9

Significance Score of Motifs in Biological Sequences

Grégory Nuel
Institute for Mathematical Sciences (INSMI), CNRS, Paris
Department of Applied Mathematics (MAP5), University of Paris Descartes
France

1. Introduction

In Bioinformatics, it is common to search biological sequences (DNA, RNA, proteins) for functional motifs such as cross-over hotspot instigators (chi), restriction sites, regulation motifs, binding sites, active sites in proteins, etc. (Beaudoing et al., 2000; Brazma et al., 1998; El Karoui et al., 1999; Frith et al., 2002; Hampson et al., 2002; Karlin et al., 1992; Leonardo Marino-Ramírez & Landsman, 2004; van Helden et al., 1998). Due to evolution pressure, functional motifs are likely to be more conserved than non-functional motifs. As a consequence, it is a natural strategy to search biological sequences for motifs which are statistically exceptional (ex: over- or under-represented).
Given $\mathcal{M}$ a motif of interest (from simple strings to complex regular expressions), a recurrent question is: "how surprising is it to observe n occurrences of $\mathcal{M}$ in my dataset ". In statistical terms, this is equivalent to compute the p-value of observation n in respect with a relevant reference model. More precisely, if $X_{1:\ell} = X_1 \dots X_\ell$ is a length ℓ random sequence generated by our reference model, and if N denotes the random number of occurrences of $\mathcal{M}$ in $X_{1:\ell}$, for any $n \geqslant 0$ our objective is to compute the significance score of observation n:

$$S(n) = \begin{cases} +\log_{10} \mathbb{P}(N \leqslant n) \text{ if } n \leqslant \mathbb{E}[N] \\ -\log_{10} \mathbb{P}(N \geqslant n) \text{ if } n > \mathbb{E}[N] \end{cases} \quad (1)$$

this score representing the p-value in a decimal log-scale, negative (resp. positive) values being associated to under- (resp. over-) representation events.
In order to compute such a score for a given motif $\mathcal{M}$ and a given dataset, one needs two essential steps:

1) Counting: count the observed number n of occurrences of motif $\mathcal{M}$ in the dataset;

2) Significance: compute the p-value of observation n with respect to a reference model.

In this chapter, we give all the necessary details to perform these two steps using state of the art approaches including some unpublished results.

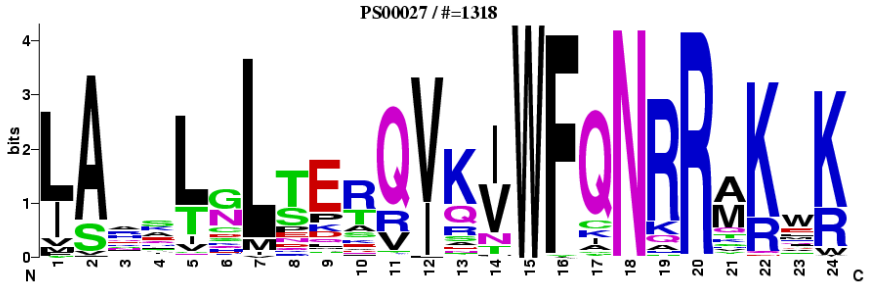
2. Counting motifs

2.1 Biological motifs

We can see on Fig. 1 various examples of the kind of biological motifs we usually deal with in Bioinformatics. In most cases, these motifs are built from a set of active sequences (putative

```
AGCGG GSTGGTGG CCWGG CTGCAG GAATTC CAYNNNNNRTG

CPRRRGRQTYTRFQTLELEKEFHF...........NHYLTRRRRIEIAHAL.........
CPRRRGRQTYTRFQTLELEKEFHF...........NHYLTRRRRIEIAHAL.........
VSVRKKRKPYSKFQTLELEKEFLF...........NAYVSKQKRWELARNL.........
-----------------LTKYFNK...........QPYPTRREIEKLAASL.........
-----------------LTKYFNK...........QPYPTRREIEKLAASL.........
-----------------LTKYFNK...........QPYPTRREIEKLAASL.........
SGKRRRRGNLPKESVQILRDWLYEhr........yNAYPSEQEKVLLSRQT.........
SKKRRHRTTFTSLQLEELEKVFQK...........THYPDVYVREQLALRT.........
SKKRRHRTTFTSLQLEELEKVFQK...........THYPDVYVREQLALRT.........
```




```
H-D-[LIVMFY]-x-H-x-[AG]-x(2)-[NQ]-x-[LIVMFY]

A  [ 3 21 25  0  0 24  1  0 ]
C  [13  1  0  0  5  0  0  0 ]
G  [ 4  0  0  0  0  1  0  2 ]
T  [ 5  3  0 25 20  0 24 23 ]
```

Fig. 1. Various kind of biological motifs. From top to bottom: strings in IUPAC (Cornish-Bowden, 1985) alphabet (DNA), multiple alignment (proteins), sequence logo (proteins), consensus pattern (proteins), and frequency matrix (DNA). Various sources including ReBase (Roberts et al., 2010), PROSITE (Sigrist et al., 2010), and JASPAR databases (Bryne et al., 2008).

or confirmed by experiments) in the form of a multiple alignment or a frequency matrix from which can be derived a consensus. This consensus could sometimes be a simple string (ex: `AGCGG` the chi site of *B. subtilis*) but in most cases it is a degenerated pattern (ex: `CAYNNNNNRTG` a restriction site in the IUPAC alphabet, PROSITE signatures). In all cases however, it is possible to consider our biological motif $\mathcal{M}$ as a (possibly large) set of strings.
Formally, let $\mathcal{M}$ be a finite set of strings over a finite alphabet $\mathcal{A}$. Ex: $\mathcal{A} = \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}$ for DNA sequences; this is the alphabet we are going to use from now on in our examples. Let $X_{1:\ell} = X_1 \dots X_\ell$ be an observed sequence of length ℓ over $\mathcal{A}$. Then the number $N(\mathcal{M}; X_{1:\ell})$ of

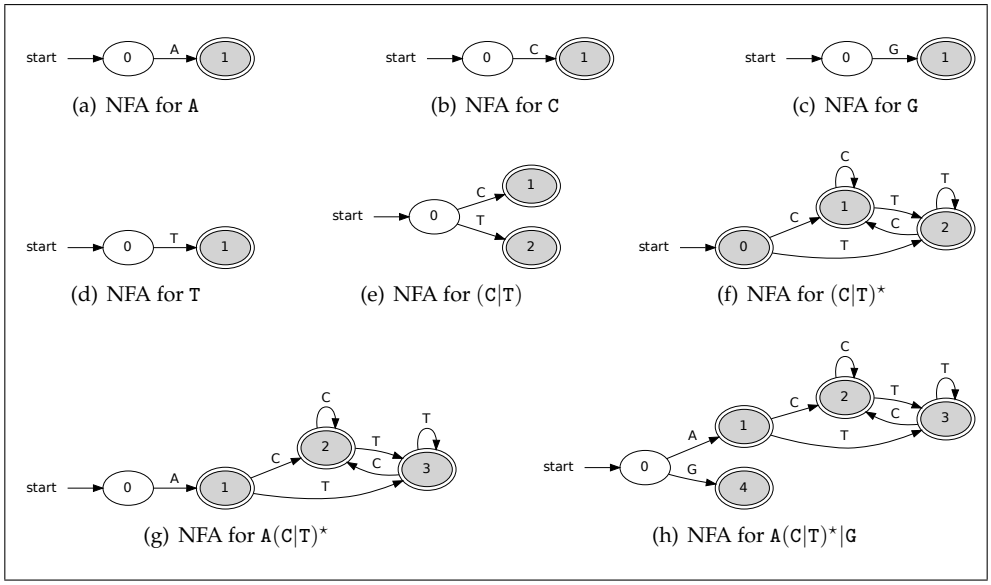


Fig. 2. Glushkov's construction for $\mathtt{A}(\mathtt{C}|\mathtt{T})^\star|\mathtt{G}$. (a), (b), (c), and (d) are singletons; (e) results from the union of (b) and (d); (f) results from the Kleene's closure of (e); (g) results from the concatenation of (a) and (f); (h) results from the union of (g) and (c).

matching positions of $\mathcal{M}$ in $X_{1:\ell}$, is defined by

$$N(\mathcal{M}; X_{1:\ell}) = \sum_{i=1}^{\ell} \mathbf{1}_{X_{1:i} \in \mathcal{A}^\star \mathcal{M}} \tag{2}$$

$\mathcal{A}^\star\mathcal{M}$ being the set of all finite sequences over $\mathcal{A}$ ending with one element of $\mathcal{W}$ (this notation will be explained in the next section), and where $\mathbf{1}_A$ is the indicator function of event A.
In the particular case where $\mathcal{M}$ contains no strings that are included into each other (which is a common assumption), the number N of matching position corresponds exactly to the number of occurrences. However, there is no need to put any restriction on $\mathcal{M}$ as long as we are interested in the number of matching positions like we do.
From now on, if the sequence $X_{1:\ell}$ is observed, we denote by the number of matching positions by n, and if the sequence $X_{1:\ell}$ is random, we simply denote by N the random number of matching positions.

2.2 Regular languages

Let us denote by $\mathcal{A}^\star$ the set of all finite sequences over $\mathcal{A}$. Any subset $\mathcal{L} \subset \mathcal{A}^\star$ is then called a *language* over $\mathcal{A}$. We denote by $\mathcal{P}(\mathcal{A}^\star)$ the set of all possible languages over $\mathcal{A}$. We denote by $\varepsilon \in \mathcal{A}^\star$ the empty sequence, and for the sake of simplicity, the singletons of $\mathcal{P}(\mathcal{A}^\star)$ will be simply denoted by their element. Ex: $\mathtt{A}$ instead of $\{\mathtt{A}\}$, $\mathtt{TGC}$ instead of $\{\mathtt{TGC}\}$, ε instead of $\{\varepsilon\}$. We define on these languages three *regular operations*:

Union (|): for all $\mathcal{L}_1, \mathcal{L}_2 \in \mathcal{P}(\mathcal{A}^\star)$, $\mathcal{L}_1|\mathcal{L}_2 = \mathcal{L}_1 \cup \mathcal{L}_2$. The neutral element of the binary operator | is $\emptyset$. Ex: $\{\mathtt{AT}, \mathtt{GA}\}|\{\mathtt{T}, \mathtt{GA}, \mathtt{TT}\} = \{\mathtt{AT}, \mathtt{T}, \mathtt{GA}, \mathtt{TT}\}$.

Require: remove first all states that are not reachable from σ or that cannot reach any element of $\mathcal{F}$
$\mathcal{W} \leftarrow \{\mathcal{F}, \mathcal{Q} \setminus \mathcal{F}\}$ and $\mathcal{P} \leftarrow \{\mathcal{F}, \mathcal{Q} \setminus \mathcal{F}\}$
while $\mathcal{W}$ is not empty **do**
  select and remove $\mathcal{V}$ from $\mathcal{W}$
  for all $a \in \mathcal{A}$ **do**
    $\mathcal{S} = \{q \in \mathcal{Q}, \delta(q, a) \in \mathcal{V}\}$
    for all $\mathcal{R} \in \mathcal{P}$ such as $\mathcal{R} \cap \mathcal{S} \neq \emptyset$ and $\mathcal{R} \nsubseteq \mathcal{S}$ **do**
      replace $\mathcal{R}$ in $\mathcal{P}$ by $\mathcal{R}_1 \leftarrow \mathcal{R} \cap \mathcal{S}$ and $\mathcal{R}_2 \leftarrow \mathcal{R} \setminus \mathcal{R}_1$
      if $\mathcal{R} \in \mathcal{W}$ **then**
        replace $\mathcal{R}$ in $\mathcal{P}$ by $\mathcal{R}_1$ and $\mathcal{R}_2$
      else
        if $|\mathcal{R}_1| \leqslant |\mathcal{R}_2|$ **then** add $\mathcal{R}_1$ to $\mathcal{W}$ **else** add $\mathcal{R}_2$ to $\mathcal{W}$ **end if**
      end if
    end for
  end for
end while

Algorithm 1. Performs Hopcroft's reduction on NFA $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$. $\mathcal{W}$ (working set) and $\mathcal{P}$ (partition set) are two sets of set of NFA states. The resulting complexity is $O(|\mathcal{Q}| \log |\mathcal{Q}|)$.

