J. Appl. Prob. **47,** 1105–1123 (2010)
Printed in England
© *Applied Probability Trust* 2010

ON THE FIRST k MOMENTS OF THE RANDOM COUNT OF A PATTERN IN A MULTISTATE SEQUENCE GENERATED BY A MARKOV SOURCE

G. NUEL,* *Paris Descartes University*

Abstract

In this paper we develop an explicit formula that allows us to compute the first k moments of the random count of a pattern in a multistate sequence generated by a Markov source. We derive efficient algorithms that allow us to deal with any pattern (low or high complexity) in any Markov model (homogeneous or not). We then apply these results to the distribution of DNA patterns in genomic sequences, and we show that moment-based developments (namely Edgeworth's expansion and Gram–Charlier type-B series) allow us to improve the reliability of common asymptotic approximations, such as Gaussian or Poisson approximations.

Keywords: Optimal Markov chain embedding; deterministic finite automaton; moment generating function; Edgeworth's expansion; Gram–Charlier series

2010 Mathematics Subject Classification: Primary 60J10
Secondary 62E15; 62E17

1. Introduction

The distribution of pattern counts in a random sequence generated by a Markov source has many applications in a wide range of fields, including reliability, insurance, communication systems, pattern matching, and bioinformatics. In the latter field, a common application is the statistical detection of patterns of interest in biological sequences such as DNA or proteins. Such approaches have successfully led to the confirmation of known biological signals (PROSITE signatures, CHI motifs, etc.) as well as the identification of new functional patterns (regulatory motifs in upstream regions, binding sites, etc.); see, e.g. [3], [8], [13], [15], [19], [20], [24], and [37].

From a statistical point of view, studying the distribution of the random count of a pattern (simple or complex) in a multistate Markov chain is a difficult problem. A great deal of effort has been spent on this problem in the last fifty years with many concurrent approaches and we give here only a few references (see [23, Chapter 6], [28], and [32], for more comprehensive reviews). Exact methods are based on a wide range of techniques, such as Markov chain embedding, moment generating functions, combinatorial methods, and exponential families [1], [6], [7], [9], [16], [27], [35], [36]. There is also a wide range of asymptotic approximations, the most popular among them being Gaussian approximations [10], [21], [30], [31], Poisson approximations [14], [17], [18], [33], and large deviations approximations [12], [26].

More recently, the connection between this problem and pattern matching theory has been pointed out by several authors [11], [22], [25], [29], [34]. Thanks to these works, it is now possible to obtain an optimal Markov chain embedding of any pattern problem through minimal

Received 21 September 2009; revision received 31 May 2010.

* Postal address: MAP5, Department of Applied Mathematics, CNRS 8145, Paris Descartes University, 49 rue des Saints-Pères, F-75006 Paris, France. Email address: gregory.nuel@parisdescartes.fr

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

deterministic finite automata (DFAs). In this paper we apply this technique to the exact computation of the first k moments of a pattern count in a random sequence generated by a Markov source. Our aim is to provide efficient algorithms to perform these computations both for low and high complexity patterns in either homogeneous or heterogeneous Markov models.

The paper is organized as follows. In a first part, we recall the principles of optimal Markov chain embedding through DFAs. We then derive from the moment generating function of the random pattern count a new expression for its first k moments, and introduce three different algorithms to compute it. The relative complexity of these algorithms with respect to previous approaches are then discussed. Finally, we apply Edgeworth's expansion and Gram–Charlier type-B series techniques to obtain near Gaussian or near Poisson approximations, and show how this allows us to improve the reliability of classical asymptotic approximations with a modest additional cost.

2. DFAs and optimal Markov chain embedding

2.1. Sequence model

Let $(X_i)_{1\leq i\leq \ell}$ be an order-($d \geq 0$) Markov chain over the cardinal $s \geq 2$ alphabet $\mathcal{A}$. For all $1 \leq i \leq j \leq \ell$, we denote by $X_i^j := X_i \cdots X_j$ the subsequence between positions i and j. For all $a_1^d := a_1 \cdots a_d \in \mathcal{A}^d$, $b \in \mathcal{A}$ and $1 \leq i \leq \ell - d$, let us denote by $\nu(a_1^d) := \mathrm{P}(X_1^d = a_1^d)$ the starting distribution and by $\pi_{i+d}(a_1^d, b) := \mathrm{P}(X_{i+d} = b \mid X_i^{i+d-1} = a_1^d)$ the transition probability towards X_{i+d}.

2.2. Pattern count

Let $\mathcal{W}$ be a finite set of words (for the purpose of simplification, we assume that $\mathcal{W}$ contains no word of length smaller or equal to d) over $\mathcal{A}$. We consider the random number N of matching positions of $\mathcal{W}$ in X_1^ℓ defined by

$$N := \sum_{i=1}^{\ell} 1_{\{X_1^i \in \mathcal{A}^* \mathcal{W}\}}, \tag{1}$$

where $\mathcal{A}^*\mathcal{W}$ is the set of all finite sequences over $\mathcal{A}$ ending with one element of $\mathcal{W}$ and 1_A is the indicator function of the event A.

2.3. Pattern cardinality

Let us define the *pattern cardinality* R to be the cardinal of the finite set $\mathcal{W}$, i.e. $R = |\mathcal{W}|$. In the simple widespread approach whereby we count all the elements of $\mathcal{W}$, the simple task of obtaining the number of pattern occurrences in a sequence results in a linear complexity with R. For more complex tasks, such as computing variance or moment generating functions, we often get complexities in $O(R^2)$ or $O(R^3)$. For patterns of modest cardinality (e.g. $R < 50$), this could be an acceptable cost. However, the computation cost quickly becomes unbearable when considering more complex patterns (e.g. $R > 50$, $R > 1000$, or more).

Fortunately, this problem can be largely reduced by exploiting classical results from pattern matching theory. The idea consists in embedding the whole set $\mathcal{W}$ into a special graph called a deterministic finite automaton (DFA), whose number of states L is usually much smaller than R. This transformation and how we can take advantage of it for probabilistic computations are recalled in the following three subsections.

2.4. Deterministic finite automata

As suggested in [11], [22], [25], and [29], we want to perform an optimal Markov chain embedding of the problem through a DFA. Here we use the notation of [29].

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

Let $(\mathcal{A}, \mathcal{Q}, \sigma, \mathcal{F}, \delta)$ be a DFA, where $\mathcal{A}$ is our finite alphabet, $\mathcal{Q}$ is a finite state space, $\sigma \in \mathcal{Q}$ is the starting state, $\mathcal{F} \subset \mathcal{Q}$ is the subset of final states, and $\delta \colon \mathcal{Q} \times \mathcal{A} \to \mathcal{Q}$ is the transition function. We recursively extend the definition of δ over $\mathcal{Q} \times \mathcal{A}^*$ thanks to the relation $\delta(p, aw) := \delta(\delta(p, a), w)$ for all $p \in \mathcal{Q}$, $a \in \mathcal{A}$, and $w \in \mathcal{A}^*$.

We assume the following two properties.

(P1) The DFA recognizes the language $\mathcal{A}^*\mathcal{W}$ (i.e. the set of all finite sequences over $\mathcal{A}$ ending with one element of $\mathcal{W}$). This means that $x \in \mathcal{A}^*\mathcal{W}$ is equivalent to $\delta(\sigma, x) \in \mathcal{F}$.

(P2) The DFA is non-d-ambiguous (a DFA having this property is also called a dth order DFA in [22]), which means that, for all $q \in \mathcal{Q}$, $\delta^{-d}(q) := \{a_1^d \in \mathcal{A}_1^d$, there exists a $p \in \mathcal{Q}$, $\delta(p, a_1^d) = q\}$ is either a singleton, or the empty set. For the sake of simplicity, we denote by $\delta^{-d}(q)$ its unique element in the singleton case.

Thanks to classical results from the theory of languages and automata, it can be proved that the set of DFAs having properties (P1) and (P2) is not empty and, hence, it is possible to consider a *minimal* DFA in the sense that it achieves properties (P1) and (P2) with the smallest possible cardinal $L = |\mathcal{Q}|$. From now on, we assume that we have built such a minimal DFA.

