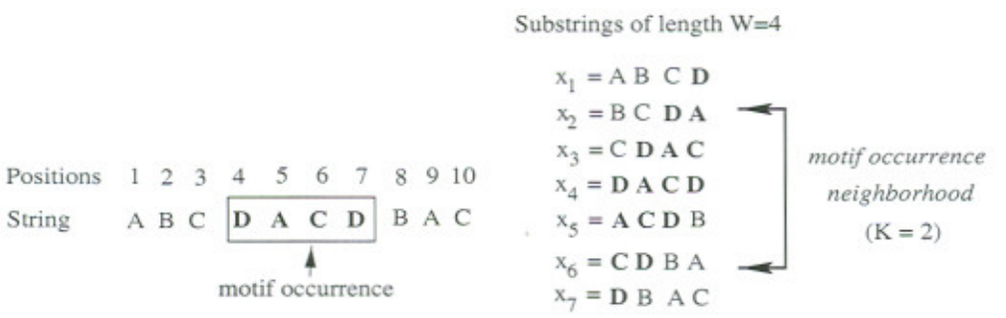

Figure 2: The neighborhood of a motif occurrence

partitioned as in the T-size scheme. We call this partitioning scheme as *η-distance threshold*.

The above two schemes for terminating the expansion of a kd-tree on a string domain can be also applied in a *hybrid* mode allowing the criterion of distance to have higher priority over the T-size.

Another problem we must face concerns the occurrence of overlappings with the already discovered motifs during the selection of a candidate motif instance. Therefore when a new motif is discovered, substrings which correspond to positions next to the motif occurrences in the original set of sequences (determining the *neighborhood* of motif occurrences) contain a portion of the discovered motif. If any of these *overlapping* substrings were used as a candidate motif model for initializing a new component of the mixture model, it would probably lead (as the performance of the EM algorithm depends very much on the initialization of the parameters [8, 15]) to the discovery of a new motif that would exhibit significant overlap with another one already being discovered. An example is illustrated in Figure 2, where the substrings $x_1, \ldots x_7$ of length $W = 4$ overlap with the discovered motif occurrence $x_4 =$ DACD and these substrings should not be subsequently considered as candidates for the discovery of additional motifs.

In order to avoid this inconvenience, a binary indicator value ϖ is introduced for each leaf node ($\tau = 1, \ldots, C$), whose value indicates the occurrence of a significant portion of a motif already being discovered ($\varpi_\tau = 1$) in the subset corresponding to that node. A parameter K ($K < W$) is used to define the neighborhood $\mathcal{N}_i$ of a motif occurrence x_i as the set of substrings x_j ($j = i - K, \ldots, i + K$) which *match* at least K contiguous characters with x_i, and thus are derived from K left and K right starting positions from the starting position i (in the original set of sequences) that corresponds to x_i, i.e.

$$\mathcal{N}_i = \{x_j\}, j = i - K, \ldots, i + K.$$

For example, setting $K = 2$ the substrings x_j ($j = 2, \ldots, 6$) in Figure 2, are included in the neighborhood of the motif occurrence $x_4 =$ DACD. Initially we set $\varpi_\tau = 0$ ($\forall \tau = 1, \ldots, C$),

and whenever a new motif g is found and its motif occurrences x_i are identified [1], the leaf nodes τ that contain substrings belonging to the neighborhood $\mathcal{N}_i$ of one of the x_i are excluded ($\varpi_\tau = 1$) from the set of candidate motifs used in the global search phase. Best results obtained for $K < W/2$.

The above strategy eliminates the possibility of overlappings among the motifs discovered and ensures proper specification of candidate motif models, while at the same time it iteratively reduces further the time complexity of the global search. In comparison with the MEME approach where substrings that correspond to the neighborhood of motif occurrences are deleted from the dataset, in our scheme the overlapping substrings are only excluded from the set of candidate motifs used in the global search phase. This constitutes a significant advantage over the MEME approach.

The proposed greedy EM algorithm

Summarizing the above ideas we have the following algorithm for learning a mixture of motifs model for the motif discovery problem.

[1] the motif occurrences x_i are determined by the z_{ig} values (Equation 5) that are near to 1 (e.g. $z_{ig} > 0.9$)

Start with a set of N unaligned sequences $S = \{S_s\}$ with length $L_s = |S_s|$ $(s = 1, \ldots, N)$, taking values from alphabet $\Sigma = \{\alpha_1, \ldots, \alpha_\Omega\}$ of length $\Omega = |\Sigma|$.