Concatenation ($\cdot$): for all $\mathcal{L}_1, \mathcal{L}_2 \in \mathcal{P}(\mathcal{A}^*)$, $\mathcal{L}_1 \cdot \mathcal{L}_2 = \{xy, x \in \mathcal{L}_1, y \in \mathcal{L}_2\}$. The neutral element of the binary operator $\cdot$ is ε. For all $\mathcal{L} \in \mathcal{P}(\mathcal{A}^\star)$, $\mathcal{L}^0 = \varepsilon$ (convention), $\mathcal{L}^1 = \mathcal{L}$, $\mathcal{L}^2 = \mathcal{L} \cdot \mathcal{L}$ and the notation extends recursively to $\mathcal{L}^k$ for any $k \geqslant 3$. Ex: $\{\mathtt{G}, \mathtt{GA}\} \cdot \{\mathtt{AT}, \mathtt{T}\} = \{\mathtt{GAT}, \mathtt{GT}, \mathtt{GAAT}\}$; $\{\mathtt{G}, \mathtt{GA}\}^3 = \{\mathtt{GGG}, \mathtt{GGGA}, \mathtt{GGAG}, \mathtt{GGAGA}, \mathtt{GAGG}, \mathtt{GAGGA}, \mathtt{GAGAG}, \mathtt{GAGAGA}\}$. For the sake of simplicity, $\cdot$ is implicitly used when the operator is omitted.. Ex: $\mathcal{A}\mathcal{L}$ means $\mathcal{A} \cdot \mathcal{L}$.

Kleene's closure (*): For all $\mathcal{L} \in \mathcal{P}(\mathcal{A}^\star)$, $\mathcal{L}^\star = \sum_{k \geqslant 0} \mathcal{L}^k$. Ex: $\{\mathtt{AT}\}^\star = \{\varepsilon, \mathtt{AT}, \mathtt{ATAT}, \mathtt{ATATAT}, \ldots\}$.

The precedence rule of these operations is: $|$ (lowest precedence), $\cdot$ (associative operator), * (highest precedence). Ex: $\mathtt{A}|\mathtt{C} \cdot \mathtt{T}^\star = (\mathtt{A}|(\mathtt{C}(\cdot \mathtt{T}^\star))$, $\mathtt{TT} \cdot \mathtt{A}|\mathtt{C}^\star \cdot \mathtt{G} = ((\mathtt{TT} \cdot \mathtt{A})|((\mathtt{C}^\star) \cdot \mathtt{G}))$.
We call *regular expression* over $\mathcal{A}$ any algebric expression over $\mathcal{P}(\mathcal{A}^\star)$ defined from singleton elements and a finite number of regular operations. The resulting language is called a *regular language*. Ex: any finite language is a regular language, $\mathcal{A}^\star$ is a regular language, $(\mathtt{A}|\mathtt{C}|\mathtt{G}|\mathtt{T})^\star\mathtt{GGATG}$ is a regular language, $\{\mathtt{AG}, \mathtt{AAGG}, \mathtt{AAAGGG}, \ldots\}$ is not a regular language.

2.3 Non-deterministic finite automaton

A *Non-deterministic Finite Automaton* (NFA) is defined as a 5-tuple $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ where: $\mathcal{A}$ is a finite alphabet, $\mathcal{Q}$ is a finite *state space*, $\sigma \in \mathcal{Q}$ is the *starting state*, $\mathcal{F} \subset \mathcal{Q}$ is the set of *final states*, and $\delta : \mathcal{Q} \times \mathcal{A} \to \mathcal{P}(\mathcal{Q})$ is the *transition function*. An element $X_{1:\ell} \in \mathcal{A}^\star$ is *accepted* by this NFA if and only if it exists a *path* from the starting state to one of the final state that sequentially use the letters $X_{1:\ell}$ in the transitions. More formally, it means that it exists a sequence of states (ie: elements of $\mathcal{Q}$) $q_0 = \sigma, q_1, q_2 \ldots, q_{\ell-1}, q_\ell \in \mathcal{F}$ such as $q_i \in \delta(q_{i-1}, X_i)$ for all $1 \leqslant i \leqslant \ell$. The *language of a NFA* is the set of all elements of $\mathcal{A}^\star$ it accepts.

Theorem 1. For any language $\mathcal{L} \in \mathcal{P}(\mathcal{A}^\star)$: $\mathcal{L}$ regular $\iff$ it exists a NFA whose language is $\mathcal{L}$.

We admit that the language of a NFA is always regular (see Hopcroft et al., 2001, for the formal proof) but we will prove the reciprocal with the Glushkov's construction (Allauzen &

Mohri, 2006). This construction provides a simple way to build the NFA directly from the regular expression of the language. The idea is to treat the regular expression as any algebraic expression with a stack of operands (NFAs) and a stack of operators (regular operations). Since a regular expression is by definition built from singleton elements of $\mathcal{A}^\star$ and the three regular operations, we only need to give the construction of a NFA corresponding to singleton elements, and the constructions corresponding to the regular operations.

Singleton: for any $X_{1:\ell} \in \mathcal{A}^\star$ we build the NFA $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ with $\mathcal{Q} = \{0, 1, \ldots, \ell\}$, $\sigma = 0$, $\mathcal{F} = \{\ell\}$, and $\delta(i-1, X_i) = \{i\}$ for all $1 \leqslant i \leqslant \ell$.

Union: the union $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ of two NFAs $(\mathcal{A}, \mathcal{Q}_1, \sigma_1, \mathcal{F}_1, \delta_1)$ and $(\mathcal{A}, \mathcal{Q}, \sigma_2, \mathcal{F}_2, \delta_2)$ is given by: $\mathcal{Q} = \mathcal{Q}_1 \cup \mathcal{Q}_2 \setminus \{\sigma_2\}$, $\sigma = \sigma_1$, $\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2$ and

$$\delta(q,a) = \begin{cases} \delta_1(\sigma_1, a) \cup \delta_1(\sigma_2, a) & \text{if } q = \sigma_1 \\ \delta_1(q,a) & \text{if } q \in \mathcal{Q}_1 \setminus \{\sigma_1\} \\ \delta_2(q,a) & \text{if } q \in \mathcal{Q}_2 \setminus \{\sigma_2\} \end{cases}. \tag{3}$$

Concatenation: the concatenation $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ of two NFAs $(\mathcal{A}, \mathcal{Q}_1, \sigma_1, \mathcal{F}_1, \delta_1)$ and $(\mathcal{A}, \mathcal{Q}, \sigma_2, \mathcal{F}_2, \delta_2)$ is given by: $\mathcal{Q} = \mathcal{Q}_1 \cup \mathcal{Q}_2 \setminus \{\sigma_2\}$, $\sigma = \sigma_1$, $\mathcal{F} = \mathcal{F}_2$ and

$$\delta(q,a) = \begin{cases} \delta_1(q,a) & \text{if } q \in \mathcal{Q}_1 \setminus \mathcal{F}_1 \\ \delta_2(\sigma_2, a) & \text{if } q \in \mathcal{F}_1 \\ \delta_2(q,a) & \text{if } q \in \mathcal{Q}_2 \setminus \{\sigma_2\} \end{cases}. \tag{4}$$

Kleene's closure: the Kleene's closure $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ of NFA $(\mathcal{A}, \mathcal{Q}_1, \sigma_1, \mathcal{F}_1, \delta_1)$ is given by: $\mathcal{Q} = \mathcal{Q}_1$, $\sigma = \sigma_1$, $\mathcal{F} = \mathcal{F}_1 \cup \{\sigma_1\}$ and

$$\delta(q,a) = \begin{cases} \delta_1(q,a) & \text{if } q \in \mathcal{Q}_1 \setminus \mathcal{F}_1 \\ \delta_1(\sigma_1, a) & \text{if } q \in \mathcal{F}_1 \end{cases}. \tag{5}$$

Using Glushkov's construction, it is then possible to build a NFA whose language correspond to the regular expression of our choice. However in general, this construction is not optimal in terms of number of states. Fortunately, the reduction algorithm (Algorithm 1) due to Hopcroft provides a (partial) solution to this problem. Note that finding a minimal NFA for a given regular expression is a difficult task in general, but that Hopcroft's reduction is a good heuristic (we will see later that in the case of DFA, Hopcroft's reduction is indeed a minimization).

2.4 Counting with NFA

NFAs provide with Algorithm 2 an extremely efficient way to look for matching positions of any motif $\mathcal{M}$ (in fact, any regular expression) in a sequence $X_{1:\ell}$. The algorithm directly results from the definition of the language of a NFA.

Let us illustrate this algorithm with a toy example: how to find all matching positions of $\mathcal{M} =$ `G(G|C)G` in $X_{1:12} =$ `AGCGGTGGGCGA` ? We first use Glushkov's construction and Algorithm 1 to obtain on Fig. 3 a minimal NFA whose language is `(A|C|G|T)*G(G|C)G`. Then we directly apply Algorithm 2 starting with $\mathcal{S} = \{0\}$:

- $i = 1$, $X_1 =$ `A`, $\mathcal{S} \leftarrow \delta(\{0\}, \texttt{A}) = \{0\}$;
- $i = 2$, $X_2 =$ `G`, $\mathcal{S} \leftarrow \delta(\{0\}, \texttt{G}) = \{0, 1\}$;

Require: $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ be a (minimal) NFA whose language is $\mathcal{A}^\star\mathcal{M}$
1: $\mathcal{S} \leftarrow \{\sigma\}$
2: **for** $i = 1 \dots \ell$ **do**
3: $\quad\mathcal{S} \leftarrow \cup_{q\in\mathcal{S}}\delta(q, X_i)$
4: $\quad$**if** $\mathcal{S} \cap \mathcal{F} \neq \emptyset$ **then**
5: $\quad\quad$report i as a matching position
6: $\quad$**end if**
7: **end for**

Algorithm 2. NFA pattern matching. Returns all matching positions of motif $\mathcal{M}$ in $X_{1:\ell}$. Complexity is $O(|\mathcal{Q}| \times \ell)$.

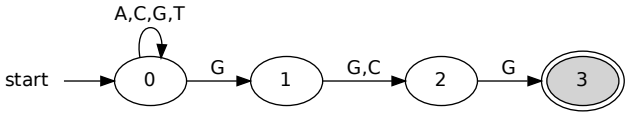


Fig. 3. Minimal NFA whose language is (A|C|G|T)*G(G|C)G.