2.5. Markov chain embedding

Theorem 1. *We consider the random sequence over $\mathcal{Q}$ defined by $\widetilde{X}_0 := \sigma$ and $\widetilde{X}_i := \delta(\widetilde{X}_{i-1}, X_i)$ for all $1 \le i \le \ell$. Then $(\widetilde{X}_i)_{i \ge d}$ is a heterogeneous order-1 Markov chain over $\mathcal{Q}' := \delta(s, \mathcal{A}^d\mathcal{A}^*)$ such that, for all $p, q \in \mathcal{Q}'$ and $1 \le i \le \ell - d$, the starting distribution $\mu_d(p) := \mathrm{P}(\widetilde{X}_d = p)$ and the transition matrix $T_{i+d}(p, q) := \mathrm{P}(\widetilde{X}_{i+d} = q \mid \widetilde{X}_{i+d-1} = p)$ are given by*

$$\mu_d(p) = \begin{cases} \nu(\delta^{-d}(p)) & \textit{if } \delta^{-d}(p) \neq \varnothing, \\ 0 & \textit{otherwise,} \end{cases}$$

$$T_{i+d}(p, q) = \begin{cases} \pi_{i+d}(\delta^{-d}(p), b) & \textit{if there exists a } b \in \mathcal{A} \textit{ such that } \delta(p, b) = q, \\ 0 & \textit{otherwise.} \end{cases}$$

In addition, we have

$$X_1 \cdots X_i \in \mathcal{A}^*\mathcal{W} \iff \widetilde{X}_i \in \mathcal{F} \quad \textit{for all } 1 \le i \le \ell \tag{2}$$

and

$$N = \sum_{i=1}^{\ell} 1_{\{X_1^i \in \mathcal{A}^*\mathcal{W}\}} = \sum_{i=1}^{\ell} 1_{\{\widetilde{X}_i \in \mathcal{F}\}}. \tag{3}$$

Proof. Thanks to (P2), it is clear that $(\widetilde{X}_i)_{i \ge d}$ is an order-1 Markov chain whose starting distribution and transition matrix are easy to obtain. Equation (2) is a direct consequence of (P1), and (3) naturally follows thanks to definition (1). See [22] or [29] for more details.

2.6. Moment generating function

Corollary 1. *The moment generating function $f(y)$ of N is given by*

$$f(y) := \sum_{n=0}^{+\infty} \mathrm{P}(N = n) y^n = \mu_d \biggl(\prod_{i=1}^{\ell-d} (P_{i+d} + y Q_{i+d}) \biggr) \mathbf{1}, \tag{4}$$

where $\mathbf{1}$ is a column vector of 1s (in the same manner, we denote by $\mathbf{0}$ the column vector of 0s)

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

and, for all $1 \le i \le \ell - d$, $T_{i+d} = P_{i+d} + Q_{i+d}$ *with* $P_{i+d}(p, q) := 1_{\{q \notin \mathcal{F}\}} T_{i+d}(p, q)$ *and* $Q_{i+d}(p, q) := 1_{\{q \in \mathcal{F}\}} T_{i+d}(p, q)$ *for all* $p, q \in \mathcal{Q}'$.

Proof. Since Q_{i+d} contains all counting transitions, we keep track of the number of occurrences by associating a dummy variable y to these transitions. We hence just have to compute the marginal distribution at the end of the sequence and sum up the contribution of each state. See [11], [22], [25], and [29] for more details.

Corollary 2. *In the particular case where* $(X_i)_{1\le i\le\ell}$ *is a homogeneous Markov chain we can drop the indices in* P_{i+d} *and* Q_{i+d}*; hence, (4) simplifies to*

$$f(y) = \mu_d (P + yQ)^{\ell-d} \mathbf{1}. \tag{5}$$

Corollary 2 can be found explicitly in [22] or [34], but its (although straightforward) generalization to the heterogeneous model (Corollary 1) appears to be a new result.

3. Main result

Lemma 1. *For all* $k \ge 0$*, we have*

$$f^{(k)}(y) = k!\, \mu_d \bigg(\sum_{1\le i_1<\cdots<i_k\le\ell-d} \prod_i A_{i,\{i_1,\ldots,i_k\}}(y) \bigg) \mathbf{1}, \tag{6}$$

where, for all $I \subset \mathbb{N}$, $A_{i,I}(y) = P_{i+d} + yQ_{i+d}$ *if* $i \notin I$ *and* $A_{i,I}(y) = Q_{i+d}$ *if* $i \in I$.

Proof. The lemma is obvious for $k = 0$. We now assume that the lemma is true for fixed rank k. When differentiating (6), the key is to see that, for all $I \subset \mathbb{N}$, $(\prod_i A_{i,I}(y))' = \sum_{j\notin I} \prod_i A_{i,I\cup\{j\}}(y)$. For each configuration $I = \{i_1, \ldots, i_{k+1}\}$, it is therefore obvious that $A_{i,I}(y)$ appears in $A'_{i,I\setminus\{j\}}$ for all $j \in I$. This explains the $k+1$ factor which combines with $k!$ to establish the lemma for rank $k + 1$.

Theorem 2. *For all* $k \ge 0$*, we have*

$$\mathrm{E}\bigg(\frac{N!}{(N-k)!}\bigg) = k!\, [g(y)]_{y^k} \quad \textit{with} \quad g(y) = \mu_d \bigg(\prod_{i=1}^{\ell-d} (T_{i+d} + yQ_{i+d}) \bigg) \mathbf{1}, \tag{7}$$

where $[g(y)]_{y^k}$ *denotes the coefficient of degree* k *in* $g(y)$.

Proof. By differentiating the moment generating function f k times we easily obtain

$$\mathrm{E}\bigg(\frac{N!}{(N-k)!}\bigg) = F^{(k)}(1).$$

Expanding the expression of $g(y)$ to degree k then allows us to identify the correct term in (6) for $y = 1$, thus proving the theorem.

Corollary 3. *In the particular case where* $(X_i)_{1\le i\le\ell}$ *is a homogeneous Markov chain, (7) simplifies to*

$$\mathrm{E}\bigg(\frac{N!}{(N-k)!}\bigg) = k!\, [g(y)]_{y^k} \quad \textit{with} \quad g(y) = \mu_d (T + yQ)^{\ell-d} \mathbf{1}. \tag{8}$$

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

4. Three algorithms

4.1. Full recursion

For all $1 \leq i \leq \ell - d$, we consider the column polynomial vector defined by

$$E_i(y) := \left(\prod_{j=i}^{\ell-d} (T_{j+d} + yQ_{j+d})\right)\mathbf{1}.$$

If we denote by $E_k(i) := [E_i(y)]_{y^k}$ its coefficient of degree k for all $k \geq 0$ then it is clear that we can rewrite the expression of $g(y)$ in (7) as $[g(y)]_{y^k} = \mu_d E_k(1)$.

Proposition 1. *We have the following results for all* $1 \leq i \leq \ell - d$*:*

(i) $E_0(i) = \mathbf{1}$*;*

(ii) $E_1(\ell - d) = Q_\ell \mathbf{1}$*;*

(iii) *if* $k \geq 1$ *and* $(\ell - d - i + 1) < k$*, then* $E_k(i) = \mathbf{0}$*;*

(iv) *if* $k \geq 1$ *and* $i < \ell - d$*, then* $E_k(i) = T_{i+d}E_k(i+1) + Q_{i+d}E_{k-1}(i+1)$.

Proof. (i) It is clear that $E_0(i) = (\prod_{j=1}^{\ell-d} T_{j+d})\mathbf{1}$, which is equal to $\mathbf{1}$ since all the T_{j+d} are stochastic matrices. Part (ii) is immediate. For part (iii), the product must contain at least k terms to have a degree k contribution. Part (iv) is easily proved by recurrence using the fact that $E_i(y) = (T_{i+d} + yQ_{i+d})E_{i+1}(y)$.

Algorithm 1. Compute the first k terms of $g(y)$ in the most general case by performing the following full recursion.

Require: the starting distribution μ_d, matrices T_i and Q_i for all $1 \leq i \leq \ell - d$, and an $O(k \times L)$ workspace to keep the current values of $E_j(i)$ for $0 \leq j \leq k$, where L denotes the cardinal of $\mathcal{Q}'$.

