Complexity Profiles of DNA Sequences Using Finite-Context Models

Armando J. Pinho[⋆], Diogo Pratas, and Sara P. Garcia

Signal Processing Lab, IEETA / DETI
University of Aveiro, 3810–193 Aveiro, Portugal
{ap,pratas,spgarcia}@ua.pt

Abstract. Every data compression method assumes a certain model of the information source that produces the data. When we improve a data compression method, we are also improving the model of the source. This happens because, when the probability distribution of the assumed source model is closer to the true probability distribution of the source, a smaller relative entropy results and, therefore, fewer redundancy bits are required. This is why the importance of data compression goes beyond the usual goal of reducing the storage space or the transmission time of the information. In fact, in some situations, seeking better models is the main aim. In our view, this is the case for DNA sequence data. In this paper, we give hints on how finite-context (Markov) modeling may be used for DNA sequence analysis, through the construction of complexity profiles of the sequences. These profiles are able to unveil structures of the DNA, some of them with potential biological relevance.

1 Introduction

Modeling plays a key role in data compression. With the invention of the first practical algorithm for arithmetic coding [1], the problem of finding out an efficient representation for a certain information source could be restated as a data modeling problem. For our purposes, a model is a mathematical description of the information source, providing a probability estimate of the next outcome. The entropy of this model sets a lower bound on the compression performance of the arithmetic encoder. This bound is tight, meaning that it is possible to generate a bitstream with average entropy as close as desired to the entropy of the model, suggesting that the effort should be made to find good models of the information sources.

For about the last ten years, we have been addressing the problem of data compression using arithmetic coding. Initially in the context of image coding and, more recently, in the context of DNA coding, we have been relying on

[⋆] This work was partially funded by FEDER through the Operational Program Competitiveness Factors - COMPETE and by National Funds through FCT - Foundation for Science and Technology in the context of project FCOMP-01-0124-FEDER-010099 (FCT reference PTDC/EIA-EIA/103099/2008).

finite-context (Markov) models for describing the data in an efficient way. Finite-context models assume that the source has Markovian properties, i.e., that the probability of the next outcome of the information source depends only on some finite number of (recent) past outcomes. This past is normally referred to as the "context", hence the name "finite-context model".

In the context of DNA data compression, these models have been usually associated with the task of providing compression when the main method fails. However, they have also been used as the main method, both for representing protein-coding regions of DNA [2] and for representing unrestricted DNA, i.e., DNA with coding and non-coding regions [3,4,5,6,7]. In this paper, we present and discuss the problem of computing complexity profiles (or information sequences) using finite-context models. Basically, a complexity profile indicates how many bits are required to represent each symbol (DNA base). These complexity profiles are of interest because they reveal structures inside the chromosomes, structures that are often associated with regulatory functions of DNA [8].

2 Finite-Context Models

Consider an information source that generates symbols, s, from an alphabet $\mathcal{A}$, and denote by $x^n = x_1 x_2 \ldots x_n$ the sequence of symbols generated by the source after n outcomes. A finite-context model of an information source (see Fig. 1 for an example where $\mathcal{A} = \{0, 1\}$) assigns probability estimates to the symbols of the alphabet, according to a conditioning context computed over a finite and fixed number, k, of past outcomes (order-k finite-context model) [9,10,11]. At instant n, we represent these conditioning outcomes by $c^n = x_{n-k+1}, \ldots, x_{n-1}, x_n$.

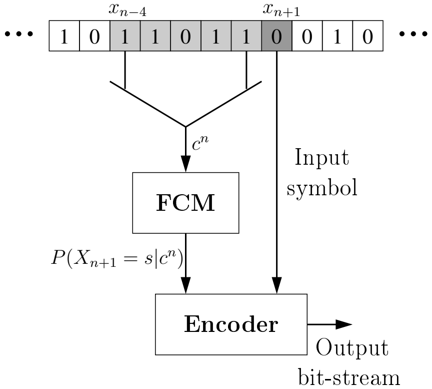

Fig. 1. Example of a finite-context model for the binary alphabet, i.e., for $\mathcal{A} = \{0, 1\}$. The probability of the next outcome, X_{n+1}, is conditioned by the k last outcomes. In this example, $k = 5$.

