Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Corso di laurea magistrale in Ingegneria Informatica

On the Discovery of Significant Motifs in Genomic Sequences

Author:
Matteo Boscariol

Advisors:
Prof. Geppino Pucci
Prof. Andrea Pietracaprina

December 9, 2013

Academic year 2013-2014

Abstract

The discovery of statistically significant motifs is important in order to make decisions that are not relying on pure chance. Testing each frequent pattern for statistical significance in isolation may lead to a high false discovery rate. In this thesis we study the statistical properties of some families of motifs of the same length. In particular, we develop a method for the approximation of the average number of frequent motifs in the family in a text where each character is independent. We give a bound on the error of the approximation and show that this bound is loose in practice. We develop a test through simulation which verifies whether the distribution of the number of frequent motifs can be approximated to a Poisson distribution. We discover that in the families we studied the real distribution can be approximated only when its average is significantly less than 1.

Acknowledgements

I would like to express my sincere gratitude to my advisors, Prof. Geppino Pucci and Prof. Andrea Pietracaprina, whose encouragement, knowledge and expertise have been an invaluable help in the production of this thesis. I would like to thank Prof. Fabio Vandin for his aid, in particular for providing the benchmark datasets used in this thesis.

I would like to thank the organizers of the Italian Olympiad in Informatics for providing me many great experiences and opportunities to improve myself since high school, with special gratitude to Prof. Roberto Grossi, Prof. Luigi Laura, and Prof. Romeo Rizzi, whose teachings helped me greatly in these years.

I would like to thank my family, which contributed in any possible way to support me in my graduation, even through many difficulties: my parents, my grandparents, my uncles and aunts, my cousins, my sister, her husband Alessandro and his family, and my nephew Luca. I hope I will be able to share the fruits of their support in the near future.

I would like to thank my friends that accompanied me in this journey, cheered me up and helped me feel in excellent company: Jenny, Matteo, Denis, Enrico, Laura, Nicola, Lorenzo, Alessandro, Giovanni, Alberto, Simone, Davide, Massimo, Carlo, and any friend I forgot to mention.

Contents

Chapter 1

Introduction

1.1 Data mining and algorithm accuracy

Data mining is playing a very important role in society. Its wide application spectrum, such as business, medicine, science and engineering, and the availability of large data repositories make it one of the most important fields of computer science. As data warehouses get larger and the available computational power grows, there is an ever increasing interest in extracting more information and knowledge from the data.

The development of efficient data mining algorithms is one of the main topics of research in the field. Depending on the problem, these algorithms usually have to search the output space for unusually frequent (or infrequent) elements in the input data. For example, in the market basket analysis, one of the main problems is to find set of items (*itemsets*) that appear in the input transactions with a certain frequency. The classical algorithms that attempt to solve this problem must face two major issues: the search space may be exponential in size (compared to the input size); and the definition of "frequent" is often left to the user through a set of parameters. The latter point is crucial, because the parameters control both the efficiency and the accuracy of the algorithms. If the definition of "frequent" is too strict, the algorithm is more conservative, but it may return few discoveries or none at all. On the other hand, if the definition is too lax, the algorithm may become too slow, the output can be exponential in the size of the input, and most of the frequent elements it discovers may be uninteresting, or may have no meaningful relation with the data.

For these reasons, there is an increasing interest in developing data mining algorithms that are efficient and retrieve information from the data with a certain accuracy. In order to do this, the algorithms often make use of additional concepts that reduce the size of the output. For example, in the market basket analysis, some algorithms focus on the closed itemsets, which are itemsets whose frequency is strictly higher than the

ones of their supersets. In this case, if an itemset is not closed, there is a superset whose frequency is the same as that itemset, and the latter can be used to describe both events. Similarly, other algorithms focus on finding maximal itemsets, which are frequent itemsets such that all their supersets are not frequent.

An important concept for any algorithm is the statistical significance of the frequent elements. An outcome is statistically significant if its occurrence is unlikely to happen "by chance". The statistical significance is an interesting property, as it gives additional confidence on the quality of the result. By establishing a random model that can approximately describe the process that generates the data, and using this random model to check whether an observed event is unlikely to happen in randomly-generated samples, we can obtain a substantial confidence on the meaningfulness of statistically significant results.

The properties of closure and maximality can be verified easily and usually produce well-defined structures, allowing the algorithms to perform substantial optimizations. On the other hand, the statistical significance of an element, while theoretically appealing, introduces new complications to the problem. First of all, the statistical significance measures are non-monotonic [8], while monotonicity is a desired property for Apriori algorithms. Then, the evaluation of the statistical significance requires the calculation of statistics that define the "score" of an element, which measures its statistical significance. Finally, the random model must be chosen with care.

The discovery of statistically significant elements is nonetheless important in order to make decisions that are not relying on pure chance. Consequently, the statistical significance is often employed in recent algorithms.

1.2 Frequent patterns in computational biology

Computational biology is the field that studies the development of theoretical models, analytical methods and algorithms for the study of biological information. Its broad definition encompasses various disciplines: computer science, applied mathematics, statistics, molecular biology, and so on. It is one of the leading applications of computer science. The field has grown considerably in the last decades, as new technologies enable the acquisition of large amounts of biological information, thus constantly introducing new problems and challenges. The vast family of problems that arise in computational biology is introduced in [24].

In computational biology, the discovery of interesting patterns in the DNA plays an important role. The goal is to find recurring patterns in a sequence that may have special biological functions. Various algorithms have been developed that consider a

large variety of patterns, from simple sequences to more complex regular expressions, considering even patterns that allow a certain number of mismatches [13].

As both the size and the number of DNA sequences grow, the interest in efficient algorithms is growing as well. Depending on the family of patterns, the number of frequent patterns in a text may be polynomial or exponential in the size of the input. Even when the number of frequent patterns is polynomial, some of them may carry uninteresting or redundant information. For this reason, recent algorithms apply the concepts of closure and maximality for the patterns, in order to remove redundancies [7].

Furthermore, there is also a substantial interest in the statistical significance of the discovered patterns. Depending on the random model, even patterns that share a common structure may have substantial variations in their expected frequencies. Thus, the discovered patterns are individually tested for their statistical significance.

There are several works in computational biology that study the statistical significance of the patterns obtained by sequence mining and sequence similarity algorithms [2, 5, 6, 10, 11, 12, 15, 16, 17, 18, 19, 20, 21, 22, 23, 25]. These works usually focus on two related metrics to determine the significance: the z-score, which measures how far is the observed frequency of a pattern from its expectation; and the p-value, which measures the probability that the frequency of a pattern is greater or equal to the observed frequency. These statistical tests usually take a limited number of patterns in consideration, specifically the patterns in the output of the algorithms, and their significance is usually determined individually through a single statistical hypothesis test for each returned pattern, which decide whether a pattern is significant according to a probability threshold.

1.3 Frequent patterns and multiple hypotheses

The evaluation of the statistical significance of frequent patterns "a posteriori", after they have been mined from the data, must be done carefully. As the sequence size increases and the families of events become bigger, some of these events may occur by chance, even when each event is considered rare individually. This may lead to a high percentage of false positives, if the significance threshold does not consider the multiplicity of the hypotheses we are testing.

For example, suppose that in a family $\mathcal{F}$ of patterns, each pattern may appear frequent in a random sequence with probability 10^{-5}, but the family has $|\mathcal{F}| = 10^6$ patterns. Then, for the linearity of the expectation, the expected number of frequent patterns in a random sequence is 10. Thus, if we use an inappropriate level of significance (for example, $\alpha = 0.05$), we may consider some patterns as significant even when they are expected to be frequent due to chance.

In order to solve this issue, there are some corrections to the significance levels of the hypothesis tests that can be considered. Some corrections ensure a bound on the probability of having any false positives (the Familywise Error Rate, or FWER), or a bound on the expected rate of false positives over the number of positives (False Discovery Rate, or FDR). The corrections that bound the FWER are often excessively conservative, while the corrections for the FDR may increase the accuracy while returning a reasonable number of discoveries.

There is also a certain interest in establishing an appropriate frequency such that all the frequent patterns can be considered statistically significant with a limited FDR. Finding an appropriate frequency is also important in order to achieve a good trade-off between efficiency and accuracy. This approach has been developed for the problem of finding statistically significant itemsets in [9]. In order to establish a frequency threshold over which any frequent itemset can be considered statistically significant, the method requires to calculate the p-value for the total number of frequent patterns. In order to obtain this value, the authors employed the Chen-Stein method, which is a powerful tool for bounding the error in approximating the real distribution with a Poisson distribution. The Chen-Stein method has already been applied successfully for the approximation of the p-value for single patterns and small sets of patterns [15], while its applicability on large, structured families of patterns seems to be virtually unexplored.

1.4 Objective and results

In section 1.3, we mentioned a recently-developed approach for identifying statistically significant frequent itemsets, whose details can be found in [9]. The purpose of this work is to check whether this approach can be also applied for the identification of statistically significant patterns in genomic sequences.

We are focused on families of patterns with the same length k, and we are interested in finding some information about the number of frequent patterns of length k: in particular, we estimate the expected number of frequent patterns, and find some conditions under which the number of frequent patterns can be approximated to a Poisson approximation through the Chen-Stein method.

In the beginning, we start with the exploration of the current state of the art in the evaluation of the statistical significance of patterns, and we obtain some specific results for our families of patterns. We will use these results to approximate the average number of frequent patterns efficiently.

We obtain an estimation of this average within a certain error bound, and we show through some simulations that the error bound is very conservative in practice. We make some considerations on the applicability of the Chen-Stein method for the number

of frequent patterns through theoretical tools and by simulating random texts that share some characteristics of the input.

We show that the Poisson approximation is applicable only when the expected number of frequent patterns is considerably less than 1. When independent random models are applicable and under certain conditions, we can find reasonable frequency levels such that the Poisson approximation holds, without the need for simulations.

1.5 Organization of the document

In chapter 2 we introduce the notation used throughout this work, give an introduction to the statistical significance, present the approach described in [9], define some of the random models commonly used in computational biology, and enunciate a theorem obtained from the Chen-Stein method, which gives a bound on the error made by approximating the distribution of certain integer random variables to a Poisson distribution.

In chapter 3 we present some known algorithms for the calculation of the exact probability of occurrence of a single pattern. One of these has been independently studied and developed, and it will be used for comparison in the next steps. We also report on the complexity of these algorithms.

In chapter 4 we report some results on the approximation of the p-value for the number of occurrences of a single pattern, focusing on simple expressions for the error bound. Some of these results are characterized by little variability among the family of patterns.

In chapter 5, through the results obtained in the previous chapter, we give a good estimate of the average number of frequent patterns with a specific length for independent models. Subsequently, we approach the issue of finding theoretical estimates for the approximation error for the distribution of the number of frequent patterns, obtained from the Chen-Stein method. We consider a naïve approach, which gives an exact value for the approximation error, as a function of the average and the variance. We also consider more elaborate approaches, which appear to be theoretically challenging but can be evaluated experimentally.

In chapter 6 we evaluate the goodness of the results obtained in the previous section through simulation, in particular to see whether there is room for improvement in the naïve approach for the approximation error. We simulate a number of large sequences, from which we obtain an empirical distribution that can be compared to the theoretical results.

In chapter 7 we summarize the theoretical and empirical results and propose future developments.

Chapter 2

Preliminaries

2.1 String and motif definitions

DNA sequences can be represented as strings of arbitrary length built on the alphabet $\Sigma = \{A, C, G, T\}$. A string s of length $|s| = k \geq 0$ on the alphabet Σ is a concatenated sequence of k characters:

$$s = s[0]s[1] \dots s[k-1], \quad s[i] \in \Sigma \quad \forall i \in \{0, \dots, k-1\}$$

Σ^* represents the set of all strings in Σ of arbitrary length, while $\Sigma^k = \{s \in \Sigma^* : |s| = k\}$ is the set of all strings of length k. Thus

$$\Sigma^* = \bigcup_{k=0}^{\infty} \Sigma^k$$

The symbol ϵ represents the empty string ($|\epsilon| = 0$).

Let $i, j \in \{0, \dots, k-1\}$. The substring of s from i to j is defined as:

$$s[i \dots j] = \begin{cases} s[i]s[i+1] \dots s[j] & \text{if } i \leq j \\ \epsilon & \text{otherwise} \end{cases}$$

We define a *motif* as a string x defined on the extended alphabet:

$$x \in (\Sigma \cup \{\circ\})^*$$

where $\circ \notin \Sigma$ is called a *wildcard* (or *don't care character*). If x does not contain any wildcard, it is called a *word* or *solid block*.

We now give some definitions of occurrence and frequency:

Definition 2.1 (Occurrence of a pattern in a text)**.** Given a string s and a motif x where $|s| = l, |x| = k \leq l$, the motif x *occurs* in string s at position $i : 0 \leq i \leq l - k$ if

$$(s[i + j] = x[j]) \vee (x[j] = \circ) \quad \forall j : 0 \leq j \leq k - 1.$$

in this context, s is called a *text* and x is called a *pattern.*

For solid blocks, the occurrence condition is simply $s[i \dots i + k - 1] = x$.

Definition 2.2 (Number of occurrences)**.** The number of occurrences of x in s is the number of distinct positions where x occurs in s:

$$N(x, s) = |\{i \in \{0, \dots, l - k\} : x \text{ occurs in } s \text{ at position } i\}|$$

Definition 2.3. Let $q \in \mathbb{N},\ q > 0$. We say that the motif x is *frequent* with quorum q in the text s if $N(x, s) \geq q$.

In some occasions, the string s will be omitted, and the number of occurrences will be represented as $N_x = N(x, s)$.

Definition 2.4. Let $\mathcal{F} \subseteq (\Sigma \cup \{\circ\})^*$ be a family of motifs. The number of frequent motifs of the family $\mathcal{F}$ in the text s is defined as:

$$Q_q(\mathcal{F}, s) = |x \in \mathcal{F} : N(x, s) \geq q|$$

When both the string and the motif family are clear from the context, we may omit them. We will often use the family $\mathcal{F} = \Sigma^k$; in this case, we will use the expression $Q_{k,q} = Q_q(\Sigma^k, s)$.

2.2 Statistical hypothesis testing

In a simple statistical hypothesis test, we are interested in finding whether our data is likely to have been generated from a random model. A hypothesis test consists in formulating a null hypothesis H_0 and an alternative hypothesis H_1, and then deciding whether to reject the null hypothesis. Usually, the null hypothesis assumes that the data is generated from a random variable or random process, with a known probability distribution, occasionally with a set of parameters that are estimated from the data itself.

After establishing a null hypothesis, the next step is to provide a suitable *test statistic*, which is a function of the data that summarizes its characteristics. The observed value t_{obs} of the test statistic is then compared to the test statistic $\mathbf{T}$ calculated on the random

process assumed by the null hypothesis. In statistics, various test statistic distributions have been formulated according to the underlying distribution assumed by the null hypothesis and the chosen test statistic.

The rejection of the null hypothesis is determined by defining a suitable critical region C in the space of the possible test statistics, and rejecting the null hypothesis when $t_{\text{obs}} \in C$. The region is chosen such that $p = \Pr(\mathbf{T} \in C) \leq \alpha$, where p is the *p-value* of the test, while α is the *significance level* of the test.

The significance level of a test is the probability of rejecting H_0 when it is true (also called Type I error). Its value is chosen arbitrarily before the test, depending on the application. Common significance levels are $\alpha = 0.01$, $\alpha = 0.02$, or $\alpha = 0.05$.

When $t_{\text{obs}} \notin C$, the test "fails to reject" the null hypothesis, which means that the test cannot reject the hypothesis with an acceptable type I error. It does not mean that H_0 is accepted, as the test is not capable of asserting that H_0 is true with any confidence.

In data mining, hypothesis testing is used to determine whether an element (for example, a pattern in a text, or an itemset in a set of transactions) has a high chance of having a particular role in the data, for example to test whether a group of items is unusually frequent in the transactions, meaning that they are likely to describe an interesting phenomenon. Similarly, an unusually frequent pattern in a strand of DNA may carry biological information. We call these elements "statistically significant".

2.2.1 Multiple hypothesis testing

Statistical hypothesis testing may be used improperly, especially if we are performing a number of different hypothesis tests. When we are testing multiple hypotheses to determine which patterns in a set of frequent patterns in a text are statistically significant, their multiplicity must be handled with care.

For example, suppose that we find that a pattern x from a certain family that occurs q times in a text, and under the null hypothesis, the probability for that pattern to occur at least q times is low, such as $p = 10^{-5}$. With the standard values of α, we may say that x is statistically significant. However, if we decided which pattern to test after mining it from the data, we may end up with some false positives: if there are 10^6 distinct patterns in the family with the same p-value as x, we would get that the expected number of frequent patterns is 10. Thus, we may end up systematically marking some patterns as statistically significant even when the text is generated according to the null hypothesis.

In order to choose appropriate significance levels for testing multiple hypotheses, some additional metrics have been defined. We report two of these metrics: the Familywise Error Rate, which is more common when the number of hypotheses is low; and the False Discovery Rate, which is suited for data mining algorithms to evaluate the accuracy.

Definition 2.5 (Familywise Error Rate)**.** Suppose we are testing m hypotheses. Let $\mathcal{H} = \{H_0^1, \ldots, H_0^m\}$ be the collection of null hypotheses. The Familywise Error Rate (FWER) is defined as:

$$FWER = \Pr\left(\bigcup_{i=1}^{m} \{H_0^i \text{ is rejected} \mid H_0^i \text{ is true}\}\right)$$

Definition 2.6 (False Discovery Rate)**.** Suppose we are testing m hypotheses. Let R be the number of rejected null hypotheses, and let V the number of Type I errors. Then the False Discovery Rate is defined as FDR $= E[V/R]$, with $V/R = 0$ when $R = 0$.

The easiest way to limit the FWER is the *Bonferroni correction*:

Definition 2.7 (Bonferroni correction)**.** Suppose we are testing m hypotheses. Let $\alpha \in (0, 1)$. The Bonferroni correction tests each hypothesis i with significance level:

$$\alpha_i = \frac{\alpha}{m}$$

The Bonferroni correction limits the FWER through the union bound:

Theorem 2.8. *The familywise error rate for m hypothesis tests with significance level $\alpha_i = \alpha/m$ is*

$$FWER \leq \sum_i \alpha_i = \alpha$$

The main drawback of this method is the very low significance level obtained when the number of hypotheses is large. In the pattern discovery environment, if we are testing the family $\mathcal{F} = \Sigma^k$, we are potentially testing up to $m = 4^k$ hypotheses a priori, while in practice we only test the patterns that are actually frequent in the text. This may lead to the discovery of a very limited number of significant patterns. Additionally, a bound on the FWER does not imply a bound on the FDR.

Another approach increases the power of the tests, while keeping the FDR under a desired threshold:

Theorem 2.9 (Benjamini and Yekutieli)**.** *Suppose we are testing m hypotheses. Let $p_{(1)} \leq \cdots \leq p_{(m)}$ be the ordered observed p-values of each test, and let $\beta \in (0, 1)$. Let*

$$l = \max\left\{ i \geq 0 : p_{(i)} \leq \frac{i}{m \sum_{j=1}^{m} \frac{1}{j}} \beta \right\}$$

The FDR for the rejection of the tests $(1), \ldots, (l)$ is upper bounded by β.

This approach is more selective than the Bonferroni correction for the tests with the lowest observed p-values, but the significance levels slowly increase after each rejected hypothesis.

