Chapter 1
Using a solver over the string pattern domain to analyze gene promoter sequences

Christophe Rigotti, Ieva Mitašiūnaitė, Jérémy Besson, Laurène Meyniel,
Jean-François Boulicaut, and Olivier Gandrillon

Abstract This chapter illustrates how inductive querying techniques can be used to support knowledge discovery from genomic data. More precisely, it presents a data mining scenario to discover putative transcription factor binding sites in gene promoter sequences. We do not provide technical details about the used constraint-based data mining algorithms that have been previously described. Our contribution is to provide an abstract description of the scenario, its concrete instantiation and also a typical execution on real data. Our main extraction algorithm is a complete solver dedicated to the string pattern domain: it computes string patterns that satisfy a given conjunction of primitive constraints. We also discuss the processing steps necessary to turn it into a useful tool. In particular, we introduce a parameter tuning strategy, an appropriate measure to rank the patterns, and the post-processing approaches that can be and have been applied.

Christophe Rigotti
Laboratoire LIRIS CNRS UMR 5205, INSA-Lyon, 69621 Villeurbanne, France e-mail: christophe.rigotti@insa-lyon.fr

Ieva Mitašiūnaitė
Faculty of Mathematics and Informatics, Vilnius University, Lithuania e-mail: ieva.mitasiunaite@gmail.com

Laurène Meyniel
Laboratoire LIRIS CNRS UMR 5205, INSA-Lyon, 69621 Villeurbanne, France e-mail: laurene.meyniel@wanadoo.fr

Jean-François Boulicaut
Laboratoire LIRIS CNRS UMR 5205, INSA-Lyon, 69621 Villeurbanne, France e-mail: jean-francois.boulicaut@insa-lyon.fr

Olivier Gandrillon
Centre de Génétique Moléculaire et Cellulaire CNRS UMR 5534, Université Claude Bernard Lyon I, 69622 Villeurbanne, France e-mail: gandrillon@cgmc.univ-lyon1.fr

1.1 Introduction

Understanding the regulation of gene expression remains one of the major challenges in molecular biology. One of the elements through which the regulation works is the initiation of the transcription by the interaction between short DNA sequences (called *gene promoters*) and multiple activator and repressor proteins called Transcription Factors (TFs). These gene promoter elements are located in sequences called promoter sequences, that are DNA sequences close to the sequences that encode the genes. In fact, on a promoter sequence various compounds can bind, having then an impact on the activation/repression of the gene associated to this promoter sequence, but among these compounds, the TFs are known to play a very important role. Therefore, many researchers are working on TFs and Transcription Factor Binding Sites (TFBSs). These are subsequences of the promoter sequences where the TFs are likely to bind. In practice, identifying patterns corresponding to putative TFBSs help the biologists to understand which TFs are involved in the regulation of the different genes.

In this study, we report our contribution to gene promoter sequence analysis and TFBS discovery by means of generic constraint-based data mining techniques over strings. Indeed, we consider that the promoter sequences are sequences of nucleotides represented by the symbols A, C, G and T (i.e., a data sequence is a string over the alphabet $\{A, C, G, T\}$ and a pattern is a substring in such sequences). Contrary to many approaches that support motif discovery in promoter sequences, we do not take into account domain knowledge about that quite specific type of strings. Instead, we use a generic solver over the string pattern domain.

The recent advances in constraint-based mining (see [2] and [7] for an overview), and more generally the current developments in the domain of inductive querying (i.e., the vision proposed in [10]), lead to the design of many mining tools based on the constraint paradigm. We have now at hand scalable complete solvers, in particular over the string domain, that can be used to find substring patterns in sequences. However, this is far from being sufficient to tackle a real application. In this chapter, we present all the necessary processing, beyond the pattern extraction, that is needed to support knowledge discovery from a biological perspective, hopefully leading to the discovery of new putative TFBSs. First, we describe the corresponding data mining abstract scenario, and then we give its concrete instantiation. Finally, we illustrate its execution by means of a typical case study. We also give technical details about aspects that are important to run the scenario in practice. This includes, in particular, the tuning of the parameters in the early exploratory mining stage, the ranking of the patterns using a measure adapted to the domain, and the designed pattern post-processing technique to exhibit putative TFBSs.

Methodological and technical details about the method and the algorithms can be found in several papers. The *Marguerite* solver over the string pattern domain has been described in details in [14, 15]. A concrete instance of the scenario is described in the journal publication [16]. This is also where our measure of interest, the so-called TZI measure, is studied in depth. Our parameter tuning method has been introduced in [1]. Last by not least, the Ph.D. thesis [13] considers all these issues in detail.

The rest of the chapter is organized as follows. In Section 1.2, we present the scenario both at an abstract and instantiated level. Then, in Section 1.3, we describe the kind of patterns and the constraints that are handled by the solver. The parameter tuning strategy is discussed in Section 1.4 and the dedicated measure to rank the patterns is introduced in Section 1.5. Then, a typical example of a real execution of the scenario is presented in Section 1.6. Finally, we conclude with a short summary in Section 1.7.

1.2 A promoter sequence analysis scenario

Let us present the scenario which has been designed and used in our case study. First, we describe it in abstract terms and then we explain how it has been instantiated into a concrete scenario.

1.2.1 A generic scenario

This abstract view describes the main steps of the general process that has been studied. It can be decomposed as a workflow containing the following sequence of operations:

- Use the results of SAGE experiments [21] to select two groups of genes, one group corresponding to genes active in a context (called the positive context), and the second group corresponding to genes active in an opposite context (called the negative context). These positive vs. negative issues are application dependent. Notice that SAGE is one technology for recording gene expression values in biological samples and that other popular approaches could be used, e.g., microarrays.
- Retrieve from a gene database the promoter sequences of the selected genes. Construct two sets D^+ and D^- of promoter sequences: one for the genes active in the positive context (D^+), and the other for the genes active in the negative context (D^-).
- Perform a differential extraction of substrings between datasets D^+ and D^-, to find substrings frequent in D^+ and not frequent in D^-.
- Compute for each extracted substring a dedicated interestingness measure.
- Select some of the patterns, according to their ranking on the measure value and/or to their support in D^+ and/or support in D^-.
- Perform a complementary post-processing:
  - Cluster the set of selected patterns (pairwise alignment).

