A p-value computation to assess the significance of huge motif clusters

J. Bourdon[1,2] and S. Carat[1,3] and Mireille Régnier[4]

[1] LINA, CNRS UMR 6241, Universit de Nantes, France
[2] SYMBIOSE-Inria team, INRIA Rennes-Bretagne-Atlantique, France
[3] Institut du thorax, INSERM U 915, Université de Nantes, France
[4] AMIB-Inria team, LIX-Ecole Polytechnique, 91 128 Palaiseau, France
mireille.regnier@inria.fr

Abstract. In computational biology, extracting a subset of sequences that possesses a common characteristic is a crucial task. Doing it properly implies the use of multiple testing arguments that allow to keep the number of false positive in control. In this paper, we focus on the problem of extracting motif clusters (*i.e.,* a subset of sequences where a given motif (or set of motifs) is present) from a large set of sequences. We provide new results based on a suitable use of generating functions to assess the significance of a (possibly huge) cluster of motifs, occurring in sequences of different lengths and different nucleotide composition. We first present an *exact* polynomial algorithm suitable to study sets containing up to 5000 sequences, and then derive a tight asymptotic expansion that allows to study huge sets of sequences (without limitations on the number of sequences). This latter formula is particularly adapted to study data arising from next generation sequencing methods.

Key words: combinatorics on words, string and combinatorial issues in computational biology and bioinformatics, text algorithms, algorithms on automata and words

1 Introduction

A now classical method to study a set of motifs (that may be given, for example, by a PWM) in a set of sequences defines a score function for these motifs and then provides, by some rule of thumb, a threshold in order to select the most significant motifs (10 top motifs, motifs with a pvalue smaller than, say, 5%,...). Such a heuristic approach makes very difficult to control the number of false positive occurring in this selection. We designed a method, relying on an extensive use of generating functions, that allows to evaluate the significance of the selection. Such an information is not available in other methods that aim at correcting pvalues in multiple tests, such as Bonferroni, Benjamin-Hochberg,...(see [Sha95] for a review on multiple testing methods).

Our method is freely available on the web [5]. It applies to assess the significance of composite patterns [EP02] or for a search on both strands [LR04].

In Section 2, one provide an expression of the probability $p_{k,M}$ of selecting a cluster of size k in a set of size M by means of a generating function. Section 3 is devoted to the presentation of an exact algorithm for computing $p_{k,M}$. In Section 4, we present an approximation formula allowing to deal with huge sets of sequences. Finally, Section 5 provides some experimental results.

2 Word counting in a set of short sequences

One assumes below that M random sequences are given. Their lengths may be different and the probability to find one (or k) motifs in sequence i actually depends on this sequence.

Definition 1. *Let N_i be the random variable that counts the number of occurrences in sequence i. One denotes,*

$$X_i = \min(N_i, 1) \ ,$$
$$X = \sum_{i=1}^{M} X_i \ .$$

Remark 1. X is associated to the *sequence number model*.

Definition 2. *For any sequence i, let $p_i(k)$ be the probability to find k $\mathcal{H}$-occurrence in this sequence. One denotes:*

$$\psi_i^{[s]}(t) = E(e^{tS_i^{[s]}}) = \sum_{k=0}^{s} p_i(k) e^{tk} \ ,$$
$$\psi^{[s]}(t) = \prod_{i=1}^{M} \psi_i^{[s]}(t) \ .$$

In the particular case where $s = 1$, one denotes:

$$\phi_i(t) = E(e^{tX_i}) = \psi_i^{[1]}(t) \ ,$$
$$\phi(t) = \prod_{i=1}^{M} \psi_i^{[1]}(t) \ .$$

Proposition 1. *The probability generating functions $(\phi_i(t))_{1 \leq i \leq M}$ and $\phi(t)$ satisfy:*

$$\phi_i(t) = p_i(e^t - 1) + 1 \text{ for } 1 \leq i \leq M \ ,$$
$$\phi(t) = \prod_{i=1}^{M} [p_i(e^t - 1) + 1] \ .$$

[5] `http://www.lina.sciences.univ-nantes.fr/bioatlanstic/MPV/`

Proposition 2. *Probability generating function $\phi(t)$ can be computed from the generating functions $(\phi_i(t))_{1\leq i\leq M}$ in $O(M^2)$ time and $O(M)$ space.*

Probability generating function $\psi^{[s]}(t)$ can be computed from $(\psi_i^{[s]}(t))_{1\leq i\leq M}$ in $O(s^2M)$ time and $O(sM)$ space.