Initialization

$E_0(\ell - d) = \mathbf{1}$, $E_1(\ell - d) = Q_\ell \mathbf{1}$, and $E_j(\ell - d) = \mathbf{0}$ for $2 \leq j \leq k$.

Recursion

```
for $i = \ell - d - 1, \ldots, 1$ do
    for $j = k, \ldots, 1$ do
        $E_j(i) = T_{i+d}E_j(i+1) + Q_{i+d}E_{j-1}(i+1)$
    end for
end for
```

Output: for all $0 \leq j \leq k$, $[g(y)]_{y^j} = \mu_d E_j(1)$.

The workspace complexity is $O(k \times L)$, and, since all matrix vector products exploit the sparse structure of the matrices, the time complexity is $O(\ell \times k \times s \times L)$, where $s \times L$ corresponds to the maximum number of nonzero terms in T_{i+d}.

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

4.2. Direct power computation

From now on we consider the particular case where the Markov model is homogeneous. According to (8), the expression of $g(y)$ in such a case then simplifies to $g(y) = \mu_d(T + yQ)^{\ell-d}\mathbf{1}$. If we denote by $M_i(y) := \mathcal{T}_k((T + yQ)^i)$ (where $\mathcal{T}_k$ is the truncature function whose value is 0 for all terms of degree greater than k), our problem is then to compute only $M_{\ell-d}(y)$ since $[g(y)]_{y^j} = [\mu_d M_{\ell-d}(y)\mathbf{1}]_{y^j}$ for all $0 \le j \le k$.

Proposition 2. *We have*

$$M_{\ell-d}(y) = \prod_{j=0}^{J} M_{2^j}(y)^{1_{\{a_j=1\}}}, \tag{9}$$

where $\ell - d = a_0 2^0 + a_1 2^1 + \cdots + a_J 2^J$ with $a_j \in \{0, 1\}$ for $0 \le j \le J := \lfloor \log_2(\ell - d)\rfloor$. (Here, for all $x \in \mathbb{R}$, $\lfloor x \rfloor$ denotes the largest integer smaller than x.)

Proof. The proof is immediate.

Since we need to compute only the terms of degree smaller than k in $M_{\ell-d}(y)$ to obtain the first k moments of N, we can speed up the computation by ignoring terms of degree greater than k in (9). Hence, we obtain Algorithm 2, where $\tau_k[p(y)]$ denotes the truncated polynomial obtained from $p(y)$ by dropping all terms of degree greater than k.

Algorithm 2. Compute the first k terms of $g(y)$ in the particular case of a homogeneous Markov model through the following direct power computation.

```
Require: the starting distribution $\mu_d$, matrices $T$ and $Q$, $\ell$, $d$, an $O(k \times L^2 \times J)$ workspace for $M_{2^j}(y)$ for $0 \le j \le J$, and a polynomial matrix $M(y)$.

Preliminary computations
Perform the binary decomposition $\ell - d = a_0 2^0 + \cdots + a_J 2^J$, $M_{2^0}(y) = (P + yQ)^1$.
for $j = 1, \ldots, J$ do
    $M_{2^j}(y) = \tau_k[M_{2^{j-1}}(y)^2]$
end for

Computing $M_{\ell-d}(y)$
$M(y) = M_0(y)$.
for $j = 0, \ldots, J$ do
    if $a_j = 1$ then $M(y) = \tau_k[M(y) \times M_{2^j}(y)]$
end for

Output: for all $0 \le j \le k$, $[g(y)]_{y^j} = [\mu_d M_{\ell-d}(y)\mathbf{1}]_{y^j}$.
```

The workspace complexity is $O(k \times L^2 \times \log_2 \ell)$ and the time complexity is $O(k^2 \times L^3 \times \log_2 \ell)$ (k^2 for the polynomial products and L^3 for the matrix products).

4.3. Partial recursion

In this subsection we assume that T is an irreducible and aperiodic matrix, and we denote by ν the magnitude of its second largest eigenvalue.

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

For all $i \geq 0$, we consider the polynomial vector $F_i(y) := (T + yQ)^i \mathbf{1}$, and, for all $k \geq 0$, we denote by $F_k(i) := [F_i(y)]_{y^k}$ the term of degree k in $F_i(y)$. By convention, $F_k(i) = \mathbf{0}$ if $i < 0$. It is then possible to rewrite the expression of $g(y)$ in (8) as $[g(y)]_{y^k} = \mu_d F_k(\ell - d)$. Additionally, let us finally recursively define the quantity $D_j^k(i)$ for all $k, i, j \geq 0$ by $D_k^0(i) := F_k(i)$ and, if $i \geq 1$ and $j \geq 1$, $D_k^j(i) := D_k^{j-1}(i) - D_k^{j-1}(i-1)$ so that

$$D_k^j(i) = \sum_{\delta=0}^{j} (-1)^\delta \binom{j}{\delta} F_k(i - \delta). \tag{10}$$

Lemma 2. *We have the following initial conditions:*

(i) *for all $i \geq 0$, $D_0^0(i) = \mathbf{1}$;*

(ii) *for all $j \geq 1$, $D_0^j(i) = (-1)^i \binom{j-1}{i} \mathbf{1}$ if $0 \leq i \leq j-1$, and $D_0^j(i) = \mathbf{0}$ if $i \geq j$;*

(iii) *for all $k \geq 1$, $D_k^0(0) = \mathbf{0}$, and $D_k^0(i) = TD_k^0(i-1) + QD_{k-1}^0(i-1)$ for $i \geq 1$.*

In addition, for all $k, j, i \geq 1$, we have the following recurrence relations:

(iv) $D_k^j(i) = D_k^{j-1}(i) - D_k^{j-1}(i-1)$*;*

(v) $D_k^j(i) = TD_k^j(i-1) + QD_{k-1}^j(i-1)$.

Proof. (i) It is clear that $D_0^0(i) = T^i \mathbf{1} = \mathbf{1}$ since T is a stochastic matrix. Part (ii) is a consequence of (i) and (10). Part (iii) is proved by recurrence. Part (iv) is simply the definition of $D_k^j(i)$. Part (v) is a consequence of (iii) and of the recursive definition of $D_k^j(i)$.

Lemma 2 provides an efficient way to compute all the $D_k^j(i)$ for $0 \leq k, j \leq K$, and $0 \leq i \leq \alpha$ (see Algorithm 3 below). However, these computations suffer numerical instability in floating point algebra. This phenomenon is emprically studied in Subsection 5.3.

Lemma 3. *For all $k \geq 1$, we have*

(i) $D_k^k(i) = \sum_{j=k}^{i} T^{i-j} Q D_{k-1}^k(j-k)$ *for $i \geq k$;*

(ii) *there exists a $\boldsymbol{C}_k \in \mathbb{R}^L$ such that $D_k^k(i) = \boldsymbol{C}_k + O(k\nu^{i/k})$ and $D_k^{k+1}(i) = \mathbf{0} + O(k\nu^{i/k})$ for all $i \geq 2k$ as $i \to \infty$.*

Proof. Part (i) is a direct application of Lemma 2(v). For $k = 1$, part (i) simply gives $D_1^1(i) = T^{i-1}Q\mathbf{1}$, which proves (ii) for $k = 1$. We assume that (ii) is true for some fixed rank k and then decompose $D_{k+1}^{k+1}(i)$ into

$$D_{k+1}^{k+1}(i) = \underbrace{T^{i-\alpha}\left(\sum_{j=k+1}^{\alpha} T^{\alpha-j} Q D_k^{k+1}(j-k-1)\right)}_{A} + \underbrace{\sum_{j=\alpha+1}^{i} T^{i-j} Q D_k^{k+1}(j-k-1)}_{B}$$

for some $\alpha \geq 2k$. Thanks to the stochasticity of T, there exists a $\boldsymbol{C}_{k+1}^\alpha \in \mathbb{R}^L$ such that $A = \boldsymbol{C}_{k+1}^\alpha + O(\nu^{i-\alpha})$, and since (ii) is true at rank k, $B = \sum_{j=\alpha}^{i} O(k\nu^{j/k})$. Elementary analysis proves that

$$\min_\alpha \left\{ \nu^{i-\alpha} + \sum_{j=\alpha}^{i} k\nu^{i'/k} \right\} = O((k+1)\nu^{i/(k+1)}),$$

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

the minimum being obtained for $\alpha = i(k-1)/k$. Part (ii) is then proved at rank $k+1$ with $\boldsymbol{C}_{k+1} = \boldsymbol{C}_{k+1}^{\alpha}$ for that particular α.