In practice, the probability that the next outcome, X_{n+1}, is $s \in \mathcal{A}$, is obtained using the estimator

$$P(X_{n+1} = s|c^n) = \frac{N_s^n + \alpha}{\sum\limits_{a \in \mathcal{A}} N_a^n + |\mathcal{A}|\alpha}, \qquad (1)$$

where $|\mathcal{A}|$ denotes the size of the alphabet, and N_s^n represents the number of times that, in the past, the information source generated symbol s having c^n as the conditioning context. The parameter α controls how much probability is assigned to unseen (but possible) events, and plays a key role in the case of high-order models. In fact, when k is large, the number of conditioning states, $|\mathcal{A}|^k$, is high, implying that statistics have to be estimated using only a few observations. This estimator reduces to Laplace's estimator for $\alpha = 1$ [12] and to the frequently used Jeffreys/Krichevsky estimator when $\alpha = 1/2$ [13,14].

Initially, when all counters are zero, the symbols have probability $1/|\mathcal{A}|$, i.e., they are assumed equally probable. The counters are updated each time a symbol is encoded. Since the context is causal, the decoder is able to reproduce the same probability estimates without needing additional information.

The block denoted "Encoder" in Fig. 1 is an arithmetic encoder. It is well known that practical arithmetic coding generates output bitstreams with average bitrates almost identical to the entropy of the model [9,10,11]. The number of bits that are required to represent symbol x_{n+1} is given by $-\log_2 P(X_{n+1} = x_{n+1}|c^n)$. Therefore, the average bitrate (entropy) of the finite-context model after encoding N symbols is given by

$$H_N = -\frac{1}{N} \sum_{n=0}^{N-1} \log_2 P(X_{n+1} = x_{n+1}|c^n) \quad \text{bps}, \qquad (2)$$

where "bps" stands for "bits per symbol".

3 Applications to DNA data

DNA sequences are sequences of symbols (bases) from a 4-symbol alphabet: adenine (A), cytosine (C), guanine (G), and thymine (T). Several specific coding methods have been proposed for compressing these sequences (see, for example, [15,16,17,18,19,20,21,2,22,23,3,4,6,7]). Most of these methods are based on searching procedures for finding exact or approximate repeats, both directly and in their reversed complemented versions (A ↔ T, C ↔ G). Although this approach has been quite effective in terms of compression rates, it also requires a significant computational effort. Low-order finite-context models are typically used in those methods as a secondary, fall back mechanism. Our goal has been to investigate DNA compression methods based only on finite-context models.

Modeling DNA data using only finite-context models has advantages over the typical DNA compression approaches that mix purely statistical (for example, finite-context models) with substitutional models (such as Lempel-Ziv based algorithms): (1) finite-context models lead to much faster performance, a characteristic of paramount importance for long sequences (for example, some human chromosomes have more than 200 million bases); (2) the overall model may be easier to interpret, because it is made of sub-models of the same type.

Initially, we proposed a three-state finite-context model for DNA protein-coding regions, i.e., for the parts of the DNA that carry information regarding how proteins are synthesized [2]. This three-state model proved to be better than a single-state model, giving additional evidence of a phenomenon that is common in these protein-coding regions, the periodicity of period three.

More recently [3,4,6,7], we investigated the performance of finite-context models for unrestricted DNA, i.e., DNA including coding and non-coding parts. In that work, we have shown that a characteristic usually found in DNA sequences, the occurrence of inverted repeats, which is used by most of the DNA coding methods (see, for example, [18,19,20]), can also be successfully integrated in finite-context models. Inverted repeats are copies of DNA sub-sequences that appear reversed and complemented in some parts of the DNA.

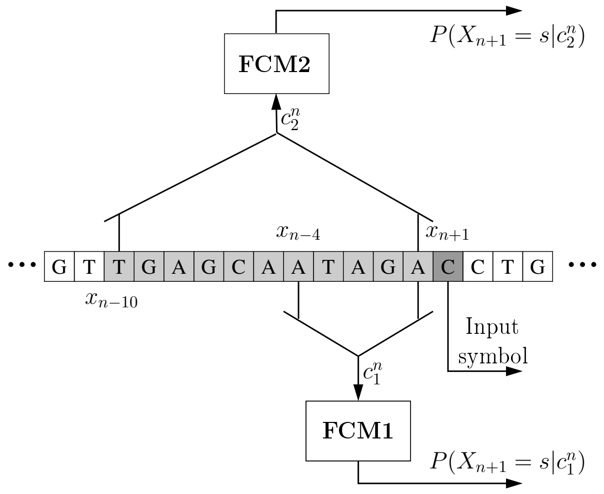

Fig. 2. Example of the use of multiple finite-context models for encoding DNA data. In this case, two models are used, one with a depth-5 context and the other using an order-11 context.