Kirsch et al. [9] state that this approach can be directly applied to the problem of returning significant frequent itemsets. The procedure can easily be adapted for the frequent patterns: after choosing an appropriate value for the quorum and mining the frequent patterns, one needs to calculate the p-value for all the frequent patterns to appear as frequently as in the input sequence (we can skip infrequent patterns by assuming that their p-value is 1, obtaining a more selective procedure), sort the values and select the l patterns with the lowest p-value. One can calculate the p-value with exact algorithms or with approximated algorithms, depending on the quality of the approximated algorithms and the time required to test all the frequent patterns. We will explore both possibilities, evaluating the computation time of an exact algorithm and an approximated algorithm. However, the quorum value that limits the number of frequent patterns to analyze may be still required.

Alternatively, the authors suggest an algorithm that can find a quorum value such that all the frequent patterns can be considered statistically significant.

Quorum threshold for statistically significant patterns Define q_{min} and q_{max} be the minimum quorum and the maximum quorum for which to test the significance of the patterns, respectively. The procedure will perform h tests, with $h = \lfloor \log_2(q_{max} - q_{min}) \rfloor + 1$. For each test $i \in 0, \dots h-1$ we define the values $\alpha_i > 0, \beta_i > 0$ such that

$$\sum_{i=0}^{h-1} \alpha_i = \alpha < 1, \qquad \sum_{i=0}^{h-1} \beta_i = \beta < 1$$

We define the null hypotheses

$$H_0^i = \{Q_{k,q_i} \text{ is drawn from the random variable } \mathbf{Q}_{k,q_i} = Q_{q_i}(\Sigma^k, \mathbf{s})\}, \quad q_i = q_{min} + 2^i$$

Where $\mathbf{s}$ is a random process that generates strings. We choose the following rejection condition for the null hypotheses:

$$\{\Pr(\mathbf{Q}_{k,q_i} > Q_{k,q_i}) < \alpha_i\} \wedge \left\{Q_{k,q_i} > (\beta_i)^{-1} E[\mathbf{Q}_{k,q_i}]\right\}$$

If at least one of the hypotheses is rejected, we choose $q^* = \min\{i \geq 0 : H_0^i \text{ is rejected}\}$. Then, we mark any pattern with frequency of at least q^* as statistically significant.

The following theorem is an adaptation of the theorem found in [9] that proves the quality of the procedure.

Theorem 2.10. *With confidence $1 - \alpha$, the FDR of the quorum threshold procedure is at most β.*

In order to apply this procedure, we need the p-value for $\mathbf{Q}_{k,q_i}$, or at least an upper bound obtained through an approximation, and naturally its mean. If we are able to

find these values efficiently, we can skip the calculation of the p-value of the frequent patterns and flag all of them as statistically significant. Thus, we are interested in finding a way to estimate these values.

2.3 Random model

Our null hypotheses assume that the observed test statistic of the input sequence is likely to be obtained from a string generated by a random model. The definition of an appropriate random model (and thus, the null hypothesis for the test) is important for the meaningfulness of the tests.

The model should adequately describe the real process that generates the text, by taking into account its known properties. When no property is known for the process, the type and the parameters of the model are typically estimated from the input sequence; however, if the chosen random model is too complex, the analysis becomes more difficult and the model may describe too accurately the sample, eventually including its noise instead of the real process.

Various random models are used in literature. In this document, we work on some simple models, assuming that the sample space consists of all the strings of the same length as the observed sequence: $\Omega = \Sigma^l$.

2.3.1 Independent random model

Definition 2.11. Let $p : \Sigma \to [0, 1]$, with $\sum_{e\in\Sigma} p(e) = 1$. A random process $\mathbf{s} \in \Omega$ is an **independent random model** with probability function p if the random variables $\{\mathbf{s}[i] : i \in \{0, \ldots, l-1\}\}$ are mutually independent, identically distributed and:

$$P(\mathbf{s}[i] = e) = p(e) \quad \forall e \in \Sigma, i \in \{0, \ldots, l-1\}$$

Occasionally, with an abuse of notation, we use the probability vector $p = (p_1, p_2, p_3, p_4)$, with $p_1 = p(A)$, $p_2 = p(C)$, $p_3 = p(G)$, $p_4 = p(T)$. When $p(e) = 1/|\Sigma| \quad \forall e \in \Sigma$, each character of the alphabet has the same probability of occurrence in any position of the string. We call this an *independent equiprobable* random model.

The advantage of an independent random model is that each position is independent from the other positions, which reduces the complexity of the analysis. An independent equiprobable random model is the simplest to analyze, though it does not contain any characteristic of the samples.

2.3.2 1-order Markov chain random model

Definition 2.12. Let $p : \Sigma \to [0,1]$ and $T : \Sigma \times \Sigma \to [0,1]$, with $\sum_{e\in\Sigma} p(e) = 1$ and $\sum_{f\in\Sigma} T(e,f) = 1 \quad \forall e \in \Sigma$. A random process $\mathbf{s} \in \Omega$ is a **1-order Markov chain random model** with initial probability function p and transition function T if:

$$P(\mathbf{s}[0] = e) = p(e) \quad \forall e \in \Sigma$$

and if:

$$\begin{aligned} P(\mathbf{s}[i] = e_i \mid \mathbf{s}[i-1] = e_{i-1}, \mathbf{s}[i-2] = e_{i-2}, \dots \mathbf{s}[0] = e_0) &= P(\mathbf{s}[i] = e_i \mid \mathbf{s}[i-1] = e_{i-1}) \\ &= T(e_{i-1}, e_i) \end{aligned}$$

for any $i > 0$ and any possible sequence of states $e_0, \dots, e_i \in \Sigma$.

In this document we always assume that $T(e,f) > 0 \ \forall e, f \in \Sigma$. This implies that the Markov chain has a single recurring class, and it is stationary, which means there exists a steady-state probability function π such that $\sum_{e\in\Sigma} \pi(e) = 1$ and

$$\lim_{n\to\infty} P(s[i+n] = f \mid s[i] = e) = \pi(f) \quad \forall e, f \in \Sigma, i \geq 0$$

We also assume that $p = \pi$, so that the unconditioned probability is the same for all the positions: $P(s[i] = e) = P(s[0] = e) = p(e) \ \forall i > 0, e \in \Sigma$.

In order to avoid excessive cluttering in the notation, and to reuse some properties of the Markov chains, with an abuse of notation we will interpret p and π as vectors of $|\Sigma|$ elements as we did in the independent models, and we will interpret T as a matrix of size $|\Sigma| \times |\Sigma|$, indexed as:

$$T_{ij} = T(e_i, e_j) \ \ \forall i, j \in \{1, \dots, |\Sigma|\}, \quad \text{with} \quad \Sigma = \{e_1, \dots, e_{|\Sigma|}\}$$

In this context, π is the steady-state probability vector, while T is the transition matrix.

2.4 The Chen-Stein method

The Chen-Stein method is a powerful tool for calculating an error bound when approximating the sum of dependent random variables to a Poisson distribution with the same mean.

In order to determine the error bound, we define the variation distance between two random variables.

Definition 2.13. Let $\mathbf{Y_0}, \mathbf{Y_1}$ be two random variables with the same domain D. The total variation distance between $\mathbf{Y_0}$ and $\mathbf{Y_1}$ is defined as

$$\|\mathcal{L}(\mathbf{Y_0}) - \mathcal{L}(\mathbf{Y_1})\| = 2 \sup_{A \subseteq D} |P(\mathbf{Y_0} \in A) - P(\mathbf{Y_1} \in A)|$$

The variation distance is essentially twice the least upper bound of the error when we calculate probabilities for the distribution of $\mathbf{Y_0}$ by using the distribution of $\mathbf{Y_1}$, and viceversa. This result can be easily applied to the complementary cumulative distribution function:

$$\begin{aligned} |P(\mathbf{Y_0} \geq c) - P(\mathbf{Y_1} \geq c)| &\leq \sup_{A \subseteq D} |P(\mathbf{Y_0} \in A) - P(\mathbf{Y_1} \in A)| \\ &= \frac{1}{2} \|\mathcal{L}(\mathbf{Y_0}) - \mathcal{L}(\mathbf{Y_1})\| \end{aligned}$$

Let $\{\mathbf{X}_\alpha : \alpha \in I\}$ be a set of dependent indicator variables, where I is the set of indices of the variables. We want to approximate the sum $\mathbf{W} = \sum_{\alpha \in I} \mathbf{X}_\alpha$ to a Poisson distribution with the same mean. For each α, we define the *neighborhood set* of $\mathbf{X}_\alpha$ as a subset of indices of indicator variables, $B_\alpha \subset I$, with $\alpha \in B_\alpha$. The neighborhood set is arbitrary, but usually it should contain the indices β such that $\mathbf{X}_\alpha$ and $\mathbf{X}_\beta$ are dependent.

The following theorem shows the role of the neighborhood set in the Poisson approximation.

Theorem 2.14. *Let $(\mathbf{X}_\alpha : \alpha \in I)$ be a collection of dependent random indicator variables, where each one denotes the occurrence of an event, with $p_\alpha = E[X_\alpha]$. Let $\mathbf{W} = \sum_{\alpha \in I} \mathbf{X}_\alpha$ be the number of occurrences, and let $\mathbf{Z}$ be a Poisson random variable with $E[\mathbf{Z}] = E[\mathbf{W}] = \lambda < \infty$. Then the total variation distance between $\mathbf{Z}$ and $\mathbf{W}$ satisfies:*

$$\|\mathcal{L}(\mathbf{Z}) - \mathcal{L}(\mathbf{W})\| \leq 2(b_1 + b_2 + b_3)$$

where

$$b_1 = \sum_{\alpha \in I} \sum_{\beta \in B_\alpha} p_\alpha p_\beta$$

$$b_2 = \sum_{\alpha \in I} \sum_{\beta \in B_\alpha} E[\mathbf{X}_\alpha \mathbf{X}_\beta]$$

$$b_3 = \sum_{\alpha \in I} s_\alpha$$

with

$$\begin{aligned} s_\alpha &= E\left| E\left[\mathbf{X}_\alpha - p_\alpha \,\middle|\, \sum_{\beta \in I - B_\alpha} \mathbf{X}_\beta \right] \right| \\ &\leq \sum_{J \subseteq I - B_\alpha} |E\left[\mathbf{X}_\alpha - p_\alpha \,|\, E_J\right]| \, P(E_J) \end{aligned}$$

where E_J is the event where the random variables outside the neighborhood assume the value 1 *if and only if their index is in J:*

$$E_J = (\mathbf{X}_\beta = 1 \ \forall \beta \in J) \wedge (\mathbf{X}_\beta = 0 \ \forall \beta \in I - B_\alpha - J)$$

Note that when the choice of B_α is such that $\mathbf{X}_\alpha$ is independent from $\{X_\beta : \beta \notin B_\alpha\}$, we have $b_3 = 0$. When possible, we try to use this result in order to have a simple analysis, but in other cases choosing a smaller neighborhood may be important in order to obtain smaller errors.

Chapter 3

Probability distribution of the number of occurrences of a pattern

This chapter deals with the calculation of the probability that a certain pattern occurs in a random string at least a certain number of times (which will be called q-occurrence probability), which is the most common way of evaluating the statistical significance of a frequent pattern. The exact calculation for solid blocks can rely on dynamic programming or other techniques, such as generating functions.

The problem of calculating the q-occurrence probability in independent models for solid blocks is defined as follows:

Problem statement Let $\mathbf{s}$ be a random string of length l generated by an independent random model with probability vector $p = (p_A, p_C, p_G, p_T)$. Let $x \in \Sigma^k$, where $k \leq l$, and $q \in \mathbb{N}$. Calculate

$$h_x(q) = \Pr(N(x, \mathbf{s}) \geq q) = \Pr(x \text{ occurs at least } q \text{ times in } \mathbf{s})$$

3.1 Finite Markov Chain Imbedding approach

In order to compare the exact probability with the approximated results, we implemented a simple algorithm based on dynamic programming that can calculate the q-occurrence probability for a word of length k to appear at least q times in $O(kql)$ time and $O(kq)$ space, when independent models are used and $|\Sigma|$ is constant. Additionally, we show that a small reinterpretation of the algorithm can reduce the dependence on the length of the text, yielding $O((kq)^3 \log l)$ time complexity and $O((kq)^2)$ space complexity.

Both of these methods are known in literature; the core approach is called Finite Markov Chain Imbedding (FMCI) [3, 4, 12], which consists on mapping the values i of a random variable $\mathbf{X}$ to a set of states C_i of a Markov chain $\{\mathbf{Y}_1, \dots \mathbf{Y}_n\}$, such that $\Pr(\mathbf{X} = i) = \Pr(\mathbf{Y}_n \in C_i)$.

Our procedure has been developed independently. We start from an initial, simple consideration regarding the relation between the probability which we have to estimate and the evolution of a deterministic finite automaton (DFA) that recognizes q occurrences of the pattern in a text. We will use this consideration to formulate a recursive formula for the calculation of the probability, obtaining a dynamic programming algorithm with $O(kql)$ time complexity.

Subsequently, we notice the similarities between a DFA and a Markov chain under some conditions. These similarities will bring to the second algorithm, whose time complexity is $O((kq)^3 \log l)$.

Initial considerations We now consider the following function, for $r \geq 0$, $0 \leq i \leq l$, and $0 \leq j \leq k-1$:

$$\begin{aligned} h_x(i,(r,j)) = \Pr\{&x \text{ occurs at least } r \text{ times in } \mathbf{s}[l-i-j \dots l-1] \\ &\mid \mathbf{s}[l-i-j \dots l-i-1] = x[0 \dots j-1] \\ &\wedge \mathbf{s}[l-i-j' \dots l-i-1] \neq x[0 \dots j'-1] \quad \forall j' > j,\ j' < k\} \end{aligned}$$

This function is the probability for x to complete r occurrences in the last i unknown characters of $\mathbf{s}$, when j is the maximum number of previous characters that match the first j characters of x, except x itself. Note that $h_x(q) = h_x(l,(q,0))$.

The idea at the base of this function is to study the evolution of a deterministic finite automaton $A = (Q, \Sigma, \delta, q_0, F)$ that recognizes the language $\mathcal{L} = \{s \in \Sigma^* : N(x,s) \geq q\}$. It is possible to construct such an automaton by defining the set of states $Q = \{0, \dots, q\} \times \{0, \dots, |x|-1\}$. Each state $(r,j) \in Q$ indicates that the automaton has already recognized $q-r$ occurrences of x, while j is the current number of matched characters of x, which is the size of the longest suffix of the past input that is also a prefix of x. The set of final states is $F = \{(0,j)\ \forall j \in \{0, \dots, |x|-1\}\}$, while the transition function can be precomputed in $O(qk|\Sigma|)$ time, by extending the Knuth-Morris-Pratt algorithm.

The KMP algorithm employs a "partial match" table (also called failure function), which can be seen as a succint representation of a finite-state automaton that recognizes at least one occurrence of the pattern in the text. From this table, it is simple to build an explicit automaton that recognizes q occurrences. A graphical example is shown in Figure 3.1.

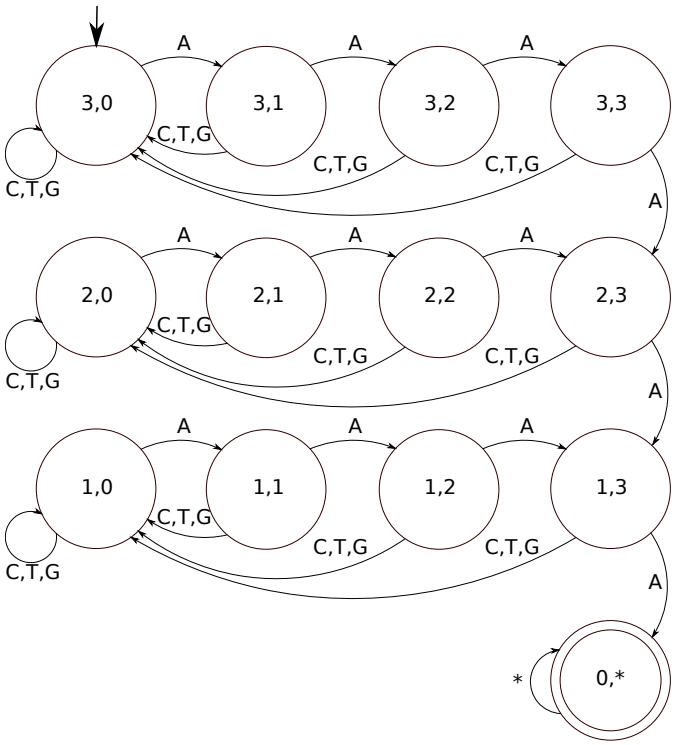

FIGURE 3.1: A DFA recognizing texts that contain the pattern "AAAA" three times. The initial state is $(3, 0)$, while the final states have been condensed in a single absorbing state $(0, *)$. Note that after an occurrence of the pattern, the automaton transitions to a state that has already recognized 3 characters of the pattern.

Dynamic programming First of all, we show that we can give a recursive definition of $h_x(i, (r, j))$, and use the limited number of subproblems to develop a simple algorithm for the q-occurrence probability.

Theorem 3.1.

$$h_x(i,(r,j)) = \begin{cases} 1 & \text{if } r = 0 \\ 0 & \text{if } (r > 0) \ \wedge \ (i = 0) \\ \sum_{c \in \Sigma} p_c \cdot h_x(i-1, \delta((r,j),c)) & \text{if } (r > 0) \ \wedge \ (i > 0) \end{cases}$$

where

$$\delta((r,j),c) = \begin{cases} (r, j+1) & \text{if } c = x[j] \ \wedge \ j < k-1 \\ (r-1, LP_x(x[1 \ldots k-1])) & \text{if } c = x[j] \ \wedge \ j = k-1 \\ (r, LP_x(x[1 \ldots j-1]c)) & \text{if } c \neq x[j] \end{cases}$$

and $LP_a(b)$ is the length of the longest prefix of a that is a suffix of b:

$$LP_a(b) = \max\{\hat{j} \leq |b| : a[0 \ldots \hat{j}-1] = b[|b| - \hat{j} \ldots |b|-1]\}$$

Proof. The pattern x always occurs at least 0 times in any string in Σ^*, thus $h(i,(0,j)) = 1\ \forall i,j$. We prove the rest of the recurrence by induction on the values of i.

Base case When $i = 0$ and $r > 0$, we cannot complete any occurrence of x with 0 characters, so $h(0,(r,j)) = 0\ \forall r > 0$.

Recursion Suppose that the recurrence holds for $i' \leq i-1$. We use the total probability theorem to decompose the definition of $h_x(i,(r,j))$:

$$h_x(i,(r,j)) = \sum_{c\in\Sigma} p_c E_{i,(r,j),c}$$

Where

$$\begin{aligned} E_{i,(r,j),c} = \Pr\{&x \text{ occurs at least } r \text{ times in } \mathbf{s}[l-i-j\ldots l-1] \\ &\mid \mathbf{s}[l-i-j\ldots l-i-1] = x[0\ldots j-1] \\ &\wedge\ \mathbf{s}[l-i-j'\ldots l-i-1] \neq x[0\ldots j'-1] \quad \forall j' > j,\ j' < k \\ &\wedge\ \mathbf{s}[l-i] = c\} \end{aligned}$$

We now study the probability $E_{i,(r,j),c}$.