Proposition 3. *For all $k \geq 1$, $0 \leq j \leq k$, and any $i \geq \alpha \geq 2k$,*

$$D_k^j(i) = \sum_{j'=0}^{k-j} \binom{i-\alpha}{j'} D_k^{j+j'}(\alpha) + O\left(k \binom{i-\alpha}{k-j} \nu^{\alpha/k}\right) \quad \textit{as } i \to \infty, \tag{11}$$

and in the particular case where $j = 0$ we obtain

$$F_k(i) = F_k(\alpha) + \sum_{j'=1}^{k} \binom{i-\alpha}{j'} D_k^{j'}(\alpha) + O\left(k \binom{i-\alpha}{k} \nu^{\alpha/k}\right) \quad \textit{as } i \to \infty.$$

Proof. A simple application of Lemma 3(ii) proves that $D_k^k(i) = D_k^k(\alpha) + O(\nu^{\alpha/k})$, which is exactly (11) for $j = k$. We then obtain the result for $j < k$ by recurrence and the facts that

$$D_k^j(i) = D_k^j(\alpha) + \sum_{i'=\alpha+1}^{i} D_k^{j+1}(i') \quad \text{and} \quad \sum_{i'=\alpha+1}^{i} \binom{i'-\alpha}{j'} = \binom{i-\alpha}{j'+1}.$$

Algorithm 3. Compute $D_k^j(\alpha)$ for all $0 \leq k, j \leq K$ as follows.

```
Require: the matrices $T$ and $Q$, a value $\alpha \geq K$, and an $O(K^2 \times L)$ workspace
to keep the current value of $D_k^j(i)$ and $D_k^j(i-1)$ for all $0 \leq k, j \leq K$.

for $i = 0, \ldots, \alpha$ do
    Initialization
    $D_0^0(i) = \mathbf{1}$
    for $j = 1, \ldots, K$ do
        $D_0^j(i) = (-1)^i \binom{j-1}{i} \mathbf{1}$ if $0 \leq i \leq j-1$, and $D_0^j(i) = \mathbf{0}$ if $i \geq j$
    end for
    for $k = 1, \ldots, K$ do
        $D_k^0(i) = \mathbf{0}$ if $i = 0$, and $D_k^0(i) = T D_k^0(i-1) + Q D_{k-1}^0(i-1)$ if
        $i \geq 1$
    end for
end for

Recursion
for $k = 1, \ldots, K$ and $j = 1, \ldots, K$ do
    update $D_k^j(i)$ with either

        $D_k^{j-1}(i) - D_k^{j-1}(i-1)$   or   $T D_k^j(i-1) + Q D_{k-1}^j(i-1)$.

end for

Output: $D_k^j(\alpha)$ for all $0 \leq k, j \leq K$.
```

The workspace complexity is $O(K^2 \times L)$ and, since all matrix vector products exploit the sparse structure of the matrices, the time complexity is $O(\alpha \times K^2 \times s \times L)$.

4.4. Comparison with known methods

To the author's knowledge, there is no record of a method that allows us to compute order-k moments of a pattern count in heterogeneous Markov sequences. This work was in fact initially motivated by this observation. In the homogeneous case however, many interesting approaches can be found in the literature. In most cases, these methods are limited to the computation of the first two moments, but several of them can be also used to obtain arbitrary order moments, as with our method.

One of these approaches involves considering the bivariate moment generating function

$$f(y,z) := \sum_{n\geq 0,\,\ell\geq d} \mathrm{P}(N_\ell = n) y^n z^\ell,$$

where N_ℓ is the random number of pattern occurrences in a sequence of length ℓ. Thanks to (5), it is easy to show that

$$f(y,z) = z^d \times \mu_d (I - z(P + yQ))^{-1}\mathbf{1},$$

where I denotes the identity matrix. It is then possible to obtain order-k moments of N_ℓ using the relation

$$\frac{\partial^k f}{\partial y^k}(1,z) = \sum_{\ell\geq d} \mathrm{E}\biggl(\frac{N_\ell!}{(N_\ell - k)!}\biggr) z^\ell.$$

Such interesting approaches have been developed by several authors, including Lladser [22] and Nicodème *et al.* [25]. In order to apply this method, we should first use a computer algebra system (CAS) to perform the bivariate polynomial inversion of the matrix $I - z(P + yQ)$ to obtain $f(y,z)$, thus resulting in a complexity of $O(L^3)$, where L is the number of states in the embedding Markov chain. Hence, we need to compute the order-k partial derivative in y of $f(y,z)$ prior to performing the fast Taylor expansion of the result up to z^ℓ. The resulting complexity is $O(\log_2 \ell \times D^3)$, where D is the degree of the denominator in $\partial^k f/\partial y^k(1,z)$. As in Algorithm 2, we obtain a cubic complexity with L^3 for linear algebra computations, and a logarithmic complexity with ℓ thanks to the binary decomposition. However, this method is much more sophisticated to implement (it requires only simple operations on polynomial matrices while the alternative approach requires a complete computer algebra system) and the D^3 term that appears in the Taylor expansion complexity in fact hides at least a cubic complexity in k which is not easy to handle. Let us note that Nicodème *et al.* [25] also suggested obtaining the asymptotic development of moments by computing only the local behavior of the generating function $f(y,z)$, which allows the computation to be performed in faster floating point arithmetic. However, this approach cannot give the exact moments, only approximations, and we still need to perform the formal inversion of an order-L bivariate polynomial matrix, which is an expensive step.

More recently, Ribeca and Raineri [34] suggested computing the full bulk of the exact distribution of N_ℓ through (5) using a power method similar to that given in Subsection 4.2, with the difference that all polynomial products are performed using fast Fourier transforms (FFTs). The drawback with FFT polynomial products is that the resulting coefficients are known with an absolute precision equal to the largest one times the relative precision of the floating point. As a consequence, the distribution is well computed only in its center part. Fortunately, this is precisely the part of the distribution that matters for moment computations. Using this approach, and a very careful implementation, we can compute the full distribution

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

with a complexity of $O(L^3 \times \log_2 \ell \times n_{\max} \log_2 n_{\max})$, where $n_{\max}$ is the maximum number of pattern occurrences in the sequence. Once again, the resulting complexity is likely to be much higher than that of Algorithm 2 since k^2 is usually far smaller than $n_{\max} \log_2 n_{\max}$. Moreover, Algorithm 2 is again much easier to implement than this sophisticated FFT approach.

Finally, we should note that both these two known approaches involve a complexity of $O(L^3)$ in time (and at least $O(L^2)$ in memory), which makes it difficult or even impossible to use them for moderate or high complexity patterns (e.g. $L = 100$ or $L = 1000$). For such patterns, Algorithm 1 appears to be a safe but slow alternative (linear complexity with sequence length ℓ) and Algorithm 3 seems to be a very promising approach since it allows us to handle such complex patterns while retaining a logarithmic complexity with ℓ as in Algorithm 2. Unfortunately, the numerical instabilities observed in practice with Algorithm 3 need to be investigated further before we can trust this approach.

5. Application to DNA patterns in genomics

5.1. Dataset

We consider an order-($d = 1$) homogeneous Markov model over $\mathcal{A} = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$, whose transition matrix estimated over the complete genome of the bacteria *Escherichia coli* is given by

$$\pi = \begin{pmatrix} 0.30 & 0.21 & 0.22 & 0.27 \\ 0.23 & 0.23 & 0.33 & 0.22 \\ 0.28 & 0.29 & 0.23 & 0.20 \\ 0.19 & 0.28 & 0.23 & 0.30 \end{pmatrix}.$$

We consider a sequence $X = X_1 \cdots X_\ell$ of length $\ell = 400\,000$, starting with $X_1 = \mathtt{A}$.