DNA is non-stationary, with regions of low information content (low entropy) alternating with regions with average entropy close to two bits per base. This alternation is modeled by most DNA compression algorithms by using a low-order finite-context model for the high entropy regions and a Lempel-Ziv dictionary-based approach for the repetitive, low entropy regions. We have been studying approaches relying only on finite-context models for representing both regions, leading us to conclude that DNA can be much better represented by Markov models than what it was previously believed.

Moreover, our studies have shown that multiple finite-context models can be more effective in capturing the statistical information along the sequence [4,6,7]. Figure 2 gives an example of these multiple models, that can operate in a competitive or cooperative way. When in competitive mode, the best of the models is chosen for encoding each DNA block, i.e., the one that requires less bits is used for representing the current block [7]. When in cooperative mode of operation, the probability estimates of the several models are combined using an adaptive mixture model [6].

4 Complexity profiles of DNA

The work of researchers such as Solomonoff, Kolmogorov, Chaitin and others [24,25,26,27,28,29], related to the problem of defining a complexity measure of a string, has been of paramount importance for several areas of knowledge. However, because it is not computable, the Kolmogorov complexity of a string A, $K(A)$, is usually approximated by some computable measure, such as Lempel-Ziv complexity measures [30], linguistic complexity measures [31] or compression-based complexity measures [32].

One of the important problems that can be formulated using the Kolmogorov theory is the definition of similarity. Following this line, Li *et al.* [33] proposed a similarity metric based on an information distance [34], defined as the length of the shortest binary program that is needed to transform strings A and B into each other. This distance depends not only on the Kolmogorov complexity of A and B, respectively $K(A)$ and $K(B)$, but also on conditional complexities, for example $K(A|B)$, that indicates how complex string A is when string B is known. Because this distance is based on the Kolmogorov complexity (not computable), they proposed a practical analog based on standard compressors, which they call the normalized compression distance [33].

According to [33], a compression method needs to be “normal” in order to be used in the normalized compression distance. One of the conditions for a compression method to be normal is that compressing string AA (the concatenation of A with A) should generate essentially the same number of bits as compressing A alone [35]. This characteristic holds, for example, in Lempel-Ziv based compressors, making them a frequent choice in this kind of applications.

The construction and analysis of DNA complexity profiles has been an important topic of research, due to its applicability in the study of regulatory functions of DNA, comparative analysis of organisms, genomic evolution and others [36,37]. For example, it has been observed that low complexity regions of DNA are often associated with important regulatory functions [38].

Several measures have also been proposed for evaluating the complexity of DNA sequences. Among those, we find the compression-based approaches the most promising and natural, because compression efficiency is clearly defined (it can be measured by the number of bits generated by the encoder).

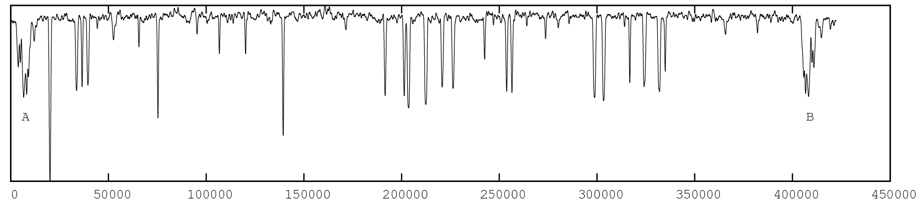

Fig. 3. Complexity profile of chromosome 1 of the *Cyanidioschyzon merolae* organism, obtained with a multiple finite-context modeling approach. We can see several regions where the complexity value goes well below the baseline level that, for an entropy-based complexity profile of DNA, can be set at two bits per DNA nucleotide. The two regions which we have marked with letters A and B correspond to telomeric inverted repeated sequences.