Case I When $c \neq x[i]$, we have that x cannot occur in position $l-i-j$. The next candidate occurrence position for x is the first position after $l-i-j$ where x matches the currently determined characters of $\mathbf{s}$, or $l-i+1$ if x does not match any of the determined characters. The condition of $E_{i,(r,j),c}$ fixes $\mathbf{s}[l-i-j\ldots l-i] = x[0\ldots j-1]c$ and that there are no partial matches that are greater than j. We calculate the next candidate occurrence by getting the largest possible match for x, which is the largest value of $\hat{j} \leq j$ such that $\mathbf{s}[l-i-\hat{j}+1\ldots l-i] = x[0\ldots\hat{j}-1]$. This is the definition of $LP_x(x[1\ldots j-1]c)$. Thus

$$\begin{aligned} E_{i,(r,j),c} = \Pr\Big\{&x \text{ occurs at least } r \text{ times in } \mathbf{s}[l-i-\hat{j}+1\ldots l-1] \\ &\mid \mathbf{s}[l-i-\hat{j}+1\ldots l-i] = x[0\ldots\hat{j}-1] \\ &\wedge\ \mathbf{s}[l-i-j'+1\ldots l-i] \neq x[0\ldots j'-1] \quad \forall j' > \hat{j},\ j' < k\Big\} \\ = h_x(i-1,(r,\hat{j})) &= h_x(i-1,\delta((r,j),c)) \end{aligned}$$

Case II When $c = x[i]$ and $j < k-1$, we do not have an occurrence of x in $l-i-j$ yet, but the pattern may still occur in this position. We have

$$\begin{aligned} E_{i,(r,j),c} &= \Pr\{x \text{ occurs at least } r \text{ times in } \mathbf{s}[l-i-j \dots l-1] \\ &\qquad \mid \mathbf{s}[l-i-j \dots l-i] = x[0 \dots j-1]x[j] \\ &\qquad\quad \wedge \mathbf{s}[l-i-j' \dots l-i] \neq x[0 \dots j'-1]x[j'] \quad \forall j' > j,\ j' < k\} \\ &= h_x(i-1,(r,j+1)) = h_x(i-1,\delta((r,j),c)) \end{aligned}$$

Case III When $c = x[i]$ and $j = k-1$, then x occurs at position $l-i-j$. We need to count $r-1$ more occurrences of x in the next positions:

$$\begin{aligned} E_{i,(r,j),c} &= \Pr\{x \text{ occurs at least } r-1 \text{ times in } \mathbf{s}[l-i-j+1 \dots l-1] \\ &\qquad \mid \mathbf{s}[l-i-j \dots l-i] = x\} \\ &= \Pr\{x \text{ occurs at least } r-1 \text{ times in } \mathbf{s}[l-(i-1)-j \dots l-1] \\ &\qquad \mid \mathbf{s}[l-(i-1)-(j+1) \dots l-(i-1)-1] = x\} \end{aligned}$$

We determine a new candidate position by getting the largest value of $\hat{j} < j+1$ such that $\mathbf{s}[l-(i-1)-\hat{j} \dots l-(i-1)-1] = x[0 \dots \hat{j}-1]$. This is equivalent to $\hat{j} = LP_x(\mathbf{s}[l-(i-1)-(j+1)+1 \dots l-(i-1)-1]) = LP_x(x[1 \dots (j+1)-1]) = LP_x(x[1 \dots k-1])$, where we removed the first known character of $\mathbf{s}$ as its position corresponds to the previously counted occurrence of x. Thus

$$\begin{aligned} E_{i,(r,j),c} &= \Pr\Big\{x \text{ occurs at least } r-1 \text{ times in } \mathbf{s}[l-(i-1)-\hat{j} \dots l-1] \\ &\qquad \mid \mathbf{s}[l-(i-1)-\hat{j} \dots l-(i-1)-1] = x[0 \dots \hat{j}-1] \\ &\qquad\quad \wedge \mathbf{s}[l-(i-1)-j' \dots l-(i-1)-1] \neq x[0 \dots j'-1] \quad \forall j' > \hat{j},\ j' < k\Big\} \\ &= h_x(i-1,(r-1,\hat{j})) = h_x(i-1,\delta((r,j),c)) \end{aligned}$$

In conclusion, for $r > 0$ and $i > 0$, we have

$$h_x(i,(r,j)) = \sum_{c \in \Sigma} p_c E_{i,(r,j),c} = \sum_{c \in \Sigma} p_c h_x(i-1,\delta((r,j),c))$$

□

Algorithm 1 Dynamic programming algorithm for the calculation of the exact probability of q-occurrence.

Input: $\langle \Sigma, p, l, x, q \rangle$, where $p \in [0,1]^{|\Sigma|}$, $\sum_{c \in \Sigma} p_c = 1$, $l \in \mathbb{N}$, $x \in \Sigma^*$, $q \in \mathbb{N}, q \leq l - |x| + 1$
Output: $h_x(q) = \Pr(x \text{ occurs at least } q \text{ times in } \mathbf{s})$

```
{Initialize the subproblem matrix}
for r ∈ {1, ..., q} do
  for j ∈ {0, ..., k − 1} do
    M(0, r, j) ← 0
  end for
end for
for i ∈ {0, ..., l} do
  for j ∈ {0, ..., k − 1} do
    M(i, 0, j) ← 1
  end for
end for
{i-major scan of the subproblems}
for i ∈ {1, ..., l} do
  for j ∈ {0, ..., k − 1} do
    for r ∈ {1, ..., q} do
      M(i, r, j) ← 0
      for c ∈ Σ do
        M(i, r, j) ← M(i, r, j) + p_c M(i, δ(r, j))
      end for
    end for
  end for
end for
return M(l, q, 0)
```

The correctness of this algorithm comes from the previous theorem. The "i-major" scan of the algorithm guarantees that each value is computed only when all the values needed by the recursive definition are already computed. The transition function δ can be precomputed in $O(|Q||\Sigma|) = O(qk|\Sigma|)$ time by using the KMP algorithm. The time complexity of the algorithm is dominated by the four nested loops that solve each subproblem. Thus, the complexity of this algorithm is $O(lqk|\Sigma|)$.

The space required by the matrix is $O(lqk)$, but it can be reduced to $O(qk)$ by keeping only the current row and the previous row of M, precisely the values of $M(\hat{i}, r, j)$ with $\hat{i} \in i - 1, i$.

Transition matrix algorithm When the input characters are i.i.d random variables with a known distribution, the DFA can be mapped to a Markov chain, with $G =$

$\{g_1, \ldots, g_{|G|}\} = Q$ as the set of states, y as the initial probability vector with

$$y_i = \begin{cases} 1 & \text{if } g_i = (q, 0) \\ 0 & \text{otherwise} \end{cases}$$

and a transition matrix $D \in \mathbb{R}^{|Q|\times|Q|}$, with

$$D_{ij} = \sum_{c\in\Sigma:\delta(g_i,c)=g_j} p_c$$

We can calculate the final probability for each state after l transition easily:

$$\Pr(X_l = q_i) = (pT^l)_i$$

Thus, we can implement an alternative algorithm with a time complexity that has a limited dependence on the length of the sequence.

Algorithm 2 FMCI algorithm for the calculation of the exact probability of q-occurrence.

Input: $\langle \Sigma, p, l, x, q \rangle$, where $p \in [0,1]^{|\Sigma|}$, $\sum_{c\in\Sigma} p_c = 1$, $l \in \mathbb{N}$, $x \in \Sigma^*$, $q \in \mathbb{N}, q \leq l-|x|+1$
Output: $h_x(q) = \Pr(x$ occurs at least q times in $\mathbf{s})$

```
for r1 ∈ {1, ..., q} do
  for j1 ∈ {0, ..., k − 1} do
    v_(r1,j1) ← 0
    for r2 ∈ {1, ..., q} do
      for j2 ∈ {0, ..., k − 1} do
        T_(r1,j1),(r2,j2) ← 0
      end for
    end for
    for c ∈ Σ do
      (r2, j2) ← δ((r1, j1), c)
      T_(r1,j1),(r2,j2) ← T_(r1,j1),(r2,j2) + p_c
    end for
  end for
end for
v_(q,0) ← 1
a ← vT^l
return ∑_j a_(0,j)
```

The construction of the transition matrix takes $O((qk)^2 + qk|\Sigma|)$ time, while most of the time is spent in the calculation of the power of this matrix. If we use a naïve algorithm

for the multiplication, with a divide-and-conquer algorithm for the exponentiation, we obtain a time complexity of $O((qk)^3 \log l)$.

Markov models and wildcards Both algorithms can be extended in the Markov random model case. Suppose that m is the order of the Markov random model. The DFA that recognizes q occurrences of the pattern is still the same, but in order to determine the transition probability from one state to another, we need to know the previous m characters. If $j < m$, the state of the DFA does not contain this information. In order to use the same procedure, we need to add more states in the DFA such that each state determines the last m characters. We can substitute each state (r, j) with a new state described by the triplet (r, j, y), with $y \in \Sigma^{m-j}$, such that $y \cdot x[0 \dots j-1]$ describes the last m characters of the chain. The main issue is that we need a large number of states to complete the DFA:

$$|Q| = (k-m)(q+1) + \sum_{j=0}^{m-1} (q+1)|\Sigma|^{m-j} \geq (k-m)(q+1) + (q+1)|\Sigma|^m$$

Thus, the number of states and the time complexity of the algorithm is exponential in m. Furthermore, the transition function δ must be redesigned so that it adds the necessary information in the state. In practice, these algorithms (especially the transition matrix power algorithm) become inefficient unless m is very small, as with 1-order Markov chains.

We have the same issue if we try to consider patterns that have wildcards. In this case, the DFA described above must be adapted in order to keep the characters that match a wildcard in the pattern, and the function LP_x that finds the new candidate position must receive these characters in input. We also lose the $O(k)$ time complexity guaranteed by the KMP algorithm for the calculation of LP_x.

3.2 Related works

This problem is not new in literature, and other approaches have been proposed, in order to provide exact calculation even for patterns with wildcards or with Markov chain random models. One approach is given in [25], which considers the number of occurrences as a sum of random variables that indicate the occurrence of the pattern in a certain position. The probability of having at least q occurrences is then calculated by using the inclusion-exclusion principle, by considering all the possible subsets of indicator variables. The subset are grouped by their size a and their first obligatory occurrence position b. The sum of all the probabilities for this group is defined as $P(a, b)$. The authors show that, for specific categories of patterns, the calculation of $P(a, b)$ (hence, the result) can be done through dynamic programming, with $O(l^3)$ time complexity.

Other works use the mathematical properties of probability generating functions. In [14] the authors analyze the language containing the texts with exactly q occurrences of a set of solid patterns, and decompose the language into smaller, basic languages. These basic languages are associated to probability generating functions with special properties, which are employed to obtain the mean number of occurrences for each pattern and the covariance matrix for the pairs of patterns. The authors also consider the q-occurrence probability for a single pattern, and for solid patterns in an independent random model they provide a method to calculate the coefficients of a linear recurrence relation of degree kq on the q-occurrence probability in texts of length l. Then, the l-th value of the recurrence can be obtained by rewriting the recurrence relation in terms of matrix multiplication and calculating the l-th power of the recurrence matrix, yielding a $O((kq)^3 \log l)$ time complexity. This solution is more complex, and it is trickier to use with Markov models.

In [5], the authors use the probability generating functions to obtain a recurrence relation for the probability distribution of $N_x(\mathbf{s})$ for independent equiprobable models, for which they need only the length of s, the number of occurrences q and the *overlap capability* of the solid pattern, which is a vector of positions on which a pattern can overlap with itself. From the recurrence relation, they obtain an algorithm that calculates the probability distribution from 0 to q in $O(kql)$ time and $O(kq)$ space. This method is limited to independent equiprobable models, and it calculates only the probability distribution; the CDF is calculated by summing each term of the probability distribution from 0 to q.

Chapter 4

Poisson approximation for the number of occurrences

Our first goal is to determine the error bound on the Poisson approximation of the distribution of the number of occurrences of a given pattern. These results can be used to calculate an approximation of the q-occurrence probability of a pattern, and for some families of patterns under some conditions, they can provide a reasonable approximation more efficiently than the exact algorithms. Furthermore, when independent models are used, we can show that the expressions for the approximate q-occurrence probability and the error bound can be used to estimate the number of frequent patterns, which will be analytically estimated in the following chapter.

4.1 Preliminary definitions

In order to apply the Chen-Stein theorem, we define the number of occurrences as

$$N(x, s) = \sum_{i=0}^{l-k} \xi_s(x, i)$$

where $\xi_s(x, i) = 1$ when x occurs in s at position i.

Let $\mathbf{s}$ be a random process as one of those described in section 2.3. We define $\mathbf{N}_x = N(x, \mathbf{s})$ which is a random variable obtained from the sum of $l - k + 1$ random indicator variables:

$$\mathbf{N}_x = \sum_{i=0}^{l-k} \mathbf{H}_{x,i} \qquad \mathbf{H}_{x,i} = \xi_{\mathbf{s}}(x, i)$$

Generally, these variables are dependent from each other, but we will show that in most cases the indicator variables share the same mean, and the number of dependent pairs is limited.

4.2 Independent equiprobable case

In this section, we assume that the random model for $\mathbf{s}$ is an independent equiprobable random model, with $l = |\mathbf{s}|$. We are given a pattern x of length k, with $1 \leq k < l$, and we want to calculate $E[\mathbf{N}_x]$ and a bound on the error on the approximation of its distribution to a Poisson random variable.

The calculation of the probability of x to occur in the text in any position is straightforward, thanks to the independence of each position:

$$\begin{aligned} E[\mathbf{H}_{x,i}] &= P(\mathbf{s}[i, i+k-1] = x[0, k-1]) \\ &= P(\mathbf{s}[i] = x[0])P(\mathbf{s}[i+1] = x[1]) \dots P(\mathbf{s}[i+k-1] = x[k-1]) \\ &= \frac{1}{4^k} \end{aligned}$$

Consequently, the mean number of occurrences is easy to obtain through the linearity of the expectation:

$$\mu_x = E[\mathbf{N}(x)] = \sum_{i=0}^{l-k} E[\mathbf{H}_{x,i}] = \sum_{i=0}^{l-k} \frac{1}{4^k} = \frac{l-k+1}{4^k}$$

Note that in this situation, the mean is the same for all the patterns of length k regardless of their structure, so we can omit the subscript: $\mu_x = \mu \ \forall x \in \Sigma^k$.

We define the index set as $I = \{0, \dots, l-k\}$ and the neighborhood set as $B_i = \{j \in I : |j-i| < k\} \ \forall i \in I$. It is easy to show that the neighborhood set includes the indexes of all the variables that are dependent from $\mathbf{H}_{x,i}$.

Theorem 4.1. *$\mathbf{H}_{x,i}$ is independent from $\{\mathbf{H}_{x,j} : j \in I - B_i\}$.*

Proof. As any position is independent from the others, we just have to show that the set of occurrences described by $\{\mathbf{H}_{x,j} : j \in I - B_i\}$ does not share any position with the occurrence of x in i. Suppose that $j < i$: for the definition of B_i, we have $j \leq i-k$, and $j+k-1 < i$. The joint probability of the event $\mathbf{H}_{x,j} = 1$ and $\mathbf{H}_{x,i=1}$ is

$$\begin{aligned} P(\mathbf{H}_{x,j} = 1 \wedge \mathbf{H}_{x,i} = 1) &= P(s[j, j+k-1] = x \wedge s[i, i+k-1] = x) \\ &= P(s[j, j+k-1] = x)P(s[i, i+k-1] = x) \\ &= P(\mathbf{H}_{x,j} = 1)P(\mathbf{H}_{x,i} = 1) \end{aligned}$$

The same approach holds when $j > i$. □

Thanks to the previous theorem, we already know from Theorem 2.14 that $b_3 = 0$. We now give some bounds for b_1 and b_2.

Theorem 4.2.

$$b_1 \leq \frac{2k-1}{l-k+1}\mu^2$$

Proof. We simply have to apply the formula for b_1, approximate the results by ignoring the reduced number of neighbors for indices near the beginning or the end of the text, and apply Theorem 4.1:

$$\begin{aligned} b_1 &= \sum_{i=0}^{l-k} \sum_{j \in B_i} E[\mathbf{H}_{x,i}]E[\mathbf{H}_{x,j}] = \sum_{i=0}^{l-k} \sum_{j \in B_i} \frac{1}{4^{2k}} \\ &\leq \sum_{i=0}^{l-k} \sum_{j=i-k+1}^{i+k-1} \frac{1}{4^{2k}} = \sum_{i=0}^{l-k} \frac{2k-1}{4^{2k}} \\ &= \frac{(l-k+1)(2k-1)}{4^{2k}} = \frac{2k-1}{l-k+1}\mu^2 \end{aligned}$$

□

Our bound on b_1 is sharp when $l \gg k$, which is usually the case in the discovery of frequent patterns.

The estimation of b_2 requires to evaluate the joint probability of positions where the pattern may overlap with itself if it occurs in both positions. This event can occur only if x can partially overlap with itself, that is when a suffix of x corresponds to its own prefix.

Definition 4.3. A pattern $x \in \Sigma^k$ is *self-overlapping with distance* $0 < d < k$, or equivalently, *periodic with period* d if:

$$x[d \dots k-1] = x[0 \dots k-d-1]$$

We define the periodicity indicator function as:

$$\varepsilon_x(d) = \begin{cases} 1 & \text{if } x \text{ is periodic with period } d \\ 0 & \text{otherwise} \end{cases}$$

The periodicity indicator function lets us give a compact form for the joint probability of $\mathbf{H}_{x,i}\mathbf{H}_{x,j}$:

Theorem 4.4.

$$E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] = \begin{cases} \frac{1}{4^{2k}} & \text{if } |j-i| \geq k \\ \varepsilon_x(|j-i|)\frac{1}{4^{k+|j-i|}} & \text{otherwise} \end{cases}$$

Proof. The case $|j-i| \geq k$ can be obtained directly from the independence of $\mathbf{H}_{x,i}$ and $\mathbf{H}_{x,j}$. When $d = |j-i| < k$, we have that the two events are overlapping. Let $j < i$; we have that $j < i < j+k < i+k$. Between i and $j+k-1$, the two occurrences overlap, and we need to verify the following conditions for the event to occur:

$$\begin{aligned} s[i \dots j+k-1] = s[j+d \dots j+k-1] &= x[d \dots k-1] \\ = s[i \dots i+k-d-1] &= x[0 \dots k-d-1] \\ \implies x[d \dots k-1] &= x[0 \dots k-d-1] \end{aligned}$$

This means that $E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] = 0$ when $\varepsilon_x(|j-i|) = 0$. If the pattern is periodic with period d, we have:

$$\begin{aligned} E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] &= P(s[j \dots j+k-1] = x \ \wedge \ s[i \dots i+k-1] = x) \\ &= P(s[j \dots i-1] = x[0 \dots d-1] \ \wedge \ x[i \dots i+k-1] = x) \\ &= \frac{1}{4^d}\frac{1}{4^k} = \frac{1}{4^{k+d}} \end{aligned}$$

The same result holds when $j > i$. □

We can now calculate a bound on the value of b_2.