- $i = 3$, $X_3 = \mathtt{C}$, $\mathcal{S} \leftarrow \delta(\{0,1\}, \mathtt{C}) = \{0,2\}$;
- $i = 4$, $X_4 = \mathtt{G}$, $\mathcal{S} \leftarrow \delta(\{0,2\}, \mathtt{G}) = \{0,1,3\}$, matching position;
- $i = 5$, $X_5 = \mathtt{G}$, $\mathcal{S} \leftarrow \delta(\{0,1,3\}, \mathtt{G}) = \{0,1,2\}$;
- $i = 6$, $X_6 = \mathtt{T}$, $\mathcal{S} \leftarrow \delta(\{0,1,2\}, \mathtt{T}) = \{0\}$;
- $i = 7$, $X_7 = \mathtt{G}$, $\mathcal{S} \leftarrow \delta(\{0\}, \mathtt{G}) = \{0,1\}$;
- $i = 8$, $X_8 = \mathtt{G}$, $\mathcal{S} \leftarrow \delta(\{0,1\}, \mathtt{G}) = \{0,1,2\}$;
- $i = 9$, $X_9 = \mathtt{G}$, $\mathcal{S} \leftarrow \delta(\{0,1,2\}, \mathtt{G}) = \{0,1,2,3\}$, matching position;
- $i = 10$, $X_{10} = \mathtt{C}$, $\mathcal{S} \leftarrow \delta(\{0,1,2,3\}, \mathtt{C}) = \{0,2\}$.
- $i = 11$, $X_{11} = \mathtt{G}$, $\mathcal{S} \leftarrow \delta(\{0,2\}, \mathtt{G}) = \{0,1,3\}$, matching position;
- $i = 12$, $X_{12} = \mathtt{A}$, $\mathcal{S} \leftarrow \delta(\{0,1,3\}, \mathtt{A}) = \{0\}$.

We hence return three matching positions: 4, 9 and 11.
One should note in this example that in twice occasions, we need to recompute a previously computed transition ($i = 7$ and $i = 11$). Obviously, this kind of event is likely to appear very often when working with longer sequences. It is hence a natural idea to store in memory previously computed transitions. This approach, known as *lazy determinization* (Green et al., 2004), speeds up considerably pattern matching (reducing the complexity from $O(|\mathcal{Q}| \times \ell)$ to $O(\ell)$) at the expense of a higher memory usage. We will see later that the amount of memory needed can increase exponentially with the NFA size $|\mathcal{Q}|$; this problem is usually addressed by allocating a fixed amount of memory to a buffer of computed transitions which is flushed when full.

3. Significance

Since we now have efficient algorithms to count the number of occurrence of a motif $\mathcal{M}$ in a sequence $X_{1:\ell}$, let us deal with the significance of an observation n.

3.1 Reference model

The choice of a reference model is obviously a key point. Since biological sequences like DNA or proteins are known to have unbalanced letter compositions, it is hence clear that our model should at least take into account this source of bias. A natural parametric approach [1] is hence to model $X_{1:\ell}$ as a i.i.d. sequence with $\mathbb{P}(X_i = a) = \pi(a)\ \forall a \in \mathcal{A}$ with all $\pi(a) \in [0,1]$ and $\sum_{a\in\mathcal{A}} \pi(a) = 1$. This model is called model M0 with parameter π.

For example, in the complete genome of HIV1 (Genbank AF033819) we observe the following counts: 3272 `A`, 1642 `C`, 2225 `G`, and 2042 `T`. The maximum likelihood estimates of a M0 model based on this observation is then: $\widehat{\pi}(\mathtt{A}) = 3272/9181 \simeq 35.64\%$, $\widehat{\pi}(\mathtt{C}) = 1642/9181 \simeq 17.88\%$, $\widehat{\pi}(\mathtt{G}) = 2225/9181 \simeq 24.23\%$, and $\widehat{\pi}(\mathtt{T}) = 2042/9181 \simeq 22.24\%$.

But if we look now to the frequencies of di-nucleotides on the same HIV1 genome, we observe considerable bias as well:

`AA` 1087	`AC` 524	`AG` 971	`AT` 690
`CA` 754	`CC` 378	`CG` 82	`CT` 427
`GA` 769	`GC` 425	`GG` 625	`GT` 406
`TA` 662	`TC` 315	`TG` 546	`TT` 519

For example, we observe $971/3272 = 29.68\%$ of `G` after a `A`, but a `G` occurs after a `C` only $82/1641 = 16.41\%$ of the time. This phenomenon is directly explained by the fact that the di-nucleotide `CG` tend to be easily methylated (see CpG island, Fatemi et al., 2005). Is hence tempting to take into account the frequencies of di-nucleotides in our reference model, or tri-nucleotides, or more, which naturally leads to Markov models.

For any $d \geqslant 0$, we denote by Md the (homogeneous) Markov model of order d defined for any $i \geqslant d+1$, $a \in \mathcal{A}^d$, and $b \in \mathcal{A}$ by:

$$\mathbb{P}(X_i = b | X_{i-d:i-1} = a) = \pi(a,b) \tag{6}$$

where π denotes the *transition matrix* of Md. This model is clearly defined conditionally to $X_{1:d}$.

The maximum likelihood estimator $\widehat{\pi}$ is then given for all $a \in \mathcal{A}^d$, and $b \in \mathcal{A}$ by:

$$\widehat{\pi}(a,b) = \frac{n_{ab}}{\sum_{b'\in\mathcal{A}} n_{ab'}} \tag{7}$$

where n_{ab} are the observed counts of word ab in the training dataset.

When working with Markov model and biological sequences, a recurrent question is: what order d should I choose for my reference model ? This is a classical model selection problem which can easily be solved using penalized likelihood criteria like BIC or AIC (Liddle, 2007). For example, using the BIC criterion, one would select $d = 1$ for the complete genome of HIV1 ($\ell \simeq 10$kb), and $d = 5$ for the complete genome of *E. coli* ($\ell \simeq 4.6$Mb). However, since our objective is the significance of motifs counts rather than the modelization of biological sequence in itself, we suggest a different approach.

First, it is critical to realize than working with a model Md as reference model allows to take into account the sequence composition bias in $(d+1)$-mers. Hence, with $d = 1$ one takes into account the composition bias in di-nucleotides, and with $d = 5$, one takes into account the composition bias in hexa-nucleotides. The decision could then be based on the information one wishes to include in the reference model; working on coding sequences, one might wish to take into account at least the codon bias hence resulting in the choice of $d \geqslant 2$. On the other

[1] An alternative non-parametric approach, the *shuffling*, consists in performing uniformly a random permutation of the original sequence; this approach is not treated here.

Require: $(\mathcal{A}, \mathcal{Q}_1, \sigma, \mathcal{F}_1, \delta_1)$ a NFA
$q_0 \leftarrow \{\sigma\}$, $L \leftarrow 1$, $\mathcal{Q}_2 \leftarrow \{q_0\}$, $\mathcal{F}_2 \leftarrow \emptyset$
for $i = 0 \dots L-1$ **do**
  for all $a \in \mathcal{A}$ **do**
    $\mathcal{S} \leftarrow \delta_1(q_i, a)$
    if $\exists j, q_j = \mathcal{S}$ **then**
      $\delta_2(q_i, a) = q_j$
    else
      $q_L \leftarrow \mathcal{S}$, $L \leftarrow L+1$, $\mathcal{Q}_2 \leftarrow \mathcal{Q}_2 \cup \{q_L\}$
      if $\mathcal{S} \cap \mathcal{F}_\infty$ **then** $\mathcal{F}_2 \leftarrow \mathcal{F}_2 \cup \{q_L\}$ **end if**
    end if
  end for
end for
Output: return $(\mathcal{A}, \mathcal{Q}_2, q_0, \mathcal{F}_2, \delta_2)$

Algorithm 3. Determinization. Build a DFA which recognizes the same language than the original NFA.

hand, it would obviously be pointless to use a reference model of order $d = 7$ to study a motif of length 8 or less.
Another critical point to keep in mind is that motif significance is by nature very sensitive to the parameters of the reference model. In order to convince us, let us a consider the following simple example with $\mathcal{M} = \texttt{GGATG}$, a reference model M0 of parameter π, and $\ell = 1,000,000$. If $\pi(\texttt{A}) = \pi(\texttt{T}) = 0.10$ and $\pi(\texttt{C}) = \pi(\texttt{G}) = 0.40$ we get $\mathbb{E}[N_\ell] = \ell \times 0.40^3 \times 0.10^2 \simeq 640.0$. Now if $\pi(\texttt{A}) = \pi(\texttt{T}) = 0.08$ and $\pi(\texttt{C}) = \pi(\texttt{G}) = 0.42$ then $\mathbb{E}[N_\ell] = \ell \times 0.42^3 \times 0.08^2 \simeq 474.2$. If we admit that the standard deviation of N_ℓ is roughly equal to $\sigma = 25$ (we will see later on how to perform such computation), an observation of $n = 550$ could be interpreted as a significant over-representation with the first parameters, and a significant under-representation with the second parameters (observation n deviates from the expectation by more than three standard deviations in both cases). The reason behind this is that parameter values are typically involved in complex products when evaluating the significance of an observation, and that such operations usually increase small variations rather than averaging them (like with sums). This problem have been investigated in Nuel (2006c) where it is shown that unwise choices of d might lead to many false positive results.

3.2 Monte-Carlo simulations

Since the theoretical distribution of N not easy to obtain, it is tempting to study it from the empirical point of view by performing simple simulations. The approach is quite straightforward:

1) generate a random dataset i according to the reference model;

2) count the number of occurrence n_i of $\mathcal{M}$ in the dataset;

3) repeat 1) and 2) until we have a sample $n_1, n_2, \dots, n_r$.

Once a reference sample have been obtained, we can derive the empirical p-value of the observation n using:

$$\widehat{\mathbb{P}}(N \leqslant n) = \frac{\sum_{i=1}^{r} \mathbf{1}_{n_i \leqslant n}}{r} \quad \text{or} \quad \widehat{\mathbb{P}}(N \geqslant n) = \frac{\sum_{i=1}^{r} \mathbf{1}_{n_i \geqslant n}}{r} \tag{8}$$

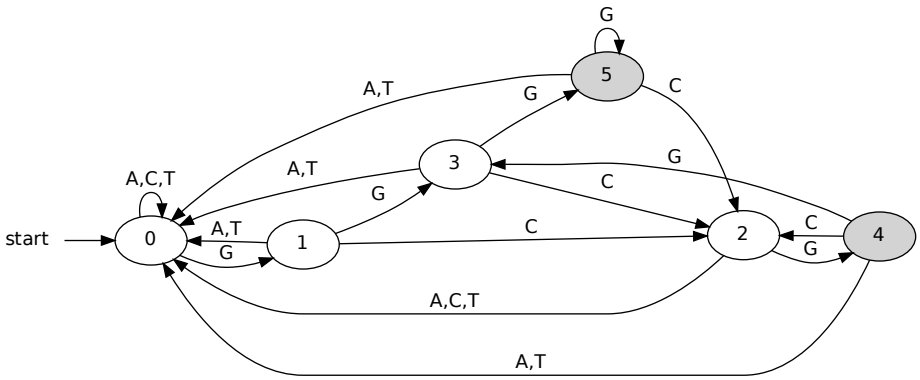


Fig. 4. Minimal DFA whose language is (A|C|G|T)*G(G|C)G.

or, alternatively, one might use this sample to derive empirical expectation, variance, and z-score:

$$\widehat{Z}(n) = \frac{n - \widehat{\mu}}{\sigma} \quad \text{with} \quad \widehat{\mu} = \frac{1}{r}\sum_{i=1}^{r} n_i \quad \text{and} \quad \widehat{\sigma}^2 = \frac{1}{r}\sum_{i=1}^{r}(n_i - \widehat{\mu})^2. \tag{9}$$

If this approach is quite simple, it suffers several drawbacks: 1) it is slow; 2) sample size must be large to obtain accurate results. Indeed, if the true p-value is p, then $\widehat{p} \sim \mathcal{B}(r, p)$ where r is the sample size. The following table gives a 90% upper bound confidence for $\widehat{p}$ for several value of r in the case where $p = 10^{-5}$:

r	10^3	10^4	10^5	10^6	10^7	10^8
bound	1.00×10^{-3}	1.00×10^{-4}	3.00×10^{-5}	1.50×10^{-5}	1.14×10^{-5}	1.04×10^{-5}

we clearly see that it requires at least $r = 10^6$ samples to obtain the first accurate digit in $\widehat{p}$, and a prohibitive $r = 10^8$ samples for the second digit. Considering that very small p-value are easily encountered in motif significance (ex: 10^{-20}, 10^{-50}, 10^{-100}), it is clear that empirical p-value have a limited interest in this context.
Empirical z-score does not suffer the same drawback but makes the implicit assumption that N has a Gaussian distribution which is highly questionable as we will see later on.
For completeness, let us point out that *importance sampling* techniques might solve the estimation problem by sampling N using a tailored dataset distribution (Chan et al., 2010). However, these sophisticated numerical techniques are slow and requires a good skills to be implemented.