One of the key advantages of DNA compression based on finite-context models is that the encoders are fast and have $\mathcal{O}(n)$ time complexity. As we mentioned already, most of the effort spent by previous DNA compressors is in the task of finding exact or approximate repeats of sub-sequences or of their inverted complements. No doubt, this approach has proved to give good returns in terms of compression gains, but normally at the cost of long compression times. Although slow encoders could be tolerated for storage purposes (compression could be ran in batch mode), for interactive applications they are certainly not appropriate. For example, the currently best performing DNA compression techniques, such as NML-1 [22] or XM [23], could take hours for compressing a single human chromosome. Compressing one of the largest human chromosomes with the techniques based on finite-context models takes less than ten minutes in a 1.66 GHz laptop computer. These DNA sequences have about 240 million bases.

Figure 3 shows an example of one of those complexity profiles (corresponding to chromosome 1 of the *Cyanidioschyzon merolae*) as generated by a multiple finite-context model DNA encoder. We can observe several regions where the complexity is very small, meaning that a reduced number of bits was required for compression those regions. Of particular interest are the two regions which we have marked with letters A and B, corresponding to telomeric inverted repeated sequences.

5 Conclusion

It has been shown that finite-context models are a powerful tool for representing DNA sequences, as demonstrated by the good compression results that they are able to provide [6,7]. However, they may also be useful in other tasks, such as in data analysis. The construction of complexity profiles is an obvious case. These information sequences allow a quick analysis of long sequences, unveiling locations of low information content, which are usually associated with DNA regions of potential biological interest. This seems to be a very promising line of research, clearly deserving further investigation.

References

1. Rissanen, J.: Generalized Kraft inequality and arithmetic coding. IBM J. Res. Develop. **20**(3) (May 1976) 198–203
2. Pinho, A.J., Neves, A.J.R., Afreixo, V., Bastos, C.A.C., Ferreira, P.J.S.G.: A three-state model for DNA protein-coding regions. IEEE Trans. on Biomedical Engineering **53**(11) (November 2006) 2148–2155
3. Pinho, A.J., Neves, A.J.R., Ferreira, P.J.S.G.: Inverted-repeats-aware finite-context models for DNA coding. In: Proc. of the 16th European Signal Processing Conf., EUSIPCO-2008, Lausanne, Switzerland (August 2008)
4. Pinho, A.J., Neves, A.J.R., Bastos, C.A.C., Ferreira, P.J.S.G.: DNA coding using finite-context models and arithmetic coding. In: Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, ICASSP-2009, Taipei, Taiwan (April 2009)
5. Pratas, D., Pinho, A.J.: Compressing the human genome using exclusively Markov models. In: Advances in Intelligent and Soft Computing, Proc. of the 5th Int. Conf. on Practical Applications of Computational Biology & Bioinformatics, PACBB 2011. Volume 93. (April 2011) 213–220
6. Pinho, A.J., Pratas, D., Ferreira, P.J.S.G.: Bacteria DNA sequence compression using a mixture of finite-context models. In: Proc. of the IEEE Workshop on Statistical Signal Processing, Nice, France (June 2011)
7. Pinho, A.J., Ferreira, P.J.S.G., Neves, A.J.R., Bastos, C.A.C.: On the representability of complete genomes by multiple competing finite-context (Markov) models. PLoS ONE **6**(6) (2011) e21588
8. Pinho, A.J., Pratas, D., Ferreira, P.J.S.G., Garcia, S.P.: Symbolic to numerical conversion of DNA sequences using finite-context models. In: Proc. of the 19th European Signal Processing Conf., EUSIPCO-2011, Barcelona, Spain (August 2011)
9. Bell, T.C., Cleary, J.G., Witten, I.H.: Text compression. Prentice Hall (1990)
10. Salomon, D.: Data compression - The complete reference. 4th edn. Springer (2007)
11. Sayood, K.: Introduction to data compression. 3rd edn. Morgan Kaufmann (2006)
12. Laplace, P.S.: Essai philosophique sur les probabilités (A philosophical essay on probabilities). John Wiley & Sons, New York (1814) Translated from the sixth French edition by F. W. Truscott and F. L. Emory, 1902.
13. Jeffreys, H.: An invariant form for the prior probability in estimation problems. Proc. of the Royal Society (London) A **186** (1946) 453–461
14. Krichevsky, R.E., Trofimov, V.K.: The performance of universal encoding. IEEE Trans. on Information Theory **27**(2) (March 1981) 199–207
15. Grumbach, S., Tahi, F.: Compression of DNA sequences. In: Proc. of the Data Compression Conf., DCC-93, Snowbird, Utah (1993) 340–350
16. Rivals, E., Delahaye, J.P., Dauchet, M., Delgrange, O.: A guaranteed compression scheme for repetitive DNA sequences. In: Proc. of the Data Compression Conf., DCC-96, Snowbird, Utah (1996) 453
17. Chen, X., Kwong, S., Li, M.: A compression algorithm for DNA sequences. IEEE Engineering in Medicine and Biology Magazine **20** (2001) 61–66
18. Matsumoto, T., Sadakane, K., Imai, H.: Biological sequence compression algorithms. In Dunker, A.K., Konagaya, A., Miyano, S., Takagi, T., eds.: Genome Informatics 2000: Proc. of the 11th Workshop, Tokyo, Japan (2000) 43–52
19. Manzini, G., Rastero, M.: A simple and fast DNA compressor. Software—Practice and Experience **34** (2004) 1397–1411