Initialization

- Apply a window of length W to the original set S creating a learning dataset of substrings $X = \{x_i\}$ $(i = 1, \ldots, n)$, where $n = \sum_{s=1}^{N}(L_s - W + 1)$.
- Apply the proposed kd-tree approach over the dataset X, find the C final consensus substrings and define the corresponding C candidate initial parameter values θ_τ, $\tau = 1, \ldots, C$ (Equation 17) used for global searching.
- Initialize $\varpi_\tau = 0$, $\forall \tau = 1, \ldots, C$, and calculate matrix Ξ of quantities $\xi_{\tau,i}$ according to Equation 18.
- Initialize the model using one component ($g = 1$) that represents the background with parameter settings (probability matrix θ_1) equal to the relative frequencies of characters $\alpha_l \in \Sigma$, i.e. $\varrho_l^1 = \frac{f_l}{\sum_{s=1}^{N} L_s}$, where f_l indicates the frequency of character α_l in the original set of sequences S.

Iterate

1. Perform EM steps (Equations 5-7) until convergence: $|\mathcal{L}^{(t)}(\Psi_g)/\mathcal{L}(\Psi_g)^{(t-1)} - 1| < 10^{-6}$. If an appropriate stopping condition (e.g. maximum number of motifs, or $\sum_{\tau=1}^{C} \varpi_\tau = C$) holds then terminate.
2. If $g \geq 2$ then from the motif occurrences x_i (with $z_{ig} > 0.9$) find their neighborhood $\mathcal{N}_i = \{x_j\}$ $(j = i - K, \ldots, i + K)$ and set $\varpi_\tau = 1$ for each leaf node τ which contains any of the x_j.
3. Insert a new candidate component $g + 1$ by searching over all θ_τ (where $\tau = 1, \ldots, C$ and $\varpi_\tau = 0$) and setting $\hat{\theta}_{g+1}$ equal to the θ_τ that maximize the log-likelihood function of Equation 14 using the already calculated quantities $\xi_{\tau,i}$ instead of $\phi(x_i; \theta_\tau)$. Compute the weight $\hat{a}$ using the obtained value $\hat{\theta}_{g+1}$ in Equation 15.
4. Perform partial EM steps (Equations 10 - 12) with initial values $\hat{a}$ and $\hat{\theta}_{g+1}$, until convergence as in step 1 and obtain the parameter values Ψ_{g+1}.
5. If $\mathcal{L}(\Psi_{g+1}) > \mathcal{L}(\Psi_g)$ then accept the new mixture model with $g + 1$ components and go to step 1, otherwise terminate.

The above algorithm ensures the monotonic increase of the log-likelihood of the learning set since EM cannot decrease the log-likelihood and the proposed partial EM solutions are accepted only if $\mathcal{L}(\Psi_{g+1}) > \mathcal{L}(\Psi_g)$. The stopping condition mentioned at step 1 depends not only on the maximum allowed number of components g (specified by the user), but also on the vector ϖ. This means that in the case $\varpi_\tau = 1$, $\forall \tau = 1, \ldots, C$, the parameter space for selecting candidate components has been entirely searched and therefore the possibility of

existence of another motif in the set of sequences is very low.

3 Experimental results

In order to evaluate the effectiveness of our method we have conducted a series of experiments considering several sets of biological sequences. We have tried to select training sets consisting of protein sequences for which there is *a priori* knowledge about the existing motifs and our objective was to examine the ability of the method in discovering them. In addition, we have measured the significance of the discovered motifs in terms of information content, in an attempt to demonstrate the capability of the greedy mixture learning approach to build more distinct (clearer) motifs which can subsequently be used as powerful diagnostic signatures.

The experiments described in this section have been conducted using both artificial and real datasets. In all cases the width W of the motifs is considered constant (specified by the user) and the proposed approach was applied only once. It must be noted that the only free parameter of our method was the λ value which is used to initialize the candidate position weight matrices (Equation 17). Good values for λ were found to be in the range $[0.6, 0.8]$.