3.3 Markov chain embedding

The key to perform any motif significance computation if first to embed the original problem into an order 1 Markov chain taking into account all the combinatoric complexity. This technique, called *Markov chain embedding* have been used by many authors in the context of motif significance Antzoulakos (2001); Boeva et al. (2005); Chang (2005); Fu (1996); Nuel (2006a), but it is only recently that its connexion to NFA and Deterministic Finite Automata (DFA) have been pointed out (Crochemore & Stefanov, 2003; Lladser, 2007; Nicodème et al., 2002; Nuel, 2008a; Nuel & Prum, 2007; Ribeca & Raineri, 2008).

We start with a NFA whose language is $\mathcal{A}^\star\mathcal{M}$ from which we build a DFA $(\mathcal{A}, \mathcal{Q}, q_0, \mathcal{F}, \delta)$ using the determinization algorithm (Algorithm 3). A DFA differs from an NFA only by the definition of its transition function: $\delta : \mathcal{Q} \times \mathcal{A} \to \mathcal{P}(\mathcal{Q})$ for a NFA, and $\delta : \mathcal{Q} \times \mathcal{A} \to \mathcal{Q}$ for a DFA. For example, we can see on Figure 4, a (minimal) DFA whose language is $(\mathtt{A}|\mathtt{C}|\mathtt{G}|\mathtt{T})^\star\mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$. This DFA has more states (6) than the corresponding NFA (4). In fact, since the state space $\mathcal{Q}_2$ of a DFA corresponds to a subset of the parts of the original NFA state space $\mathcal{Q}_1$, we have $|\mathcal{Q}_2| \leqslant 2^{|\mathcal{Q}_1|}$. Fortunately, this upper bound is seldom reached in practice.

Theorem 2 (Markov chain embedding for Model M0)**.** Let $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ be a (minimal) DFA whose language is $\mathcal{A}^\star\mathcal{M}$. Let $X_{1:\ell}$ be a random sequence generated by the M0 model of parameter π. We consider the sequence $Z_{0:\ell}$ recursively defined by $Z_0 = \sigma$, and $Z_i = \delta(Z_{i-1}, X_i)$ for all $1 \leqslant i \leqslant \ell$. Then $Z_{0:\ell}$ is an order 1 Markov chain whose transition matrix $\mathbf{T}$ is defined for all $p, q \in \mathcal{Q}$ by:

$$\mathbf{T}(p, q) = \sum_{a \in \mathcal{A}, \delta(p,a)=q} \pi(a) \tag{10}$$

and having the following property for all $1 \leqslant i \leqslant \ell$: $X_{1:i} \in \mathcal{A}^\star\mathcal{M} \iff Z_i \in \mathcal{F}$.

For example, if we consider the DNA motif $\mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$ and the corresponding DFA of Figure 4, we get the following transition matrix:

$$\mathbf{T} = \begin{pmatrix} \pi(\mathtt{A}) + \pi(\mathtt{C}) + \pi(\mathtt{T}) & \pi(\mathtt{G}) & 0 & 0 & 0 & 0 \\ \pi(\mathtt{A}) + \pi(\mathtt{T}) & 0 & \pi(\mathtt{C}) & \pi(\mathtt{G}) & 0 & 0 \\ \pi(\mathtt{A}) + \pi(\mathtt{C}) + \pi(\mathtt{T}) & 0 & 0 & 0 & \pi(\mathtt{G}) & 0 \\ \pi(\mathtt{A}) + \pi(\mathtt{T}) & 0 & \pi(\mathtt{C}) & 0 & 0 & \pi(\mathtt{G}) \\ \pi(\mathtt{A}) + \pi(\mathtt{T}) & 0 & \pi(\mathtt{C}) & \pi(\mathtt{G}) & 0 & 0 \\ \pi(\mathtt{A}) + \pi(\mathtt{T}) & 0 & \pi(\mathtt{C}) & 0 & 0 & \pi(\mathtt{G}) \end{pmatrix}.$$

In order to extend Theorem 2 to order Md with $d > 0$ it is necessary to build DFA $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ be a (minimal) DFA whose language is $\mathcal{A}^\star\mathcal{M}$ and with the property that for all $q \in \mathcal{Q}$, $\text{past}(q) = \{a \in \mathcal{A}^d, \exists p \in \mathcal{Q}, \delta(p, a) = q\}$ is either empty or a singleton. A DFA having this property is called a order d DFA by Lladser (2007), and is called non d-ambiguous by Nuel (2008a). The construction of such a (minimal) DFA is not very complicated but is a bit technical. A possible approach suggested by Nuel (2008a) consists in starting from a DFA without this property and duplicating any "ambiguous" state. Another more straightforward approach consists in adding the elements of $\mathcal{A}^\star\mathcal{A}^d$ to the original language with a specific label for the final states corresponding to each elements of $\mathcal{A}^d$, and to keep these labels during minimization and determinization algorithms.

Theorem 3 (Markov chain embedding for Model Md)**.** Let $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ be a (minimal) order d DFA whose language is $\mathcal{A}^\star\mathcal{M}$. Let $X_{1:\ell}$ be a random sequence generated by the Md model of parameter π. We consider the sequence $Z_{d:\ell}$ recursively defined by $Z_d = \delta(\sigma, X_{1:d})$, and $Z_i = \delta(Z_{i-1}, X_i)$ for all $1 \leqslant i \leqslant \ell$. Then $Z_{d:\ell}$ is an order 1 Markov chain whose transition matrix $\mathbf{T}$ is defined for all $p, q \in \mathcal{Q}$ by:

$$\mathbf{T}(p, q) = \sum_{a \in \mathcal{A}, \delta(p,a)=q} \pi(\text{past}(p), a) \tag{11}$$

and having the following property for all $1 \leqslant i \leqslant \ell$: $X_{1:i} \in \mathcal{A}^\star\mathcal{M} \iff Z_i \in \mathcal{F}$.

One should note that $Z_{d:\ell}$ is defined on $\delta(\sigma, \mathcal{A}^d\mathcal{A}^\star)$ which could be slightly smaller than $\mathcal{Q}$. This subset corresponds to the states of $\mathcal{Q}$ having a order d past. If we consider the DFA

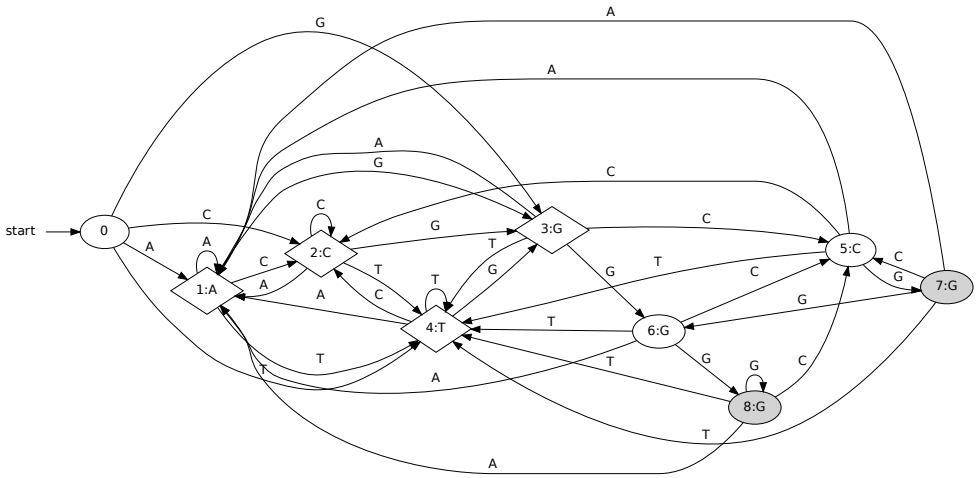


Fig. 5. Minimal order 1 DFA whose language is $(\mathtt{A}|\mathtt{C}|\mathtt{G}|\mathtt{T})^{\star}\mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$. The order 1 past of each state is indicated in the state itself. Diamond-shaped states correspond to the elements of $\delta(0, \mathcal{A}^1)$.

of Figure 5, $d = 1$, and with $X_1 = \mathtt{A}$, we see that the Markov chain $Z_{d:\ell}$ is defined on $\{1,2,3,4,5,6,7,8\}$ by $Z_1 = 1$ and the following transition matrix:

$$\mathbf{T} = \begin{pmatrix} \pi(\mathtt{A},\mathtt{A}) & \pi(\mathtt{A},\mathtt{C}) & \pi(\mathtt{A},\mathtt{G}) & \pi(\mathtt{A},\mathtt{T}) & 0 & 0 & 0 & 0 \\ \pi(\mathtt{C},\mathtt{A}) & \pi(\mathtt{C},\mathtt{C}) & \pi(\mathtt{C},\mathtt{G}) & \pi(\mathtt{C},\mathtt{T}) & 0 & 0 & 0 & 0 \\ \pi(\mathtt{G},\mathtt{A}) & 0 & 0 & \pi(\mathtt{G},\mathtt{T}) & \pi(\mathtt{G},\mathtt{C}) & \pi(\mathtt{G},\mathtt{G}) & 0 & 0 \\ \pi(\mathtt{T},\mathtt{A}) & \pi(\mathtt{T},\mathtt{C}) & \pi(\mathtt{T},\mathtt{G}) & \pi(\mathtt{T},\mathtt{T}) & 0 & 0 & 0 & 0 \\ \pi(\mathtt{C},\mathtt{A}) & \pi(\mathtt{C},\mathtt{C}) & 0 & \pi(\mathtt{C},\mathtt{T}) & 0 & 0 & \pi(\mathtt{C},\mathtt{G}) & 0 \\ \pi(\mathtt{G},\mathtt{A}) & 0 & 0 & \pi(\mathtt{G},\mathtt{T}) & \pi(\mathtt{G},\mathtt{C}) & 0 & 0 & \pi(\mathtt{G},\mathtt{G}) \\ \pi(\mathtt{G},\mathtt{A}) & 0 & 0 & \pi(\mathtt{G},\mathtt{T}) & \pi(\mathtt{G},\mathtt{C}) & \pi(\mathtt{G},\mathtt{G}) & 0 & 0 \\ \pi(\mathtt{G},\mathtt{A}) & 0 & 0 & \pi(\mathtt{G},\mathtt{T}) & \pi(\mathtt{G},\mathtt{C}) & 0 & 0 & \pi(\mathtt{G},\mathtt{G}) \end{pmatrix}.$$

From now on, we assume that our motif problem with Md reference model is embedded into the Markov chain $Z_{d:\ell}$ whose transition matrix is decomposed into $\mathbf{T} = \mathbf{P} + \mathbf{Q}$ where matrices $\mathbf{P}$ and $\mathbf{Q}$ are defined for all p, q by: $\mathbf{P}(p,q) = \mathbf{T}(p,q)\mathbf{1}_{q \notin \mathcal{F}}$, and $\mathbf{Q}(p,q) = \mathbf{T}(p,q)\mathbf{1}_{q \in \mathcal{F}}$.