Proof. $\phi(t)$ can be viewed as a product of M polynomials of degree 1. At step i, a polynomial of size i is created and multiplied by a polynomial with 2 elements. Therefore, the cost is $O(i)$. Summing over i yields $O(M^2)$. The space complexity is given by the size of polynoms to be maintained, upper bounded by M.

$\psi^{[s]}(t)$ can be viewed as a product of M polynomials of degree s. The cost of every multiplication is s^2, or $s^{\frac{\log 3}{\log 2}}$ if one uses Karatsuba's algorithm.

3 Exact computation algorithm

In this section, we present a polynomial algorithm that computes the probability that a cluster of k components is extracted from M components. The core of the algorithm consists in computing the coefficients of polynomial $\mu(u)$ defined by the change of variable $u = e^t$ in $\phi(t)$. One has

$$\mu(u) = \prod_{i=1}^{M}(p_i u + 1 - p_i).$$

In the sequel, $[u^k]P(u)$ denotes the coefficient of u^k in polynom P. Then we consider $\mu_n(u) = \prod_{i=1}^{M}(p_i u + 1 - p_i)$, which obviously satisfies the recurrence $\phi_0(u) = 1$, and, for $0 < n \leq M$, $\mu_n(u) = (p_n u + 1 - p_n)\mu_{n-1}(u)$. We set $\mu(u) = \mu_M(u)$. This steadily leads to a dynamic programming algorithm to compute the coefficients is trivial. Indeed, one has

- $[u^0]\mu_n(u) = (1 - p_n)[u^0]\mu_{n-1}(u)$ for all $0 < n \leq M$;
- $[u^k]\mu_n(u) = (1 - p_n)[u^k]\mu_{n-1}(u) + p_n[u^{k-1}]\mu_{n-1}(u)$ for all $0 < n \leq M$ and all $0 < k \leq n$.

The algorithm involves two imbricated loops for indices n and k. Its complexity is thus of order $O(M^2 \times \text{cost operation})$

One of the major problem when one implements such computations concerns the precision needed to establish the results (in terms of number of decimals). Here we use a GNU Multiprecision[6] library in conjunction with a Multi-Precision Floating-Point[7] library. It is necessary to have a lower bound for the number of digits needed to get precise results. For that, we just notice the smallest coefficient is either $[u^0]\mu(u)$ or $[u^M]\mu(u)$. There exists a close formula for these two coefficients. One has,

$$[u^0]\mu(u) = \prod_{i=1}^{M}(1 - p_i) \quad \text{and } [u^M]\mu(u) = \prod_{i=1}^{M} p_i.$$

[6] `http://gmplib.org`
[7] `http://www.mpfr.org`

Consequently, the number of digits needed to perform a complete and precise computation of all the coefficients of $\phi(u)$ with D significant digits for the base number in a scientific notation is $D+\max\{-\sum_{i=1}^{M}\log_{10}(1-p_i), -\sum_{i=1}^{M}\log_{10} p_i\}$. This number of digits is thus in $O(M)$. The precision obviously impacts the theoretical complexity of the algorithm. First notice that with GMP, multiplications of two b-bits multi-precision numbers are performed using Karatsuba's algorithm that has a $O(b^{\frac{\log 3}{\log 2}})$ complexity. Nevertheless, operations are evaluated only when the results are about to be displayed, conserving only a $O(b)$ step for storing efficiently the operands. In our case, this reduces the entire complexity of the main loop $O(M^2 b)$ by increasing the complexity of the display step whose complexity is of order $O(Mb^{\frac{\log 3}{\log 2}})$. Here, b being $O(M)$, the global complexity is $O(M^3)$.

The method has been implemented using C++ language. Several versions as well as an online version is freely available at `http://www.lina.sciences.univ-nantes.fr/bioatlanstic/MPV/`.

Due to its precision, and therefore its complexity, our program cannot deal with more than 5000 sequences approximately. This is sufficient for numbers of applications. Nevertheless, next generation sequencing methods now produce huge sets of sequences (bigger than 5000 sequences) that need to be analyzed. Next, we present a tight approximation formula that makes an intensive use of large deviation theory.

Figure 1 depicts an example of result obtained by using MPV on a set of 800 p-values.

4 Large deviations for short sequences

Our aim here is to provide an approximate computation that relies on large deviations [DZ98]. It is noticeable that, so far, the only available results on large deviations consider either long sequences or short sequences with the same length and distribution [Wat95]. We drop here these two conditions.