20. Korodi, G., Tabus, I.: An efficient normalized maximum likelihood algorithm for DNA sequence compression. ACM Trans. on Information Systems **23**(1) (January 2005) 3–34
21. Behzadi, B., Le Fessant, F.: DNA compression challenge revisited. In: Combinatorial Pattern Matching: Proc. of CPM-2005. Volume 3537 of LNCS., Jeju Island, Korea, Springer-Verlag (June 2005) 190–200
22. Korodi, G., Tabus, I.: Normalized maximum likelihood model of order-1 for the compression of DNA sequences. In: Proc. of the Data Compression Conf., DCC-2007, Snowbird, Utah (March 2007) 33–42
23. Cao, M.D., Dix, T.I., Allison, L., Mears, C.: A simple statistical algorithm for biological sequence compression. In: Proc. of the Data Compression Conf., DCC-2007, Snowbird, Utah (March 2007) 43–52
24. Solomonoff, R.J.: A formal theory of inductive inference. Part I. Information and Control **7**(1) (March 1964) 1–22
25. Solomonoff, R.J.: A formal theory of inductive inference. Part II. Information and Control **7**(2) (June 1964) 224–254
26. Kolmogorov, A.N.: Three approaches to the quantitative definition of information. Problems of Information Transmission **1**(1) (1965) 1–7
27. Chaitin, G.J.: On the length of programs for computing finite binary sequences. Journal of the ACM **13** (1966) 547–569
28. Wallace, C.S., Boulton, D.M.: An information measure for classification. The Computer Journal **11**(2) (August 1968) 185–194
29. Rissanen, J.: Modeling by shortest data description. Automatica **14** (1978) 465–471
30. Lempel, A., Ziv, J.: On the complexity of finite sequences. IEEE Trans. on Information Theory **22**(1) (January 1976) 75–81
31. Gordon, G.: Multi-dimensional linguistic complexity. Journal of Biomolecular Structure & Dynamics **20**(6) (2003) 747–750
32. Dix, T.I., Powell, D.R., Allison, L., Bernal, J., Jaeger, S., Stern, L.: Comparative analysis of long DNA sequences by per element information content using different contexts. BMC Bioinformatics **8**(Suppl. 2) (2007) S10
33. Li, M., Chen, X., Li, X., Ma, B., Vitányi, P.M.B.: The similarity metric. IEEE Trans. on Information Theory **50**(12) (December 2004) 3250–3264
34. Bennett, C.H., Gács, P., Vitányi, M.L.P.M.B., Zurek, W.H.: Information distance. IEEE Trans. on Information Theory **44**(4) (July 1998) 1407–1423
35. Cilibrasi, R., Vitányi, P.M.B.: Clustering by compression. IEEE Trans. on Information Theory **51**(4) (April 2005) 1523–1545
36. Nan, F., Adjeroh, D.: On the complexity measures for biological sequences. In: Proc. of the IEEE Computational Systems Bioinformatics Conference, CSB-2004, Stanford, CA (August 2004)
37. Pirhaji, L., Kargar, M., Sheari, A., Poormohammadi, H., Sadeghi, M., Pezeshk, H., Eslahchi, C.: The performances of the chi-square test and complexity measures for signal recognition in biological sequences. Journal of Theoretical Biology **251**(2) (2008) 380–387
38. Gusev, V.D., Nemytikova, L.A., Chuzhanova, N.A.: On the complexity measures of genetic sequences. Bioinformatics **15**(12) (1999) 994–999