Theorem 4.5.

$$b_2 \leq 2\mu_x \sum_{d=1}^{k-1} \varepsilon_x(d)\frac{1}{4^d}$$

Proof. We simply apply Theorem 4.4 in the formula of b_2, following the same procedure we used for the calculation of b_1:

$$\begin{aligned} b_2 &= \sum_{i \in I} \sum_{j \in B_i/\{i\}} E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] \\ &= \sum_{i \in I} \sum_{j \in B_i/\{i\}} \varepsilon_x(|j-i|)\frac{1}{4^{k+|j-i|}} \\ &\leq \sum_{i \in I} \frac{2}{4^k} \sum_{d=1}^{k-1} \varepsilon_x(d)\frac{1}{4^d} \\ &= 2\frac{l-k+1}{4^k} \sum_{d=1}^{k-1} \varepsilon_x(d)\frac{1}{4^d} \\ &= 2\mu \sum_{d=1}^{k-1} \varepsilon_x(d)\frac{1}{4^d} \end{aligned}$$

Pattern	Average	Error $(b_1 + b_2)$
`AAAAAAAAAA`	0.476829	0.317893
`AATAATAATA`	0.476829	0.015146
`AAAAAAAAAT`	0.476829	$8.64004 \cdot 10^{-6}$
`AAAAAAAAAAAAAAA`	0.000465648	0.000310432
`AATAATAATAATAAT`	0.000465648	$1.47825 \cdot 10^{-5}$
`AAAAAAAAAAAAAAT`	0.000465648	$1.25764 \cdot 10^{-11}$
`AAAAAAAAAAAAAAAAAAAA`	$4.5473 \cdot 10^{-7}$	$3.03153 \cdot 10^{-7}$
`AATAATAATAATAATAATAA`	$4.5473 \cdot 10^{-7}$	$1.44359 \cdot 10^{-8}$
`AAAAAAAAAAAAAAAAAAAT`	$4.5473 \cdot 10^{-7}$	$1.61294 \cdot 10^{-17}$

TABLE 4.1: Expected number of occurrences and approximation error bounds for the occurrence probability of some patterns. The random text of length $l = 500000$ is generated in an independent equiprobable model.

□

The value of b_2 depends heavily on the values of the periodicity function ε_x. Its minimum value is 0, which occurs when the pattern is aperiodic, while it assumes its maximum value when the pattern has period 1, in the degenerate case $x[0] = x[1] = \cdots = x[k-1]$, where:

$$b_2 \leq 2\mu \sum_{d=1}^{k-1} \frac{1}{4^d} = 2\mu \frac{(1/4) - (1/4^k)}{1 - 1/4} < \frac{2}{3}\mu$$

In conclusion, we obtained that in the independent equiprobable case all the patterns have the same mean $\mu = (l - k + 1)/4^k$, while the total Poisson approximation error bound $b_1 + b_2$ lies between $\frac{2k-1}{l-k+1}\mu^2$ and $\frac{2}{3}\mu + \frac{2k-1}{l-k+1}\mu^2$. From this result, we can infer that the error bound is small when the mean is sufficiently low (much less than 1), or for aperiodic patterns when $l \gg 2k\mu^2$, or equivalently when $l \ll 4^{2k}/(2k)$, thus the patterns must be adequate with respect to the length of the text.

Some values for the expected number of occurrences and the error bounds on the approximation are shown in Table 4.1, calculated by using the previous theorems. The results confirm that the quality of the error bound degrades for patterns with a small period.

A logarithmic-scale comparison between the Poisson approximation and the exact probability can be found in Figure 4.1. The figure shows the probability of occurrence for three different patterns with different periodicity, in an independent equiprobable model. We can see that the probabilities tend to diverge significantly when the patterns have a small period compared to the length of the pattern. However, the error bound given in Table 4.1 largely overestimates the error in the complementary cumulative distribution function.

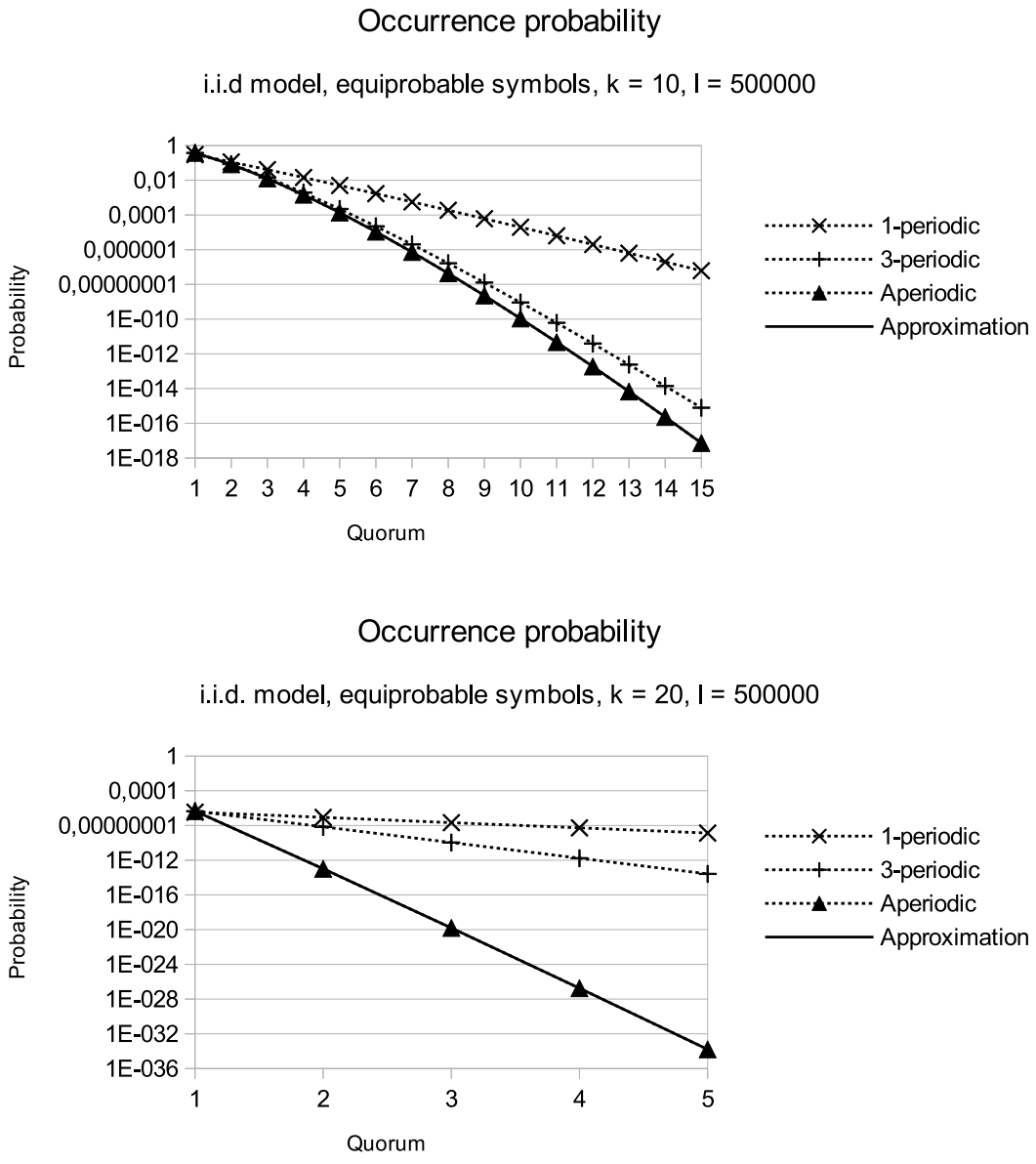

FIGURE 4.1: Logarithmic scale comparison between the Poisson approximation for the occurrence probability and the exact probabilities of a 1-periodic pattern, a 3-periodic pattern and an aperiodic pattern of length $k = 10$ and $k = 20$, in a text of length $l = 500000$ generated with an independent equiprobable random model.

4.3 Independent non-equiprobable case

We adapt the previous results for independent random processes with character probability $p_e : e \in \Sigma$. First of all, we define the probability of an occurrence in position i:

$$\begin{aligned} E[\mathbf{H}_{x,i}] &= P(\mathbf{s}[i] = x[0])P(\mathbf{s}[i+1] = x[1]) \dots P(\mathbf{s}[i+k-1] = x[k-1]) \\ &= p_{x[0]}p_{x[1]} \cdots p_{x[k-1]} \end{aligned}$$

The probability is independent from i, so we define $p_x = p_{x[0]}p_{x[1]} \cdots p_{x[k-1]}$. The mean number of occurrences can be calculated as usual:

$$\mu_x = E[\mathbf{N}_x] = \sum_{i=0}^{l-k} E[\mathbf{H}_{x,i}] = \sum_{i=0}^{l-k} p_x = (l-k+1)p_x$$

We define $I = \{0, \dots, l-k\}$ and $B_i = \{j \in I : |j-i| < k\}\ \forall i \in I$, as before. Theorem 4.1 still holds, thus $b_3 = 0$. We can easily adapt the calculation of b_1 and b_2:

Theorem 4.6.

$$b_1 \leq \frac{2k-1}{l-k+1}\mu_x^2$$

$$b_2 \leq 2\mu_x \sum_{d=1}^{k-1} \varepsilon_x(d)\, p_{x[0\dots d-1]}$$

Proof. The calculation of b_1 is now straightforward:

$$\begin{aligned} b_1 &= \sum_{i=0}^{l-k} \sum_{j \in B_i} E[\mathbf{H}_{x,i}]E[\mathbf{H}_{x,j}] = \sum_{i=0}^{l-k} \sum_{j \in B_i} p_x^2 \\ &\leq (l-k+1)(2k-1)p_x^2 = \frac{2k-1}{l-k+1}\mu_x^2 \end{aligned}$$

For the calculation of b_2:

$$\begin{aligned} b_2 &= \sum_{i \in I} \sum_{j \in B_i/\{i\}} \varepsilon_x(d)\, p_x\, p_{x[0\dots d-1]} \\ &\leq \sum_{i \in I} 2 \sum_{d=1}^{k-1} \varepsilon_x(d)\, p_x\, p_{x[0\dots d-1]} \\ &= 2p_x(l-k+1) \sum_{d=1}^{k-1} \varepsilon_x(d)\, p_{x[0\dots d-1]} \\ &= 2\mu_x \sum_{d=1}^{k-1} \varepsilon_x(d)\, p_{x[0\dots d-1]} \end{aligned}$$

□

In this case, the value of b_2 depends both on the periodicity of x and the characters it is composed of. The worst case is when $x[0] = x[1] = \cdots = x[k-1] = \arg\max\{p_e : e \in \Sigma\}$. If we define $p_{max} = \max\{p_e : e \in \Sigma\}$, we can give a general bound to the value of b_2:

$$b_2 \leq 2\mu_x \sum_{d=1}^{k-1} p_{max}^d = 2\mu_x \frac{p_{max} - p_{max}^k}{1 - p_{max}} < \frac{2p_{max}}{1 - p_{max}} \mu_x$$

4.4 1-order Markov chain

With 1-order Markov chains, we lose the benefit of independence between positions, but if the transition matrix has certain properties, then we can use a limited neighborhood while keeping the value of b_3 under control.

We assume that the random model is a 1-order Markov chain, with transition matrix T, and that the Markov chain has a stationary vector π. We also assume that the initial probability vector p is equal to the stationary vector. With these conditions, we have that the probability of occurrence in a certain position is the same for all the positions:

$$\begin{aligned}
E[\mathbf{H}_{x,i}] &= p_x = P(s[i \dots i+k-1] = x) \\
&= P(s[i] = x[0]) \prod_{j=1}^{k-1} P(s[i+j] = x[j] \mid s[i \dots i+j-1] = x[0 \dots j-1]) \\
&= P(s[i] = x[0]) \prod_{j=1}^{k-1} P(s[i+j] = x[j] \mid s[i+j-1] = x[j-1]) \\
&= (p \cdot T^i)_{x[0]} \prod_{j=1}^{k-1} T_{x[j-1],x[j]} \\
&= \pi_{x[0]} \prod_{j=1}^{k-1} T_{x[j-1],x[j]}
\end{aligned}$$

We can obtain the mean number of occurrences as usual:

$$\mu_x = E[\mathbf{N}(x)] = (l - k + 1)p_x$$

In general, the joint probability of occurence for two patterns can be calculated as follows:

Theorem 4.7.

$$E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] = \begin{cases} p_x & \text{if } i = j \\ \varepsilon_x(d)\, p_x T_{x[d-1],x[0]} \frac{p_{x[0\dots d-1]}}{\pi_{x[0]}} & \text{if } d = |j-i| < k,\ i \neq j \\ \frac{(p_x)^2}{\pi_{x[0]}} (T^{d-k+1})_{x[k-1],x[0]} & \text{if } d = |j-i| \geq k \end{cases}$$

Proof. The first case, where $i = j$, is trivial. Now suppose that $j < i$, without loss of generality. In the second case, with $0 < d = |j - i| < k$, the two occurrences of x are overlapping. Thus, the two events can happen simultaneosly only when $\varepsilon_x(d) = 1$. In this case, we can see that x occurs in position i and j if and only if $x[0 \dots d-1] \cdot x$ occurs in position j. The probability of this event is:

$$\begin{aligned} p_{x[0\dots d-1]\cdot x} &= \pi_{x[0]} \left(\prod_{r=1}^{d-1} T_{x[r-1],x[r]} \right) T_{x[d-1],x[0]} \prod_{r=1}^{k-1} T_{x[r-1],x[r]} \\ &= p_{x[0\dots d-1]} T_{x[d-1],x[0]} \prod_{r=1}^{k} T_{x[r-1],x[r]} \\ &= p_{x[0\dots d-1]} T_{x[d-1],x[0]} \frac{p_x}{\pi_{x[0]}} \end{aligned}$$

In the third case, the occurrences do not overlap, and as we assumed that $T(e, f) > 0 \ \forall e, f \in \Sigma$, this event has a nonzero probability of occurrence. Assuming that $j < i$, there are $d - k = i - j - k \geq 0$ characters between the two occurrences in the text, so there are $d - k + 1$ transitions between the last character of the first occurrence and the first character of the second occurrence (between $s[j + k - 1]$ and $s[i]$). The first occurrence influences the probability of the first character of the second occurrence; the probability of this event is:

$$\begin{aligned} E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] &= p_x \left[(T^{d-k+1})_{x[k-1],x[0]} \right] \prod_{r=1}^{k} T_{x[r-1],x[r]} \\ &= p_x \left[(T^{d-k+1})_{x[k-1],x[0]} \right] \frac{p_x}{\pi_{x[0]}} \\ &= \frac{(p_x)^2}{\pi_{x[0]}} \left[(T^{d-k+1})_{x[k-1],x[0]} \right] \end{aligned}$$

□

In the third case, we have that $\mathbf{H}_{x,i}$ is dependent on $\mathbf{H}_{x,j}$, unless we have $(T^{d-k+1})_{x[k-1],x[0]} = \pi_{x[0]}$. However, with some Markov chains, we can assume that the transition probability converges quickly to the stationary probability. We can use this to limit the size of the neighborhood set and keep b_3 under control.

We now choose a value of $\varphi \in (0, 1)$, and we find the minimum value c such that the transition probabilities after c steps differ from the stationary probability by no more than φ. First of all, we show that further steps will still satisfy the condition.

Theorem 4.8. *Let* $0 < T_{ij} < 1 \ \forall i, j$, $\varphi \in (0, 1)$ *and* $c = \min\{c' : |(T^{c'})_{ij} - \pi_j| < \varphi\}$. *Then, for any* $c' > c$*:*

$$|(T^{c'})_{ij} - \pi_j| < \varphi$$

Proof. We prove the upper bound $(T^{c'})_{ij} < \pi_j + \varphi$, by splitting the power of the transition matrix in two factors, $T^{c'-c}T^c$ then we bound the values for T^c and exploit the stochasticity of the transition matrix, which implies that the rows of any power of T must sum to 1:

$$\begin{aligned}(T^{c'})_{ij} &= (T^{c'-c}\ T^c)_{ij} = \sum_k (T^{c'-c})_{ik}\ (T^c)_{kj} \\ &\leq \sum_k (T^{c'-c})_{ik}(\pi_j + \varphi) \\ &= (\pi_j + \varphi)\sum_k (T^{c'-c})_{ik} = \pi_j + \varphi\end{aligned}$$

The same approach holds for the lower bound. □

We now define $B_i = \{j \in I : |j - i| \leq k + c\}$. We adapt the definition of b_3 to our case:

$$b_3 = \sum_{i=0}^{l-k} s_i$$

with

$$\begin{aligned}s_i &= E\left|E\left[\mathbf{H}_{x,i} - p_x \,\middle|\, \sum_{j\in I-B_i} \mathbf{H}_{x,j}\right]\right| \\ &\leq \sum_{J\subseteq I-B_i} |E\left[\mathbf{H}_{x,i} - p_x \,|\, E_J\right]|\, P(E_J)\end{aligned}$$

Where E_J is the event where each random variable outside the neighborhood has value 1 if and only if its index is in J. We now show that it is possible to approximate each term of the summation, regardless of E_J, in order to obtain a bound on b_3. Through this bound, we obtain the following theorem:

Theorem 4.9. *Let $b_3^{max} \in (0, 1)$. If we choose φ such that:*

$$0 < \varphi \leq \frac{b_3^{max}}{3|\Sigma|^k} \min\{\pi_b : b \in \Sigma\}$$

and define the neighborhood as $B_i = \{j \in I : |j - i| \leq k + c\}$, with $c = \min\{c' : |(T^{c'})_{ij} - \pi_j| < \varphi\}$, then $b_3 \leq b_3^{max}$.

Proof. From the hypothesis, we note that there exists a value of $R > 0$ such that:

$$\varphi \leq R\pi_b\ \forall b \in \Sigma$$

Let $i \in I$. In order to evaluate s_i, we split the event E_J in two events: $E_J = E_{JL} \wedge E_{JR}$, where

$$E_{JL} = (\mathbf{H}_{x,\beta} = 1\ \forall \beta \in J : \beta < i) \wedge (\mathbf{H}_{x,\beta} = 0\ \forall \beta \in I - B_\alpha - J : \beta < i)$$

$$E_{JR} = (\mathbf{H}_{x,\beta} = 1 \ \forall \beta \in J : \beta > i) \wedge (\mathbf{H}_{x,\beta} = 0 \ \forall \beta \in I - B_\alpha - J : \beta > i)$$

Essentially, E_{JL} and E_{JR} represent the conditioning occurrence events whose positions are lower or higher than i, respectively. We only consider the indices i such that neither E_{JL} nor E_{JR} are empty; the results can be easily extended to the other cases.

The purpose is to calculate the conditioned probabilty as:

$$\Pr(\mathbf{H}_{x,i} = 1 | E_J) = \frac{\Pr(\mathbf{H}_{x,i} = 1 \wedge E_J)}{\Pr(E_J)}$$

and give an upper bound on the conditional probability by finding an upper bound for the numerator and a lower bound for the denominator; to do this, we calculate these probabilities by considering each event in order of position; to avoid considering each possible configuration of E_{JL} and E_{JR}, we determine the last position that may be "directly dependent" on E_{JL}, and the first such position in E_{JR}. As we defined the neighborhood set as $B_i = \{j \in I : |j - i| \leq k + c\}$, the positions are $\gamma_1 = (i - k - c - 1) + k - 1 = i - c - 2$, and $\gamma_2 = i + k + c + 1$, respectively. Subsequently, we apply the law of total probability.