3.4 Main results

We present here the main results that are then used to derive exact computations and various approximations of $S(n)$. In all this section, we assume that N is the random number of occurrences of $\mathcal{M}$ in $X_{1:\ell}$, a sequence generated by a Md model ($X_{1:d}$ being fixed) with $d \geqslant 0$. we denote by $\mathbf{T} = \mathbf{P} + \mathbf{Q}$ be the transition $(L \times L)$ matrix of the Markov chain embedding of the corresponding problem. We also introduce two vectors: $\mathbf{u}$ a $1 \times L$ vector filled with '0' and having a '1' in the position corresponding to $X_{1:d}$, and $\mathbf{v}$ a $L \times 1$ vector of '1'.

Proposition 4 (probability generating function)**.** If we denote by $G(y) = \mathbb{E}[y^N]$ the probability generating function (pgf) of N, then we have:

$$G(y) = \sum_{n \geqslant 0} \mathbb{P}(N = n) y^n = \mathbf{u}(\mathbf{P} + y\mathbf{Q})^{\ell - d}\mathbf{v}. \tag{12}$$

Proof. The first equality derives directly from the definition of $G(y)$. For the second equality now, it is clear that $\mathbf{u}(\mathbf{P}+\mathbf{Q})^{\ell-d}$ gives the marginal distribution of Z_ℓ. We then connect this distribution to N by counting the number of times we use the transitions of $\mathbf{Q}$ with the dummy variable y so that $\mathbf{u}(\mathbf{P}+y\mathbf{Q})^{\ell-d}$ gives the joint distribution of (Z_ℓ, N). Finally, we sum up the contributions of all states using the product with $\mathbf{v}$. □

For example, let us consider $\mathcal{M} = \texttt{G(G|C)G}$ and $X_{1:12}$ generated by a M0 model with parameters $\pi(\texttt{A}) = \pi(\texttt{T}) = 0.10$ and $\pi(\texttt{C}) = \pi(\texttt{G}) = 0.40$. Proposition 4 hence gives:

$$G(y) = \begin{pmatrix}1&0&0&0&0&0\end{pmatrix} \times \begin{pmatrix} 0.6 & 0.4 & 0 & 0 & 0 & 0 \\ 0.2 & 0 & 0.4 & 0.4 & 0 & 0 \\ 0.6 & 0 & 0 & 0 & 0.4y & 0 \\ 0.2 & 0 & 0.4 & 0 & 0 & 0.4y \\ 0.2 & 0 & 0.4 & 0.4 & 0 & 0 \\ 0.2 & 0 & 0.4 & 0 & 0 & 0.4y \end{pmatrix}^{12} \times \begin{pmatrix}1\\1\\1\\1\\1\\1\end{pmatrix} \tag{13}$$

$$= 0.33369 + 0.31148y + 0.19357y^2 + 0.09681y^3 + 0.04140y^4 + 0.01569y^5 + 0.00528y^6 + 0.00157y^7 + 0.00042y^8 + 0.00008y^9 + 0.00002y^{10}. \tag{14}$$

From this result, we have the whole distribution of N: support is $\{0, 1, \ldots, 10\}$, $\mathbb{P}(N=0) = 0.33369$, $\mathbb{P}(N=1) = 0.31148$, …, $\mathbb{P}(N=10) = 0.00002$. We can also easily derive moments of N from this distribution: $\mathbb{E}[N] = 1.28$, $\sigma[N] = 1.29$.

Lemma 5 (derivatives of the pgf)**.** For any $k \leqslant 0$, the order k derivative of the pgf G is given by:

$$G^{(k)}(y) = k![z^k]\mathbf{u}(\mathbf{P}+y\mathbf{Q}+z\mathbf{Q})^{\ell-d}\mathbf{v} \tag{15}$$

where the $[z^k]$ operator denotes the extraction of the coefficient of z^k in the expression.

Proof. The formal proof can be found in Nuel (2010) in a slightly less general case. Here we prove it only for the first two derivatives in the particular case where $\ell - d = 3$. Starting from $G(y) = \mathbf{u}(\mathbf{P}+y\mathbf{Q})^3\mathbf{v}$ we get:

$$G'(y) = \mathbf{u}\left(\mathbf{Q}(\mathbf{P}+y\mathbf{Q})^2 + (\mathbf{P}+y\mathbf{Q})\mathbf{Q}(\mathbf{P}+y\mathbf{Q}) + (\mathbf{P}+y\mathbf{Q})^2\mathbf{Q}\right)\mathbf{v} \tag{16}$$

and

$$G''(y) = 2\mathbf{u}\left(\mathbf{Q}^2(\mathbf{P}+y\mathbf{Q}) + \mathbf{Q}(\mathbf{P}+y\mathbf{Q})\mathbf{Q} + (\mathbf{P}+y\mathbf{Q})\mathbf{Q}^2\right)\mathbf{v} \tag{17}$$

which are easily connected to the terms coefficients of z^1 and z^2 in $\mathbf{u}(\mathbf{P}+y\mathbf{Q}+z\mathbf{Q})^{\ell-d}\mathbf{v}$. □

If we denote for all $k \geqslant 0$ the k-th *factorial moment* of N by $F_k = \mathbb{E}[N!/(N-k)!]$, then, by the definition of the pgf, it is clear that $F_k = G^{(k)}(0)$, and thanks to Lemma 5 we get:

$$F_k = k![z^k]\mathbf{u}(\mathbf{T}+z\mathbf{Q})^{\ell-d}\mathbf{v}. \tag{18}$$

And if we now denote the moment generating function (mgf) of N by $M(t) = \mathbb{E}[e^{tN}] = G(e^t)$, and the cumulant generating function (cgf) of N by $\Lambda(t) = \log\mathbb{E}[e^{tN}] = \log M(t) = \log G(e^t)$, we get directly the k-th moment of N: $\mathbb{E}[N^k] = M^{(k)}(0)$; and the k-th cumulant of N: $\kappa_k = \Lambda^{(k)}(0)$.

Corollary 6 (characteristics moments)**.** If we denote by $\mu = \kappa_1$ the *expectation* of N, by $\sigma = \sqrt{\kappa_2}$ the *standard deviation* of N, by $\gamma_1 = \kappa_3/\sigma^3$ the *skewness* of N, and by $\gamma_1 = \kappa_4/\sigma^4$ the *excess kurtosis* of N, then we get: $\mu = F_1$, $\sigma^2 = F_2 + F_1 - F_1^2$,

$$\gamma_1 = \frac{3F_2 - 3F_1^2 + F_3 - 3F_1F_2 + 2F_1^3 + F_1}{\sigma^3}, \tag{19}$$

and

$$\gamma_2 = \frac{7F_2 - 7F_1^2 + 6F_3 - 18F_1F_2 + 12F_1^3 + F_4 - 4F_1F_3 - 3F_2^2 + 12F_1^2F_2 - 6F_1^4 + F_1}{\sigma^4}. \tag{20}$$

Proof. On just need to compute the derivatives $\Lambda^{(1)}(0)$, $\Lambda^{(2)}(0)$, $\Lambda^{(3)}(0)$, and $\Lambda^{(4)}(0)$. □

If we consider again $\mathcal{M} = \mathtt{G(G|C)G}$ and $X_{1:12}$ generated by a M0 model with parameters $\pi(\mathtt{A}) = \pi(\mathtt{T}) = 0.10$ and $\pi(\mathtt{C}) = \pi(\mathtt{G}) = 0.40$. Eq. (18) hence gives:

$$\sum_{k\leqslant 0} \frac{F_k}{k!} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0.6 & 0.4 & 0 & 0 & 0 & 0 \\ 0.2 & 0 & 0.4 & 0.4 & 0 & 0 \\ 0.6 & 0 & 0 & 0 & 0.4+0.4y & 0 \\ 0.2 & 0 & 0.4 & 0 & 0 & 0.4+0.4y \\ 0.2 & 0 & 0.4 & 0.4 & 0 & 0 \\ 0.2 & 0 & 0.4 & 0 & 0 & 0.4+0.4y \end{pmatrix}^{12} \times \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \tag{21}$$

$$\begin{aligned} &= 1 + 1.28y + 1.01683y^2 + 0.61211y^3 + 0.29709y^4 + 0.11835y^5 \\ &\quad + 0.03845y^6 + 0.00992y^7 + 0.00193y^8 + 0.00025y^9 + 0.00002y^{10}. \end{aligned} \tag{22}$$

From this result, we can get all factorial moments of N: $\mathbb{E}(1) = F_0 = 1$, $\mathbb{E}(N) = F_1 = 1.28$, $\mathbb{E}(N(N-1)) = F_2 = 2.033664$, $\mathbb{E}(N(N-1)(N-2)) = F_3 = 3.6726374$, $\mathbb{E}(N(N-1)(N-2)(N-3)) = F_4 = 7.1302266, \ldots, \mathbb{E}(N!/(N-10)!) = F_{10} = 60.881161$. Thanks to Corollary 6 we get the following characteristic moments: $\mu = 1.28$, $\sigma = 1.294320$, $\gamma_1 = 1.163783$, $\gamma_2 = 1.492661$.

3.5 Exact computations

As we have seen above, Proposition 4 provides a way to obtain the whole distribution of N by computing $G(y) = \mathbf{u}(\mathbf{P} + y\mathbf{Q})^{\ell-d}\mathbf{v}$ from which we can easily derive $S(n)$ for any $n \geqslant 0$:

$$S(n) = \begin{cases} +\log_{10}\left(\sum_{k=0}^{n}[y^k]G(y)\right) & \text{if } n \leqslant \mathbb{E}[N] \\ -\log_{10}\left(\sum_{k=n}^{+\infty}[y^k]G(y)\right) & \text{if } n > \mathbb{E}[N] \end{cases}.$$

From the algorithmic point of view, there are basically two approaches to compute $S(n)$ using Expression (12). The first one, called *power*, consists in computing $(\mathbf{P} + y\mathbf{Q})^{\ell-d}$ using the power method and a binary decomposition of $\ell - d$. Ex: if $\ell - d = 1097$ then $\ell - d = 2^{10} + 2^6 + 2^3 + 2^0$. We then just have to recursively compute $\mathbf{D}_k(y) = (\mathbf{P} + y\mathbf{Q})^{2^k}$ using the relation $\mathbf{D}_{k+1}(y) = \mathbf{D}_k(y) \times \mathbf{D}_k(y)$ for all $k \geqslant 0$. Since in the computation of $S(n)$ we are only interested in terms of degree n or less (or n or more), we can easily truncate [2] all polynomials at degree n thus dramatically reducing the computational costs of polynomial products. We end

[2] In the case of over-representation, all contributions of degree n or more are summed into the term of degree n.

up with a $O(\log_2 \ell \times n^2 \times L^3)$ complexity in time where L is the order of the transition matrix $\mathbf{T} = \mathbf{P} + \mathbf{Q}$. The corresponding memory complexity is $O(\log_2 \ell \times n \times L^2)$. Since the length ℓ of the dataset appears in a logarithmic scale in these complexity, the power approach is obviously suitable for large datasets (ex: $\ell = 10^6$ or $\ell = 10^9$). Unfortunately, the cubic complexity with L (quadratic in memory) prevents the approach to deal with complex motifs with high L. One should also note that the quadratic complexity in n could really be a problem when dealing with frequent motifs and/or large datasets. In order to overcome this problem, Ribeca & Raineri (2008) suggested to use fast Fourier transforms (FFT) to perform all polynomial product hence replacing n^2 by $n \log_2 n$ in the time complexity. However appealing at first glance, this approach is not recommended in practice since the FFT products in floating-point arithmetics induce numerical instabilities that make totally unreliable the smallest coefficients of the polynomials. And unfortunately, these coefficients are precisely the one needed to study the tail distribution of N.