5.2. Some moments

In this subsection we compute the first $k = 4$ moments of several DNA patterns. We then use these moments to compute the expectation $m = m_1$, the standard deviation $\sigma = \sqrt{m_2}$, the skewness $\gamma_1 = m_3/m_2^{3/2}$, and the excess kurtosis $\gamma_2 = m_4/m_2^2 - 3$, where $m_i := \mathrm{E}((N - m_1)^i)$ is the centered moment of order i. A negative or positive skewness indicates that the mass of the distribution is concentrated on the right or, respectively, left side of the expectation. A skewness of 0 indicates a balanced distribution. A negative or positive excess kurtosis indicates that the distribution is more flat or, respectively, more peaked than the Gaussian distribution. A Gaussian distribution has a excess kurtosis of 0.

In Table 1 we can see the value of these quantities for several DNA patterns. The first three patterns have been arbitrarily chosen, but pattern GCTGGTGG is a well-known functional pattern in the *E. coli* bacteria genome: the crossover hotspot instigator, also called the CHI motif (see [13] for more details on CHI motifs in bacteria).

For the first three simple patterns, we can see how the additional information of the skewness and excess kurtosis gives us a better description of their distribution. For example, we know from theory that highly overlapping patterns are distributed according to compound Poisson approximations. This is exactly why we observe an increased skewness and kurtosis from pattern GCTGGT (nonoverlapping) to pattern GGGGGG (highly self-overlapping).

If we now consider the more complex patterns of the second part of Table 1, we can observe how the running time of Algorithm 2 quickly increases with L. This is obviously not a surprise since we expect a cubic complexity in this parameter with this approach. We should however note that it is nevertheless possible to deal with moderately complex patterns like GNNGNNGG,

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

TABLE 1: First four moments of several DNA patterns computed through the power algorithm (running time indicated in seconds). The background model is the order-($d = 1$) homogeneous Markov model defined in Subsection 5.1 and the sequence length is $\ell = 400\,000$. The special letter 'N' means 'any nucleotide' (this is standard notation in DNA sequences).

Pattern	L	Expectation	Standard deviation	Skewness	Excess kurtosis	Time
GCTGGT	9	70.09	8.364	0.119 10	0.014 13	0.09
AGAGAG	9	84.89	9.791	0.127 80	0.019 03	0.09
GGGGGG	9	65.91	10.260	0.202 90	0.053 63	0.09
GCTGGTGG	11	3.782	1.945	0.514 20	0.264 30	0.11
GCTGGNGG	14	20.79	4.559	0.219 20	0.048 01	0.11
GNTGGNGG	21	79.55	9.014	0.115 70	0.013 90	0.49
GNTGNNGG	28	340.1	18.680	0.056 28	0.003 31	1.10
GNNGNNGG	63	1508.0	42.290	0.032 83	0.001 36	15.80

which in fact contains a total of $4^4 = 256$ simple patterns. Another interesting observation is that both the skewness and kurtosis get closer to 0 when we add more N symbols into the pattern. This is due to the fact that adding more N makes the pattern more frequent (this can be seen with the geometrically increasing expectation) and that Gaussian approximations for pattern problems are well known to work better for frequent patterns.

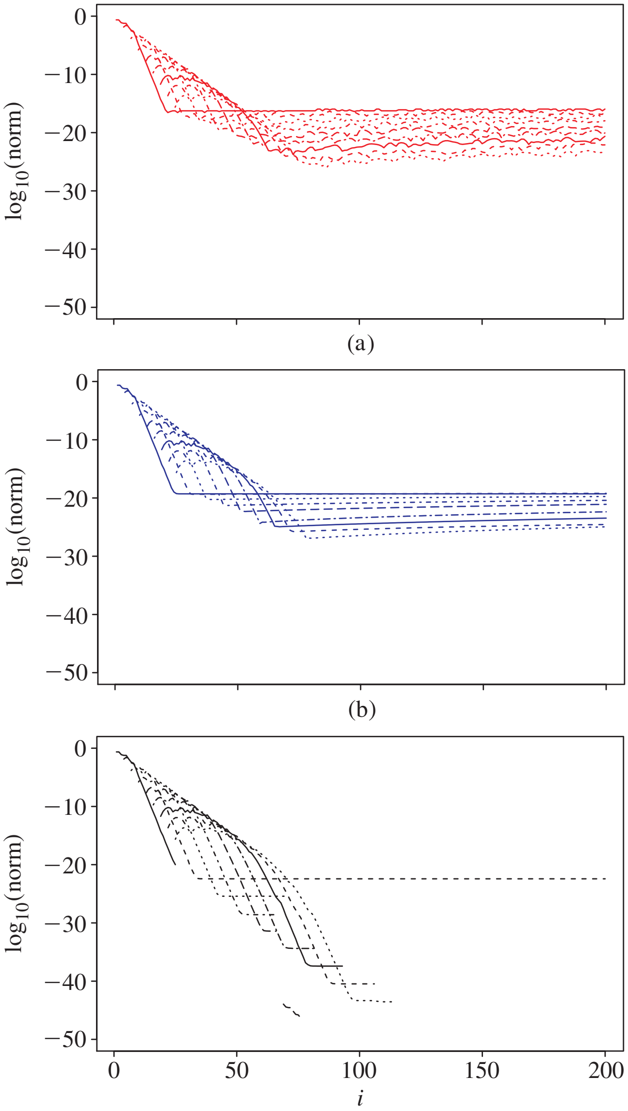
5.3. Numerical stability of the partial recursion

In Figure 1 we plot the results of an empirical study of the convergence of $D_k^{k+1}(i)$ towards $\mathbf{0}$, obtained by computing $\|D_k^{k+1}(i)\|_\infty$ for several k through Algorithm 3. We consider here three ways of updating $D_k^j(i)$: using only $D_k^{j-1}(i) - D_k^{j-1}(i-1)$ (Figure 1(a)), using only $TD_k^j(i-1) + QD_{k-1}^j(i-1)$ (Figure 1(b)), and taking the update which displays the smallest norm (Figure 1(c)). If these three alternative approaches give similar results when $\|D_k^{k+1}(i)\|_\infty \geq 10^{-15}$, differences start to appear for smaller values. The differential recurrence relation (Figure 1(a)) quickly starts to accumulate machine precision residuals and results in noisy curves that increase slowly. When using the matrix recurrence relation (Figure 1(b)), a similar problem arises, although it appears slightly later and with far less noise. Surprisingly, the last approach which combines the two updating methods at each step benefits from a synergistic effect and displays a far better stability. Similar behavior has been observed for a wide range of tested patterns (data not shown).

5.4. Near Gaussian approximations

Gaussian approximations for random pattern counts are widely used in the literature. These approximations are typically used when exact computations are untractable either because of the pattern complexity and/or because of the length of the considered sequence and/or the high number of observed occurrences. For such problems, Gaussian approximations are supposed to work better for frequent patterns. In practice, however, the quality of these approximations decrease dramatically when considering extreme events, which means that these approximations are not recommanded for precisely computing small p-values (see [28] for more details). Here, we want to advance these approximations by taking advantage of higher-order moments to obtain near Gaussian approximations. This well-known technique is described in detail in Appendix B.

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press



(a)

(b)

(c)

FIGURE 1: Plot of $\log_{10} \|D_k^{k+1}(i)\|_\infty$ for $1 \leq k \leq 9$ (from left to right) against $1 \leq i \leq 100$ for the pattern $\mathcal{W}$ = GNTGNNGG over the DNA alphabet $\mathcal{A} = \{$A, C, G, T$\}$ (where the N symbol refers to 'any letter') using an order-($d = 1$) Markov model. The curves are obtained through Algorithm 3 using recurrence relation Lemma 2(iv) only (*top*), Lemma 2(v) only (*middle*), and Lemma 2(iv) and (v), keeping the $D_k^j(i)$ displaying the smallest norm (*bottom*). The missing values (large contiguous regions) correspond to $\|D_k^{k+1}(i)\|_\infty = 0$ in our floating point computations.