For all the experimental datasets we have also applied the MEME approach using the available software from the corresponding Web site [2]. MEME uses three different models describing the distribution of motifs among the sequences. In all the experiments we have selected the "any number of repetitions" model [2, 3] as it is equivalent to the one used in our approach. After filling an appropriate form, the MEME Web site processes the submitted biological datasets and returns the results (through e-mail) describing in detail the motifs discovered for the submitted set of sequences.

It must be noted also that apart from the experiments described in the following subsections, additional experiments have been conducted using protein families available from the PROSITE database [10]. The reason why they are not cited here is that the PROSITE database contains deterministic (gapped) motifs of protein families which are not convenient for the evaluation of the proposed greedy EM algorithm. Hence, we decided to work with the PRINTS database [1] where the available information about fingerprints can be easily exploited to assess the performance of our method.

3.1 Experiments with artificial datasets

In the artificial datasets used in our experiments each motif has an associated randomly generated "seed substring" and copies of the motif (motif occurrences) are created by randomly

[2] The Web site of MEME/MAST system version 3.0 can be found at *http://meme.sdsc.edu/meme/website/*

motif	*starting position*	*seed motifs*											
			1	2	3	4	5	6	7	8	9	10	
1	1-20	$\star\star\star$	K	L	I	M	A	T	I	S	M	A	$\star\star\star$
2	31-50	$\star\star\star$	P	E	G	T	H	T	I	S	M	A	$\star\star\star$
3	61-80	$\star\star\star$	A	R	N	D	C	Q	E	G	H	I	$\star\star\star$
4	91-110	$\star\star\star$	E	G	H	I	L	K	M	F	P	S	$\star\star\star$
5	121-140	$\star\star\star$	W	Y	V	T	R	Q	A	N	P	V	$\star\star\star$
6	151-170	$\star\star\star$	A	N	C	E	H	L	M	P	T	Y	$\star\star\star$

Table 1: The motif distribution in the first series of artificial datasets

performing a number of substitutions (*mutations*) on the motif's seed substring with a mutation probability p_m. In fact, the mutation operation inserts a degree of noise within the motif description and as a result, the greater the probability value p_m, the harder the motif identification problem. For simplicity, without loss of generality, we have chosen to construct each artificial sequence using mutated copies of all the motifs (single occurrence for each one) at random positions (assuming no overlapping occurrences).

Two series of experiments have been made with the artificial datasets. In the first series we measured the impact of the proposed kd-tree approach for candidate selection on the performance of the whole algorithm. We created artificial datasets using six (6) different seed substrings of length $W = 10$ (Table 1), where the first two substrings are identical in half length, therefore making the problem of discovering them harder. Ten such sequences of variable length (between 190 and 220) were constructed by randomly locating and mutating copies of these substrings (ensuring no overlapping), while randomly filling the rest positions with characters from the amino acids alphabet Σ. Assuming three different values of the mutation probability $p_m = \{0, 0.1, 0.15\}$ three artificial datasets were constructed.

Table 2 provides comparative results for the above three artificial datasets obtained from the application of the greedy EM algorithm with and without the kd-tree approach. In particular, the second column displays the performance of the greedy EM method with global search over all the input substrings (no kd-tree) in terms of the number (M) of the discovered motifs. The next column presents the same result with the employment of the kd-tree method by applying the T-size scheme, as well as the hybrid scheme (T-size and η-distance threshold), respectively. For both of these cases results were obtained considering four different values for the size of the subsets in the tree nodes ($T = \{100, 50, 20, 10\}$), while for the hybrid scheme the parameter η was fixed at 0.75. In addition, the number of obtained centroids (leaf nodes) C is presented, which are used in the global searching phase.

The observations that can be derived from these results concerning the candidate selection

Problem	*Greedy EM*	*kd-tree Greedy EM*							
		T-size scheme				*Hybrid scheme*			
p_m						$\eta = 0.75$			
		$T = 100$	$T = 50$	$T = 20$	$T = 10$	$T = 100$	$T = 50$	$T = 20$	$T = 10$
0	M=6	C=32	C=64	C=134	C=285	C=880	C=880	C=880	C=886
($n = 1974$)		M=1	M=3	M=4	M=6	M=6	M=6	M=6	M=6
0.1	M=6	C=32	C=64	C=130	C=264	C=894	C=894	C=894	C=894
($n = 1963$)		M=1	M=2	M=4	M=6	M=6	M=6	M=6	M=6
0.15	M=6	C=32	C=64	C=128	C=260	C=916	C=916	C=916	C=916
($n = 1964$)		M=1	M=1	M=2	M=5	M=6	M=6	M=6	M=6