The joint probability can be expressed as follows:

$$\begin{aligned}\Pr(\mathbf{H}_{x,i} = 1 \wedge E_J) = \sum_{a\in\Sigma}\sum_{b\in\Sigma} &\Pr(E_{JL}) \cdot \Pr(s[\gamma_1] = a \mid E_{JL}) \\ &\cdot \Pr(\mathbf{H}_{x,i} = 1 \mid (s[\gamma_1] = a) \wedge E_{JL}) \\ &\cdot \Pr(s[\gamma_2] = b \mid \mathbf{H}_{x,i} = 1 \wedge (s[\gamma_1] = a) \wedge E_{JL}) \\ &\cdot \Pr(E_{JR} \mid s[\gamma_2] = b \wedge \mathbf{H}_{x,i} = 1 \wedge (s[\gamma_1] = a) \wedge E_{JL})\end{aligned}$$

We now use the Markov property:

$$\begin{aligned}\Pr(H_{x,i} = 1 \wedge E_J) = \sum_{a\in\Sigma}\sum_{b\in\Sigma} &\Pr(E_{JL}) \cdot \Pr(s[\gamma_1] = a | E_{JL}) \\ &\cdot \Pr(\mathbf{H}_{x,i} = 1 | s[\gamma_1] = a) \\ &\cdot \Pr(s[\gamma_2] = b | \mathbf{H}_{x,i} = 1) \\ &\cdot \Pr(E_{JR} | s[\gamma_2] = b)\end{aligned}$$

We now give some upper bounds to the terms that do not depend on E_J:

$$\Pr(\mathbf{H}_{x,i} = 1 | s[\gamma_1] = a) = (T^{i-\gamma_1})_{a,x[0]} \frac{p_x}{\pi_{x[0]}} \leq \frac{p_x}{\pi_{x[0]}}(\pi_{x[0]} + \varphi) = p_x + \frac{\varphi}{\pi_{x[0]}}$$

$$\begin{aligned}\Pr(s[\gamma_2] = b | \mathbf{H}_{x,i} = 1) &= \Pr(s[\gamma_2] = b \mid s[i + k - 1] = x[k - 1]) \\ &= (T^{\gamma_2 - (i+k-1)})_{x[k-1],b} \leq \pi_b + \varphi\end{aligned}$$

Finally, we obtain:

$$\begin{aligned}
\Pr(H_{x,i}=1\wedge E_J) &\leq \left[p_x+\frac{\varphi}{\pi_{x[0]}}\right]\sum_{a\in\Sigma}\sum_{b\in\Sigma}\Pr(E_{JL})\cdot\Pr(x[\gamma_1]=a\mid E_{JL})\\
&\quad\cdot(\pi_b+\varphi)\cdot\Pr(E_{JR}\mid s[\gamma_2]=b)\\
&=\left[p_x+\frac{\varphi}{\pi_{x[0]}}\right]\Pr(E_{JL})\sum_{b\in\Sigma}(\pi_b+\varphi)\cdot\Pr(E_{JR}\mid s[\gamma_2]=b)\\
&\quad\cdot\sum_{a\in\Sigma}\Pr(x[\gamma_1]=a\mid E_{JL})\\
&=\left[p_x+\frac{\varphi}{\pi_{x[0]}}\right]\Pr(E_{JL})\sum_{b\in\Sigma}(\pi_b+\varphi)\cdot\Pr(E_{JR}\mid s[\gamma_2]=b)\\
&\leq\left[p_x+\frac{\varphi}{\pi_{x[0]}}\right]\Pr(E_{JL})\left[(1+R)\Pr(E_{JR})\right]
\end{aligned}$$

We now need to give an upper bound to the denominator:

$$\begin{aligned}
\Pr(E_J) &= \sum_{a\in\Sigma}\sum_{b\in\Sigma}\Pr(E_{JL})\cdot\Pr(x[\gamma_1]=a\mid E_{JL})\\
&\quad\cdot(T^{\gamma_2-\gamma_1})_{a,b}\cdot\Pr(E_{JR}\mid s[\gamma_2]=b)\\
&\geq\sum_{a\in\Sigma}\sum_{b\in\Sigma}\Pr(E_{JL})\cdot\Pr(x[\gamma_1]=a\mid E_{JL})\\
&\quad\cdot(\pi_b-\varphi)\cdot\Pr(E_{JR}\mid s[\gamma_2]=b)\\
&=\Pr(E_{JL})\sum_{b\in\Sigma}(\pi_b-\varphi)\cdot\Pr(E_{JR}|s[\gamma_2]=b)\\
&\geq\Pr(E_{Jl})\left[(1-R)\Pr(E_{JR})\right]
\end{aligned}$$

Finally we can obtain the upper bound on the conditioned probability:

$$\begin{aligned}
\Pr(H_{x,i}=1|E_J)=\frac{\Pr(H_{x,i}=1\wedge E_J)}{\Pr(E_J)} &\leq\left[p_x+\frac{\varphi}{\pi_{x[0]}}\right]\frac{1+R}{1-R}\leq(p_x+R)\frac{1+R}{1-R}\\
&=p_x+p_x\frac{2R}{1-R}+R\frac{1+R}{1-R}
\end{aligned}$$

We can obtain a lower bound with the same procedure:

$$\begin{aligned}
\Pr(H_{x,i}=1|E_J)=\frac{\Pr(H_{x,i}=1\wedge E_J)}{\Pr(E_J)} &\geq\left[p_x-\frac{\varphi}{\pi_{x[0]}}\right]\frac{1-R}{1+R}\\
&\geq(p_x-R)\frac{1-R}{1+R}\\
&=p_x-p_x\frac{2R}{1+R}-R\frac{1-R}{1+R}
\end{aligned}$$

Thus, the absolute value for the conditional expectation is at most:

$$\begin{aligned}|E[H_{x,i} - p_x|E_J]| &= |\Pr(H_{x,i} = 1|E_J) - p_x| \\ &\leq \max\left\{p_x\frac{2R}{1-R} + R\frac{1+R}{1-R},\ p_x\frac{2R}{1+R} + R\frac{1-R}{1+R}\right\} \\ &\leq \frac{R}{1-R}(2p_x + 1 + R)\end{aligned}$$

When R is sufficiently small, we get the approximate bound:

$$|E[H_{x,i} - p_x|E_J]| \leq R(2p_x + 1)$$

This bound does not depend on E_J, thus:

$$b_3 = \sum_{\alpha\in I} s_\alpha \leq \sum_{x\in\Sigma^k} R(2p_x + 1) = |\Sigma|^k R + 2R\sum_{x\in\Sigma^k} p_x \leq 3R|\Sigma|^k$$

Thus, if we want to limit b_3 to a specified maximum value b_3^{max}, we can impose:

$$\varphi \leq \frac{b_3^{max}}{3|\Sigma|^k}\min\{\pi_b : b \in \Sigma\}$$

□

4.5 Extension to structured motifs

The Chen-Stein method is flexible, and can also be applied to patterns that contain wildcards. Its applicability is only limited to how the pattern may self-overlap.

In order to maintain the tractability of the problem, we consider a small family of motifs in the form $w = w_1 \circ^t w_2$, where the solid patterns w_1, w_2 are separated by t wildcard characters, and $|w_1| = |w_2| = m \leq t$ (thus $|w| = k = t + 2m$).

The probability of occurence is still the same for each position. With independent random models, we get:

$$p_w = p_{w_1}p_{w_2}$$

And $p_w = 1/4^{2m}$ for the equiprobable random model. These probabilities are equivalent to the probabilities obtained by the solid pattern $w' = w_1w_2$. with a With 1-order Markov chains, the wildcards introduce a slight difference:

$$p_w = p_{w_1}(T^{t+1})_{w_1[m-1],w_2[0]}\frac{p_{w_2}}{\pi_{w_2[0]}}$$

When t is sufficiently large, the probability converges to $p_{w_1}p_{w_2}$. The mean number of occurrences follows the usual expression: $\mu_w = (l-k+1)p_w$.

Generally, two occurrences i and j of the same motif are said to be overlapping when $|j-i| < k$, and overlapping occurrences are usually dependent on each other. However, we can see that the pairs of occurrences i, j are independent when each character of any pattern overlaps only with wildcards of the other pattern. The particular form of the pattern w lets us define a smaller neighborhood set:

$$B_i = \{j \in i : |j-i| < m \vee t < |j-i| < k\}$$

We now present some bounds for b_1 and b_2 in independent models. The calculation of b_1 is straightforward:

$$b_1 = \sum_{\alpha\in I}\sum_{\beta\in B_\alpha} p_w^2 \le (l-k+1)(6m-3)\cdot p_w^2 = \frac{(6m-3)}{l-k+1}\mu_w^2 \le \frac{2k-1}{l-k+1}\mu_w^2$$

We can observe that b_1 is less than the error bound for a solid pattern x of the same length as w if their expectations were the same. However, the average number of occurrences of a solid pattern of length k is generally smaller than the average number of occurrences of a motif with the same length.

The error bound b_2 can be expressed as:

$$b_2 = \sum_{\alpha\in I}\sum_{\beta\in B_\alpha-\{\alpha\}} E[\mathbf{H}_{w,i}\mathbf{H}_{w,j}]$$

where

$$E[\mathbf{H}_{w,i}\mathbf{H}_{w,j}] = \begin{cases} p_x & \text{if } i = j \\ \varepsilon_w(d)\, p_w p_{w_1[0\ldots d-1]} p_{w_2[0\ldots d-1]} & \text{if } d = |j-i| < m,\ i \neq j \\ \varepsilon_w(d) p_w p_{w_2} p_{w_1[0\ldots m+t-d-1]} & \text{if } d = |j-i|,\ t < d \le m+t \\ \varepsilon_w(d) p_w p_{w_2} p_{w_1[k-d\ldots m-1]} & \text{if } d = |j-i|,\ m+t < d < k \end{cases}$$

With the periodicity indicator function defined as:

$$\varepsilon_w(d) = \begin{cases} 1 & \text{if } (w[i] = w[i+d]) \vee (w[i] = \circ) \vee (w[i+d] = \circ), \quad \forall\ 0 \le i \le k-d-1 \\ 0 & \text{otherwise} \end{cases}$$

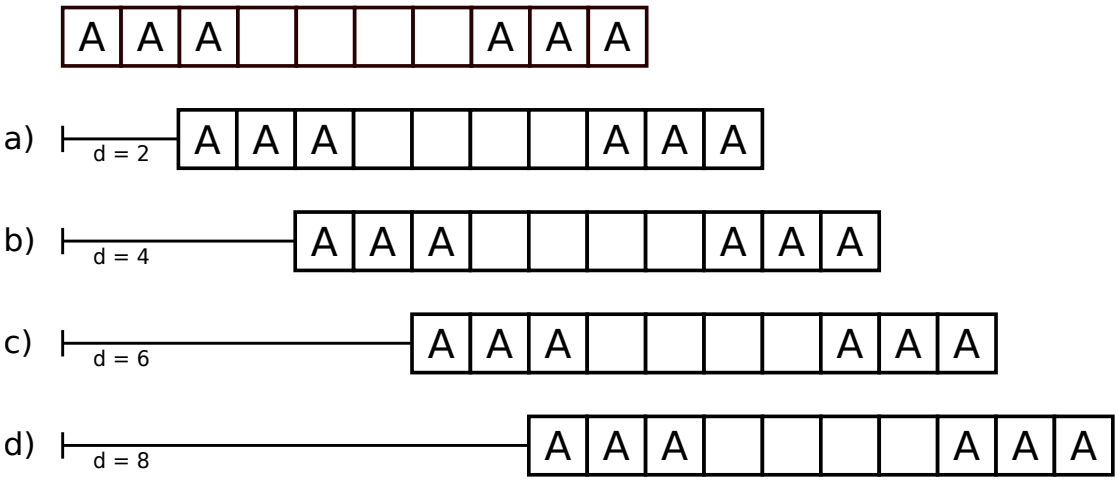

FIGURE 4.2: Graphical representation of various possible overlaps. *a*) Both solid words w_1 and w_2 overlap. *b*) No solid word overlaps. In independent models, these events are independent. *c*) The right side of w_1 overlaps with the left side of w_2. *d*) The left side of w_1 overlaps with the right side of w_2.

By replacing each term of b_2 with its definition, ignoring the reduced number of neighbors for indices near the ends of the text:

$$\begin{aligned} b_2 &= \sum_{\alpha\in I}\sum_{\beta\in B_\alpha-\{\alpha\}} E[\mathbf{H}_{x,i}\mathbf{H}_{x,j}] \\ &\le 2(l-k+1)\left[\sum_{d=1}^{m-1}\varepsilon_x(d)\,p_w p_{w_1[0\ldots d-1]}p_{w_2[0\ldots d-1]}\right. \\ &\quad + \sum_{d=t+1}^{m+t}\varepsilon_x(d)p_w p_{w_2}p_{w_1[0\ldots m+t-d-1]} \\ &\quad \left. + \sum_{d=m+t+1}^{k-1}\varepsilon_x(d)p_w p_{w_2}p_{w_1[k-d\ldots m-1]}\right] \end{aligned}$$

The expression can be simplified in the independent equiprobable model:

$$b_2 \le 2(l-k+1)\left[\sum_{d=1}^{m-1}\frac{\varepsilon_x(d)}{4^{2m+2d}} + \sum_{d=t+1}^{m+t}\frac{\varepsilon_x(d)}{4^{3m+(m+t-d)}} + \sum_{d=m+t+1}^{k-1}\frac{\varepsilon_x(d)}{4^{3m+(m+d-k)}}\right]$$

For those patterns that always overlap, such as $A^m \circ^t A^m$, in the independent equiprobable model, we get:

$$\begin{aligned}
b_2 &\leq 2(l-k+1)\left[\sum_{d=1}^{m-1}\frac{1}{4^{2m+2d}} + \sum_{d=t+1}^{m+t}\frac{1}{4^{3m+(m+t-d)}} + \sum_{d=m+t+1}^{k-1}\frac{1}{4^{3m+(m+d-k)}}\right] \\
&= 2(l-k+1)\left[\frac{1}{4^{2m}}\sum_{d=1}^{m-1}\frac{1}{4^{2d}} + \frac{1}{4^{3m}}\sum_{d=0}^{m-1}\frac{1}{4^{d}} + \frac{1}{4^{3m}}\sum_{d=1}^{m-1}\frac{1}{4^{d}}\right] \\
&\leq 2(l-k+1)\left[\frac{1}{4^{2m}}\frac{1}{15} + \frac{1}{4^{3m}}\left(1+\frac{2}{3}\right)\right] \\
&= 2(l-k+1)p_w\left[\frac{1}{15} + \frac{5}{3}\sqrt{p_w}\right] \\
&= \mu_w\left[\frac{2}{15} + \frac{10}{3}\sqrt{p_w}\right]
\end{aligned}$$

In this case, the error bound has slightly improved from the solid pattern case, but only approximately by a constant factor. As for b_1, the value of b_2 depends on the average number of occurrences, thus we expect that the error bound is higher than the bound for a solid pattern with the same length.

4.6 Related works

The problem of studying the probability distribution of the number of occurrences of a pattern has been extensively studied. In [20], the authors briefly analyze the Poisson approximation for the number of occurrences of a pattern in Markovian models. The authors focus on the asymptotic analysis of the error bound: they suppose that the neighborhood consists of the positions for which there are less than c characters between the occurrences, and they show that $b_1 \leq (l-k+1)(2c+2k-3)p_x^2$, while b_2 depends on the overlap capability and in the worst case $b_2 = O(lkp_x)$. In their asymptotic framework with $|x| = k = \Theta(\log n)$, $n \to \infty$, the asymptotic bound for b_2 does not converge to 0.

This issue is addressed by evaluating the overlapping occurrences separately. The authors define a *clump* as a set of overlapping occurrences. The start of the clump is an occurrence of the pattern that does not overlap with any previous occurrence. If Y_i denotes that a new clump starts in position i, it is easy to show that if $|j-i| < k$, then Y_i and Y_j are mutually exclusive. Thus, the authors show that the number of clumps can be approximated to a Poisson distribution, with an error of $O(lkp_x^2)$ in independent models. In order to obtain the number of occurrences instead of the number of clumps, the authors must use the compound Poisson process approximation [1]; for this method, they need to characterize each clump with its length, thus defining a set of indicator variables $Z_{i,r}$ that indicate that a clump of r occurrences starts at position i. For some simple situations shown in [1] and [24], in independent models and when the pattern has a single principal period (that is, x is periodic only with period d and its multiples), it is

shown that the number of occurrences can be approximated to the sum of **N** geometric random variables, where **N** is a random variable with a Poisson distribution.

In [20] the general case is considered, including Markovian models and patterns with more than one principal period. In [15] the method is extended to sets of m words of various lengths. This extension can be applied to obtain an approximation for the number of occurrences of a motif.

We used the Poisson approximation directly in our evaluation; the quality of the approximation depends on the number of occurrences and the overlap capability of the pattern, but the procedure is simpler and the results we obtained lead to an elegant approximation for the average number of frequent patterns in independent models, which is shown in the next chapter.

Chapter 5

Poisson approximation for the number of frequent patterns

In this chapter we analyze the number of frequent patterns of a given length k in a random text of given length l. First of all, we justify our interest in the number of frequent patterns through an example in the discovery of frequent patterns in a text. Subsequently, we try to obtain some of the values that are required by the procedure described in 1.3, namely an approximation for the mean and an approximation for the complementary CDF for the number of frequent patterns. For the approximation of the mean, we use the results obtained in the previous chapter; for the complementary CDF, we try to apply the Chen-Stein theorem to the number of frequent patterns, and develop a simulation that allows us to test the error bounds for any neighborhood. Finally, we devise a statistical test to compare the empirical distribution obtained from the simulation to the Poisson distribution, in order to validate the results given by the Chen-Stein method and evaluate the possibility of further expansions.

5.1 Rationale and example

The work related to the q-occurrence probability of a single pattern is often used in statistical tests to determine whether the observed pattern frequency is significant. However, as we noted in chapter 1, a frequent pattern may appear significant with a single hypothesis test, even when the expected number of frequent patterns of the same kind is high.

For example, in an independent random model with $l = 10^6$ and $k = 10$, the average number of occurrences for any pattern x of length k is $\mu = (10^6 - 9)/(4^{10}) \approx 1$. For aperiodic patterns, the error in the Poisson approximation is $b_1 \leq (2k-1)\mu^2/(l-k+1) \approx$

$2 \cdot 10^{-5}$. Thus, the probability for one of those patterns to have at least 7 occurrences is at most $P(\text{Poisson}(1) \geq 7) + b_1 \approx 1.032 \cdot 10^{-4}$.

We might then be tempted to consider all the aperiodic patterns that occur at least 7 times in this text to be statistically significant if we set a significance level $\alpha = 0.05$ and we test each frequent pattern "a posteriori". Unfortunately, we can show that there are at least $\frac{2}{3} \cdot 4^k$ aperiodic patterns, and that the expected number of frequent aperiodic patterns is at least $\frac{2}{3} \cdot 4^k \cdot (P(\text{Poisson}(1) \geq 7) - b_1) \approx 44$.

Reducing the significance level through a Bonferroni correction, if we suppose we are testing $m = \frac{2}{3} \cdot 4^k$ patterns, would give a significance level of $\alpha/m \approx 7.15 \cdot 10^{-8}$, which would be insufficient to mark any pattern that appears 7 times as statistically significant, regardless of how many frequent patterns occur. The Benjamini and Yekutieli procedure for limiting the FDR, on the other hand, would sort the patterns in increasing order of p-value: as some of the frequent patterns may occur more than 7 times, the procedure would start considering these patterns as statistically significant, and increase its threshold in order to increase the power of the tests while maintaining a chosen FDR.

In section 1.3 we also introduced a new method that can determine a quorum value such that all the frequent patterns are statistically significant within a chosen FDR. For this method, we need the mean number of frequent patterns and the p-value for the observed number of frequent patterns. This drives our search for an approximation for $E[\mathbf{Q}_{k,q}]$ and $\Pr(\mathbf{Q}_{k,q} \geq n)$.

5.2 Approximation for the mean

Let $\mathbf{Q}_{k,q}$ be a random variable corresponding to the number of patterns of length k that occur at least q times in a text. We define $\mathbf{X}_{x,q}$ as a random indicator variable, such that $E[\mathbf{X}_{x,q}] = P(\mathbf{X}_{x,q} = 1) = P(\mathbf{N}_x \geq q)$. Then $\mathbf{Q}_{k,q} = \sum_{x \in \Sigma^k} \mathbf{X}_{x,q}$.