We can see in Figure 2 the relative error (on a log scale) of several Edgeworth approximations for the distribution of pattern GCTGGT. The solid line shows the reliability of plain Gaussian approximations (which correspond to an order-($s = 0$) Edgeworth expansion). Unsurprisingly, this approximation works better around the expectation ($\mathrm{E}(N) = 70.09$ according to Table 1),

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

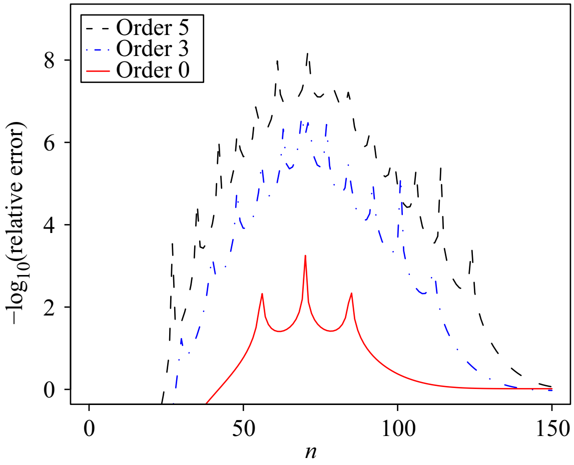


FIGURE 2: The relative error on a decimal log scale for Edgeworth's expansion of order $s = 0$ (*solid line*), $s = 3$ (*dash–dot line*), and $s = 5$ (*dashed line*) for pattern GCTGGT in an order-1 homogeneous Markov model (parameter estimated on the complete genome of *E. coli*) of length $\ell = 400\,000$.

providing two exact digits in the range [54, 85], and one exact digit in the range [50, 92]. Beyond these limits, we get too far in the tail distribution (i.e. too far from the expectation) to obtain reliable results. This behavior is exactly what we expect from the central limit theory.

If we now consider an order $s = 3$ Edgeworth expansion (which uses moments up to order $k = 5$) depicted with a dash–dot line in Figure 2, we see a dramatic improvement in both the accuracy of the approximation (up to six exact digits) and in the range of reliability (at least one exact digit on [28, 118]). We can even obtain a further improvement by considering an order-($s = 5$) expansion (see the dashed line in Figure 2), which uses moments up to order $k = 7$. In both cases, however, the reliability of these approximations decreases dramatically when the probability of the event of interest decreases.

We observe very similar behavior for pattern AGAGAG and pattern GGGGGG, and thus we omit the corresponding figures.

We should note the presence of fairly regular peaks in Figure 2 which are characteristics of Taylor expansions and similar polynomial approximations when looking at the relative error in the proper scale.

Thanks to this work, we see that, for a modest additional cost (computing moments up to order $k = 5$ or $k = 7$ instead of simple first and second moments), we can dramatically improve the reliability of Gaussian approximations for pattern problems. However, we should note that if the improvement is significant, it first affects the region closest to the mean. This is not surprising for a central limit approximation, as such an approximation works best in the center of the distribution. As a consequence, our advice would be to use higher-order developments when considering more extreme distribution events. But even better advice would be to rely preferably on a tail distribution approximation (e.g. large deviations) when considering such extreme events.

5.5. Near Poisson approximations

A very common alternative to Gaussian approximations for random pattern counts is Poisson approximations. These approximations are known to be quite accurate for nonoverlapping

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

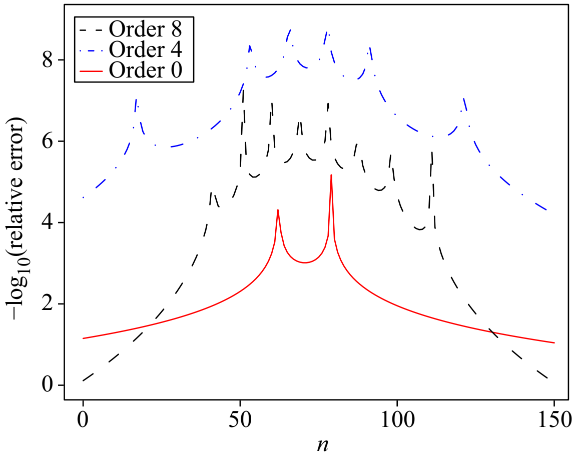


Figure 3: The relative error on a decimal log scale for the Gram–Charlier type-B approximation of order $s = 0$ (*solid line*), $s = 4$ (*dash–dot line*), and $s = 8$ (*dashed line*) for pattern GCTGGT in an order-1 homogeneous Markov model (parameter estimated on the complete genome of *E. coli*) of length $\ell = 400\,000$.

patterns, but also to fail for highly self-overlapping patterns for which compound Poisson approximations are known to perform better. Poisson approximations are supposed to perform better for rare patterns; in practice, however, it appears that they also perform well for frequent patterns, with the advantage over Gaussian approximations of being suitable for the computation of small p-values (see [28] for more details). Here we want to evaluate the reliability of near Poisson approximations based on the Gram–Charlier type-B series described in Appendix C.

For the nonoverlapping pattern GCTGGT, we can see in Figure 3 that the plain Poisson approximation (order-($s = 0$) Gram–Charlier type-B series) already gives very good results with at least one exact digit on all the distribution, and up to four or five of them in the region close to the expectation. This interesting result is dramatically improved by the order-($s = 4$) approximations, which give at least four exact digits on all the considered range and more that eight exact digits around the expectation. Surprisingly, the order-($s = 8$) approximation is less reliable than the previous approximation, and gives even worse results than the plain Poisson approximation in the tail distributions. This is due to the fact that the coefficients c_k computed according to (15) below accumulate large terms that compensate each other. This is a typical scenario for large relative errors in floating point arithmetic. We can solve this problem by either performing computations with an arbitrary number of digits (usually slow), or explicitly computing the expected relative error using the current machine precision and discarding the unreliable coefficients.

If we now consider the self-overlapping pattern AGAGAG, we know from theory that Poisson approximations are not supposed to perform well. This is the reason why in Figure 4 we observe that the plain Poisson approximations works only on a very limited range of the distribution (approximately on [69, 103]). Once again, however, an order-($s = 4$) or order-($s = 8$) Gram–Charlier expansion dramatically improves the reliability of the approximations, giving up to six exact digits close to the expectation and at least one exact digit on a much wider range (up to [24, 150] for order $s = 8$). We should note that in this case, the numerical issue observed for

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

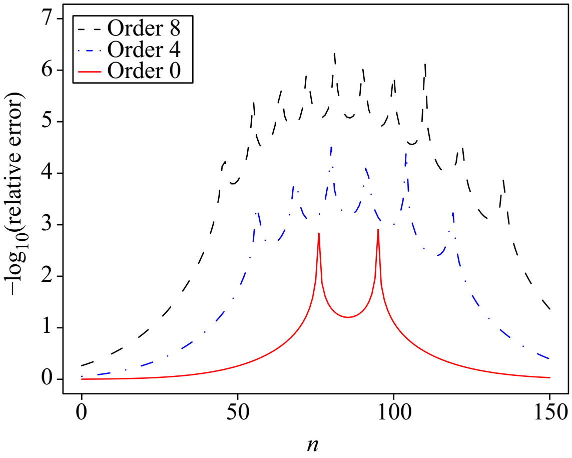


FIGURE 4: The relative error on a decimal log scale for the Gram–Charlier type-B approximation of order $s = 0$ (*solid line*), $s = 4$ (*dash–dot line*), and $s = 8$ (*dashed line*) for pattern AGAGAG in an order-1 homogeneous Markov model (parameter estimated on the complete genome of *E. coli*) of length $\ell = 400\,000$.

high-order approximations for the previous pattern does not occur. We obtain a very similar result for the even more self-overlapping pattern GGGGGG, and thus we omit the corresponding figure.

As with near Gaussian approximations, we see that near Poisson approximations can dramatically improve the reliability of Poisson approximations for a very modest cost (e.g. computing moments up to order $k = 4$ or $k = 8$).

6. Conclusion

In this paper we derived from the explicit expression of the moment generating function of a pattern random count N a new formula that allows us to compute an arbitrary number k of moments of N. We also introduced three efficient algorithms to perform this computation. The first algorithm allows the computation of pattern count moments of arbitrary order in the framework of the heterogeneous Markov model, which is a completely new result (to the author's knowledge). The second algorithm, suitable for homogeneous models and low complexity patterns, appears to have a better or similar complexity to state-of-the art known algorithms, but with a far simpler implementation. Finally, the third algorithm uses partial recursions, exploiting the sparse structure of the transition matrix to provide a logarithmic complexity with the sequence length even for high complexity patterns. This very promising approach however suffers from numerical instabilities in floating point arithmetic that need to be investigated further.