4.1 Large deviations formulae

The expectation and variance of the number of motifs occurrences in a set of short sequences steadily follow from Proposition 1 above.

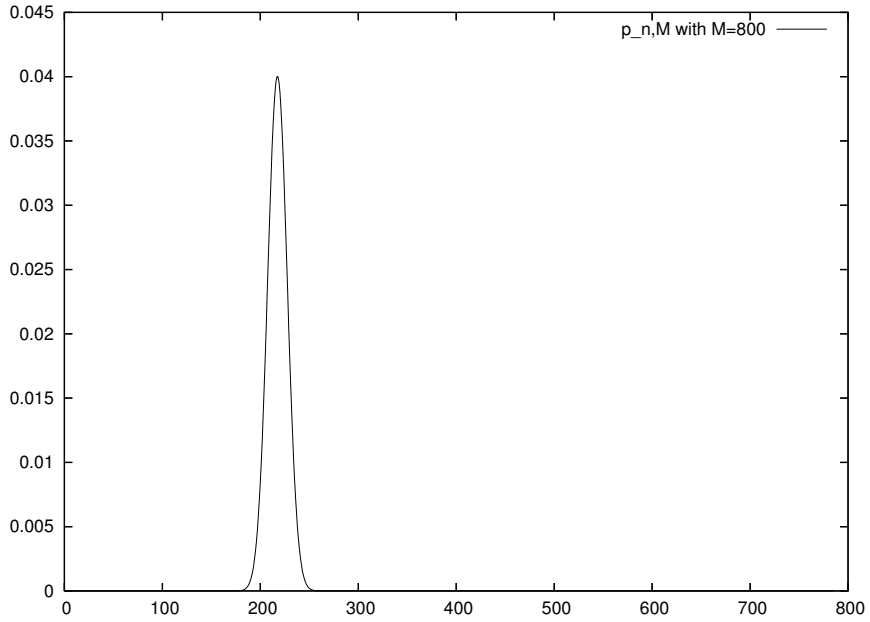
Definition 3. *Let p_i denote the probability $p_i^{(1)}$, for sequence i. One denotes, for each integer j,*

$$\sigma_j = \sum_{i=1}^{i=M} p_i^j \ . \tag{1}$$

Remark 2. It follows from generating functions properties that

$$E(X) = \sigma_1 \ ,$$
$$V(X) = \sigma_1 - \sigma_2 \ .$$

Fig. 1. An example on $p_{k,M}$ computations for $M = 800$ shows a bell shape corresponding to an asymptotic gaussian approximation.



Our aim is the derivation of the probability to have at least k positive sequences out of n, where $a = \frac{k}{n}$ is (significantly) greater than the expectation σ_1.

Definition 4. *Given a real value a, $0 < a < 1$, let $h_a(t)$ be the function*

$$h_a(t) = \log(E(e^{t(X-a)})) = \frac{1}{n} \log \phi(t) - at \ . \tag{2}$$

Let t_a be the root of smallest modulus of Equation $h'_a(t) = 0$,

$$1 - a = \frac{1}{n} \sum_{i=1}^{n} \frac{1 - p_i}{p_i(e^t - 1) + 1} \ . \tag{3}$$

Following the method detailed in [RD04], one searches for the roots of smallest modulus of Equation $h'_a(t) = 0$, in order to use the saddle point method [FS09]. We get the following approximate expression for it.

Proposition 3. *The root of smallest modulus of Equation (3) is* real. *It is called the* fundamental *root. It can be approximated by $\tilde{t}_a$ where*

$$\tilde{t}_a = \log[1 + \frac{a - \sigma_1}{\sigma_1 - \sigma_2} + \frac{(a - \sigma_1)^2(\sigma_2 - \sigma_3)}{(\sigma_1 - \sigma_2)^3}] \ . \tag{4}$$

Proof. A Taylor expansion of (3) yields

$$\begin{aligned} n(a-1) &= -\sum_i (1-p_i)[1+\sum_{j\geq 1}(-1)^j p_i^j (e^t-1)^j] \\ &= -n(1-\sigma_1) + n\sum_j (-1)^{j+1}(\sigma_j - \sigma_{j+1})(e^t-1)^j \end{aligned}$$

Under the two conditions that $p_i(e^t-1)$ is small (p_i is small) and that $|(\sigma_j - \sigma_{j+1})(e^t-1)|$ are upper bounded by some number smaller than 1, - a condition to be checked in the computations-, we get to Equation below where X stands for e^t-1.

$$0 = (\sigma_1 - a) + (\sigma_1 - \sigma_2)X - (\sigma_2 - \sigma_3)X^2 \ .$$

One choses among the two solutions the one that satisfies $t_a = 0$ when $a = p$ yields (4).