– In each cluster, perform a multiple alignment of the patterns in the cluster, to obtain a consensus motif (centroid) for each cluster.
– Search these consensus motifs in a database of known TF binding sites (e.g., $Transfac^{®}$ database [12]), to look for their corresponding TFs and the known functions of these TFs (if any).

The workflow of the whole process is depicted Figure 1.1. Notice that numerous efforts have given rise to a variety of computational methods to discover putative TFBSs in sets of promoters of co-regulated genes (see [16] for an overview). Among them two families can be distinguished: statistical or stochastic approaches, and combinatorial approaches [20]. Concerning the family of statistical and stochastic approaches, a recent review of the most widely used algorithms exhibits rather limited results [19]. The scenario presented in this chapter uses a combinatorial approach, and its main originality w.r.t. the other combinatorial algorithms, which allow to extract patterns from several datasets (e.g., SPEXS [3] or DRIM [9]), is that the maximal support threshold is set explicitly. This is particularly interesting, when there is a clear semantic cut between a positive and negative datasets, i.e., the negative dataset has an opposite biological sense (presence/absence of a mutation; addition or not of a given drug, etc.), and does not just represent random background.

1.2.2 Instantiation of the abstract scenario

We focus the search on putative TFBSs that could be used to regulate the transcription of the genes associated to promoter sequences of the positive dataset (D^+) while they are not likely to have an important impact on the regulation of the genes associated to the other set. To collect the sets D^+ and D^-, the method starts with a classical operation used in molecular biology: the search for differentially expressed genes[1], using SAGE experiments. This allows to obtain two groups of genes from which we derive two sets of associated promoter sequences using a promoter database. To look for putative TFBSs regulating the overexpressed genes, we choose the first set (the promoters of the overexpressed genes) to be used as a positive set, and the second set as a negative one[2]. The promoter sequences are sequences of compounds called bases. There are four different bases, commonly represented by the symbols A, C, G and T, and a sequence is simply represented by a string over the alphabet $\{A,C,G,T\}$. Then the method consists in finding patterns that are substrings occurring in at least α_{min} promoters from the positive set and in at most α_{max} promoters from the negative set, where the parameter α_{min} (resp. α_{max}) is supposed to be a large

[1] It consists in comparing two biological situations, Sit_1 and Sit_2, in order to obtain two groups of genes: one that is up-regulated, and the other one that is down-regulated, when going from Sit_1 to Sit_2.

[2] Notice that if we exchange the positive and negative datasets, then we could find putative TFBSs regulating the underexpressed genes.

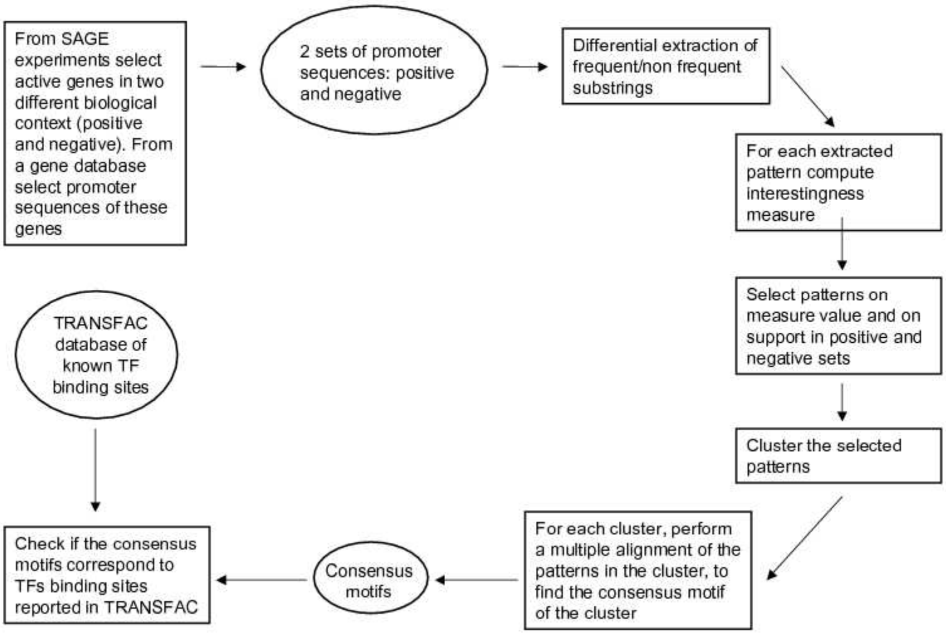
From SAGE experiments select active genes in two different biological context (positive and negative). From a gene database select promoter sequences of these genes
2 sets of promoter sequences: positive and negative
Differential extraction of frequent/non frequent substrings
For each extracted pattern compute interestingness measure
Select patterns on measure value and on support in positive and negative sets
Cluster the selected patterns
For each cluster, perform a multiple alignment of the patterns in the cluster, to find the consensus motif of the cluster
Consensus motifs
Check if the consensus motifs correspond to TFs binding sites reported in TRANSFAC
TRANSFAC database of known TF binding sites

Fig. 1.1 Workflow of the abstract scenario.

(resp. small) threshold value. Typical sizes of the promoter sequences are about a few thousands of symbols, and the positive and negative datasets contain each a few tens of such sequences.

We consider two kinds of patterns: Exact Matching Patterns (EMPs) and Soft Matching Patterns (SMPs). Both are strings of bases, but they differ in the way their *supports* are defined. The support of an EMP in a dataset is the number of sequences of the dataset that contain at least one exact occurrence of this EMP. Let α_{dist} be a given threshold, termed the *soft matching* threshold, then the support of a SMP is the number of sequences containing at least one soft occurrence of the pattern, where a soft occurrence is a part of the sequence different from the pattern in at most α_{dist} positions (i.e., the Hamming distance between this part of the sequence and the pattern is at most α_{dist}). Both SMPs and EMPs are necessary: SMPs allow to gather the degenerated TFBSs while EMPs are dedicated to pick out the conserved ones.

The two kinds of patterns are extracted using a solver over the string pattern domain called *Marguerite* (see Section 1.3). This tool performs a differential extraction of patterns between the two sets of sequences D^+ and D^-. To run an extraction, the user has to set the four following constraints: L the length of the patterns, α_{min} their minimal support in D^+, α_{max} their maximal support in D^-, and α_{dist} the soft matching threshold (for SMPs). *Marguerite* is complete in the sense that it finds all possible patterns satisfying the constraints according to the user setting. In the case of SMPs, the solver enforces an additional constraint: the patterns must have at-least one exact occurrence in D^+. This additional constraint enables to focus on SMPs that appear at-least one time in a non-degenerated way. Concerning the use of the solver, setting four parameters is not an easy task, so we developed and used a dedicated parameter tuning tool (see Section 1.4).