We should note that our main result can be easily extended to mixed moments of several pattern counts. For brevity, we give here such a result only for the particular case of two patterns, $\mathcal{W}_1$ and $\mathcal{W}_2$, in a homogeneous model. We assume that the final states of the DFA could be partitioned into $\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2$ such that $\mathcal{F}_1$ and $\mathcal{F}_2$ count the number, N_1 and N_2, of the occurrences of $\mathcal{W}_1$ and $\mathcal{W}_2$, respectively. This is always possible by duplicating states. We

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

consider

$$f(y_1, y_2) := \sum_{n_1, n_2 \geq 0} \mathrm{P}(N_1 = n_1,\ N_2 = n_2) y_1^{n_1} y_2^{n_2},$$

and we then have

$$f(y_1, y_2) = \mu_d (P + y_1 Q_1 + y_2 Q_2)^{\ell - d} \mathbf{1}.$$

By introducing

$$g(y_1, y_2) := \mu_d (T + y_1 Q_1 + y_2 Q_2)^{\ell - d} \mathbf{1}$$

we obtain for any $k_1, k_2 \geq 0$,

$$\mathrm{E}\left(\frac{N_1!}{(N_1 - k_1)!} \frac{N_2!}{(N_2 - k_2)!} \right) = k_1! \, k_2! \, [g(y_1, y_2)]_{y_1^{k_1} y_2^{k_2}}.$$

As an application, we have considered the distribution of DNA patterns in genomic sequences. In this particular framework, we have shown how order-($k = 3$) and order-($k = 4$) moments allow us to obtain a better description of the distribution (with quantities like skewness and excess kurtosis). We have also considered moment-based approximations, namely Edgeworth's expansion (near Gaussian approximations) and Gram–Charlier type-B series (near Poisson approximations). For both approximations, we have seen how the additional information provided by a couple of higher-order moments can dramatically improve the reliability of these common approximations. As a perspective, it seems to be very promising to develop near geometric or compound Poisson distributions with Gram–Charlier type-B series.

Appendix A. Moments and cumulants

For any random variable X and any $k \geq 0$, we define the following quantities: $g_k := 1/k!\ \mathrm{E}(X!/(X - k)!)$, the coefficient of degree k in the polynomial $g(y)$ defined in Section 3; $m'_k := \mathrm{E}(X^k)$, the moment of order k; $m_k := \mathrm{E}((N - m'_1)^k)$, the centered moment of order k; and κ_k, the cumulant of order k defined by

$$h(t) := \log \mathrm{E}(\mathrm{e}^{tN}) = \sum_{k \geq 1} \kappa_k (t^k / k!).$$

Cumulants and moments are connected through the following formula:

$$\kappa_k = m'_k - \sum_{l=1}^{k-1} \binom{k-1}{l-1} \kappa_l m'_{k-l}.$$

Using this formula, we obtain $\kappa_1 = \mathrm{E}(X)$, $\kappa_2 = m_2 = \mathbb{V}(X)$, $\kappa_3 = m_3$, and $\kappa_4 = m_4 - 3m_2^2$. The skewness, γ_1, and excess kurtosis, γ_2, can be expressed in terms of the cumulants: $\gamma_1 = \kappa_3 / \kappa_2^{3/2}$ and $\gamma_2 = \kappa_4 / \kappa_2^2$.

Appendix B. Edgeworth's expansion

We take Edgeworth's expansion directly from [5], except for the explicit order-5 expansion given in (14) below, which is a new contribution to the author's knowledge (only order-3 explicit expansions seem to be available in the literature).

Let X be a centered random variable ($\mathrm{E}(X) = 0$) that admits finite moments of all orders (we denote by σ^2 the variance of X), and let $\Phi(t) := \mathrm{E}(\mathrm{e}^{\mathrm{i}X})$ (where i denotes an imaginary

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

complex number) be its characteristic function. Let φ be the characteristic function of X/σ; we have $\varphi(t) = \Phi(t/\sigma)$. The definition of cumulants (see Appendix A) then allows us to write the expansion:

$$\log\phi(t) = \log\Phi\left(\frac{t}{\sigma}\right) \sim \sum_{k=2}^{\infty}\frac{\kappa_k}{\sigma^k k!}(\mathrm{i}t)^k.$$

Then by defining $S_k := \kappa_k/\sigma^{2k-2}$ we obtain

$$\phi(t) \sim \exp\left\{\sum_{r=1}^{\infty}\frac{S_{r+2}\sigma^r}{(r+2)!}(\mathrm{i}t)^{r+2}\right\}. \tag{12}$$

The Fourier transform of expansion (12) then gives

$$q(x) = Z(x)\left(1 + \sum_{s=1}^{\infty}\sigma^s\left\{\sum_{\{k_m\}_s} H_{s+2r}(x)\prod_{m=1}^{s}\frac{1}{k_m!}\left(\frac{S_{m+2}}{(m+2)!}\right)^{k_m}\right\}\right), \tag{13}$$

where $q(x) := \sigma p(\sigma x)$ is the probability distribution function (PDF) of X/σ ($p(x)$ being the PDF of X), $Z(x) = \exp(-x^2/2)/\sqrt{2\pi}$ is the PDF of a standard Gaussian variable, $\{k_m\}_s$ is the set of all nonnegative integer solutions of the Diophantine equation $k_1 + 2k_2 + \cdots + sk_s = s$, $r = k_1 + k_2 + \cdots + k_s$, and the $H_k(x)$ are the Hermite polynomials defined recursively by $H_0(x) := 1$ and $H_k(x) := xH_{k-1}(x) - H'_{k-1}(x)$ for all $k \geq 1$.

The sets of $\{k_m\}_s$ for $1 \leq s \leq 5$ are

$$\{k_m\}_1 = \{1\}, \qquad \{k_m\}_2 = \{20, 01\}, \qquad \{k_m\}_3 = \{300, 110, 001\},$$
$$\{k_m\}_4 = \{4000, 2100, 0200, 1010, 0001\},$$
$$\text{and}\quad \{k_m\}_5 = \{50000, 31000, 12000, 20100, 01100, 10010, 00001\},$$

and the explicit expression of (13) up to order $s = 5$ (such an explicit expression can be found up to $s = 3$ in [4]) is

$$\begin{aligned}\frac{q(x)}{Z(x)} \simeq{}& 1 + \sigma\left\{H_3(x)\frac{S_3}{3!}\right\}\\ &+ \sigma^2\left\{H_4(x)\frac{S_4}{4!} + H_6(x)\frac{S_3^2}{2!\,3!^2}\right\} + \sigma^3\left\{H_5(x)\frac{S_5}{5!} + H_7(x)\frac{S_3S_4}{3!\,4!} + H_9(x)\frac{S_3^3}{3!^4}\right\}\\ &+ \sigma^4\left\{H_6(x)\frac{S_6}{6!} + H_8(x)\left(\frac{S_3S_5}{3!\,5!} + \frac{S_4^2}{2!\,4!^2}\right) + H_{10}(x)\frac{S_3^2S_4}{2!\,3!^2 4!} + H_{12}(x)\frac{S_3^4}{4!\,3!^4}\right\}\\ &+ \sigma^5\left\{H_7(x)\frac{S_7}{7!} + H_9(x)\left(\frac{S_4S_5}{4!\,5!} + \frac{S_3S_6}{3!\,6!}\right) + H_{11}(x)\left(\frac{S_3^2S_5}{2!\,3!^2 5!} + \frac{S_3S_4^2}{2!\,3!4!^2}\right)\right.\\ &\qquad\left. + H_{13}(x)\frac{S_3^3S_4}{3!^4 4!} + H_{15}(x)\frac{S_3^5}{5!\,3!^5}\right\}. \end{aligned} \tag{14}$$

Appendix C. Gram–Charlier type-B series for near Poisson distributions

Gram–Charlier type-B series for near Poisson distributions is initially taken from [2], but we derive new recurrence relations that are more adapted to a modern computational framework than the explicit (and sometimes erroneous) formulae given in the original paper.

