*Bioinformatics*, 32(6), 2016, 835–842
doi: 10.1093/bioinformatics/btv679
Advance Access Publication Date: 16 November 2015
Original Paper

OXFORD

Sequence analysis

# The intrinsic combinatorial organization and information theoretic content of a sequence are correlated to the DNA encoded nucleosome organization of eukaryotic genomes

**Filippo Utro[1,*], Valeria Di Benedetto[2], Davide F.V. Corona[3] and Raffaele Giancarlo[2]**

[1]Computational Genomics Group, IBM T.J. Watson Research Center, Yorktown Heights, NY, USA, [2]Dipartimento di Matematica ed Informatica, Università di Palermo and [3]Dipartimento STEBICEF, Dulbecco Telethon Institute c/o Università di Palermo, Palermo, Italy

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on July 1, 2015; revised on November 6, 2015; accepted on November 9, 2015

## Abstract

**Motivation**: Thanks to research spanning nearly 30 years, two major models have emerged that account for nucleosome organization in chromatin: statistical and sequence specific. The first is based on elegant, easy to compute, closed-form mathematical formulas that make no assumptions of the physical and chemical properties of the underlying DNA sequence. Moreover, they need no training on the data for their computation. The latter is based on some sequence regularities but, as opposed to the statistical model, it lacks the same type of closed-form formulas that, in this case, should be based on the DNA sequence only.

**Results**: We contribute to close this important methodological gap between the two models by providing three very simple formulas for the sequence specific one. They are all based on well-known formulas in Computer Science and Bioinformatics, and they give different quantifications of how complex a sequence is. In view of how remarkably well they perform, it is very surprising that measures of sequence complexity have not even been considered as candidates to close the mentioned gap. We provide experimental evidence that the intrinsic level of combinatorial organization and information-theoretic content of subsequences within a genome are strongly correlated to the level of DNA encoded nucleosome organization discovered by Kaplan *et al*. Our results establish an important connection between the intrinsic complexity of subsequences in a genome and the intrinsic, i.e. DNA encoded, nucleosome organization of eukaryotic genomes. It is a first step towards a mathematical characterization of this latter 'encoding'.

**Supplementary information**: Supplementary data are available at *Bioinformatics* online.

**Contact**: futro@us.ibm.com.

## 1 Introduction

Kornberg (1981), in a very influential article, posed the problem of establishing whether nucleosome positioning obeys statistical laws or is dictated by sequence-specific rules. Such a problem is at the heart of studies concerned with the understanding, as well as the prediction, of nucleosome organization and positioning in genomic DNA and it has been the object of investigation in many disciplines, e.g. biology, physics, bioinformatics. An account on the state of the

art for nucleosome organization, as well as novel results, can be found in two recent articles (Giancarlo *et al.*, 2015; Minary and Levitt, 2014). Among the many outstanding results, the two discussed next are, in our view, both milestones and the most relevant in order to place the contribution given in this article in the proper light.

Kornberg and Stryer (1988) proposed a statistical model, referred to as the Barrier Model and recently generalized by Möbius and Gerland (2010) that would account for nucleosome organization in genomes. Genome-wide experiments have shown its validity (Mavrich *et al.*, 2008a,b). It has the great virtue of being formalized by very elegant mathematical formulas, which can be computed with no training on DNA sequences. Moreover, it had no predecessor of its kind nor has it been the object of controversies after its proposal.

Consequent remarkable findings by Segal *et al.* (2006) and Kaplan *et al.* (2008) established that nucleosome organization is DNA-encoded. The authors went also one step further and showed that, via machine learning techniques, a computer program could learn those sequence rules and later be used as a predictor of nucleosome positioning. That study had been preceded by results that pointed to specific sequence features as being associated to nucleosome depletion or enrichment in genomic regions (see Giancarlo *et al.*, 2015; Minary and Levitt, 2014, for summary and relevant references). However, it was the first study that provided an answer to the sequence-specificity question in general terms and on a genomic scale, as opposed to the specifics of preceding results, e.g. regular patterns occurring in DNA or base composition.

Unfortunately, although what is meant by DNA-encoded in that context is semantically clear, mathematically it is void of meaning since no code was exhibited. Moreover, large scale investigations that have tried to identify universal sequence dictated rules, a requirement less stringent than a code, for nucleosome positioning tended to exclude their existence (Valouev *et al.*, 2008). Simple 'intrinsic' sequence rules are emerging, but they are very specific and very few (Peckham *et al.*, 2007; Tillo and Hughes, 2009) and there is no evidence that they are exhaustive. On the contrary, recent results indicate that the role of $k$-mers involved in nucleosome enrichment or depletion may be more complex than what expected (Giancarlo *et al.*, 2015). In summary, although the lab experiments show the existence of such an encoding, in terms of rules and properties of the associated alphanumeric sequences not much progress has been made. When compared with the Barrier Model, the DNA-encoded one lacks a rigorous set of mathematical formulas, easy to compute and requiring no training on the DNA sequences, that could (even in part) be used to explain the mathematical nature of that encoding. Finding such an object would be important methodologically in order not to transform sequence-specific nucleosome positioning into a potentially endless list of specific rules. Prediction of nucleosome positions based on it, it is perceived as a minor point.