In order to assess the significance of a pattern we used the notion of Twilight Zone (TZ) [11] to build a Twilight Zone Indicator (TZI). A twilight zone is a zone in a parameter space, where we are likely to obtain patterns produced by the random background. For a pattern ϕ of length L, the indicator $TZI(\phi)$ is an estimate of the minimum number of patterns of length L, due to the random background, that are likely to be extracted together with ϕ, in the most stringent conditions (i.e., with the strongest constraints, that still lead to the extraction of ϕ). The computation of the TZI is detailed in Section 1.5. It is based on the same hypothesis made in [11]: the data sequences are composed of independent and uniformly distributed nucleotides, and the possible overlap of the occurrences of the patterns is considered to have a limited impact on the number of extracted patterns. In addition, we suppose that the positive and the negative datasets are independent.

During the next step, the biologist browses and ranks the patterns (according to the TZI measure, and to the support of the patterns in D^+ and D^-) and then he/she selects some promising ones.

On these selected patterns, the following post-processing is applied (see Section 1.6.3). First the similar patterns are grouped by performing a hierarchical clustering. Then, for each cluster we compute the average of the TZI of the patterns in the cluster, and in each cluster, the patterns are aligned with a multiple alignment tool (*MultAlin* [5]) to build a *consensus pattern* of the cluster. Finally, the concensus patterns are checked w.r.t. the $Transfac^{®}$ [12] database, to find out if they are known TFBSs, close to some known TFBSs or unknown.

1.3 The *Marguerite* solver

We introduce the solver *Marguerite* which supports inductive querying on strings. It has been used in the scenario described in this chapter. We define more precisely the patterns and constraints handled by this solver. More details can be found in [14, 15].

Let Σ be a finite alphabet (in the scenario $\Sigma = \{A, C, G, T\}$), then a string ϕ over Σ is a finite sequence of symbols from Σ. The language of patterns $\mathscr{L}$ is Σ^*, i.e, the set of all strings over Σ. A string dataset D is a multi-set[3] of strings from Σ^*. The

[3] The dataset may contain several times the same string.

length of a string ϕ is denoted $|\phi|$. A substring ϕ' of ϕ is a sequence of contiguous symbols in ϕ.

An *exact occurrence* of a pattern ϕ is simply a substring of a string in D that is equal to ϕ. The *exact support* of ϕ, denoted $supp_E(\phi,D)$, is the number of strings in D that contain at least one exact occurrence of ϕ. Notice that multiple occurrences of a pattern in the same string do not change its support.

Let α_{dist} be a positive integer, then an (α_{dist})*-soft occurrence* of a pattern ϕ is a substring ϕ' of a string in D, having the same length as ϕ and such that $hamming(\phi,\phi') \leq \alpha_{dist}$, where $hamming(\phi,\phi')$ is the Hamming distance between ϕ and ϕ' (i.e., the number of positions where ϕ and ϕ' are different). The (α_{dist})-*soft support* of ϕ is the number of strings in D that contain at least one (α_{dist})-soft occurrence of ϕ. It is denoted $supp_S(\phi,D,\alpha_{dist})$.

Example 1. If $D = \{atgcaaac, acttggac, gatagata, tgtgtgtg, gtcaactg\}$, then we have $supp_E(gac,D) = 1$ since only string *acttggac* contains *gac*, and we also have $supp_S(gac,D,1) = 3$ because *acttggac*, *gatagata* and *gtcaactg* contain some 1-soft occurrences of *gac*.

Definition 1 (Frequency constraints). In the case of the exact support, given a threshold value f, the minimal (resp. maximal) frequency constraint is $MinFr(\phi, D, f) \equiv supp_E(\phi,D) \geq f$ (resp. $MaxFr(\phi,D,f) \equiv supp_E(\phi,D) \leq f$). For the (α_{dist})-soft support, the constraints are defined as $MinFr(\phi,D,f) \equiv supp_S(\phi,D,\alpha_{dist}) \geq f \wedge supp_E(\phi,D) \geq 1$ and $MaxFr(\phi,D,f) \equiv supp_S(\phi,D,\ \alpha_{dist}) \leq f$.

Notice that, in the case of the soft support, our definition of *MinFr* enforces the presence of at least one exact occurrence, in order to discard patterns that only occur as degenerated instances.

The generic conjunction of constraints handled by *Marguerite* is:
$\mathscr{C} \equiv MinFr(\phi,D^+,\alpha_{min}) \wedge MaxFr(\phi,D^-,\alpha_{max}) \wedge |\phi| = L$, where D^+ and D^- are string databases, α_{min} and α_{max} are frequency thresholds, and L is a user defined pattern length.

The algorithms used by *Marguerite* [14, 15] are based on the generic algorithm FAVST [6], designed for the efficient extraction of strings under combination of constraints, taking advantage of the so-called Version Space Tree (VST) [8] data structure. *Marguerite* extends FAVST to degenerated patterns discovery through similarity and soft-support constraints. It is implemented in $C/C++$ and can be used to compute both Exact Matching Patterns (EMPs) and Soft Matching Patterns (SMPs) in a complete way (i.e., all patterns satisfying the constraints are outputted).

1.4 Tuning the extraction parameters

In an exploratory data mining task based on pattern extraction, one of the most commonly used parameter tuning strategies, in the early exploration stage, is to run a few experiments with different settings, and to simply count the number of patterns that are obtained. Then, using some domain knowledge, the user tries to guess some potentially interesting parameter settings. After that stage, the user enters a more iterative process, in which she/he also looks at the patterns themselves and at their scores (according to various quality measures), and uses her/his knowledge of the domain to focus on some patterns and/or to change the parameters by some *local* variations of their values.

To support this early exploratory stage, so that the user can guess promising initial parameter settings, we decided to probe the parameter space in a more systematic way, so that it could be possible to provide graphics that depict the extraction landscape, i.e., the number of patterns that will be obtained for a wide range of parameter values. This idea is very simple, and many (if not all) of the practitioners have one day written their own script/code to run such sets of experiments. However, in many cases, the cost of running real extractions for hundreds of different parameter settings is clearly prohibitive.