Another interesting approach called *full recursion*, consists in computing $\mathbf{v}_i = (\mathbf{P} + y\mathbf{Q})^i \mathbf{v}$ for all $0 \leqslant i \leqslant \ell - d$ recursively using the relation $\mathbf{v}_{i+1} = (\mathbf{P} + y\mathbf{Q})\mathbf{v}_i$. There are two main interests for this approach: 1) we have only products between polynomials of degree 1 and polynomials of degree n (by dropping terms of degree greater than n like in the power approach); 2) we can take full advantage of the sparse structure (only $L \times |\mathcal{A}|$ non-zero terms in the worst case) of the transition matrix $\mathbf{T} = \mathbf{P} + \mathbf{Q}$. The resulting complexity is $O(\ell \times L \times |\mathcal{A}| \times n)$ in time, and $O(L \times n)$ in memory. Since these complexities are linear with L, this approach is able to handle very complex motifs. The drawback is that the approach can be very slow when dealing with large ℓ and n. It exists a sophisticated version of this recursion called *partial recursion* (see Nuel & Dumas, 2010) which allows to replace $\ell \times n$ by $\log \ell \times n^2$ in the time complexity. However, the quadratic complexity in n and numerical instabilities in floating-point arithmetic restrains its use to small n (ex: $n \leqslant 10$).

For completeness, let us point out another approach to the problem. The idea is that we can derive from Expression (12) the following expression:

$$G(y,z) = \sum_{n \geqslant 0} \sum_{\ell \geqslant d} \mathbb{P}(N_\ell = n) y^n z^\ell = \mathbf{u} z^d (\mathbf{I} - \mathbf{P}z + yz\mathbf{Q})^{-1} \mathbf{v} \tag{23}$$

where $\mathbf{I}$ is the identity matrix and N_ℓ the number of matching position in $X_{1:\ell}$. It is then possible to obtain $\mathbb{P}(N_\ell = n)$ for any ℓ and n using (fast) Taylor expansions of $G(y,z)$. For the mathematician, this approach is so "natural" that it is often referred as the "golden" approach to the problem of motif significance (Nicodème et al., 2002). However, this approach suffers several severe drawbacks that dramatically limits its practical interest: 1) the approach needs sophisticated computer algebra systems to be implemented (rather than simple floating point arithmetic for the previous approaches); 2) the explicit computation of $(\mathbf{I} - \mathbf{P}z + yz\mathbf{Q})^{-1}$ could be very time (and memory) consuming; 3) even if the explicit computation of the inverse matrix is avoided (which is highly advisable), the coefficient extraction using state of the art techniques (like high-order lifting for example) is often slower than the much simpler alternative developed above (see Nuel & Dumas, 2010, for details).

Considering either the power or the recursion approaches we obtain easy to implement algorithms allowing to compute the exact value of $S(n)$ in all cases except when dealing with high complexity motifs (large L) and/or frequent motifs (large n). But even if we stick to more tractable cases, exact computations could be slow. The question hence is: is it possible to compute fast and reliable approximations of $S(n)$?

ℓ	expectation	std. dev.	skewness	e. kurtosis	time (s)
12	1.280000	1.294320	1.163783	1.492661	0.01
120	15.104000	4.585724	0.361328	0.149974	0.02
1200	153.344000	14.648033	0.113920	0.014936	0.03
12000	1535.744000	46.367282	0.036014	0.001492	0.04
120000	15359.744000	146.640798	0.011394	−0.000410	0.05

Table 1. Characteristic moments the number N of occurrences of motif $\mathcal{M} = \mathtt{G(G|C)G}$ in a sequence $X_{1:\ell}$ generated by a M0 model with parameters $\pi(\mathtt{A}) = \pi(\mathtt{T}) = 0.10$ and $\pi(\mathtt{C}) = \pi(\mathtt{G}) = 0.40$. Computation performed using the power approach.

3.6 Near-Gaussian approximations

Since the random count N is basically defined by Eq. (2) as large sum of Bernouilli variables, the idea of approximating the distribution of N using Gaussian approximation sounds appealing. Indeed, Gaussian approximations are historically the first ones to have been suggested for this problem (Cowan, 1991; Kleffe & Borodovski, 1997; Pevzner et al., 1989; Prum et al., 1995). From the theoretical point of view, Central Limit Theorems (CLT) for weakly dependent variables ensure that N is asymptotically normal distributed. On Table 1, we can see the characteristic moments of N for motif $\mathcal{M} = \mathtt{G(G|C)G}$ and various value of the sequence lengths ℓ. According to theory, we observe that the skewness and excess kurtosis both decease toward 0 when ℓ grows (a normal distribution has null skewness and excess kurtosis). But it is also clear that N is not normally distributed for small values of ℓ. As a consequence, the quality of a Gaussian approximation for $S(n)$ is expected to be questionable at finite distance.

In order to overcome this issue, Nuel (2010) suggested to consider near Gaussian approximations instead of simple Gaussian approximations for this problem. The idea is simply to perform a higher order asymptotic development that exploits more than the two first moments of N. This technique is known as the Edgeworth's expansion. Blinnikov & Moessner (1998) gives a general (and rather complicated) formula for this expansion. For practical purpose, we present the result only up to order 3 expansions.

Proposition 7 (Edgeworth's expansion)**.** If we denote by $\varphi(z) = \exp(-z^2/2)/\sqrt{2\pi}$ the probability distribution function (pdf) of a standard Gaussian, then for all $n \geqslant 0$ we have the following approximation:

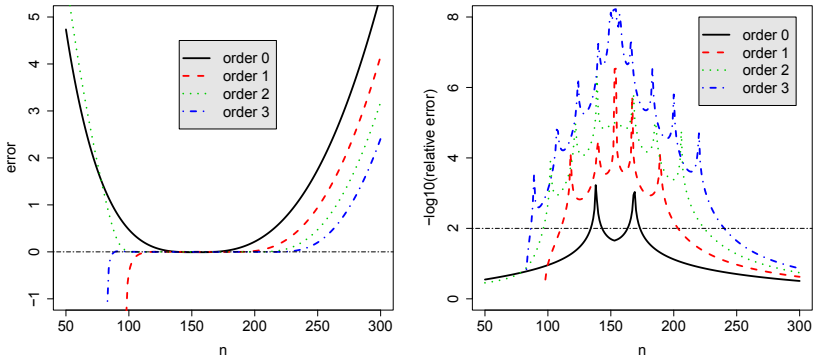
$$\mathbb{P}(N = n) \simeq \frac{\varphi(z)}{\sigma}\left(C_0(z) + \sigma C_1(z) + \sigma^2 C_2(z) + \sigma^3 C_3(z)\right) \tag{24}$$

with

$$C_0(z) = 1 \quad C_1(z) = \frac{S_3}{6}H_3(z) \quad C_2(z) = \frac{S_4}{24}H_4(z) + \frac{S_3^2}{72}H_6(z) \tag{25}$$

$$C_3(z) = \frac{S_5}{120}H_5(z) + \frac{S_3 S_4}{144}H_7(z) + \frac{S_3^3}{1296}H_9(z) \tag{26}$$

where $\mu = \mathbb{E}[N]$, $\sigma = \sqrt{\mathbb{V}[N]}$, $z = (n-\mu)/\sigma$, $S_k = \kappa_k/\sigma^{2k-2}$ for all $k \geqslant 1$, and where $H_k(z)$ are the Hermite polynomials defined recursively by $H_0(z) = 1$ and $H_k(z) = zH_{k-1}(z) - H'_{k-1}(z)$ for all $k \geqslant 1$.



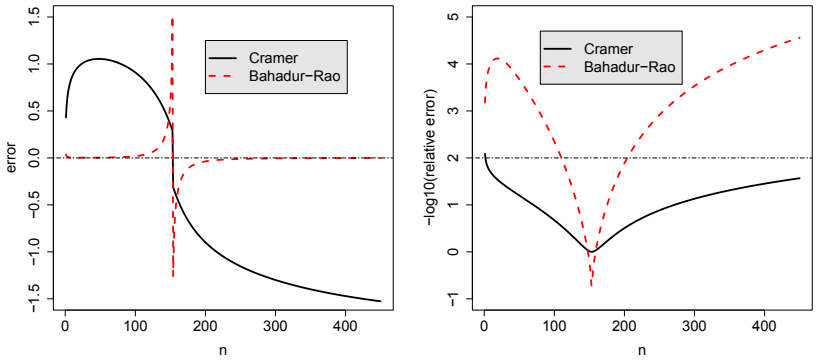
(a) Error (b) Relative error

Fig. 6. Reliability of NG approximations for $\mathcal{M} = \mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$ on a random sequence $X_{1:\ell}$ generated by a M0 model with parameters $\pi(\mathtt{A}) = \pi(\mathtt{T}) = 0.10$ and $\pi(\mathtt{C}) = \pi(\mathtt{G}) = 0.40$, and with $\ell = 1200$. The error $\mathrm{NG}_h(n) - S(n)$ is given on Figure (a); and the relative error (log-scale) $-\log_{10}|\mathrm{NG}_h(n) - S(n)|/|S(n)|$ on Figure (b). The horizontal rule indicates the null error on Figure (a), and the threshold corresponding to two correct digits on Figure (b).

For $h \in \{0, 1, 2, 3\}$ we define the Near Gaussian (NG) approximation of order h of $S(n)$ by:

$$\mathrm{NG}_h(n) = \begin{cases} +\log_{10}\left(\sum_{k=0}^{n} \frac{1}{\sigma}\varphi\left(\frac{k-\mu}{\sigma}\right)\sum_{j=0}^{h}\sigma^j C_j\left(\frac{k-\mu}{\sigma}\right)\right) & \text{if } n \leqslant \mathbb{E}[N] \\ -\log_{10}\left(\sum_{k=n}^{+\infty} \frac{1}{\sigma}\varphi\left(\frac{k-\mu}{\sigma}\right)\sum_{j=0}^{h}\sigma^j C_j\left(\frac{k-\mu}{\sigma}\right)\right) & \text{if } n > \mathbb{E}[N] \end{cases}. \tag{27}$$

We can see on Figure 6 the reliability of NG approximations. In solid black, the order 0 approximation corresponds to the classical Gaussian approximation. Unsurprisingly, this central limit approximation is accurate for the center of the distribution (n close to the expectation $\mu = 153.3$), the reliability quickly deceases when $|n - \mu|$ increases.
Central limit theorems (CLT) for N have established long ago that N should be asymptotically Gaussian distributed. The problem however with CLT theorems is that the quality of the resulting approximation dramatically decreases at finite distance when considering tail distribution events. Here we try to overcome the issue by considering Near-Gaussian approximations that exploits higher moments of N to improve the quality of the approximations. In order to do this, a critical problem is first to obtain the first k-th moments of N. Of course we can access these moments by computing the full distribution of N, but if it is possible to do so, why bothering with approximations. We hence need an method to compute the moments of N whose complexity should be somehow significantly smaller than the complete exact computations. With higher order approximation, we can see a dramatic improvement of reliability of the results, with a noticeable increase of the region where at least two digits are correct (up to $n \in [80; 240]$ for NG_3).



(a) Error (b) Relative error

Fig. 7. Reliability of CB and BR approximations for $\mathcal{M} = \mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$ on a random sequence $X_{1:\ell}$ generated by a M0 model with parameters $\pi(\mathtt{A}) = \pi(\mathtt{T}) = 0.10$ and $\pi(\mathtt{C}) = \pi(\mathtt{G}) = 0.40$, and with $\ell = 1200$. The error $\mathrm{CB}(n) - S(n)$ or $\mathrm{BR}(n) - S(n)$ is given on Figure (a); and the relative error (log-scale) $-\log_{10}|\mathrm{CB}(n) - S(n)|/|S(n)|$ or $-\log_{10}|\mathrm{BR}(n) - S(n)|/|S(n)|$ on Figure (b). The horizontal rule indicates the null error on Figure (a), and the threshold corresponding to two correct digits on Figure (b).