Table 2: Comparative results for estimating the impact of the proposed kd-tree partitioning approach

methodology are very useful. Clearly, the proposed greedy EM learning algorithm using global searching over all input substrings ($C = n$) has the ability of discovering all the significant motifs in all artificial sets of sequences. As it was expected, the application of the kd-tree method results in less candidate motifs and therefore speeds-up the identification of the appropriate initial motif parameters during the insertion of a new component in the mixture model. The T-size kd-tree scheme leads to the creation of a small number of centroids for $T \geq 20$. For smaller T values the number of centroids created is larger and sufficient enough for constructing a richer parameterized search space, but for harder problems ($p_m > 0.1$) this information is inadequate to find all motif occurrences. Adding the η-distance threshold termination criterion the results are excellent in all cases. As it is shown in Table 2, the enforcement of the η-distance threshold scheme plays the most significant role in the hybrid scheme providing almost the same number of centroids independently of the T value.

The above experiments indicate that the hybrid scheme for kd-trees is able to provide excellent component candidates for the mixture of motifs model which greatly accelerate the global search phase. Even if the hybrid scheme produces greater number of centroids in comparison with the T-size scheme, it results in great improvement in time complexity during the searching procedure in comparison with searching for candidates over all the substrings dataset (number of produced centroids is less than 50% of dataset size n). In the following experiments the applied greedy EM approach uses the hybrid kd-tree scheme with $T = N$ and $0.5 < \eta < 0.8$.

The aim of the second series of experiments is the comparison of our greedy EM approach with the MEME method in terms of the ability to discover the real number of motifs in artificial datasets. For this reason we have created artificial datasets from a new set of six (6) seed substrings of length $W = 20$. As it is illustrated in Table 3 the last two seed substrings (5

motif	*starting position*	*seed motifs*																				
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
1	1-40	* * *	A	V	E	R	Y	I	N	T	E	R	E	S	T	I	N	G	V	I	E	W	* * *
2	61-90	* * *	D	E	S	T	I	N	A	T	E	D	Y	D	E	D	I	C	A	T	E	D	* * *
3	111-140	* * *	P	R	I	M	E	R	S	T	A	G	E	D	E	S	I	G	N	I	N	G	* * *
4	161-190	* * *	W	A	I	T	I	N	G	F	T	H	E	G	I	A	N	T	T	R	A	M	* * *
5	211-240	* * *	S	T	E	W	A	R	T	E	L	E	N	A	M	I	L	E	N	A	E	M	* * *
6	261-290	* * *	A	R	I	S	T	I	D	I	S	K	N	A	M	I	L	E	N	A	E	M	* * *

Table 3: The motif distribution in the second series of artificial datasets

Problem	*number of motifs found*	
p_m	*greedy EM*	*MEME*
0	6	4
0.1	6	4
0.2	6	4

Table 4: Comparative results on artificial datasets using the greedy EM and the MEME approaches

and 6) are exactly the same in half of their length (from position 11 to 20). This fact makes the problem of identifying them as two distinct motifs very difficult. In analogy with the previous datasets, twenty ($N = 20$) artificial sequences of variable length were constructed, assuming a maximum and a minimum length of 330 and 310 respectively. For three different values of the mutation probability ($p_m = \{0, 0.1, 0.2\}$), we have created three different datasets used in our experimental study.

Given the artificial sets of sequences constructed as described above, the locations of all possible motifs are known and hence the greedy algorithm was applied in order to verify its effectiveness in discovering motif occurrences in such hard problems. The same datasets were also applied using the MEME approach and the corresponding results were depicted. Table 4 displays the comparative results for these two approaches where the superiority of the greedy EM algorithm is obvious in discovering all the incorporated motifs in the three datasets. The MEME approach does not achieve the identification of all the motifs. Due to its limitations, by considering the motifs 5 and 6 as one motif, and after erasing their occurrences from the dataset of substrings, MEME is unable to identify them in future steps. On the other hand, the greedy EM algorithm always works with the original dataset and is able to identify the two motifs.