Instead of running real experiments, a second way is to develop an analytical model, that estimates the number of patterns satisfying the constraint $\mathscr{C}$, with respect to the distribution of the symbols and the structure (number of strings and size) of the datasets, and with respect to the values of the parameters used in $\mathscr{C}$. In this approach, an important effort has to be made on the design of the model, and in most cases this is a non-trivial task. For instance, to the best of our knowledge, in the literature there is no analytical model of the number of patterns satisfying $\mathscr{C} \equiv MinFr(\phi, D^+, \alpha_{min}) \wedge MaxFr(\phi, D^-, \alpha_{max}) \wedge |\phi| = L$ when soft-occurrences are used to handle degenerated patterns (even in the simple case where $\alpha_{dist} = 1$). Designing an analytical model to handle this case is certainly not straightforward, in particular because of the specific symbol distribution that has to be incorporated in the model.

We developed a third approach based on the following key remark. When a pattern ϕ is given, together with the distribution of the symbols, the structure of the datasets and the values of the parameters in $\mathscr{C}$, we can compute $P(\phi\ sat.\ \mathscr{C})$ the probability that ϕ satisfies $\mathscr{C}$ in this dataset. In most cases, designing a function to compute $P(\phi\ sat.\ \mathscr{C})$ is rather easy in comparison to the effort needed to exhibit an analytical model that estimates the number of patterns satisfying the constraint $\mathscr{C}$. Having at hand a function to compute $P(\phi\ sat.\ \mathscr{C})$, the next step is then to estimate the total number of patterns that will be extracted, but without having to compute $P(\phi\ sat.\ \mathscr{C})$ for all patterns in the pattern space. Therefore, we propose a simple pattern space sampling approach, that leads to a fast and accurate estimate of the number of patterns that will be extracted. Finally, we can compute such an estimate for a large number of points in the parameter space and provide views of the whole extraction landscape.

To determine $P(\phi\ sat.\ \mathscr{C})$, we first compute the different frequencies of occurrence of the symbols. We consider that all occurrences of the symbols are independent, and then, for a given pattern ϕ we can easily compute the probability that ϕ occurs in a string of a given length. If we suppose that all strings in the dataset have the same length, the probability to appear in each string is the same, and we can use a binomial law to obtain the probability for this pattern to satisfy the constraint $MinFr(\phi, D^{+}, \alpha_{min})$ and the probability to satisfy the constraint $MaxFr(\phi, D^{-}, \alpha_{max})$. Finally, if we suppose that the datasets D^{+} and D^{-} are independent, we can multiply these two probabilities to obtain $P(\phi \; sat. \; \mathscr{C})$.

Let $S_{\mathscr{C}}$ be the set of patterns in $\mathscr{L}$ that satisfy the constraint $\mathscr{C} \equiv MinFr(\phi, D^{+}, \alpha_{min}) \wedge MaxFr(\phi, D^{-}, \alpha_{max}) \wedge |\phi| = L$, using $P(\phi \; sat. \; \mathscr{C})$ we can estimate $|S_{\mathscr{C}}|$ by sampling the pattern space as follows. Let us associate to each pattern ϕ a random variable X_{ϕ}, such that $X_{\phi} = 1$ when ϕ satisfy $\mathscr{C}$ and $X_{\phi} = 0$ otherwise. Then $|S_{\mathscr{C}}| = \sum_{\phi \in \mathscr{L}} X_{\phi}$. Considering the expected value of $|S_{\mathscr{C}}|$, by linearity of the expectation operator we have $E(|S_{\mathscr{C}}|) = \sum_{\phi \in \mathscr{L}} E(X_{\phi})$. Since $E(X_{\phi}) = 1 \times P(X_{\phi} = 1) + 0 \times P(X_{\phi} = 0)$, then $E(|S_{\mathscr{C}}|) = \sum_{\phi \in \mathscr{L}} P(\phi \; sat. \; \mathscr{C})$. Let S_L be the set of patterns in $\mathscr{L}$ that satisfy $|\phi| = L$. As $P(\phi \; sat. \; \mathscr{C}) = 0$ for all patterns that do not satisfy $|\phi| = L$, we have $E(|S_{\mathscr{C}}|) = \sum_{\phi \in S_L} P(\phi \; sat. \; \mathscr{C})$.

Computing this sum over S_L could be prohibitive, since we want to obtain the values of $E(|S_{\mathscr{C}}|)$ for a large number of points in the parameter space. Thus we estimate $E(|S_{\mathscr{C}}|)$ using only a sample of the patterns in S_L. Let S_{samp} be such a sample, then we use the following value as an estimate of $E(|S_{\mathscr{C}}|)$:

$$\frac{|S_L|}{|S_{samp}|} \times \sum_{\phi \in S_{samp}} P(\phi \; sat. \; \mathscr{C})$$

In practice, many techniques can be used to compute the sample. In our experiments, we use the following process:

- Step 1: build an initial sample S_{samp} of S_L (sampling with replacement) of size 5% of $|S_L|$ and compute the estimate of $E(|S_{\mathscr{C}}|)$.
- Step 2: go on sampling with replacement to add 1000 elements to S_{samp}. Compute the estimate, and if the absolute value of the difference between the new estimate and the previous one is greater than 5% of the previous estimate, then iterate on Step 2.

1.5 An objective interestingness measure

The notion of Twilight Zone (TZ) [11] has been originally proposed to characterize the *subtle motifs*, i.e., motifs that can not be distinguished (no statistically significant difference) from random patterns (patterns due to the random background). In this context, the TZ was defined as the set of values of the scoring function for which we can expect to have some random patterns exhibiting such score values. Let us consider the notion of extraction parameters in a broad sense, including structural properties of the dataset (e.g., number of sequences, length of the sequences) and mining constraints (e.g., selection threshold according to one or several measures, length of the patterns). Then, the TZ can be seen as a region (or a set of regions) in the parameter space, where we are likely to obtain random patterns among the extracted patterns, these random pattern having scores as good (or event better) than the *true* patterns.