Theorem 1. *The distribution of word occurrences in a set of sequences satisfies the large deviation property*

$$-\lim_{n\to\infty} \frac{1}{n} \log(Prob(X \geq na) = I(a) \tag{5}$$

where

$$I(a) = at_a + \sum_{j=1}^{\infty} (-1)^j \sigma_j (e^{t_a}-1)^j \ , \tag{6}$$

that can be approximated as

$$a\tilde{t_a} - (a-\sigma_1)[1 - \frac{\sigma_2(\sigma_1 - a)}{\sigma_1^2}] \ . \tag{7}$$

Proof. Parameter $I(a)$ is called the *large deviation rate* [DZ98]. It is given [RD04] as

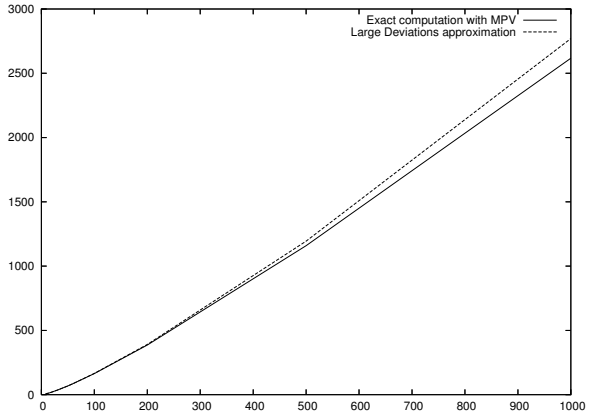
$$I(a) = -h_a(t_a) \ .$$

Expanding $\phi(t)$ as an analytic series steadily yields (6). Substituting the approximate value $\tilde{t}_a$ for t_a in (4) yields (7).

4.2 Experiments and computational precision

We provide here some evidences allowing to compare the results obtained by MPV and by the large deviation approximation. Our goal is to assess the tightness of our approximation, in the domain where MPV can be used in order to justify large deviation use in the domains that are beyond the scope of MPV.

We used a set of 3000 sequences, The probability to find the motifs ranged between 0.0009 and 0.0000001027. The mean was 0.8947805412 and the variance 0.8942941533. We display the value $nI(a)$ that is the logarithm of the pvalue (multiplied by -1).



k	8	10	12	15	20	30	50	60	100	200	500	1000
LD	7	9	11	14	21	35	68	86	164	386	1159	2617
MPV	6	8	10	14	20	35	68	86	165	391	1194	2770

Table 1. Large deviations (LD) versus MPV: log. p-value for different number k of occurences

We used for $\tilde{t}_a$ the *first order* approximations of (4) and

$$\tilde{t}_a = \log[1 + \frac{a - \sigma_1}{\sigma_1 - \sigma_2}] \ . \tag{8}$$

We used for and $I(a)$ the approximation defined in (5).

For small values, a little difference arises. It is due to the fact that, using (only) $I(a)$ we neglected a second order term in the development of the p-value. A drift occurs for k occurrences when k is larger than 100. This is due to the approximation done on $\tilde{t}_a$ and $I(a)$. The second order term for these values -that dominate $-\log(pvalue)$ in this range- increases when a increases.

5 Application to a classical pattern matching problem

One classical problem in pattern matching theory is to decide whether a motif occurs in a sequence or not. The usual way for taking such a decision consists in computing a p-value for the best score obtained and to compare this p-value to an arbitrary threshold. This threshold corresponds to the expected frequency of false positives involved by this choice. When using such a method for sets of sequences, all problems relative to multiple testing appears. Some possible strategies an are

1. keep the sequences with the k best p-values;
2. use p-values adjustments (Bonferroni, Benjamini-Hochberg,...) and keep all the sequences whose adjusted p-values are below a given threshold. Such adjustments ensure that the expected frequency of false positive is below this threshold.

With these methods, one cannot have access to a precise approximation of the number of false positive. In this paper, we provide such an approximation and suggest a new method for selecting the sequences containing an occurrence of the motif when imposing a fixed frequency of false positive (as in the case of a single sequence). We then compare our selection method to the classical ones.