3.2 Experiments with real datasets

The real datasets used in our experiments were obtained from the PRINTS database [1] which contains protein motif *fingerprints* [3]. A fingerprint of a protein family consists of a set of motifs that characterize this family and can be used as a classifier to predict the membership degree to that family, according to the complete (true family members) or partial (subfamily members) occurrence of this set of motifs in an unknown sequence. Thus, a fingerprint can be considered as a diagnostic signature of the family. The identification of the fingerprints within the PRINTS database has been made using database scanning algorithms from sequence analysis tools [1].

The current release of PRINTS (32.0) contains 1600 families with 9800 individual motifs. Each database entry is composed of the original set of sequences containing the family, its characteristic fingerprint composed of a set of motifs, the true-positive and the subfamily sequences that correspondingly match completely or partially the fingerprint of the family. The final motifs that generate a fingerprint are included in all the true-positive protein sequences and can be used as diagnostic rules for the family.

During the experiments conducted with the PRINTS database we have considered the set of true-positive examples of each family as the original training dataset. The objective was twofold. First we measured the effectiveness of the proposed greedy EM approach in terms of its ability to identify the motifs of the known fingerprints, and second we tried to figure out the possibility of discovering additional motifs within a real biological family and therefore to extend the definition of a given fingerprint. Moreover, the same datasets were examined with the MEME method. In this spirit, the objective of our experiments exceeds the simple comparison of two motif discovery techniques in real-world examples and proceeds towards the identification of additional unknown motifs that may potentially improve the established diagnostic signature of a family.

In our experimental study we have selected six (6) families from the PRINTS database, as illustrated in Table 5. These describe fingerprints for ribosomal proteins (PR00058, PR00061), secretion pathway protein C (PR00810) and glutathione S-transferases (PR001266, PR001267, PR001268). The selection of the motif length W was based on existing knowledge about the fingerprints design in the PRINTS database. Therefore, we have set W to be the smallest motif length in the set of motifs comprising a fingerprint and we have selected PRINTS families containing motifs of approximately equal length. It must be noted also that during experiments, discovered motifs with small number of occurrences were considered as redundant and thus were removed from the final set of the detected motifs (both in the greedy EM and the

[3] The PRINTS database is available at: *http://bioinf.man.ac.uk/dbbrowser/PRINTS/*

Accession number	*Name*	*Number of sequences*	*Number of motifs*	*Description*
PR00058	RIBOSOMALL5	16 (297)	6	A 6-element fingerprint that provides a signature for L5 ribosomal proteins
PR00061	RIBOSOMALL19	24 (120)	4	A 4-element fingerprint that provides a signature for L19 ribosomal proteins
PR00810	BCTERIALGSPC	6 (286)	2	A 2-element fingerprint that provides a signature for general secretion pathway protein C
PR01266	CSTRNSFRASEA	24 (222)	4	A 4-element fingerprint that provides a signature for alpha-class glutathione S-transferases
PR01267	CSTRNSFRASEM	22 (218)	4	A 4-element fingerprint that provides a signature for mu-class glutathione S-transferases
PR01268	CSTRNSFRASEP	19 (209)	3	A 3-element fingerprint that provides a signature for pi-class glutathione S-transferases

Table 5: Real datasets selected from the PRINTS database

MEME method), since we are working with fingerprints, i.e. sets of of motifs matching all the sequences in each family (*real-motifs*).

Each real-motif discovered in a dataset was evaluated in terms of the *information content* (IC) [3] of the proposed model. The information content for a real-motif j can be formally defined as follows:

$$IC_j = \sum_{k=1}^{W} \sum_{\alpha_l \in \Sigma} p_{lk}^j \log_2 \frac{p_{lk}^j}{\rho_l^1} \tag{19}$$

where ρ_l^1 indicates the overall background probability of letter α_l in the dataset. IC indicates the mutual information between motif model and single aminoacid frequencies. This score becomes maximal if the motif is well conserved and differs significantly from the background distribution. Thus, higher IC scores indicate clearer motif representations.

Table 6 summarizes the comparative results obtained using the six protein families. The superiority of the greedy EM algorithm over MEME is obvious not only in terms of the greater number of real-motifs discovered but also in terms of the degree of motif conservations as indicated by the IC scores. As in the case of the artificial datasets, when MEME considers as motifs more than one real-motifs located in equivalent number of sites upon the set of sequences, it is unable to identify them during future steps because of erasing their occurrences in the dataset of substrings. Therefore, the protein family fingerprints discovered by the MEME algorithm in most cases do not contain a great number of real-motifs and the results

are poor. It must be noted also that even if the "one motif occurrence per sequence" motif model was applied, MEME was unable to reach the number of motifs discovered by the proposed greedy EM algorithm.