We want to calculate the mean number of occurrences for the number of frequent patterns. We can use the results obtained from the Poisson approximation of the individual patterns to obtain upper and lower bounds for the average. For the upper bound we have:

$$\begin{aligned} E[\mathbf{Q}_{k,q}] &= \sum_{x \in \Sigma^k} E[\mathbf{X}_{x,q}] = \sum_{x \in \Sigma^k} P(\mathbf{N}_x \geq q) \\ &\leq \sum_{x \in \Sigma^k} [P(\text{Poisson}(\mu_x) \geq q) + b_1(x) + b_2(x)] \end{aligned}$$

The Poisson approximation gives an absolute error, so we can use it also to get a lower bound to the mean:

$$E[\mathbf{Q}_{k,q}] \geq \sum_{x \in \Sigma^k} [P(\text{Poisson}(\mu_x) \geq q) - b_1(x) - b_2(x)]$$

We can separate the three terms of the sum in order to obtain the approximated cumulative q-occurrence probabilities, and the cumulative errors b_1 and b_2. We define:

$$M = \sum_{x \in \Sigma^k} P(\text{Poisson}(\mu_x) \geq q)$$

and

$$R = R_1 + R_2, \quad R_1 = \sum_{x \in \Sigma^k} b_1(x), \; R_2 = \sum_{x \in \Sigma^k} b_2(x)$$

So that we can write in a more compact notation:

$$|E[\mathbf{Q}_{k,q}] - M| \leq R$$

5.2.1 Independent equiprobable case

We remind that, in the independent equiprobable case, we have $\mu = \mu_x = (l - k + 1)/4^k \quad \forall x \in \Sigma^k$. Thus, the frequency distribution of each pattern is approximated to the same Poisson distribution:

$$M = 4^k \Pr\left\{\text{Poisson}\left(\frac{l-k+1}{4^k}\right) \geq q\right\}$$

Even the values of b_1 are the same for all x, so we obtain:

$$R_1 = \sum_{x \in \Sigma^k} b_1(x) \leq 4^k \frac{2k-1}{l-k+1}\mu^2 = (2k-1)\mu$$

The values of b_2 depend on the periodicity of x. If we use the value of b_2 in the worst case, when x has period 1, we would obtain:

$$R_2 = \sum_{x \in \Sigma^k} b_2(x) < \frac{2}{3}\mu 4^k = \frac{2}{3}(l-k+1)$$

This cumulative error bound is $O(l)$ when k is constant, which is excessively large. We can get a better bound by substituting $b_2(x)$ and swapping the summations:

$$\begin{aligned}\sum_{x\in\Sigma^k} b_2(x) &\le \sum_{x\in\Sigma^k} 2\mu \sum_{d=1}^{k-1} \varepsilon_x(d)\frac{1}{4^d} \\ &= 2\mu \sum_{d=1}^{k-1} \sum_{x\in\Sigma^k} \varepsilon_x(d)\frac{1}{4^d} \\ &= 2\mu \sum_{d=1}^{k-1} \sum_{x\in\Sigma^k:\varepsilon_x(d)=1} \frac{1}{4^d} \\ &= 2\mu \sum_{d=1}^{k-1} \frac{1}{4^d} \left|\{x\in\Sigma^k : \varepsilon_x(d)=1\}\right|\end{aligned}$$

The size of the set in the last term is the number of patterns with period d. It is easy to obtain the following result:

Theorem 5.1. *The number of patterns of length k that are periodic with period d is $|\Sigma|^d$.*

Proof. From Definition 4.3, a pattern x is periodic with period d if

$$x[d\dots k-1] = x[0\dots k-d-1]$$

We can rewrite the equivalence as follows:

$$\begin{cases} x[d\dots 2d-1] = x[0\dots d-1] \\ x[2d\dots 3d-1] = x[2d-1\dots 3d-1] \\ \dots \\ x[\lfloor k/d\rfloor d\dots k-1] = x[(\lfloor k/d\rfloor -1)d\dots k-d-1] \end{cases}$$

We can easily substitute the right side of each equation:

$$\begin{cases} x[d\dots 2d-1] = x[0\dots d-1] \\ x[2d\dots 3d-1] = x[0\dots d-1] \\ \dots \\ x[\lfloor k/d\rfloor d\dots k-1] = x[0\dots (k \bmod d)-1] \end{cases}$$

Thus, all the characters of $x[d\dots k-1]$ depend only on $x[0\dots d-1]$, and all the characters of $x[0\dots d-1]$ are necessary for the determination of x. Thus there is a one-to-one correspondence between the set of patterns with period d and Σ^d, and the thesis follows immediately. □

By applying this result, we obtain:

$$R_2 = \sum_{x \in \Sigma^k} b_2(x) \leq 2\mu \sum_{d=1}^{k-1} \frac{1}{4^d} 4^d \leq 2(k-1)\mu$$

Thus, we can estimate the average number of frequent patterns within the cumulative error bound:

$$R_1 + R_2 = \sum_{x \in \Sigma^k} [b_1(x) + b_2(x)] \leq (4k-3)\mu$$

Note that R_2 is almost the same as R_1, which means this is asymptotically the best bound we can get from the Poisson approximations of the individual pattern, as b_1 is the same for all the patterns.

The error bound is independent from the quorum, which means that the theoretical bound does not scale well: as we increase the quorum, the average decreases (as its approximation M) while the bound is the same as the $q = 1$ case. If we want to have a relative error of at most ε, the parameters must satisfy:

$$\frac{(4k-3)\mu}{4^k \Pr\left[\text{Poisson}(\mu) \geq q\right]} = \frac{4k-3}{l-k+1} \frac{\mu^2}{\Pr\left[\text{Poisson}(\mu) \geq q\right]} \leq \varepsilon$$

This means that as the value of q is increased, we can only accurately estimate the average number of frequent patterns when such patterns are rare. This is a pessimistic bound: it is easy to see that the real and approximate q-occurrence probabilities for any pattern converge to 0 when q is increased, thus the error tends to 0. We will show in some experiments in the next chapter that the expression is often a good approximation of $E[\mathbf{Q}_{k,q}]$: in our experiments, the approximation is relatively close to the sample average, while it tends to underestimate the sample average as q is increased.

5.2.2 Independent non-equiprobable case

In this case, the average frequency of a pattern μ_x can assume different values for each pattern. However, all the permutations of x have the same μ_x. We can use this to reduce the number of q-occurrence probabilities to calculate.

We can partition the set of patterns in $O(k^3)$ classes, such that patterns in the same class are permutations of each other. We can identify each class with the tuple (k_A, k_C, k_G, k_T) which represents the number of occurrences of each symbol in x (thus, we have $k_A + k_C + k_G + k_T = k$ and $k_e \geq 0 \ \forall e \in \Sigma$).

The occurrence probability of a pattern in the permutation class (k_A, k_C, k_G, k_T) in a certain position is:

$$p_{(k_A,k_C,k_G,k_T)} = p_A^{k_A} p_C^{k_C} p_G^{k_G} p_T^{k_T}$$

Through this notation, we obtain the value of M and R_1. Both of these values can be computed by summing $O(k^3)$ terms.

$$\begin{aligned} M &= \sum_{x\in\Sigma^k} \Pr\{\text{Poisson}(\mu_x) \geq q\} \\ &= \sum_{k_A,k_C,k_G,k_T} \frac{k!}{k_A!k_C!k_G!k_T!} \Pr\left\{\text{Poisson}\left((l-k+1)p_{(k_A,k_C,k_G,k_T)}\right) \geq q\right\} \end{aligned}$$

$$\begin{aligned} R_1 &= \sum_{x\in\Sigma^k} b_1(x) \\ &\leq \sum_{x\in\Sigma^k} \frac{2k-1}{l-k+1}\mu_x^2 \\ &= (2k-1)(l-k+1) \sum_{k_A,k_C,k_G,k_T} \frac{k!}{k_A!k_C!k_G!k_T!} (p_{(k_A,k_C,k_G,k_T)})^2 \end{aligned}$$

The estimation of R_2 requires more work:

$$\begin{aligned} R_2 &= \sum_{x\in\Sigma^k} b_2(x) \\ &\leq \sum_{x\in\Sigma^k} 2\mu_x \sum_{d=1}^{k-1} \varepsilon_x(d) p_{x[0\ldots d-1]} \\ &= 2\sum_{d=1}^{k-1} \sum_{x\in\Sigma^k:\varepsilon_x(d)=1} \mu_x p_{x[0\ldots d-1]} \end{aligned}$$

The inner summation depends not only on the number of patterns with period d or on their permutation class, but when d does not divide k, it depends also on the order in which the characters appear in the first $(k \bmod d)$ positions. In fact:

$$\begin{aligned} R_2 &\leq 2\sum_{d=1}^{k-1} \sum_{x\in\Sigma^k:\varepsilon_x(d)=1} \mu_x p_{x[0\ldots d-1]} \\ &= 2(l-k+1)\sum_{d=1}^{k-1} \sum_{x\in\Sigma^k:\varepsilon_x(d)=1} p_{x[0\ldots d\lfloor k/d\rfloor-1]} \cdot p_{x[0\ldots(k \bmod d)-1]} \cdot p_{x[0\ldots d-1]} \\ &= 2(l-k+1)\sum_{d=1}^{k-1} \sum_{x\in\Sigma^k:\varepsilon_x(d)=1} [p_{x[0\ldots d-1]}]^{\lfloor k/d\rfloor+1} \cdot p_{x[0\ldots(k \bmod d)-1]} \end{aligned}$$

In order to calculate R_2 efficiently, we can obtain an upper bound either by ignoring the last factor completely, or by substituting it with the product of the $k \bmod d$ highest probabilities available for the current periodic sequence. In the first case, the formula becomes:

$$R_2 \leq 2(l-k+1)\sum_{d=1}^{k-1}\sum_{(d_A,d_C,d_G,d_T)}\frac{d!}{d_A!d_C!d_G!d_T!}[p_{x[0\ldots d-1]}]^{\lfloor k/d\rfloor+1}$$

Whether we choose to ignore the last factor, or we substitute it with the product of the highest available probabilities, the approximation of R_2 requires to sum $O(k^4)$ terms, instead of the $O(k^3)$ terms for the computation of M and R_1.

5.2.3 Extension to structured motifs

The expression for the bounds of b_1 and b_2 lead to an easy expression of M, R_1 and R_2 even for structured motifs of the form described in Section 4.5.

We define the language of all the structured motifs with solid block length m and with t wildcards:

$$\mathcal{L}_{m,t} = \Sigma^m \cdot \{\circ^t\} \cdot \Sigma^m = \{w_1 \circ^t w_2 : w_1, w_2 \in \Sigma^m\}$$

For independent equiprobable models, the approximated average is

$$M = 4^{2m}\Pr\{\text{Poisson}(\mu_w) \geq q\}$$

with $\mu = \frac{l-k+1}{4^{2m}}$.

The calculation of R_1 can exploit the independence as usual:

$$R_1 = \sum_{w\in\mathcal{L}_{m,t}} b_1(w) \leq 4^{2m}\frac{6m-1}{l-k+1}\mu_w^2 = (6m-1)\mu_w$$

The calculation of R_2, on the other hand, has to be decomposed in three families of possible overlaps:

$$\begin{aligned}
R_2 &= \sum_{w\in\mathcal{L}_{m,t}} b_2(x) \\
&\le \sum_{w\in\mathcal{L}_{m,t}} \frac{2(l-k+1)}{4^{2m}} \left[\sum_{d=1}^{m-1} \frac{\varepsilon_w(d)}{4^{2d}} + \sum_{d=t+1}^{m+t} \frac{\varepsilon_w(d)}{4^{m+(m+t-d)}} + \sum_{d=m+t+1}^{k-1} \frac{\varepsilon_w(d)}{4^{m+(m+d-k)}}\right] \\
&= \frac{2(l-k+1)}{4^{2m}} \left[\sum_{d=1}^{m-1} \sum_{w\in\mathcal{L}_{m,t}:\varepsilon_w(d)=1} \frac{1}{4^{2d}} \right. \\
&\quad \left. + \sum_{d=t+1}^{m+t} \sum_{w\in\mathcal{L}_{m,t}:\varepsilon_w(d)=1} \frac{1}{4^{m+(m+t-d)}} + \sum_{d=m+t+1}^{k-1} \sum_{w\in\mathcal{L}_{m,t}:\varepsilon_w(d)=1} \frac{1}{4^{m+(m+d-k)}}\right]
\end{aligned}$$

We now need to evaluate the number of patterns that satisfy $\varepsilon_w(d) = 1$ in the three terms.

Case $d < m$: Both w_1 and w_2 must partially overlap with themselves, thus w_1 and w_2 must be periodic with period d. There is no other constraint between them, thus the number of pairs with period d is 4^{2d}.

Case $t + 1 < d \le m + t$: In this case, a suffix of w_1 is overlapping with a prefix of w_2. Once we choose w_1, we only need to determine the remanining $m + t - d$ characters of w_2 that are not overlapped with w_1. Thus the number of patterns is $4^{m+(m+t-d)}$.

Case $m + t < d < k$: In this case, a prefix of w_1 is overlapping with a suffix of w_2. We follow the same procedure as the previous case, and we obtain that the number of patterns is $4^{m+(d-m-t)} = 4^{m+(m+d-k)}$.

$$\begin{aligned}
R_2 &\le \frac{2(l-k+1)}{4^{2m}} \left[\sum_{d=1}^{m-1} \frac{4^{2d}}{4^{2d}} + \sum_{d=t+1}^{m+t} \frac{4^{m+(m+t+d)}}{4^{m+(m+t-d)}} + \sum_{d=m+t+1}^{k-1} \frac{4^{m+(m+d-k)}}{4^{m+(m+d-k)}}\right] \\
&= \frac{2(l-k+1)}{4^{2m}} \left[m-1+m+m-1\right] = 2(3m-2)\mu_x
\end{aligned}$$

Thus, the total error bound is

$$R_1 + R_2 \le (12m-5)\mu_w = (12m-5)\frac{l-k+1}{4^{2m}}$$

5.3 Poissonicity of the number of frequent patterns

We illustrated some efficient ways to approximate the expected number of frequent patterns in a random text under some conditions. Even when these methods are not directly applicable, such as with Markov chain models where the order of the characters is very important, we can get an estimate through simulation. A simulation requires to generate a large amount of texts of the same size l by using the desired random model, and to extract the number of frequent patterns from all texts and aggregating the results.

Independently on how we obtained the expectation, we are also interested in checking whether the number of frequent patterns can be approximated by a Poisson process, and if so, which conditions the parameters must satisfy.

Let $\mathbf{Q}_{k,q} = \sum_{x \in \Sigma^k} \mathbf{X}_{x,q}$. We want to see if we can approximate it to a Poisson random variable with mean $\lambda = E[\mathbf{Q}_{k,q}]$, by applying the Chen-Stein method. Unfortunately, for any pattern pair x, y the events $\mathbf{X}_{x,q}$ and $\mathbf{X}_{y,q}$ are always potentially dependent, even when the patterns cannot overlap.

For example, suppose that $kq = O(l)$. When a pattern x is frequent, any other pattern y that does not overlap with x will have a very limited number of positions available, thus the probability that y is frequent conditioned to the event that x is frequent tends to reduce dramatically. Conversely, if y can be overlapped almost completely with x, such as when $y[0 \dots k-2] = x[1 \dots k-1]$, then y tends to have a much higher chance of being frequent when x is frequent.

If we apply the Chen-Stein method by using the neighborhood $B_\alpha = I$, we obtain:

$$b_1 = \sum_{x \in \Sigma^k} \sum_{y \in \Sigma^k} E[\mathbf{X}_{x,q}] E[\mathbf{X}_{y,q}] = \sum_{x \in \Sigma^k} E[\mathbf{X}_{x,q}] \sum_{y \in \Sigma^k} E[\mathbf{X}_{y,q}] = \lambda^2$$

$$\begin{aligned} b_2 &= \sum_{x \in \Sigma^k} \sum_{y \in \Sigma^k, y \neq x} E[\mathbf{X}_{x,q} \mathbf{X}_{y,q}] = E\left[\sum_{x \in \Sigma^k} \sum_{y \in \Sigma^k, y \neq x} \mathbf{X}_{x,q} \mathbf{X}_{y,q} \right] \\ &= E\left[\mathbf{Q}_{k,q}^2 - \sum_{x \in \Sigma^k} \mathbf{X}_{x,q}^2 \right] = E\left[\mathbf{Q}_{k,q}^2 - \sum_{x \in \Sigma^k} \mathbf{X}_{x,q} \right] \\ &= E\left[\mathbf{Q}_{k,q}^2 \right] - \lambda \end{aligned}$$

The error bound becomes:

$$b_1 + b_2 = \lambda^2 + E\left[\mathbf{Q}_{k,q}^2 \right] - \lambda = 2\lambda^2 + \mathrm{Var}(\mathbf{Q}_{k,q}) - \lambda$$

The bound is greater than 1 when $\lambda > 1$, even if the variance of $\mathbf{Q}_{k,q}$ is small. This essentially means that the bound is small only when $\mathbf{Q}_{k,q} > 0$ is a rare event, which restricts the application of the method to a limited number of situations.

5.3.1 Reduced Neighborhood Set

In order to obtain a better error bound, we need to find a neighborhood set such that the values of b_1 and b_2 decrease while b_3 remains small. Intuitively, the pairs of patterns that overlap with each other have a positive dependence: when one of the patterns is frequent, the other pattern may appear in positions that overlap with the occurrences of the first pattern with higher probability. Keeping these patterns outside their respective neighborhood may increase the value of b_3 substantially.

On the other hand, the pairs of patterns that do not overlap should have a negative correlation, as one pattern cannot appear in any position that is partially occupied by the occurrence of the other. However, when $k \cdot q \ll l$, the number of unavailable positions for a pattern when the other is frequent should be reasonably small, thus we expect that the probability does not change significantly and b_3 remains small.

For this reason, we try to use the reduced neighborhood:

$$B_x = \{y \in \Sigma^k : \ x \text{ and } y \text{ can overlap}\}$$

The first issue we encounter is that even with a reduced neighborhood, the neighborhood size is still large:

Theorem 5.2. *For $k > 2$, the number of neighbors of any pattern x is within the following bounds:*

$$\frac{7}{16}\left|\Sigma^k\right| \leq |B_x| \leq \frac{2}{3}\left|\Sigma^k\right|$$

Proof. We start by proving the lower bound. The neighborhood of x contains the following patterns:

$$B_x^{(1)} = \{y \in \Sigma^k : y[0] = x[k-1] \vee y[k-1] = x[0]\} \subset B_x$$

With simple set operations, we can calculate its size:

$$\begin{aligned} |B_x^{(1)}| &= |\{y \in \Sigma^k : y[0] = x[k-1]\}| + |\{y \in \Sigma^k y[k-1] = x[0]\}| + \\ &\quad - |\{y \in \Sigma^k : y[0] = x[k-1] \wedge y[k-1] = x[0]\}| \\ &= \frac{1}{2}\left|\Sigma^k\right| - \frac{1}{16}\left|\Sigma^k\right| = \frac{7}{16}\left|\Sigma^k\right| \end{aligned}$$

For the upper bound, note that B_x is the union of the sets of patterns y that can overlap with x by a certain number of positions:

$$\begin{aligned} B_x = &\bigcup_{d=1}^{k} \{y \in \Sigma^k : y[0, d-1] = x[k-d, k-1]\} \\ &\cup \bigcup_{d=1}^{k} \{y \in \Sigma^k : y[k-d, k-1] = x[0, d-1]\} \end{aligned}$$

For symmetry, both terms have the same size. Consequently:

$$\begin{aligned} |B_x| &\leq 2 \left| \bigcup_{d=1}^{k} \{y \in \Sigma^k : y[0, d-1] = x[k-d, k-1]\} \right| \\ &\leq 2 \sum_{d=1}^{k} \left| \{y \in \Sigma^k : y[0, d-1] = x[k-d, k-1]\} \right| \\ &= 2 \sum_{d=1}^{k} 4^{k-d} = 2 \sum_{i=0}^{k-1} 4^d = 2 \frac{4^k - 1}{4 - 1} \leq \frac{2}{3} \left| \Sigma^k \right| \end{aligned}$$

□

With a reduced neighborhood of this size, we do not expect big improvements on the value of b_1 and b_2. Furthermore, the calculation of b_1 and b_2 becomes difficult. In particular, we do not have any closed form expression for the probability that two generic patterns are frequent.