As a benchmark of our method, we use data arising from a ChIP-chip experiment on estrogen receptor ESR1 [CMS$^+$06]. Here, one possesses a large set of promotor sequences (approximately 10000 sequences) that have been annoted as "positive in ChIP" (*i.e.,* ESR1 -or a complex composed by ESR1- binds to the promotor of the associated genes). Such experiments entail a large number of false positives and false negatives. The question is how to select properly the sequences for which one is sure that ESR1 binds, up to a fixed false positive frequency. The binding site of ESR1 has been described and it is now included in some public motifs databases (motif MA00112 in JASPAR database [SAE$^+$04]). For this, we have computed the maximal p-value of ESR1 motif in all the sequences. The expected number of false positive equals $f_K := \sum_{k=0}^{K} p_{k,M}$. Finally, we use our method for determining the smallest number K such that $f_K > 0.99$. Here, $p_{k,M}$ is the probability of selecting a subset of k sequences from a set of M sequences. As a consequence, the subset composed by the sequences with the K smallest p-values contains an occurrence of ESR1 with an expected frequency of false positive equal to 1%. In our application, one has $M = 10054$, $K = 3791$. This has to be compared to the classical selection methods. First, one can compute the real expected frequency of false positive when the k ($k = 10, 100, 1000$) best p-values are chosen. Here, $f_1 0 = 2.10^{-808}$, $f_1 00 = 9.10^{-655}$ and $f_1 000 = 1.10^{-73}$. Second, one can apply a p-value adjustment method. Here, Bonferroni method does not apply. Indeed, each p-value p_i is replaced by $\min(Mp_i, 1)$. Due to a too large number M of sequences, all the p-values are greater than 1 (and than 1%) resulting on the selection of no sequences. Benjamini-Hochberg adjustment applies. It ensures that selecting the best 692 sequences implies an expected frequency of false positive smaller than 1%. Indeed, one has $f_{692} = 3.10^{-185} < 1\%$!

6 Conclusion and Future Work

In this paper, we describe a novel method for selecting a subset of sequences containing a motif when a p-value for the maximal score in the sequence is given. This results in a more precise control on the number of false positive in the selected set. We also provide an algorithm that computes an exact significance of the selected set together with an approximation formula, based on large deviation results, adapted to the study of larger sets of sequences. The classical way of using our method consists in performing an exact computation when it is possible and then use an approximation formula in the other cases.

Notice that our selection method also applies to a large number of other areas as soon as one has to decide between two states (positive or not, present

or absent,...) and when a p-value for taking such a decision is given (result of a statistical test, occurrence of a motif,...).

Finally, our generating function modeling of the problem can certainly be adapted to study some decision problems with more than two possible states. This will allow to study for instance co-occurrences pattern problems. There, one possesses several p-values per sequence (one for each pattern). The question is how can we extract, with a good control over the number of false positive, a significant cluster of sequences that possesses an occurrence of (most of) all patterns.

References

CMS+06. J. S. Carroll, C. A. Meyer, J. Song, W. Li, T. R. Geistlinger, J. Eeckhoute, A. S. Brodsky, E. K. Keeton, K. C. Fertuck, G. F. Hall, Q. Wang, S. Bekiranov, V. Sementchenko, E. A. Fox, P. A. Silver, T. R. Gingeras, X. S. Liu, and M. Brown. Genome-wide analysis of estrogen receptor binding sites. *Nat. Genet.*, 38:1289–1297, Nov 2006.

DZ98. Amir Dembo and Ofer Zeitouni. *Large deviations techniques and applications*. Springer, New York, 2nd ed. edition, 1998.

EP02. Eleazar Eskin and Pavel A Pevzner. Finding composite regulatory patterns in DNA sequences. *Bioinformatics (Oxford, England)*, 18 Suppl 1:S354–363, 2002. PMID: 12169566.

FS09. Philippe Flajolet and Robert Sedgewick. *Analysis of Algorithms*. Cambridge University Press, 2009.

LR04. M. Lescot and M. Régnier. Motif statistics on plants datasets. *Biophysics*, 48(1):16, 2004. Proc. Moscow Conference on Computational Molecular Biology, MCCMB'03.

RD04. Mireille Régnier and Alain Denise. Rare events and conditional events on random strings. *Discrete Mathematics and Theoretical Computer Science*, 6(2):191–214, 2004.

SAE+04. A. Sandelin, W. Alkema, P. Engström, W. W. Wasserman, and B. Lenhard. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res.*, 32:D91–94, Jan 2004.

Sha95. J P Shaffer. Multiple hypothesis testing. *Annual Review of Psychology*, 46(1):561–584, 1995.

Wat95. M. Waterman. *Introduction to Computational Biology*. Chapman and Hall, London, 1995.