On the other hand, the proposed greedy EM algorithm specifies an iterative procedure of adding new components together with a combined scheme of local and global search that results in better fitting of multiple-component mixture models, since it overcomes the problem of poor initialization of the component parameters. As illustrated in Table 6 the number of the discovered motifs in these protein families are not only greater with respect to MEME but also it is greater than the number of motifs specified in the PRINTS database. This means that the proposed method has led to the discovery of larger fingerprints (containing more motifs) and thus constitutes a promising tool for biological sequence analysis.

4 Conclusions

In this paper we have proposed a greedy EM algorithm for solving the multiple motif discovery problem in biological sequences. Our approach describes the problem through likelihood maximization by mixture learning using the EM algorithm. It learns a mixture of motifs model in a greedy fashion by iteratively adding new components. This is achieved through a combined scheme of local and global search which ensures fine tuning of the parameter vector of the new component. In addition a hierarchical clustering procedure is proposed based on the notion of kd-trees, which results in partitioning the (usually) large datasets (containing all substrings of length W) into a remarkable smaller number of candidate motif-models used for global searching. As it has been experimentally shown, this partitioning technique constitutes an effective strategy which manages to significantly reduce the time complexity for global searching without affecting the performance of the whole algorithm.

We have studied the performance of the proposed algorithm in several artificial and real biological datasets, including hard problems of almost indiscernible motif instances. Comparative results have also been provided through the application of the MEME approach which exhibits analogies to our method providing also an iterative algorithm of learning mixture models. The differences between the two approaches have already been highlighted throughout this paper, while experiments have shown the superiority of the greedy EM in discovering larger number of more distinguishable (clearer) motifs as suggested by the information content measure. The results obtained from the experimental study with the PRINTS database have also proved the ability of the greedy method in expanding protein fingerprints (larger number of discovered motifs) that is of great biological interest. It must be noted that our approach has been developed mainly in an attempt to overcome some limitations of the MEME scheme,

Problem *Acc. number*	*Greedy EM* *Motif*	*IC*	*MEME* *Motif*	*IC*
PR00058	I	72.4434	I′	63.2093
	II	72.4559	II′	56.9837
($W = 20$)	III	72.1314	III′	49.4185
	IV	67.3621	IV′	46.0057
	V	69.4684		
	VI	69.7300		
	VII	65.3427		
	VIII	57.0378		
PR00061	I	70.6732	I′	59.8767
($W = 24$)	II	65.2676	II′	62.6215
	III	61.6727	III′	53.1911
PR00810	I	37.2183	I′	34.3275
	II	35.2812	II′	33.9981
($W = 10$)	III	34.7850	III′	31.8698
	IV	33.0999	IV′	30.0946
	V	33.2078		
	VI	29.8507		
PR01266	I	61.9772	I′	57.7944
	II	59.2620	II′	57.3155
($W = 15$)	III	57.0082	III′	55.9137
	IV	56.6980	IV′	53.9237
	V	54.5780	V′	43.7600
	VI	50.3692	VI′	38.6581
	VII	45.3550		
	VIII	38.6579		
	IX	44.5795		
PR01267	I	52.8780	I′	52.4928
	II	52.4925	II′	51.2751
($W = 13$)	III	48.8996	III′	46.3277
	IV	50.4470	IV′	46.2217
	V	48.7069	V′	45.0163
	VI	48.5278	VI′	41.6475
	VII	44.3744		
	VIII	41.9956		
	IX	42.1630		
	X	40.3605		
PR01268	I	59.7265	I′	57.5664
	II	57.0125	II′	56.2638
($W = 17$)	III	58.7029	III′	53.1302
	IV	57.5660		
	V	56.2634		
	VI	54.2335		
	VII	51.4565		
	VIII	55.0629		
	IX	53.1297		

Table 6: Comparative results between the greedy EM and the MEME for the PRINTS database

such as erasing input data each time a new motif is discovered using the assumption that this motif is correct, and limiting the model exclusively to the two-component case. Our technique actually overcomes these limitations based on recent methods for incremental mixture density estimation.