From the computational point of view, the order h approximation requires the cumulants of N up to order $h+2$. Using the power approach, the resulting complexity is hence $O(\log_2 \ell \times h^2 \times L^3)$ in time and $O(\log_2 \ell \times (h+2) \times L^2)$ in memory. Using the recursion, the complexity resulting complexity is $O(\ell \times L \times |\mathcal{A}| \times h)$ in time, and $O(L \times h)$ in memory. In both cases, the computational time drops significantly from the exact computations.
Thanks to NG approximations, we hence have a fast and reliable way to compute an approximation of $S(n)$ when n falls in the center of the distribution (ex: $|S(n)| \leqslant 3.0$), but NG approximations unfortunately remain totally unreliable for tail distribution events (ex: $|S(n)| > 3.0$), which are moreover often precisely the event of interest. Fortunately we have a solution to this problem.

3.7 Bahadur-Rao

We want here to study specifically the tail distribution of N with events on the form $\mathbb{P}(N \geqslant n)$ with large n (or $\mathbb{P}(N \leqslant n)$ with small n). For all $t > 0$ let us first notice that we can use the Markov inequality to write: $\mathbb{P}(N \geqslant n) = \mathbb{P}(e^{tN} \geqslant e^{tn}) \leqslant \mathbb{E}[e^{tN}]/e^{tn} = \exp(\Lambda(t) - tn)$. By taking the smallest of these bounds for $t > 0$ we hence get: $\log \mathbb{P}(N \geqslant n) \leqslant \Lambda(\tau) - \tau n$ with τ defined by $\Lambda'(\tau) = n$. This upper bound, known as the Chernoff's Bound (CB), is often surprisingly sharp for events located in the tail distribution. By dealing symmetrically with $\mathbb{P}(N \leqslant n)$ and $t < 0$ we hence obtain the following approximation for $S(n)$:

$$\mathrm{CB}(n) = \delta_n \frac{\tau n - \Lambda(\tau)}{\log(10)} \tag{28}$$

where $\delta_n = -1$ if $n \leqslant \mathbb{E}[N]$, and $\delta_n = +1$ if $n > \mathbb{E}[N]$.

From the computational point of view, the solution τ of $\Lambda'(\tau) = n$ can be easily determined numerically using (for example) using the Newton-Raphson sequence (Press et al., 1992). Starting for a first guess t_0 (ex: $t_0 = 0$), one performs $t_{i+1} = t_i + (n - \Lambda'(t_i))/\Lambda''(t_i)$ for $i \geqslant 0$ until convergence to τ. The computation of Λ, Λ', and Λ'' being possible thanks to Lemma 5 and the following formulas:

$$\Lambda(t) = G(e^t) \quad \Lambda'(t) = \frac{e^t G'(e^t)}{G(e^t)} \quad \Lambda''(t) = \frac{e^{2t} G''(e^t)}{G(e^t)} - \frac{e^{2t} G'(t)^2}{G(e^t)^2} + \frac{e^t G'(e^t)}{G(e^t)} \tag{29}$$

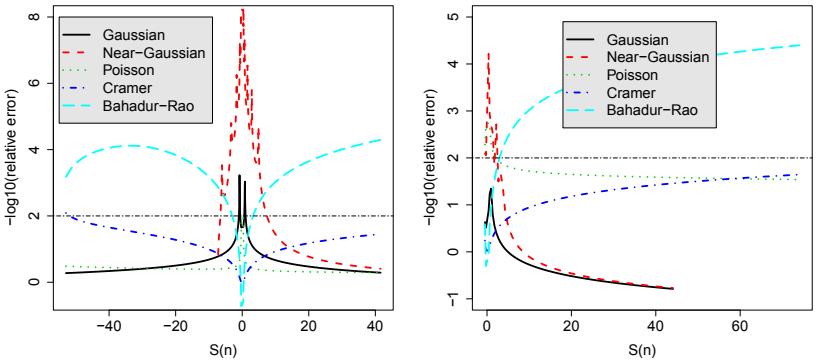
with $G(e^t) = [z^0]\mathbf{u}(\mathbf{P} + e^t\mathbf{Q} + z\mathbf{Q})^{\ell-d}\mathbf{v} = \mathbf{u}(\mathbf{P} + e^t\mathbf{Q})^{\ell-d}\mathbf{v}$, $G'(e^t) = [z^1]\mathbf{u}(\mathbf{P} + e^t\mathbf{Q} + z\mathbf{Q})^{\ell-d}\mathbf{v}$, and $G''(e^t) = 2[z^2]\mathbf{u}(\mathbf{P} + e^t\mathbf{Q} + z\mathbf{Q})^{\ell-d}\mathbf{v}$.
Moreover, this bound can be further refined using the Bahadur-Rao Theorem (Bahadur & Rao, 1960) and gives the following approximation for $S(n)$:

$$\mathrm{BR}(n) = \mathrm{CB}(n) + \delta_n \log 10 \left((1 - e^{-|\tau|}) \sqrt{2\pi\Lambda''(\tau)} \right). \tag{30}$$

From the computational point of view, $\mathrm{CB}(n)$ and $\mathrm{BR}(n)$ can be computed either with the power approach with complexities $O(\log_2 \ell \times L^3)$ in time and $O(\log_2 \ell \times L^3)$ in memory; or with the recursion approach with complexities $O(\ell \times L \times |\mathcal{A}|)$ in time and $O(L \times |\mathcal{A}|)$ in memory.
On Figure 7 we can see the reliability of the approximations $\mathrm{CB}(n)$ and $\mathrm{BR}(n)$. Unsurprisingly, the farther from the center of the distribution, the better are both approximations. We also observe that $\mathrm{BR}(n)$ is a dramatic improvement over $\mathrm{CB}(n)$ since it obtains at least two correct digits of $S(n)$ for all n but on $[120, 200]$. At the end of previous section, we have seen that the order 3 NG approximation achieves the same precision for region $[80; 240]$, hence, by combining both $\mathrm{NG}_3(n)$ (for the center of the distribution) and $\mathrm{BR}(n)$ (for the tail distributions), one can achieve at least two correct digits of $S(n)$ on the whole bulk of the distribution for a modest computational cost.

4. Discussion

Obtaining the distribution of motif count in random sequences is a very challenging problem that has attracted considerable attention from mathematicians and computer scientists in the last fifty years. Recently however, a significant advance has been obtained by connecting the well-known theory of pattern matching and automata to the Markov chain embedding technique Lladser (2007); Nuel (2008a); Nuel & Prum (2007). Thanks to this finding, it is now possible to deal with simple (runs of 1 in binary sequences, single words, etc.) or complex motifs (PROSITE signature, gapped motifs, etc.) using the same general framework.
Using exact approaches, it is possible to obtain efficiently the first moments of any motif count N, and even the complete distribution of N. As a consequence, the computation of $S(n)$ is now tractable for a wide range of motif problems including large datasets or complex motifs. However, the case of complex frequent motifs in large datasets remains an open problem (Nuel & Dumas, 2010).
As an alternative to exact computations, a wide range of approximations have been developed (see Lothaire, 2005; Nuel, 2006b; Reignier, 2000, for a review). We can basically classify these approximations in three categories: 1) Gaussian approximations (Cowan, 1991; Kleffe & Borodovski, 1997; Nuel, 2010; Pevzner et al., 1989; Prum et al., 1995); 2) Poisson approximations Erhardsson (2000); Geske et al. (1995); Godbole (1991); Reinert & Schbath (1999); Roquain & Schbath (2007); 3) large deviations approximations Denise et al. (2001); Nuel (2004).



(a) G(G|C)G, frequent motif (b) A(A|T)A, rare motif

Fig. 8. Relative error in log-scale for various approximations of $S(n)$ ($n = 0, \ldots, 200$) in a sequence $X_{1:\ell}$ generated by a M0 model with parameters $\pi(\mathtt{A}) = \pi(\mathtt{T}) = 0.10$ and $\pi(\mathtt{C}) = \pi(\mathtt{G}) = 0.40$.

In this chapter we deliberately left aside the Poisson-based approximations and considered only two of these approximations: the (Near-) Gaussian approximations with $\mathrm{NG}_h(n)$, and the large deviations based approximations with $\mathrm{CB}(n)$ and $\mathrm{BR}(n)$. The reason why Poisson-based approximations are not considered here is basically practical, these approximations cannot be directly derived from the formalism of this manuscript and require the introduction of many tedious notions like clumps, overlapping words and so on. However, we compare here the performance of all these approximations (including compound Poisson approximations) in the case where $X_{1:\ell}$ generated by a M0 model with parameters $\pi(\mathtt{A}) = \pi(\mathtt{T}) = 0.10$ and $\pi(\mathtt{C}) = \pi(\mathtt{G}) = 0.40$ i.i.d. DNA sequence, and for two motifs: the frequent $\mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$, and the rare $\mathtt{A}(\mathtt{A}|\mathtt{T})\mathtt{A}$.

We can see on Figure 8 the relative error (in log-scale) for all approximations. For Gaussian approximations, performances are only good in the very center of the distribution (for n very close to $\mathbb{E}(n)$) for the frequent motif $\mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$, and performances are poor almost everywhere for the rare motif $\mathtt{T}(\mathtt{A}|\mathtt{T})\mathtt{T}$. This observation to consistent with the well known claim that "Gaussian approximations a more suitable for frequent motif" (Lothaire, 2005). It has however to be pointed out that even in the most favorable case (with highly frequent motif), Gaussian approximations totally fail to capture the tail distribution of N and hence not suitable for the highly significant observations we usually encounter in biological sequences (Nuel, 2006b). If we consider now the near-Gaussian approximation, taking into account more moments of N dramatically improve the result for both motifs, but the failure to deal with extreme distribution events remains.

Compound Poisson approximations are known to be extremely sensitive to the relative abundance of the motif of interest in the sequence, being more accurate for rare motifs (Lothaire, 2005; Roquain & Schbath, 2007). It is hence not a surprise to see that Poisson approximations are totally unreliable for the frequent motif $\mathtt{G}(\mathtt{G}|\mathtt{C})\mathtt{G}$. For the rare motif $\mathtt{T}(\mathtt{A}|\mathtt{T})\mathtt{T}$ we naturally obtain much better results but like for Gaussian approximations, and even in this favorable case, reliability decreases in the tail distribution. Considering that Poisson

approximations are not easily generalizable to motifs defined by regular expressions, that their computations could be complicated and time consuming, and that their reliability is highly questionable in some configurations, it seems advisable to avoid their use is most cases.
With large deviations based approximations, we unsurprisingly get a low reliability in the center of the distribution, but a high reliability in the tail distribution. With Bahadur-Rao precise approximations, the improvement over the classical Chernoff's bound is quite impressive, and the complementarity with Near-Gaussian approximations clearly shows that a combination of both approaches could be a very efficient way to obtain reliable approximations of $S(n)$ for all n.
In this chapter we gave all the necessary ingredients to assess the significance score of motif in a biological sequence using state of the art results, including several unpublished ones: Lemma 5 which is an extension of the results of Nuel (2010), and the complete "Bahadur-Rao" Section which provides interesting improvements over previous large deviations work (Denise et al., 2001; Nuel, 2004).
Let us finally point out that for the sake of compactness, we have left aside some interesting questions and extensions like: approximate matching Hopcroft et al. (2001), renewal occurrences (Nuel, 2006b; Roquain & Schbath, 2007), joint distributions (Nuel, 2008b; Stefanov & Szpankowski, 2007), dataset with many sequences (Nuel et al., 2010), and sensitivity to parameter estimation (Nuel, 2006c). Even if some results are already available for these problems, many questions still have to be answered in the exciting and challenging field of the distribution of motifs in random sequences.