We can now define a Twilight Zone Indicator (TZI) to rank the patterns in the case of differential extractions. Let ϕ be a pattern, occurring in $support^+(\phi)$ sequences of the positive dataset, and in $support^-(\phi)$ sequences of the negative dataset. Then, TZI(ϕ) is an estimate of the number of random patterns, having the same length as ϕ, that will be extracted using $\alpha_{min} = support^+(\phi)$ and $\alpha_{max} = support^-(\phi)$, i.e., using the most selective constraints that still permit to obtain ϕ (since for larger α_{min} and/or lower α_{max} threshold values, ϕ will not satisfy the constraints and will not be retained during the extraction). The higher is $TZI(\phi)$, the *deeper* is ϕ in the twilight zone, and thus likely to be retrieved among a larger collection of patterns due to the random background that cannot statistically be distinguished from ϕ. Then, in practice, we will select patterns having a low TZI, to expect to have patterns that are not due to the random background.

At first glance, the number of patterns satisfying $\alpha_{min} = support^+(\phi)$ and $\alpha_{max} = support^-(\phi)$ could be obtained using the sampling based technique presented in Section 1.4. Unfortunately, if this approach can help the user to find estimates of the number of patterns in wide ranges of parameter values, the extracted patterns themselves can represent many much more $(support^+, support^-)$ pairs, than the number of $(\alpha_{min}, \alpha_{max})$ pairs considered during the parameter setting stage. For instance, it can make sense for the expert to explore the α_{min} setting between 20 and 40, while real patterns that are extracted using $\alpha_{min} \in [20, 40]$ could have support larger than 40, and not only in $[20, 40]$. In order to avoid the cost of computing the sampling based estimate for each extracted pattern, we now discuss an alternative way to obtain such an estimate. This second estimate is less accurate, in the sense that it does not take into account the difference among the frequencies of the symbols, but it uses a direct analytical estimate, i.e., without sampling. It can be much more relevant in practice.

We consider that all the sequences have the same length, denoted G. In this context, we want to estimate the number of SMP patterns of length L that will be extracted under the thresholds α_{min}, α_{max} and α_{dist}. Let us notice that estimating the number of EMP is a particular case, where α_{dist} is set to 0. As in [11], we suppose that the data sequences are composed of independent and uniformly distributed symbols, having the same occurrence probability, and that the overlapping of the occurrences of the patterns has a negligible impact on the number of patterns extracted (since $L \ll G$). Additionally, as in the previous section, we suppose that the two datasets are independent.

Occurrences at a given position

The data sequences are gene promoter sequences. On such a given vocabulary, we have 4^L different possible strings of length L. The hypotheses made on the distribution of the symbols imply that the probability that a pattern ϕ of length L has an exact occurrence starting at a given position in a sequence[4] is:

$P(\textit{exact occ. of } \phi \textit{ at one position}) = \frac{1}{4^L}$.

From an exact occurrence of ϕ, one can construct the soft occurrences of ϕ within an Hamming distance α_{dist} by placing k substitutions in $\binom{L}{k}$ possible ways, with $k \in \{0, \ldots \alpha_{dist}\}$. Since we have 4 symbols, then for each position were we have a substitution, we have 3 different possible substitutions. Thus, for a pattern ϕ, there are $\sum_{k=0}^{\alpha_{dist}} \binom{L}{k} \times 3^k$ strings that are soft occurrences of ϕ. Then, the probability that a pattern has a soft occurrence starting at a given position in a sequence is:

$P(\textit{soft occ. of } \phi \textit{ at one position}) = \frac{\sum_{k=0}^{\alpha_{dist}} \binom{L}{k} \times 3^k}{4^L}$.

In the following, we also need the probability that a pattern ϕ has a *strict* soft occurrence starting at a given position (a *strict* soft occurrence of ϕ is a soft occurrences of ϕ that is not an exact occurrence). In this case we have simply:

$P(\textit{strict soft occ. of } \phi \textit{ at one position}) = \frac{\sum_{k=1}^{\alpha_{dist}} \binom{L}{k} \times 3^k}{4^L}$.

Occurrences in a random sequence

In a sequence, there are $(G - L + 1)$ possible positions to place the beginning of an occurrence of ϕ. Since $L \ll G$, for the sake of simplicity, we approximate a number of possible positions by G. Then, considering that the occurrence overlap has a negligible impact, the probability that there is no soft occurrence of ϕ in a random sequence is:

$P(\textit{no soft occ. of } \phi \textit{ in a seq.}) = (1 - P(\textit{soft occ. of } \phi \textit{ at one position}))^G$.

The probability that there is at least one soft occurrence of ϕ in a sequence is:

$P(\textit{exists soft occ. of } \phi \textit{ in a seq.}) = 1 - (1 - P(\textit{soft occ. of } \phi \textit{ at one position}))^G$.

Similarly, the probability that there is at least one strict soft occurrence of ϕ is:

$$P(\textit{exists strict soft occ. of } \phi \textit{ in a seq.}) = \\ 1 - (1 - P(\textit{strict soft occ. of } \phi \textit{ at one position}))^G.$$

Finally, the probability that there is at least one exact occurrence is:

$P(\textit{exists exact occ. of } \phi \textit{ in a seq.}) = 1 - (1 - \frac{1}{4^L})^G$.

Minimum support constraint

To determine $P(\phi \textit{ sat. min. supp.})$, i.e., the probability of ϕ to satisfy the minimum support constraint, let us define X as the number of sequences, in the positive

[4] Except the last $L - 1$ positions.

dataset, that contains at least one exact occurrence of ϕ. The probability $P(\phi$ *sat. min. supp.*$)$ can be decomposed using the conditional probability of ϕ *sat. min. supp.* given the value of X, as follows:

$$P(\phi \textit{ sat. min. supp.}) = \sum_{i=1}^{N^+} (P(X=i) \times P(\phi \textit{ sat. min. supp.} | X=i)) \tag{1.1}$$

Notice that the sum starts at $i = 1$, and not at $i = 0$, since the pattern must have at least one exact occurrence in the positive dataset (see Section 1.3).