Ongoing research is mainly focused on working with multiple motifs of variable length. This can be viewed as a problem of expanding an existing model and determining the correct number of its parameters (the optimum width of the motif). Several model selection techniques can be adopted for this reason that have been proposed mainly for Gaussian mixture models, such as the likelihood ratio test (LRT), the minimum description length (MDL), the Markov chain Monte Carlo (MCMC), the Bayesian information criterion (BIC), the asymptotic information criterion (AIC) and some recent Bayesian approaches [18, 15].

Another direction of future work is the application of our greedy EM approach to classification problems of biological families. In particular our aim is to develop a modification of the method in order to deal with biological sequences of several categories (families). It is expected that such an approach will constitute a powerful tool for the construction of highly accurate classification systems for biological sequences.

References

[1] T. K. Attwood, M. D. R. Croning, D. R. Flower, A. P. Lewis, J. E. Mabey, P. Scordis, J. Selley, and W. Wright. PRINT-S: the database formerly known as PRINTS. *Nucleic Acids Research*, 28(1):225–227, 2000.

[2] T. L. Bailey. *Discovering motifs in DNA and protein sequences: The appoximate common substring problem.* PhD thesis, University of California, San Diego, 1995.

[3] T. L. Bailey and C. Elkan. Unsupervised Learning of Multiple Motifs in Biopolymers Using Expectation Maximization. *Machine Learning*, 21:51–83, 1995.

[4] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, 1975.

[5] G. Berejano and G. Yona. Variations on probabilistic suffix trees: statistical modeling and prediction of protein families. *Bioinformatics*, 17(1):23–43, 2001.

[6] A. Brāzma, I. Jonasses, I. Eidhammer, and D. Gilbert. Approaches to the automatic discovery of patterns in biosequences. *Journal of Computational Biology*, 5(2):277–303, 1998.

[7] B. Bréjova, C. DiMarco, T. Vinař, S. R. Hidalgo, G. Holguin, and C. Patten. Finding patterns in biological sequences. Project Report for CS798g, University of Waterloo, 2000.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood estimation of a mixing distribution. *J. Roy. Statist. Soc. B*, 39:1–38, 1977.

[9] R. Durbin, S. Eddy, A. Krough, and G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Protein and Nucleic Acid.* Cambridge University Press, 1998.

[10] K. Hofmann, P. Bucher, L. Falquet, and A. Bairoch. The PROSITE database, its status in 1999. *Nucleic Acids Research*, 27:215–219, 1999.

[11] R. Hughey and A. Krogh. SAM: sequence alignment and modeling software system. Technical Report UCSC-CRL-96-22, University of California, Santa Cruz, CA, 1998.

[12] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwland, and J. C. Wootton. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, 226:208–214, 1993.

[13] J. Q. Li and A. R. Barron. Mixture density estimation. In *Advances in Neural Information Processing Systems 12.* The MIT Press, 2000.

[14] A. Likas, N. Vlassis, and J. J. Verbeek. The global k-means clustering algorithm. Technical Report IAS-UVA-01-02, Computer Science Institute, University of Amsterdam, The Netherlands, 2001.

[15] G. M. McLachlan and D. Peel. *Finite Mixture Models.* New York: John Wiley & Sons, Inc., 2001.

[16] R. A. Render and H. F. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26(2):195–239, 1984.

[17] I. Rigoutsos, A. Floratos, L. Parida, Y. Gao, and D. Platt. The Emergency of Pattern Discovery Techniques in Computational Biology. *Metabolic Engineering*, 2:159–177, 2000.

[18] S. J. Roberts, D. Husmeier, I. Rezek, and W. Penny. Bayesian Approaches to Gaussian Mixture Modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1133–1142, 1998.

[19] A. J. Smola, O. L. Mangasarian, and B. Schölkopf. Space kernel feature analysis. Technical Report, Data Mining Institute, University of Wisconsini, Madison, 1999.

[20] R. F. Sproull. Refinements to nearest-neighbor searching in k-dimensional trees. *Algorithmica*, 6:579–589, 1991.

[21] J. J. Verbeek, N. Vlassis, and B. Kröse. Efficient greedy learning of Gaussian mixtures. Technical Report IAS-UVA-01-04, Computer Science Institute, University of Amsterdam, The Netherlands, 2001.

[22] N. Vlassis and A. Likas. A greedy EM algorithm for Gaussian mixture learning. *Neural Processing Letters*, 15(1), 2002. In print.