Another issue is that with this neighborhood, $b_3 > 0$. Finding a non-trivial upper bound for b_3 seems to be challenging. If we reconsider each term of b_3:

$$\begin{aligned} s_\alpha &= E \left| E \left[X_\alpha - p_\alpha \left| \sum_{\beta \in I - B_\alpha} X_\beta \right. \right] \right| \\ &= \sum_{i=0}^{|I-B_\alpha|} \left| E \left[X_\alpha - p_\alpha \left| \sum_{\beta \in I - B_\alpha} X_\beta = i \right. \right] \right| \cdot \Pr \left(\sum_{\beta \in I - B_\alpha} X_\beta = i \right) \\ &= \sum_{i=0}^{|I-B_\alpha|} \left| \Pr \left[X_\alpha = 1 \left| \sum_{\beta \in I - B_\alpha} X_\beta = i \right. \right] - p_\alpha \right| \cdot \Pr \left(\sum_{\beta \in I - B_\alpha} X_\beta = i \right) \end{aligned}$$

by analyzing the values s_α, we may need a bound to the probability that a certain number of patterns outside its neighborhood are frequent, and a bound to the conditional probability.

5.3.2 Simulation for determining the error bound

Due to the issues that arose in the calculation of the approximation error of $\mathbf{Q}_{k,q}$, we proceed to estimate these values in various cases by generating random sequences with the distribution given by the random models, and compare these values to a "goodness of fit" test, which gives a measure of confidence for the similarity between a theoretical distribution and an empirical distribution.

One of the issues of the simulation is that we might need a large number of sample sequences in order to get a reasonable confidence interval. However, with a suitable number of trials, we can still get an idea about when the distribution can be approximated to a Poisson distribution.

Our method follows the one proposed in [9]. However, in the frequent itemset case, the procedures that extract the frequent itemsets with a certain threshold may return an exponential number of itemsets (up to $\binom{n}{k}$ itemsets). In that context, it is required to fix an initial support value $\tilde{s}$ in order to limit the number of frequent itemsets.

In our context, there may still be a need to keep the number of frequent patterns to a reasonable level, but the number of frequent words can be at most polynomial in the input. In a collection $\mathcal{C}$ of Δ texts, each one of length l, there are $O(l\Delta/q)$ words of fixed length k that occur at least q times in one text.

Thus, if Δ is not too large, or if k is small, we can keep all the frequent patterns we encounter in memory. In order to insert and access the frequent patterns quickly, we use a *trie* to keep the patterns we encounter.

Estimation of b_1 We remind the expression of b_1:

$$b_1 = \sum_{x \in \Sigma^k} \sum_{y \in B_x} E[\mathbf{X}_{x,q}] E[\mathbf{X}_{y,q}]$$

For the estimation of b_1, we would need to estimate the probabilities of *all* the patterns, not only those that appear to be frequent. This is independent from the neighborhood. However, we will assume that those patterns that are not frequent in all the Δ trials have an estimated probability of 0. We define

$$W_q = \bigcup_{t \in \mathcal{C}} W_{q,t}, \qquad W_{q,t} = \{x \in \Sigma^k : x \text{ appears at least } q \text{ times in } t\}$$

for each of these patterns, we calculate the empirical frequency:

$$f_x = \frac{|\{t \in \mathcal{C} : x \text{ appears at least } q \text{ times in } t\}|}{\Delta}$$

The calculation of f_x for all the patterns can be done as follows. For each value of q that we want to analyze, we keep a global trie $\mathcal{T}_q$ that contains for each pattern the number of texts in which that pattern occurs q times. For each text t, we build a trie that contains all the k-words in t with their number of occurrences. Then, we increase the count in $\mathcal{T}_q$ of all the patterns that occur at least q times by scanning the current trie.

The value of b_1 is then estimated as

$$\hat{b}_1 = \sum_{x \in \Sigma^k \cap W_q} \sum_{y \in B_x \cap W_q} f_x f_y$$

Thus, we have to calculate at most $|W_q|^2$ products to determine $\hat{b}_1$. In the special case where the neighborhood is $B_x = \Sigma^k$, we showed that $b_1 = E^2[\mathbf{Q}_k, q]$, thus it can be estimated by the square of the average number of frequent patterns in the text.

Even if we choose $B_x = \{y \in \Sigma^k : x \text{ and } y \text{ can overlap}\}$, we showed that the number of neighbors is $|B_x| > \frac{7}{16}\Sigma^k$. In addition, if a pattern is frequent, we expect that its neighbors are more likely to occur than non-neighbor patterns that have the same unconditioned probability. Thus, it is reasonable to expect that $\hat{b}_1$ will not be much less than in the case where all pairs of patterns are neighbors. Thus, we can skip the calculation of b_1 when there are too many frequent patterns, as we expect it to be too large.

Estimation of b_2 We remind the expression of b_2:

$$b_2 = \sum_{x \in \Sigma^k} \sum_{y \in B_x - \{x\}} E[\mathbf{X}_{x,q} \mathbf{X}_{y,q}]$$

As with b_1, for the pairs of patterns where one of them is not frequent, their joint probabilities are estimated as 0. For the pairs of frequent patterns, we estimate the probability as:

$$f_{x,y} = \frac{|\{t \in \mathcal{C} : x \text{ and } y \text{ appear at least } q \text{ times in } t\}|}{\Delta}$$

Instead of repeating this search for each pair of neighboring patterns, we can simply count the number of frequent neighboring pairs in each text:

$$\begin{aligned}\hat{b}_2 &= \sum_{\alpha\in I}\sum_{\beta\in B_\alpha-\{\alpha\}} f_{x,y} \\ &= \frac{1}{\Delta}\sum_{\alpha\in I}\sum_{\beta\in B_\alpha-\{\alpha\}} |\{t\in\mathcal{C} : x \text{ and } y \text{ appear at least } q \text{ times in } t\}| \\ &= \frac{1}{\Delta}\sum_{t\in C}\sum_{\alpha\in I}\sum_{\beta\in B_\alpha-\{\alpha\}} \mathbf{1}(x \text{ and } y \text{ appear at least } q \text{ times in } t)\end{aligned}$$

This also means that, in order to obtain a value of $\hat{b}_2 < 1$, the average number of frequent neighboring pairs in each text must be less than 1 (or 0.5, if we count the unordered pairs).

A practical implication of this result is that we do not need to accumulate the frequent patterns in a global trie to calculate $\hat{b}_2$. For each text, we build a trie that contains all the k-words in t with their number of occurrences, then we remove all the patterns in the trie whose number of occurrences is less than q.

If the neighborhood set is $B_x = \Sigma^k$, we should simply count the number n of frequent patterns in the text, and the number of neighboring pairs (except the pairs that have the same pattern twice) is $n^2 - n$. Alternatively, we can use the mean and the variance directly, as reported at the beginning of this section.

Otherwise, for each pair we need to count the number of frequent patterns in its neighborhood. As with b_1, if we expect that the neighborhood of a pattern will probably contain roughly half the number of frequent patterns, we can skip the calculation of b_2 when there are too many frequent patterns.

Estimation of b_3 We remind the expression of b_3 and use the definition of expectation:

$$s_x = \sum_{i=0}^{|\Sigma^k-B_x|} \left| \Pr\left[\mathbf{X}_{x,q}=1 \;\middle|\; \sum_{y\in\Sigma^k-B_x}\mathbf{X}_{y,q}=i\right] - p_x \right| \cdot \Pr\left(\sum_{y\in\Sigma^k-B_x}\mathbf{X}_{y,q}=i\right)$$

The estimation of b_3 when a pattern is not independent from those outside its neighborhood requires the calculation of the conditional expectation. Even in this case, we assume that $s_x = 0$ when x is not a frequent pattern. For each pattern that is frequent in at least one of the generated text, we calculate its corresponding estimate $\hat{s}_x$ as follows:

$$\hat{s}_x = \sum_{i=0}^{|I-B_x|} \left| \frac{f_{(x,i)^+}}{f_{(x,i)}} - p_x \right| \cdot \frac{f_{(x,i)}}{\Delta}$$

Where $f_{(x,i)^+}$ is the number of texts where x is frequent and i patterns outside its neighborhood are frequent; $f_{(x,i)}$ is the number of texts where i patterns outside the neighborhood of x are frequent; and p_x is the ratio between the number of texts where x is frequent and Δ.

In order to calculate b_3, we keep the values of $f_{(x,i)}$ and $f_{(x,i)^+}$ in a map, indexed by the pattern and the number of frequent patterns outside its neighborhood. Whenever a new pair is accessed, we initialize the values in the map to 0.

For each frequent pattern and for each random text, we calculate the number i of frequent non-neighbour patterns of x, we increment $f_{(x,i)}$, and if x is frequent in the text, we also increment $f_{(x,i)^+}$. Finally, we calculate b_3 by summing the expression $\left|\frac{f_{(x,i)^+}}{f_{(x,i)}} - p_x\right| \cdot \frac{f_{(x,i)}}{\Delta}$ for each pair in the map.

The complexity of this procedure is considerable, as potentially we have to store $O(|W_q| \min(\Delta, |W_q|))$ elements (one for each frequent pattern and for each value of i obtained from each text, which can be up to the minimum between Δ and the maximum number of frequent patterns in any text). Even if we achieve a constant time for accessing each element with a hash map (assuming that k is constant), we still have to count for each pattern and for each text the number of frequent patterns outside its neighborhood. A simple linear scan on the frequent patterns for each text would require $O(|W_q| \cdot \sum_{t \in \mathcal{C}} |W_{q,t}|)$ time.

5.4 Goodness of fit for the Poisson approximation

We saw in section 5.3 that our approaches to approximating $\mathbf{Q}_{k,q}$ to a Poisson variable returned high values of b_1 and b_2 unless the mean is much less than 1. Even by using the reduced neighborhood set, we expect to obtain values of b_1 and b_2 that are not much less than their counterparts in the complete neighborhood, and we also have to estimate the value of b_3.

We now move to a more practical approach in order to determine whether $\mathbf{Q}_{k,q}$ can be approximated to a Poisson distribution. First of all, we remember that, in a Poisson distribution with mean λ, the variance is λ. If the estimator of the variance in our simulations deviates significantly from the average, we can reasonably conclude that $\mathbf{Q}_{k,q}$ cannot be approximated to a Poisson variable.

In addition, even when the estimated variance is close to λ, we can use the Pearson's chi-squared test, a more rigorous test.

Pearson's chi-squared test The Pearson's chi-squared test is a statistical test for probability distributions. Suppose we have n random samples drawn independently from

an unknown probability distribution. We classify the samples into k classes, such that each value in the domain of the samples is classified in one of those classes. For each class i, we calculate the empirical frequency f_i ($\sum_{i=1}^{k} f_i = 1$). We would like to test whether the unknown distribution can be adequately fitted to a known distribution g, whose parameters are obtained from some estimators obtained from the samples. The null hypothesis in this test is

$$H_0 = \{\text{the } n \text{ samples are drawn from } g\}$$

For each class, we calculate g_i, the theoretical probability that a random variable with distribution g belongs to class i. The test statistic in a Pearson's chi-squared test is:

$$\hat{\chi}^2 = \sum_{i=1}^{k} \frac{(f_i - ng_i)^2}{ng_i}$$

Under the null hypothesis, it can be shown that the distribution of $\hat{\chi}^2$ is approximately distributed as the sum of ν independent, standard normal variables. This distribution is called *chi-squared distribution with ν degrees of freedom*, χ^2_ν.

The number of degrees of freedom is $\nu = k - r > 0$, where r is the number of constraints or relations that are used to estimate the values g_i from the data. As $\sum_{i=1}^{k} g_i = 1$, there is always at least one constraint, thus $r > 1$. Usually, r is equal to the number of parameters of the distribution g that have been estimated from the data, plus one. In our tests, g is a Poisson distribution with mean λ estimated by the empirical average of the samples, thus $\nu = k - 2$.

Given the statistical significance α of the test, the rejection region for the test is

$$C = \{x > 0 : \Pr(\chi^2_\nu \geq x) \leq \alpha\}$$

We reject the null hypothesis when $\hat{\chi}^2 \in C$, thus when $\hat{\chi}^2 \geq \min\{x > 0 : \Pr(\chi^2_\nu \geq x) \leq \alpha\}$.

For this kind of tests, we need to classify the samples into k classes. It is advised to group the data into classes such that the expected frequency ng_i for each class is at least 5, in order to avoid excessively skewed results.

Chapter 6

Experimental results

In this chapter, we show the results of simulations that try to validate the results obtained in Chapter 5. The tests have been implemented in a program written in C++, in order to achieve good memory efficiency and performance. Some operations, such as the calculation of the CDF of the Poisson distribution and the matrix power, have been realized with the aid of the Boost libraries [1].

We report some tests on the number of frequent patterns of length $k = 10$ in a text of length $l = 500000$, generated with different random models. We chose these values of l and k in order to show the behavior of the procedures when the text is long and the number of patterns that occur at least once in the text is close to $|\Sigma^k|$. These values show the differences between the random models and between various values of the quorum.

In independent non-equiprobable random models and in 1-order Markov chain random models, we use the empirical distribution of the human metabotropic glutamate receptor 1 [2], from which we obtain the following steady-state probability vector and transition matrix:

$$\pi = \begin{pmatrix} p_A & p_C & p_G & p_T \end{pmatrix} = \begin{pmatrix} 0.296322 & 0.182711 & 0.189131 & 0.331836 \end{pmatrix}$$

$$T = \begin{pmatrix} T_{A,A} & T_{A,C} & T_{A,G} & T_{A,T} \\ T_{C,A} & T_{C,C} & T_{C,G} & T_{C,T} \\ T_{G,A} & T_{G,C} & T_{G,G} & T_{G,T} \\ T_{T,A} & T_{T,C} & T_{T,G} & T_{T,T} \end{pmatrix} = \begin{pmatrix} 0.324454 & 0.154579 & 0.222238 & 0.298729 \\ 0.359398 & 0.225585 & 0.0318679 & 0.383149 \\ 0.306249 & 0.185568 & 0.226014 & 0.282169 \\ 0.230807 & 0.182599 & 0.225139 & 0.361456 \end{pmatrix}$$

[1] `http://www.boost.org/`

[2] `http://www.ncbi.nlm.nih.gov/nuccore/NG_012839.1?from=5001&to=414953&report=fasta`

6.1 Sample average and mean estimation

We now compare the average we estimated in section 5.3 to the sample average obtained from random texts, generated with independent models.

6.1.1 Independent equiprobable model

We remember that the expected average is simply $4^k P(\text{Poisson}(\mu) \geq q)$, while the error bound is at most $R \leq (4k-3)\mu = (4k-3)(l-k+1)/4^k$. In Table 6.1, we can see that the sample average is always within the error bound. In particular, we observe that the absolute difference between the two estimations tends to be higher for small values of q.

From Figure 4.1, we know that the relative error for overlapping patterns diverges as the quorum is increased. However, in this situation, the number of samples is insufficient and does not allow to measure the difference between the approximated average and the sample average.

Quorum	Approximate average ($\pm$17.6427)	Sample average
2	87304.7	87300.8
3	13308.3	13306.6
4	1547.12	1545.17
5	145.1	145.067
6	11.395	11.33
7	0.769345	0.796
8	0.0455409	0.045
9	0.00239958	0.004
10	0.000113906	0

TABLE 6.1: Expected average number of frequent patterns of length $k = 10$ in a text of length $l = 500000$, compared with a sample average of $\Delta = 1000$ random texts generated by an independent equiprobable model.

6.1.2 Independent non-equiprobable model

When a non-equiprobable model is assumed for the sample sequence, as we discussed before, the probability of occurrence for a pattern varies according to the distribution of the characters in the pattern. Thus, we expect that some patterns have a higher average number of occurrences, which will influence the average number of frequent patterns.

In Table 6.2, we see that while in the equiprobable case there were no frequent patterns with at least 10 occurrences, in the non-equiprobable case there are about 43 patterns with at least 10 occurrences on average.

Another interesting issue is that the cumulative error R has increased, and the relative difference between the sample average and the estimated average is substantial with

high quorum values. This behavior is predicted by Figure 4.1, in addition to the fact that degenerate patterns, such as patterns with period 1, may have a higher probability than others. In fact, our approximated average tends to underestimate the real average.

Quorum	Approximate average (±50.19)	Sample average
2	103260	103257
3	32292.3	32289.7
4	10901.2	10895.9
5	3947	3945.5
6	1511.67	1512.9
7	603.75	603.745
8	248.167	248.217
9	103.69	103.696
10	43.5208	43.441
11	18.1498	18.255
12	7.45114	7.429
13	2.98991	3.069
14	1.167	1.256
15	0.441774	0.491
16	0.161969	0.191
17	0.0574894	0.08
18	0.0197592	0.038
19	0.00658009	0.018
20	0.00212472	0.009
21	0.000665784	0.004
22	0.000202616	0.003
23	5.99309e-05	0.003
24	1.72411e-05	0.003
25	4.82715e-06	0.003

TABLE 6.2: Expected average number of frequent patterns of length $k = 10$ in a text of length $l = 500000$, compared with a sample average of $\Delta = 1000$ random texts generated by an independent non-equiprobable model.

6.2 Simulation for determining the error bound

We now show the estimated values of b_1, b_2 and b_3 under independent models and the 1-order Markov chain model. We remember that we skip the calculation of the bound when we use a reduced neighborhood and the number of neighboring pairs is too large, and we skip the calculation of b_1 and b_3 when the estimated value of b_2 is greater than 1.

The results for the independent equiprobable model are shown in Table 6.3. The full neighborhood error bound clearly shows that the bound is exceedingly high when the average is greater to 1, and even for $q = 7$, where the value is lower but close to

1. Additionally, in the reduced neighborhood, we see that the estimated value of b_3 constitutes a big part of the error bound.

		Full neighborhood	Reduced neighborhood		
Quorum	Sample Avg.	$b_1 + b_2$	b_1	b_2	b_3
2	87300.8	1.52428e+10			
3	13306.6	3.54131e+08			
4	1545.17	4.77539e+06			
5	145.067	42103.4		11121.4	
6	11.33	256.938		67.622	
7	0.796	1.28643	0.333924	0.328	0.743814
8	0.045	0.00206802	0.001069	0	0.001912
9	0.004	1.9988e-05	1.2e-05	0	8e-06
10	0	0	0	0	0

TABLE 6.3: Error bounds for the Poisson approximation for the number of frequent patterns of length $k = 10$ in texts of length $l = 500000$, estimated by using $\Delta = 1000$ random texts generated by an independent equiprobable model.