5. References

Allauzen, C. & Mohri, M. (2006). A unified construction of the glushkov, follow, and antimirov automata, *in* R. KrA?lovic & P. Urzyczyn (eds), *Mathematical Foundations of Computer Science 2006*, Vol. 4162 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 110–121.

Antzoulakos, D. L. (2001). Waiting times for patterns in a sequence of multistate trials, *J. Appl. Prob.* 38: 508–518.

Bahadur, R. R. & Rao, R. R. (1960). On deviations of the sample mean, *The Annals of Math. Statistics.* 31(4): 1015–1027.

Beaudoing, E., Freier, S., Wyatt, J., Claverie, J.-M. & Gautheret, D. (2000). Patterns of variant polyadenylation signal usage in human genes, *Genome Res.* 10(7): 1001–1010.

Blinnikov, S. & Moessner, R. (1998). Expansions for nearly Gaussian distributions, *Astron. Astrophys. Suppl. Ser.* 130: 193–205.

Boeva, V., ClA©ment, J., RA©gnier, M. & Vandenbogaert, M. (2005). Assessing the significance of sets of words, *Combinatorial Pattern Matching 05, Lecture Notes in Computer Science, vol. 3537*, Springer-Verlag.

Brazma, A., Jonassen, I., Vilo, J. & Ukkonen, E. (1998). Predicting gene regulatory elements in silico on a genomic scale, *Genome Res.* 8(11): 1202–1215.

Bryne, J., Valen, E., Tang, M., Marstrand, T., Winther, O., da Piedade, I., Krogh, A., Lenhard, B. & Sandelin, A. (2008). JASPAR, the open access database of transcription factor-binding profiles: new content and tools in the 2008 update., *Nucleic Acids Res.* 36: 102–106.

Chan, H. P., Zhang, N. R. & Chen, L. H. Y. (2010). Importance sampling of word patterns in dna and protein sequences, *J. of Comput. Biol.* 17(12): 1697–1709.

Chang, Y.-M. (2005). Distribution of waiting time until the rth occurrence of a compound pattern, *Statistics and Probability Letters* 75(1): 29–38.

Cornish-Bowden (1985). IUPAC-IUB symbols for nucleotide nomenclature, *Nucl. Acids Res.* 13: 3021–3030.

Cowan (1991). Expected frequencies of dna patterns using whittle's formula, *J. Appl. Prob.* 28: 886–892.

Crochemore, M. & Stefanov, V. (2003). Waiting time and complexity for matching patterns with automata, *Info. Proc. Letters* 87(3): 119–125.

Denise, A., RA©gnier, M. & Vandenbogaert, M. (2001). Assessing the statistical significance of overrepresented oligonucleotides, *Lecture Notes in Computer Science* 2149: 85–97.

El Karoui, M., Biaudet, V., Schbath, S. & Gruss, A. (1999). Characteristics of chi distribution on different bacterial genomes, *Res. Microbiol.* 150: 579–587.

Erhardsson, T. (2000). Compound Poisson approximation for counts of rare patterns in Markov chains and extreme sojourns in birth-death chains, *Ann. Appl. Probab.* 10(2): 573–591.

Fatemi, M., Pao, M., Jeong, S., Gal-Yam, E., Egger, G., Weisenberger, D. & Jones, P. (2005). Footprinting of mammalian promoters: use of a cpg dna methyltransferase revealing nucleosome positions at a single molecule level, *Nucleic Acids Res* 33(20): 176.

Frith, M. C., Spouge, J. L., Hansen, U. & Weng, Z. (2002). Statistical significance of clusters of motifs represented by position specific scoring matrices in nucleotide sequences, *Nucl. Acids. Res.* 30(14): 3214–3224.

Fu, J. C. (1996). Distribution theory of runs and patterns associated with a sequence of multi-state trials, *Statistica Sinica* 6(4): 957–974.

Geske, M. X., Godbole, A. P., Schaffner, A. A., Skrolnick, A. M. & Wallstrom, G. L. (1995). Compound poisson approximations for word patterns under markovian hypotheses, *J. Appl. Probab.* 32: 877–892.

Godbole, A. P. (1991). Poissons approximations for runs and patterns of rare events, *Adv. Appl. Prob.* 23.

Green, T. J., Gupta, A., Miklau, G., Onizuka, M. & Suciu, D. (2004). Processing xml streams with deterministic automata and stream indexes, *ACM Trans. Database Syst.* 29: 752–788.

Hampson, S., Kibler, D. & Baldi, P. (2002). Distribution patterns of over-represented k-mers in non-coding yeast DNA, *Bioinformatics* 18(4): 513–528.

Hopcroft, J. E., Motwani, R. & Ullman, J. D. (2001). *Introduction the automata theory, languages, and computation, 2d ed.*, ACM Press, New York.

Karlin, S., Burge, C. & Campbell, A. (1992). Statistical analyses of counts and distributions of restriction sites in DNA sequences, *Nucl. Acids. Res.* 20(6): 1363–1370.

Kleffe, J. & Borodovski, M. (1997). First and second moment of counts of words in random texts generated by markov chains, *Bioinformatics* 8(5): 433–441.

Leonardo Marino-Ramírez, John L. Spouge, G. C. K. & Landsman, D. (2004). Statistical analysis of over-represented words in human promoter sequences, *Nuc. Acids Res.* 32(3): 949–958.

Liddle, A. R. (2007). Information criteria for astrophysical model selection, *Monthly Notices of the Royal Astronomical Society: Letters* 377: 74–78.

Lladser, M. E. (2007). Mininal markov chain embeddings of pattern problems, *Information Theory and Applications Workshop*, pp. 251–255.

Lothaire, M. (ed.) (2005). *Applied Combinatorics on Words*, Cambridge University Press, Cambridge.

Nicodème, P., Salvy, B. & Flajolet, P. (2002). Motif statistics, *Theoretical Com. Sci.* 287(2): 593–617.

Nuel, G. (2004). Ld-spatt: Large deviations statistics for patterns on markov chains, *J. Comp. Biol.* 11(6): 1023–1033.

Nuel, G. (2006a). Effective p-value computations using Finite Markov Chain Imbedding (FMCI): application to local score and to pattern statistics, *Algorithms for Molecular Biology* 1(1): 5.

Nuel, G. (2006b). Numerical solutions for patterns statistics on markov chains, *Stat. App. in Genet. and Mol. Biol.* 5(1): 26.

Nuel, G. (2006c). Pattern statistics on markov chains and sensitivity to parameter estimation, *Algorithms for Molecular Biology* 1(1): 17.

Nuel, G. (2008a). Pattern Markov chains: optimal Markov chain embedding through deterministic finite automata, *J. of Applied Prob.* 45(1): 226–243.

Nuel, G. (2008b). Waiting time distribution for pattern occurrence in a constrained sequence: an embedding markov chain approach, *Discrete Mathematics and Theoretical Computer Science* 10: 3.

Nuel, G. (2010). On the first k moments of the random count of a pattern in a multi-states sequence generated by a markov source, *Journal of Applied Probability* 47: 1–19.

Nuel, G. & Dumas, J.-G. (2010). Sparse approaches for the exact distribution of patterns in long multi-states sequences generated by a markov source, *submitted to J. Applied. Prob.* . arXiv:1006.3246v1.

Nuel, G. & Prum, B. (2007). Analyse statistique des séquences biologiques: modélisation markovienne, alignements et motifs, *Hermes editions, Paris*.

Nuel, G., Regad, L., Martin, J. & Camproux, A.-C. (2010). Exact distribution of a pattern in a set of random sequences generated by a markov source: applications to biological data, *Algorithms for Molecular Biology* 5: 15.

Pevzner, P., Borodovski, M. & Mironov, A. (1989). Linguistic of nucleotide sequences: The significance of deviation from mean statistical characteristics and prediction of frequencies of occurrence of words, *J. Biomol. Struct. Dyn.* 6: 1013–1026.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. & Flannery, B. P. (1992). *Numerical Recipes in C*, Cambridge University Press.

Prum, B., Rodolphe, F. & de Turckheim, E. (1995). Finding words with unexpected frequencies in dna sequences, *J. R. Statist. Soc. B* 11: 190–192.

Reignier, M. (2000). A unified approach to word occurrences probabilities, *Discrete Applied Mathematics* 104(1): 259–280.

Reinert, G. & Schbath, S. (1999). Compound poisson and poisson process approximations for occurrences of multiple words in markov chains, *J. of Comp. Biol.* 5: 223–254.

Ribeca, P. & Raineri, E. (2008). Faster exact Markovian probability functions for motif occurrences: a DFA-only approach, *Bioinformatics* 24(24): 2839–2848.

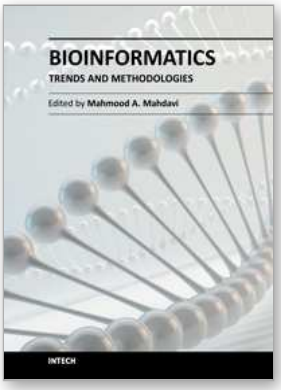
Roberts, R., Vincze, T., Posfai, J. & Macelis, D. (2010). REBASE – a database for dna restriction and modification: enzymes, genes and genomes, *Nucl. Acids Res.* 38: 234–236.

Roquain, E. & Schbath, S. (2007). Improved compound poisson approximation for the number of occurrences of any rare word family in a stationary markov chain, *Adv. in Appl. Probab.* 39(1): 128–140.

Sigrist, C., Cerutti, L., de Castro, E., Langendijk-Genevaux, P., Bulliard, V., Bairoch, A. & Hulo, N. (2010). PROSITE, a protein domain database for functional characterization and annotation, *Nucleic Acids Res.* 38.

Stefanov, V. T. & Szpankowski, W. (2007). Waiting Time Distributions for Pattern Occurrence in a Constrained Sequence, *Discrete Mathematics and Theoretical Computer Science* 9(1): 305–320.

van Helden, J., André, B. & Collado-Vides, J. (1998). Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies, *J. Mol. Biol.* 281(5): 827–842.



Bioinformatics - Trends and Methodologies
Edited by Dr. Mahmood A. Mahdavi

ISBN 978-953-307-282-1
Hard cover, 722 pages
Publisher InTech
Published online 02, November, 2011
Published in print edition November, 2011

Bioinformatics - Trends and Methodologies is a collection of different views on most recent topics and basic concepts in bioinformatics. This book suits young researchers who seek basic fundamentals of bioinformatic skills such as data mining, data integration, sequence analysis and gene expression analysis as well as scientists who are interested in current research in computational biology and bioinformatics including next generation sequencing, transcriptional analysis and drug design. Because of the rapid development of new technologies in molecular biology, new bioinformatic techniques emerge accordingly to keep the pace of in silico development of life science. This book focuses partly on such new techniques and their applications in biomedical science. These techniques maybe useful in identification of some diseases and cellular disorders and narrow down the number of experiments required for medical diagnostic.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Grégory Nuel (2011). Significance Score of Motifs in Biological Sequences, Bioinformatics - Trends and Methodologies, Dr. Mahmood A. Mahdavi (Ed.), ISBN: 978-953-307-282-1, InTech, Available from: http://www.intechopen.com/books/bioinformatics-trends-and-methodologies/significance-score-of-motifs-in-biological-sequences

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the Creative Commons Attribution 3.0 License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.