The variable X follows a binomial distribution $B(N^+,\ P($*exists exact occ. of* ϕ *in a seq.*$))$, where N^+ is the number of sequences in the positive dataset. Thus we have:

$$P(X=i) = \binom{N^+}{i} \times P(\textit{exists exact occ. of } \phi \textit{ in a seq.})^i \\ \times (1 - P(\textit{exists exact occ. of } \phi \textit{ in a seq.}))^{N^+-i}.$$

$P(\phi$ *sat. min. supp.*$|X = i)$ is the probability that ϕ satisfies the minimum support constraint, given that exactly i sequences contain at least one exact occurrence of ϕ. This also means that $(N^+ - i)$ sequences do not have any exact occurrence of a pattern. Then, according to i, there are two cases:

1. If $i \geq \alpha_{min}$ then $P(\phi$ *sat. min. supp.*$|X = i)) = 1$ since the constraint is already satisfied by the i sequences that contain each at least one exact occurrence of ϕ.
2. If $i < \alpha_{min}$ then $P(\phi$ *sat. min. supp.*$|X = i)$ is equal to the probability that at least $(\alpha_{min} - i)$ of the $(N^+ - i)$ remaining sequences contain at least one strict soft occurrence. This number of sequences that contain at least one strict soft occurrence of ϕ also follows a binomial distribution $B(N^+ - i, P($*exists strict soft occ. of* ϕ *in a seq.*$))$. Then we have:

$$P(\phi \textit{ sat. min. supp.} | X=i)) = \textstyle\sum_{z=\alpha_{min}-i}^{N^+-i} (\binom{N^+-i}{z} \\ \times P(\textit{exists strict soft occ. of } \phi \textit{ in a seq.})^z \\ \times (1 - P(\textit{exists strict soft occ. of } \phi \textit{ in a seq.}))^{N^+-i-z}).$$

It means that we can provide $P(\phi$ *sat. min. supp.*$)$ by computing the sum in Equation 1.1 and $P(\phi$ *sat. min. supp.*$|X = i)$ according to the two cases above.

Maximum Support constraint

Let Y be the number of sequences that support ϕ in the negative dataset. A pattern ϕ satisfies the maximum support constraint with threshold α_{max} if $Y \leq \alpha_{max}$. The variable Y follows a binomial distribution $B(N^-,\ P($*exists soft occ. of* ϕ *in a seq.*$))$, where N^- is the number of sequences in the negative dataset. Then the probability that ϕ satisfies the maximum support constraint is:

$$P(\phi \textit{ sat. max. supp.}) = \textstyle\sum_{z=0}^{\alpha_{max}} \binom{N^-}{z} \\ \times P(\textit{exists soft occ. of } \phi \textit{ in a seq.})^z \\ \times (1 - P(\textit{exists soft occ. of } \phi \textit{ in a seq.}))^{N^- - z}.$$

Conjunction of Minimum Support and Maximum Support constraints

Given our hypothesis that the positive and negative datasets are independent, the probability that a pattern satisfies a conjunction of minimum support and maximum support constraints is:

$$P(\phi \textit{ sat. min. and max. supp.}) = P(\phi \textit{ sat. min. supp.}) \times P(\phi \textit{ sat. max. supp.}).$$

Number of expected patterns and Twilight Zone Indicator

Let $ENP(\ L, \alpha_{min}, \alpha_{max}, \alpha_{dist})$ be the Expected Number of Patterns of length L that will be extracted under the thresholds α_{min}, α_{max} and α_{dist}. Since there are 4^L possible patterns of length L, and given the hypothesis that the overlapping of the occurrences of the patterns has a negligible impact on the number of extracted patterns, we can approximate $ENP(L, \alpha_{min}, \alpha_{max}, \alpha_{dist})$ by $P(\phi \textit{ sat. min. and max. supp.}) \times 4^L$.

Finally, let ϕ be a pattern, occurring in $support^+(\phi)$ sequences of the positive dataset, and in $support^-(\phi)$ sequences of the negative dataset for a given α_{dist} threshold. Then, TZI(ϕ) is defined as $ENP(|\phi|, support^+(\phi), support^-(\phi), \alpha_{dist})$.

1.6 Execution of the scenario

In this section, we present a typical concrete execution of the whole scenario, in the context of the study of the TFs and TFBSs involved in the activation/repression of genes in reaction to the presence of the v-erbA oncogene, a chemical compound involved in the cell self-renewal process.

1.6.1 Data preparation

Using the SAGE technique [21], we identified two sets of genes: a set R of 29 genes repressed by v-ErbA and a set A of 21 genes activated by v-ErbA. Then, we collected the promoter sequences of all these genes (taking 4000 bases for each promoter). These promoter sequences have been extracted as described in [4]. Finally, we have built two datasets: D^+ (resp. D^-) containing the promoter sequences of the genes of set R (resp. A).

These two datasets represent two biologically opposite situations. As a result, we assume that computing string patterns that have a high support in D^+ and a small support in D^- is a way to identify putative binding sites of transcription factors involved in this activation/repression process induced by v-ErbA.

1.6.2 Parameter tuning

Patterns having slightly degenerated occurrences can be interesting in our context. Therefore, we look for SMP patterns using $\alpha_{dist} = 1$ for the soft support definition. The estimates are computed according to the sampling technique presented in Section 1.4 with respective frequencies of 0.23, 0.26, 0.27, 0.24 for symbols A, C, G and T. Representative graphics depicting portions of the extraction landscape, are presented in Figure 1.2, on the right.

A typical use of such graphics is, for instance, to look for points, in the parameter space, corresponding to a large support on D^+, but a low support on D^-, a large pattern size, and a rather small number of expected patterns. Such a point can be used as an initial guess of the parameters to perform the extractions. For instance, we may consider pattern size $= 10$, minimal support on D^+ of 15, and maximal support on D^- of 5. The graphic in the middle on the right for Figure 1.2 indicates that, for this setting, only about 1 pattern is expected.

Additionally, in Figure 1.2 on the left, we give the real numbers of extracted patterns. In practice, these graphics are not easily accessible, since in these experiments the running time of a single extraction with *Marguerite* (on a Linux platform with an Intel 2Ghz processor and 1Gb of RAM) ranges from tens of minutes to several hours[5], while for an estimate (graphics on the right) only a few tens of seconds is needed. Even though the global trends correspond to the estimates on the right, there are differences in some portions of the parameter space. For example, for the setting *pattern size* $= 10$, *minimal support* $= 15$, and *maximal support* $= 5$, we have about 100 extracted patterns while we expected only one. Such a difference suggests that these 100 patterns capture an underlying structure of the datasets, and that they are not simply due to the random background.