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

Let $\psi(i) := \mathrm{e}^{-\lambda}\lambda^i/i!$ be the PDF of a Poisson distribution of parameter λ, and let Δ be the differential operator defined by $\Delta\psi(i) := \psi(i) - \psi(i-1)$. Our objective is to approximate the PDF F of a discrete nonnegative random variable X with

$$F(i) \simeq \sum_{j=0}^{s} c_j \Delta^j \psi(i).$$

To this end, we use a moment method and find a solution $(c_0, c_1, \ldots, c_s)$ of

$$\sum_{j=0}^{s} c_j P_k^j(\lambda) = \mathrm{E}(X^k) \quad \text{for all } 0 \leq k \leq s$$

with $P_k^j(\lambda) := \sum_{i\geq 0} i^k \Delta^j \psi(i)$ for all $j, k \geq 0$.

It is clear that we have $P_0^0(\lambda) = 1$, and we have the following recurrence relation for all $k, j \geq 0$:

$$P_{k+1}^0(\lambda) = \lambda\biggl[P_k^0(\lambda) + \frac{\mathrm{d}P_k^0}{\mathrm{d}\lambda}(\lambda)\biggr] \quad \text{and} \quad P_k^{j+1}(\lambda) = -\frac{\mathrm{d}P_k^j}{\mathrm{d}\lambda}(\lambda).$$

We hence find that $c_0 = 1$, and we derive the following recurrent relation for $k \geq 1$:

$$c_k = \frac{1}{P_k^k(\lambda)}\biggl(\mathrm{E}(X^k) - \sum_{j=0}^{k-1} c_j P_k^j(\lambda)\biggr).$$

Note that $P_k^k(\lambda)$ is always a scalar. If we now define $g_k := 1/k!\ \mathrm{E}(X!/(X-k)!)$ then we can show, by recurrence, for all $k \geq 1$ that we finally have

$$c_k = -\frac{(k-1)}{k!} g_1^k + \sum_{j=2}^{k} (-1)^j \frac{g_1^{k-j} g_j}{(k-j)!}. \tag{15}$$

The explicit first five terms of this formula are

$$c_2 = g_2 - \frac{g_1^2}{2}, \qquad c_3 = -g_3 + g_1 g_2 - \frac{g_1^3}{3}, \qquad c_4 = g_4 - g_1 g_3 + \frac{g_1^2 g_2}{2} - \frac{g_1^4}{8}$$

$$c_5 = -g_5 + g_1 g_4 - \frac{g_1^2 g_3}{2} + \frac{g_1^3 g_2}{6} - \frac{g_1^5}{30},$$

$$c_6 = g_6 - g_1 g_5 + \frac{g_1^2 g_4}{2} - \frac{g_1^3 g_3}{6} + \frac{g_1^4 g_2}{24} - \frac{g_1^6}{144}.$$

References

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

[2] Aroian, L. A. (1937). The type B Gram–Charlier series. *Ann. Math. Statist.* **8,** 183–192.

[3] Beaudoing, E. *et al.* (2000). Patterns of variant polyadenylation signal usage in human genes. *Genome Res.* **10,** 1001–1010.

[4] Bernardeau, F. and Kofman, L. (1995). Properties of the cosmological density distribution function. *Astrophys. J.* **443,** 479–498.

[5] Blinnikov, S. and Moessner, R. (1998). Expansions for nearly Gaussian distributions. *Astron. Astrophys. Suppl. Ser.* **130,** 193–205.

[6] Boeva, V., Clément, J., Régnier, M. and Vandenbogaert, M. (2005). Assessing the significance of sets of words. In *Combinatorial Pattern Matching 05* (Lecture Notes Comput. Sci. **3537**), Springer, Berlin.

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press

[7] Boeva, V. *et al.* (2007). Exact p-value calculation for heterotypic clusters of regulatory motifs and its application in computational annotation of cis-regulatory modules. *Algorithms Molecular Biol.* **2,** 13.

[8] Brāzma, A., Jonassen, I., Vilo, J. and Ukkonen, E. (1998). Predicting gene regulatory elements in silico on a genomic scale. *Genome Res.* **8,** 1202–1215.

[9] Chang, Y.-M. (2005). Distribution of waiting time until the rth occurrence of a compound pattern. *Statist. Prob. Lett.* **75,** 29–38.

[10] Cowan, R. (1991). Expected frequencies of DNA patterns using Whittle's formula. *J. Appl. Prob.* **28,** 886–892.

[11] Crochemore, M. and Stefanov, V. T. (2003). Waiting time and complexity for matching patterns with automata. *Inform. Process. Lett.* **87,** 119–125.

[12] Denise, A., Régnier, M. and Vandenbogaert, M. (2001). Assessing the statistical significance of overrepresented oligonucleotides. In *Algorithms in Bioinformatics* (Lecture Notes Comput. Sci. **2149**), Springer, Berlin, pp. 85–97.

[13] El Karoui, M., Biaudet, V., Schbath, S. and Gruss, A. (1999). Characteristics of Chi distribution on different bacterial genomes. *Res. Microbiol.* **150,** 579–587.

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

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

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

[17] Geske, M. X. *et al.* (1995). Compound Poisson approximations for word patterns under Markovian hypotheses. *J. Appl. Prob.* **32,** 877–892.

[18] Godbole, A. P. (1991). Poisson approximations for runs and patterns of rare events. *Adv. Appl. Prob.* **23,** 851–865.

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

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

[21] Kleffe, J. and Borodovsky, M. (1997). First and second moment of counts of words in random texts generated by Markov chains. *Comput. Appl. Biosci.* **8,** 433–441.

[22] Lladser, M. E. (2007). Minimal Markov chain embeddings of pattern problems. In *Proc. 2007 Inform. Theory Appl. Workshop*, University of California, San Diego, pp. 251–255.

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

[24] Mariño-Ramírez, L., Spouge, J. L., Kanga, G. C. and Landsman, D. (2004). Statistical analysis of over-represented words in human promoter sequences. *Nuc. Acids Res.* **32,** 949–958.

[25] Nicodème, P., Salvy, B. and Flajolet, P. (2002). Motif statistics. *Theoret. Comput. Sci.* **287,** 593–617.

[26] Nuel, G. (2004). LD-SPatt: large deviations statistics for patterns on Markov chains. *J. Comput. Biol.* **11,** 1023–1033.

[27] Nuel, G. (2006). Effective p-value computations using finite Markov chain imbedding (FMCI): application to local score and to pattern statistics. *Algorithms Molecular Biol.* **1,** 5.

[28] Nuel, G. (2006). Numerical solutions for patterns statistics on Markov chains. *Statist. Appl. Genetics Molecular Biol.* **5,** 26.

[29] Nuel, G. (2008). Pattern Markov chains: optimal Markov chain embedding through deterministic finite automata. *J. Appl. Prob.* **45,** 226–243.

[30] Pevzner, P., Borodovski, M. Y. and Mironov, A. 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.

[31] Prum, B., Rodolphe, F. and de Turckheim, E. (1995). Finding words with unexpected frequencies in deoxyribonucleic acid sequences. *J. R. Statist. Soc. B* **57,** 205–220.

[32] Reignier, M. (2000). A unified approach to word occurrences probabilities. *Discrete Appl. Math.* **104,** 259–280.

[33] Reinert, G. and Schbath, S. (1999). Compound Poisson and Poisson process approximations for occurrences of multiple words in Markov chains. *J. Comput. Biol.* **5,** 223–253.

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

[35] Stefanov, V. T. and Pakes, A. G. (1997). Explicit distributional results in pattern formation. *Ann. Appl. Prob.* **7,** 666–678.

[36] Stefanov, V. T. and Szpankowski, W. (2007). Waiting time distributions for pattern occurrence in a constrained sequence. *Discrete Math. Theoret. Comput. Sci.* **9,** 305–320.

[37] Van Helden, J., André, B. and Collado-Vides, J. (1998). Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J. Molecular Biol.* **281,** 827–842.

https://doi.org/10.1239/jap/1294170523 Published online by Cambridge University Press