The results for the independent non-equiprobable model are shown in Table 6.4. As shown in the previous section, the average number of frequent patterns is considerably higher than the average number in the equiprobable case. Thus, when the probabilities for each character are unbalanced, we expect more frequent pattterns as q is increased. The error bound for the full neighborhood become interesting when the average number is less than 0.1. In the reduced neighborhood, the values of b_3 are less dominant than the values in the equiprobable model. However, the error bound of $b_1 + b_2 + b_3$ is still remarkably similar to the full neighborhood error bound.

Finally, we consider the results for the 1-order Markov chain, shown in Table 6.5. The average number of frequent pattern is considerably higher than the average number in independent models. For the full neighborhood, the error bound is low only when the average is close to 0.1, as with the independent non-equiprobable case (from which we obtain that the quorum should be greater than 25 in this case). The value of b_3 in the reduced neighborhood is often substantially lower than the values of b_1 and b_2, and there is a slight improvement in the error bound $b_1 + b_2 + b_3$ when the average number is close to 1. This might suggest that in 1-order Markov chains, the patterns outside the neighborhood of the pattern x have a smaller influence on the occurrences of x than in the other cases.

6.3 Goodness of fit

We now use the Pearson's chi-square test for the goodness of fit, described in Section 5.4 to determine whether a Poisson approximation fits the data even when the error bound is high. We remind that the method requires to partition the sample space in classes

		Full neighborhood	Reduced neighborhood		
Quorum	Sample Avg.	$b_1 + b_2$	b_1	b_2	b_3
2	103257	2.1324e+10			
3	32289.7	2.08526e+09			
4	10895.9	2.37457e+08			
5	3945.5	3.114e+07			
6	1512.9	4.58028e+06			
7	603.745	729943			
8	248.217	123581			
9	103.696	21625		9684.22	
10	43.441	3815.9		1759.04	
11	18.255	680.451		322.552	
12	7.429	114.157		55.516	
13	3.069	19.7908		9.854	
14	1.256	3.33097		1.7	
15	0.491	0.561651	0.230693	0.304	0.039182
16	0.191	0.102702	0.035915	0.064	0.004288
17	0.08	0.0104777	0.0063	0.002	0.003936
18	0.038	0.00148059	0.00144	0	8e-06
19	0.018	0.000341694	0.000322	0	4e-06
20	0.009	8.99279e-05	8.1e-05	0	0
21	0.004	1.9988e-05	1.6e-05	0	0
22	0.003	1.1994e-05	9e-06	0	0
23	0.003	1.1994e-05	9e-06	0	0
24	0.003	1.1994e-05	9e-06	0	0
25	0.003	1.1994e-05	9e-06	0	0

TABLE 6.4: Error bounds for the Poisson approximation for the number of frequent patterns of length $k = 10$ in texts of length $l = 500000$, estimated by using $\Delta = 1000$ random texts generated by an independent non-equiprobable model, following the distribution of human_grm1.

such that the expected number of samples that belong to the class is at least 5, and that the empirical value $\hat{\chi}^2$ must be tested against a chi-square distribution with $n - 2$ degrees of freedom, where n is the number of classes.

For each tested quorum value q, we report the sample average, the sample variance, the number of classes n generated by the procedure that partitions the space, the empirical value $\hat{\chi}^2$, and the value of $\Pr(\chi^2_{n-2} > \hat{\chi}^2)$, where χ^2_{n-2} is a random variable with a chi-square distribution with $n - 2$ degrees of freedom (this value will be called p-value in this section).

In our tests, the Poisson distribution does not always guarantee to find an appropriate partition of $n > 2$ classes that satisfy the condition, especially when q is high and thus the average number of frequent patterns λ is low. For this reason, when our partition procedure returns $n \leq 2$ classes, we do not report the p-value.

		Full neighborhood	Reduced neighborhood		
Quorum	Sample Avg.	$b_1 + b_2$	b_1	b_2	b_3
2	116801	2.72851e+10			
3	41539.1	3.45099e+09			
4	14736.6	4.34345e+08			
5	5414.03	5.86296e+07			
6	2098.79	8.81266e+06			
7	864.218	1.49481e+06			
8	377.41	285269			
9	175.108	61483.2			
10	85.199	14594.9		6022.89	
11	43.267	3778.95		1609.66	
12	22.964	1070.01		471.294	
13	12.59	325.235		147.434	
14	7.051	103.286		48.428	
15	4.083	35.387		17.29	
16	2.405	12.177		6.032	
17	1.525	4.79329		2.36	
18	0.974	1.91166	0.908084	0.918	0.051532
19	0.632	0.750007	0.38656	0.326	0.030268
20	0.457	0.35124	0.203733	0.136	0.011116
21	0.333	0.163163	0.108799	0.05	0.00151
22	0.263	0.091385	0.068549	0.02	0.003376
23	0.203	0.055385	0.040819	0.012	0.003304
24	0.156	0.0304738	0.024336	0.006	0
25	0.126	0.0199902	0.015876	0.004	0

TABLE 6.5: Error bounds for the Poisson approximation for the number of frequent patterns of length $k = 10$ in texts of length $l = 500000$, estimated by using $\Delta = 1000$ random texts generated by an 1-order Markov chain model, following the distribution of human_grm1.

In Table 6.6, we report the results for the independent equiprobable model. In this case, when $q = 2$ the approximation is poorly fitting: the variance is significantly lower than λ, which is the variance of the Poisson distribution, and the p-value is remarkably small. When $q > 2$, however, the sample variance is closer to the sample average, and the p-value is decisely higher than the usual significance values; thus the Poisson approximation appears to be a decent fit even for some large values of λ in independent equiprobable models.

In Table 6.7, we report the results for the independent non-equiprobable model. In this situation, the Poisson approximation performs worse than in the equiprobable model: the p-values start to be reasonably high when $q \geq 14$, or when the average is close or less than 1. Also the variance seems close to the average for these values of q. We can conclude that in the non-equiprobable model, when the probabilities differ significantly from the equiprobable model, the error bound that we obtained from the naïve approach cannot be improved.

Quorum	Sample Avg.	Sample Variance	Classes	$\hat{\chi}^2$	p-value
2	87300.8	47022.8	145	284.682	1.90538e-11
3	13306.6	14966.3	151	188.262	0.0162396
4	1545.17	1811.86	127	143.067	0.128499
5	145.067	159.612	54	53.6492	0.410872
6	11.33	11.5306	18	11.1316	0.801292
7	0.796	0.815199	5	4.385	0.22278
8	0.045	0.043018	2	0.0236524	
9	0.004	0.00398799	1	0	
10	0	0	1	0	

TABLE 6.6: Goodness of fit for the Poisson approximation for the number of frequent patterns of length $k = 10$ in texts of length $l = 500000$, estimated by using $\Delta = 1000$ random texts generated by an independent equiprobable random model.

Quorum	Sample Avg.	Sample Variance	Classes	$\hat{\chi}^2$	p-value
2	103257	56979.5	140	242.137	1.0492e-07
3	32289.7	45599.2	143	224.598	9.53969e-06
4	10895.9	25775.7	140	1032.99	6.97554e-137
5	3945.5	10028.8	155	1226.62	6.84481e-167
6	1512.9	4033.18	125	1369.86	2.1071e-209
7	603.745	1530.5	94	1112.53	8.20245e-175
8	248.217	605.686	66	1011.79	1.70043e-170
9	103.696	222.941	47	650.682	2.36554e-108
10	43.441	85.0996	33	460.135	6.81127e-78
11	18.255	32.2162	23	378.632	2.41168e-67
12	7.429	11.2062	15	114.326	2.64538e-18
13	3.069	4.02226	9	46.3872	7.34918e-08
14	1.256	1.4319	6	8.4145	0.0775217
15	0.491	0.570489	4	10.3533	0.00564684
16	0.191	0.22074	3	0.494901	0.481749
17	0.08	0.0776777	2	0.0175593	
18	0.038	0.0365926	2	0.0141596	
19	0.018	0.0176937	2	0.00148004	
20	0.009	0.00892793	2	0.000183622	
21	0.004	0.00398799	1	0	
22	0.003	0.00299399	1	0	
23	0.003	0.00299399	1	0	
24	0.003	0.00299399	1	0	
25	0.003	0.00299399	1	0	

TABLE 6.7: Goodness of fit for the Poisson approximation for the number of frequent patterns of length $k = 10$ in texts of length $l = 500000$, estimated by using $\Delta = 1000$ random texts generated by an independent non-equiprobable random model, following the distribution of human_grm1.

Quorum	Sample Avg.	Sample Variance	Classes	$\hat{\chi}^2$	p-value
2	116801	58243.6	141	269.859	2.00229e-10
3	41539.1	42781.9	154	153.544	0.449686
4	14736.6	25945.5	151	449.43	6.9024e-32
5	5414.03	11627.9	153	837.799	1.12166e-95
6	2098.79	4906.05	137	1028.24	2.61393e-137
7	864.218	1933.1	106	829.702	1.62603e-113
8	377.41	770.176	79	579.701	3.83452e-78
9	175.108	332.709	58	454.812	7.76275e-64
10	85.199	162.394	43	422.063	9.63566e-65
11	43.267	78.1459	33	341.015	6.65791e-54
12	22.964	38.287	24	223.196	3.09532e-35
13	12.59	20.8087	19	195.101	2.75833e-32
14	7.051	10.9033	15	149.793	2.26907e-25
15	4.083	6.12824	11	118.416	2.81786e-21
16	2.405	3.01399	8	31.4464	2.08286e-05
17	1.525	1.66704	6	17.0793	0.00186556
18	0.974	0.988312	5	6.13192	0.105366
19	0.632	0.583159	4	6.97926	0.0305121
20	0.457	0.390542	4	13.5805	0.00112466
21	0.333	0.274385	3	16.0111	6.29708e-05
22	0.263	0.216047	3	19.2538	1.14442e-05
23	0.203	0.175967	3	10.1966	0.00140702
24	0.156	0.137802	3	7.92729	0.00486946
25	0.126	0.114238	3	4.95806	0.0259693

TABLE 6.8: Goodness of fit for the Poisson approximation for the number of frequent patterns of length $k = 10$ in texts of length $l = 500000$, estimated by using $\Delta = 1000$ random texts generated by an 1-order Markov chain random model, following the distribution of human_grm1.

In Table 6.8, we report the results for the independent non-equiprobable model. We notice that the situation is similar to the independent non-equiprobable model, however for $q \geq 17$ the p-values have an erratic behavior, with low p-values when $q = 21$ and $q = 22$, when there is a reduction in the number of classes from 4 to 3. The variance is comparable to the average when $q \geq 17$.

Chapter 7

Conclusion

7.1 Results

We explored various statistical aspects in the extraction of the frequent patterns in genomic sequences. Our intention was to evaluate whether the Chen-Stein method could be successfully applied to the extraction of frequent patterns as it was done in [9] for the itemsets in market basket analysis, which gave an interesting result that allowed the authors increase the efficiency and accuracy of existing algorithms.

In our framework, we discovered that the Chen-Stein method is used for approximating the distribution of the number of occurrences of a pattern to a compound Poisson distribution, while the simple Poisson distribution we described is more adequate for aperiodic patterns.

We intended to compare the exact distribution to the Poisson distribution, in order to evaluate the discrepancies between the two and see whether a Poisson approximation is sufficient for our purposes. We developed two simple algorithms for the calculation of the exact complementary cumulative distribution function, based on the DFA for the recognition of strings with the required number of occurrences of the pattern; we later discovered that these algorithms are known in literature under the name of Finite Markov Chain Imbedding. We still were interested in this and other methods for the exact calculation for the complementary CDF for a single pattern, most importantly to compare the difference in the complexity between the exact methods and the approximation methods. Furthermore, by analyzing their complexity, we conclude that they are not viable methods for the calculation of the number of frequent patterns.

Subsequently, we analyzed the applicability of the Chen-Stein method for some families of patterns in different random models. We observed that the Poisson approximation for the number of occurrences can be successfully applied to patterns that cannot self-overlap. Through the comparison between the exact probability and the approximate

probability in independent random models, while the discrepancy in the complementary CDF is significantly lower than the error bound $b_1 + b_2$, we noticed that the relative error rapidly diverges for patterns with high periodicity.

We then proceeded to investigate the distribution of the number of frequent patterns. We discovered that the results obtained through the Chen-Stein method for individual patterns could be useful for the estimation of the average number of frequent patterns. For independent models, we developed a fast way to approximate the average number of frequent patterns, together with an error bound. For equiprobable models, the method requires to calculate just one value for the complementary CDF of a Poisson random variable. For non-equiprobable independent models, the approximation requires just $O(k^{|\Sigma|-1})$ terms. The method is particularly interesting when the alphabet size is small, which is the case in genomic sequences.

Applying the Chen-Stein method to the number of frequent patterns immediately appeared to be a challenging problem: while in market basket analysis the random models that are commonly used let the authors establish a compact neighborhood set, in genomic sequences each pattern is intrinsically dependent to each other, even when they cannot overlap.

The naïve application of the Chen-Stein theorem, where the neighborhood set corresponds to the set of patterns, lets us apply the approximation only if the average number of occurrences is much less than 1. This has been established both by an analytical evaluation (which also depends on the variance of the number of frequent patterns) and by an estimation through simulation. The quorum value threshold obtained this way may be still quite reasonable for some random models.

We evaluated a different neighborhood set, which excludes those patterns that cannot overlap from their respective neighborhoods. We noticed that the size of the neighborhood is still rather large, coherently with the fact that it is sufficient for the two patterns to share a single character between the end of one pattern and the beginning of the other, in order for them to be neighbors. Unsurprisingly, the estimated values of b_1 and b_2 did not decrease substantially, while b_3 starts being greater than 0.

Finally, we decided to check whether a Poisson approximation may be suitable even when the average is about 1 or higher, in order to potentially exclude any significant improvement with other neighborhoods. The application of the Pearson's chi-square test and some estimates of the variance let us conclude that no significant improvements can be made over the naïve approach for non-equiprobable models, while some improvement may be obtained for the independent equiprobable model.

These results let us apply the method described in Section 1.3: we provided an approximation for the average number of frequent patterns in independent random models, which can be used instead of simulating the texts, and we gave some indications about

when the distribution can be approximated to a Poisson distribution. These can be used to obtain a quorum threshold after which every frequent pattern can be marked as statistically significant with a low FDR.

7.2 Further developments

There are various aspects that can be expanded in further studies. First of all, the compound Poisson approximation for single patterns may be used to improve the results we obtained for the approximation of the average number of frequent patterns. The key issue is being able to obtain expressions that lead to an acceptable complexity for the approximation and for the error.

It is also interesting to expand the calculation of the average number of frequent patterns for non-independent models. Unfortunately, in Markov chain random models, the occurrence probability of each pattern is heavily influenced by the order of the characters in the pattern; thus it is more difficult to partition the patterns in classes with equal probabilities. An approximate approach that partition the patterns in bigger classes with similar probabilities may have some range of applicability. Another possibility is to sample the space of patterns and calculate the probability (exact or approximate) of q-occurrence, in order to estimate the number of frequent patterns.

Further improvements may be made for motifs: we analyzed some simple structured motifs in our work with fixed length, however in computational biology there is a substantial interest in various types of motifs, which often reach the complexity of regular expressions, or even allow for some errors in the pattern match in order to recognize patterns with slight variability. This requires additional study on the types of motifs that are of interest in molecular biology.

Finally, there is an interest in finding patterns that appear to be frequent in a significant number of texts in a collection of texts. Some of our results may be adapted to this new problem, while the simulation may become cumbersome when the number of texts in the collection is considerable.

Bibliography

[1] Arratia, R., Goldstein, L., and Gordon, L. Poisson approximation and the Chen-Stein method. *Statistical Science 5*, 4 (1990), pp. 403–424.

[2] Arratia, R., and Reinert, G. Poisson process approximation for repeats in one sequence and its application to sequencing by hybridization. In *Combinatorial Pattern Matching*, D. Hirschberg and G. Myers, Eds., vol. 1075 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1996, pp. 209–219.

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

[4] Fu, J. C., et al. *Distribution theory of runs and patterns and its applications: a finite Markov chain imbedding approach*. World Scientific, 2003.

[5] Gentleman, J. F., and Mullin, R. C. The distribution of the frequency of occurrence of nucleotide subsequences, based on their overlap capability. *Biometrics 45*, 1 (1989), pp. 35–52.

[6] Göke, J., Schulz, M. H., Lasserre, J., and Vingron, M. Estimation of pairwise sequence similarity of mammalian enhancers with word neighbourhood counts. *Bioinformatics 28*, 5 (Mar. 2012), 656–663.

[7] Grossi, R., Pietracaprina, A., Pisanti, N., Pucci, G., Upfal, E., and Vandin, F. MADMX: A Strategy for Maximal Dense Motif Extraction. *Journal of Computational Biology 18*, 4 (Apr. 2011), 535–545.

[8] Hämäläinen, W. Statapriori: an efficient algorithm for searching statistically significant association rules. *Knowledge and Information Systems 23*, 3 (2010), 373–399.

[9] Kirsch, A., Mitzenmacher, M., Pietracaprina, A., Pucci, G., Upfal, E., and Vandin, F. An efficient rigorous approach for identifying statistically significant frequent itemsets. *J. ACM 59*, 3 (June 2012), 12:1–12:22.

[10] Kleffe, J., and Borodovsky, M. First and second moment of counts of words in random texts generated by markov chains. *Computer applications in the biosciences : CABIOS 8*, 5 (1992), 433–441.

[11] Lippert, R. A., Huang, H., and Waterman, M. S. Distributional regimes for the number of k-word matches between two random sequences. *Proceedings of the National Academy of Sciences 99*, 22 (2002), 13980–13989.

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

[13] Parida, L. *Pattern Discovery in Bioinformatics: Theory & Algorithms.* CRC Press, 2007.

[14] Régnier, M. A unified approach to word occurrence probabilities. *Discrete Appl. Math. 104*, 1-3 (Aug. 2000), 259–280.

[15] Reinert, G., and Schbath, S. Compound Poisson and Poisson process approximations for occurrences of multiple words in Markov chains. *Journal of Computational Biology 5*, 2 (1998).

[16] Reinert, G., Schbath, S., and Waterman, M. S. Probabilistic and statistical properties of words: An overview. *Journal of Computational Biology 7* (2000), 1–46.

[17] Robin, S. A compound poisson model for word occurrences in dna sequences. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 51*, 4 (2002), 437–451.

[18] Robin, S., and Daudin, J.-J. Exact distribution of word occurrences in a random sequence of letters. *Journal of Applied Probability 36*, 1 (1999), 179–193.

[19] Robin, S., Daudin, J.-J., Richard, H., Sagot, M.-F., and Schbath, S. Occurrence probability of structured motifs in random sequences. *Journal of Computational Biology 9*, 6 (2002), 761–773.

[20] Schbath, S. Compound poisson approximation of word counts in dna sequences. *ESAIM P & S 1* (1997), 1–16.

[21] Schbath, S. Statistics of motifs. *Atelier de formation 1502* (2006).

[22] Van Helden, J. Metrics for comparing regulatory sequences on the basis of pattern counts. *Bioinformatics 20*, 3 (2004), 399–406.

[23] Van Helden, J., Rios, A. F., and Collado-Vides, J. Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic acids research 28*, 8 (2000), 1808–1818.

[24] Waterman, M. S., et al. *Introduction to computational biology: maps, sequences and genomes.* Chapman & Hall Ltd, 1995.

[25] Zhang, J., Chen, X., and Li, M. Computing exact p-value for structured motif. In *Combinatorial Pattern Matching*, B. Ma and K. Zhang, Eds., vol. 4580 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2007, pp. 162–172.