1.6.3 Post-processing and biological pattern discovery

Hierarchical clustering of SMPs

The hierarchical clustering of the SMPs patterns is performed using the *hclust* function of the package *stat* of the *R* environment [18]. The proximity between clusters is computed using the complete linkage method. To improve the quality and efficiency of the clustering, we process the SMPs by groups of patterns having the same length. To construct a distance matrix, we estimate the dissimilarity of

[5] Notice that for experiments using EMP (exact support) on these datasets, with similar parameter values, the running time is only about a few tens of seconds to a few minutes.

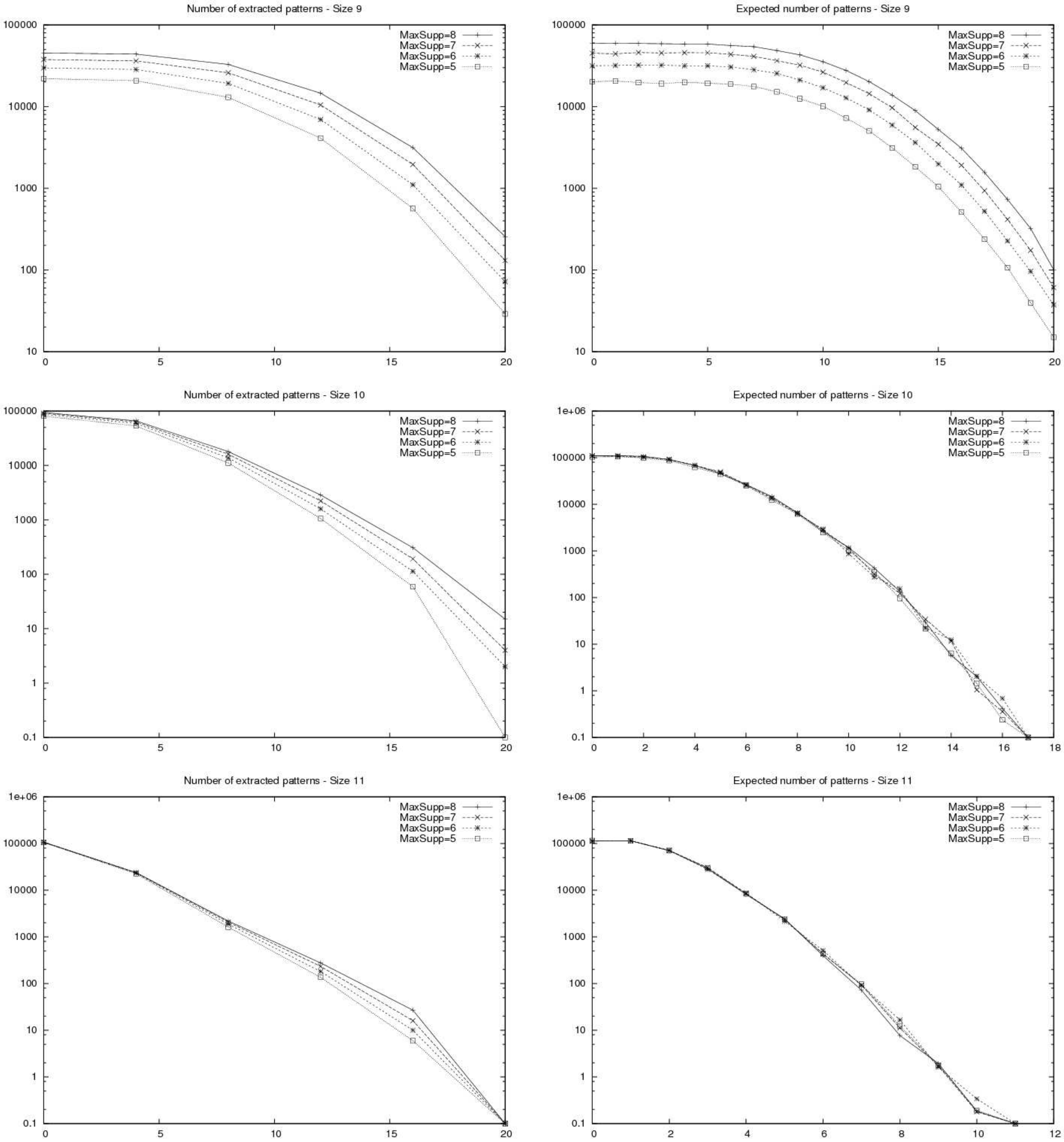

Fig. 1.2 Expected and real numbers of extracted patterns. The minimal support α_{min} corresponds to the horizontal axis, and the number of patterns corresponds to the vertical axis (log scale).

each pair of SMPs as follows. For each pair, we compute its optimal pairwise global alignment [17] with the following parameters: the score for a mismatch is 1, the score for a match is 0, the insertions and deletions inside an alignment are not allowed, the terminal gaps are not penalized, and the length of an alignment (terminal gaps are not included in the alignment length) must be at least a half of the length of the patterns in the pair. Finally, the dissimilarity of a pair of SMPs is simply the score of its best alignment (i.e., alignment having the lowest score).

Finding a consensus pattern within a cluster

To find the consensus pattern of each cluster of SMPs we align the patterns in each cluster using the multiple alignment tool *MultAlin* [5]. We use the following alignment scoring parameters: gap creation and extension penalty is -5, terminal gaps are not penalized, score for a match is 2, and score for any mismatch is 0. Once a consensus SMP is computed we consult $Transfac^{®}$ [12] to check whether it is a known TFBS. Figure 1.3 gives an example of a cluster, whose consensus SMP that has been selected because of its rather low TZI value (i.e., not likely to be due to the random background), and that is reported by $Transfac^{®}$ as a binding site of the TF c-Myb-isomorf1. In the consensus pattern in this figure, the bases that are highly conserved appear as uppercase letters in the consensus, and the weakly conserved ones appear as lowercase. Positions with no conserved bases are indicated as dots.

	Alignment	*TZI*
	`.CGGCCGTT...`	23.94
	`.GCGCCGTT...`	0.68
	`...GCCGTTAT.`	4.4
	`....CCGTTCGT`	4.4
	`...GCCGTTCG.`	23.75
	`....CCGTTAGG`	0.68
	`TTGGCCGT....`	23.75
	`...GCCGTAAC.`	107.37
	`..TGCCGTAA..`	0.58
Consensus	`...gCCGTt...`	
Transfac:	`c-Myb-isoform1`	
Mean of TZI:	21.06	

Fig. 1.3 A cluster of SMPs and its consensus computed by a multiple alignment.

Biological interpretation

The application of the scenario therefore allowed us to identify a c-Myb binding site as a signature motif of many newly identified v-ErbA repressed target genes compared with v-ErbA activated target genes. This suggests a potential role for c-Myb in the v-ErbA induced transformation. To determine the role of c-Myb in this transformation process, we used a gene reporter assay to test the ability of v-ErbA to transactivate c-Myb [4]. This experiment demonstrated that v-ErbA can indeed functionally interacts directly or indirectly with the transcriptional activity of endogenous c-Myb in T2ECs, constituting an experimental validation of the *in silico* extracted motif.

1.7 Conclusion

In this chapter, we presented a complete scenario that has been designed and used to support knowledge discovery from promoter sequences. Indeed, it can be applied to suggest putative TFBSs. The description of this application has been made at different levels: the corresponding abstract scenario, its concrete instantiation and a typical execution on a real dataset. To perform the main extraction step, we propose to use a solver developped for inductive querying over the string pattern domain. We also discussed all the additional processing required to use a solver, i.e., a data mining algorithm, in such a realistic context. This includes a parameter tuning tool, a support to pattern ranking and typical post-processing facilities dedicated to this kind of discovery task.

Acknowledgments. This work has been partly funded by EU contract IST-FET IQ FP6-516169 (Inductive Queries for Mining Patterns and Models) and by the French contract ANR-07-MDCO-014 Bingo2 (Knowledge Discovery For and By Inductive Queries). We would like to thank Stéphane Schicklin for his contribution to pattern matching against the $Transfac^{®}$ database, and the anonymous reviewers for their helpful comments and suggestions.

References

1. Besson, J., Rigotti, C., Mitasiunaité, I., Boulicaut, J.F.: Parameter tuning for differential mining of string patterns. In: Proceedings IEEE Workshop DDDM'08 co-olocated with ICDM'08, pp. 77–86 (2008)
2. Boulicaut, J.F., De Raedt, L., Mannila, H. (eds.): Constraint-Based Mining and Inductive Databases, *LNCS*, vol. 3848. Springer (2005). 400 pages
3. Brazma, A., Jonassen, I., Vilo, J., Ukkonen, E.: Predicting gene regulatory elements in silico on a genomic scale. Genome Res. **8**(11), 1202–1215 (1998)

4. Bresson, C., Keime, C., Faure, C., Letrillard, Y., Barbado, M., Sanfilippo, S., Benhra, N., Gandrillon, O., Gonin-Giraud, S.: Large-scale analysis by SAGE revealed new mechanisms of v-erba oncogene action. BMC Genomics **8**(390) (2007)
5. Corpet, F.: Multiple sequence alignment with hierarchical clustering. Nucl. Acids Res. **16**(22), 10,881–10,890 (1988)
6. Dan Lee, S., De Raedt, L.: An efficient algorithm for mining string databases under constraints. In: Proceedings KDID'04, pp. 108–129. Springer (2004)
7. De Raedt, L.: A perspective on inductive databases. SIGKDD Explorations **4**(2), 69–77 (2003)
8. De Raedt, L., Jaeger, M., Lee, S.D., Mannila, H.: A theory of inductive query answering. In: Proceedings IEEE ICDM'02, pp. 123–130 (2002)
9. Eden, E., Lipson, D., Yogev, S., Yakhini, Z.: Discovering motifs in ranked lists of DNA sequences. PLOS Computational Biology **3**(3), 508–522 (2007)
10. Imielinski, T., Mannila, H.: A database perspective on knowledge discovery. CACM **39**(11), 58–64 (1996)
11. Keich, U., Pevzner, P.A.: Subtle motifs: defining the limits of motif finding algorithms. Bioinformatics **18**(10), 1382–1390 (2002)
12. Matys, V., Fricke, E., Geffers, R., Gössling, E., Haubrock, M., Hehl, R., Hornischer, K., Karas, D., Kel, A.E., Kel-Margoulis, O.V., Kloos, D.U., Land, S., Lewicki-Potapov, B., Michael, H., Münch, R., Reuter, I., Rotert, S., Saxel, H., Scheer, M., Thiele, S., E., Wingender: Transfac : transcriptional regulation, from patterns to profiles. Nucl. Acids Res. **31**(1), 374–378 (2003)
13. Mitasiunaite, I.: Mining string data under similarity and soft-frequency constraints: Application to promoter sequence analysis. Ph.D. thesis, INSA Lyon (2009)
14. Mitasiunaite, I., Boulicaut, J.F.: Looking for monotonicity properties of a similarity constraint on sequences. In: Proceedings of ACM SAC'06 Data Mining, pp. 546–552 (2006)
15. Mitasiunaite, I., Boulicaut, J.F.: Introducing softness into inductive queries on string databases. In: Databases and Information Systems IV, *Frontiers in Artificial Intelligence and Applications*, vol. 155, pp. 117–132. IOS Press (2007)
16. Mitasiunaite, I., Rigotti, C., Schicklin, S., Meyniel, L., j. F. Boulicaut, Gandrillon, O.: Extracting signature motifs from promoter sets of differentially expressed genes. In Silico Biology **8**(43) (2008)
17. Needleman, S., Wunsch, C.: A general method applicable to the search for similarities in the amino acid sequence of two proteins. J. Mol. Biol. **48**(3), 443–453 (1970)
18. The R Project for Statistical Computing: http://www.r-project.org/
19. Tompa, M., Li, N., Bailey, T.L., Church, G.M., Moor, B.D., Eskin, E., Favorov, A.V., Frith, M.C., Fu, Y., Kent, W.J., Makeev, V.J., Mironov, A.A., Noble, W.S., Pavesi, G., Pesole, G., Rgnier, M., Simonis, N., Sinha, S., Thijs, G., van Helden, J., Vandenbogaert, M., Weng, Z., Workman, C., Ye, C., Zhu, Z.: Assessing computational tools for the discovery of transciption factor binding sites. Nat. Biotechnol. **23**(1), 137–144 (2005)
20. Vanet, A., Marsan, L., Sagot, M.F.: Promoter sequences and algorithmical methods for identifying them. Res. Microbiol. **150**(9-10), 779–799 (1999)
21. Velculescu, V.E., Zhang, L., Vogelstein, B., Kinzler, K.: Serial analysis of gene expression. Science **270**(5235), 484–487 (1995)