Here we address such a shortcoming by providing three formulas with the required characteristics. They are either well known, or variants of well known, formulas in Computer Science and Bioinformatics. They both quantify the intrinsic complexity of a sequence. In particular, we consider (i) empirical entropy, which measures information-theoretic content of a sequence (ii) two linguistic complexity measures that measure the combinatorial richness of a sequence as quantified by the number of distinct subsequences composing it. Following the same experimental set-up as in Kaplan *et al.* (2008), we find a quite elegant correspondence between fundamental notions in Mathematics and Computer Science and Biology: there is indeed a relationship between the intrinsic complexity of subsequences in a genome and the intrinsic, i.e. DNA-encoded, nucleosome organization of eukaryotic genomes. This is new and unique, as far as the role of sequence specificity in nucleosome positioning is concerned.

## 2 Methods

This section is dedicated to the presentation of the methods needed to perform this research, as follows. Section 2.1 introduces the complexity measures used here. For their study, Kaplan *et al.* (2008) devised a methodology to test whether an algorithm could distinguish nucleosome depleted regions (NDRs, for short) from nucleosome enriched ones (NER). It is outlined in Section 2.2. Section 2.3 is dedicated to the description of a general method that allows constructing nucleosome occupancy maps, via a generic algorithm as the one described in Section 2.2. Finally, Section 2.4 describes a procedure to assess the level of agreement between two nucleosome occupancy maps.

### 2.1 Sequence complexity measures

The complexity of a sequence can be formally defined by resorting to techniques and ideas coming from the following related areas: sequence combinatorics and linguistic complexity (De Luca and Varricchio, 1999; Trifonov, 1990), Shannon information theory (Cover and Thomas, 1991), and Kolmogorov-Chaitin algorithmic complexity (Li and Vitányi, 1997). In terms of actual numeric evaluations, measures stemming from this latter area are not computable and can only be heuristically approximated via data compression programs (see Giancarlo *et al.*, 2009, 2012). However, the corresponding heuristics offer no performance guarantee on how good the corresponding approximations are. Therefore, for the experiments in this article, only measures obtained from the first two mentioned areas are used.

#### 2.1.1 Combinatorial and linguistic

Let $\Sigma$ be a finite alphabet of symbols. Given a sequence $x$ of length $n$, defined over the alphabet $\Sigma$, and an integer $i \leq n$, let LCS($i$) be the number of distinct subsequences of length $i$ that are present in $x$, normalized by $n$. Now, fix an integer $k \leq n$ and let:

$$\mathrm{LC}(k) = \sum_{i=1}^{k} \mathrm{LCS}(i).$$

A few remarks are in order. LC is a normalization of a measure due to De Luca and Varricchio (1999) and it is related to the linguistic sequence complexity introduced by Trifonov (1990). Both LC and LCS measure the complexity of a sequence based on how many distinct subsequences are present in it. The lower that number, the less complex the sequence is. In order to illustrate this point, we provide an example for LCS(3). Consider two sequences $x = AAAAAAAT$ and $y = TTTTAAAA$. We have that $\mathrm{LCS}(3) = \frac{2}{8} = 0.25$ for $x$ since $AAA$ and $AAT$ are the only distinct 3-mers that occur in it. $\mathrm{LCS}(3) = \frac{4}{8} = 0.5$ for $y$, since $TTT$, $TTA$, $TAA$ and $AAA$ are the only distinct 3-mers in it. The first sequence is less complex than the second.

#### 2.1.2 Information theoretic

The empirical entropy $H_0$ of a sequence $x$ is defined as follows:

$$H_0(x) = -\sum_{i=1}^{|\Sigma|} \frac{n_i}{n} \log_2 \frac{n_i}{n},$$

where $n_i$ is the number of occurrences of symbol $a_i$ in $x$. It is worthy of mention that there is an important difference between empirical

entropy and the entropy defined in a probabilistic setting (Cover and Thomas, 1991). Indeed, as detailed in (Ferragina *et al.*, 2005), Shannon entropy is an expected value taken on a probabilistic process that may emit a possibly infinite ensemble of sequences, while empirical entropy is defined point-wise for any sequence: it measures the amount of information needed to optimally encode it, without any reference to 'a probabilistic model' generating sequences. That is, it is a punctual and intrinsic measure of information characterizing a sequence alone, rather than a measure of uncertainty characterizing a model generating sequences. As in the domain investigated here no obvious generative model is available, a punctual measure of information content seems to be best suited for the experiments performed here. Clearly, the lower the value of $H_0(x)$, the less complex $x$ is, all other things being equal. In order to exemplify this, consider again $x = AAAAAAAT$ and $y = TTTTAAAA$. $H_0(x) = 0.535$ and $H_0(y) = 1$. Again, $x$ is less complex than $y$.

It is also well known that entropy estimation, in particular for DNA sequences, is a very rich area of investigation. The interested reader may find relevant methodologies in (Giancarlo *et al.*, 2009, 2012, 2014). Therefore, it is quite natural to ask whether known techniques would bring better results with respect to the very simple empirical entropy used here, in particular when one resorts to compressive estimates of entropy. That is, estimation via the use of data compression programs. During the preliminary stages of this study, this possibility has been considered. In particular, experiments (receiver operating curve (ROC) analysis, described in Section 3.3) have been performed with the use of higher order empirical entropies and two data compressors: XM (Cao *et al.*, 2007) and Arithmetic Coding (Witten *et al.*, 1987). The first data compressor is among the best for DNA sequences. The second is quite effective and also offer the advantage to have specific parameters that control how fast the compressor learns statistics about the sequence to be compressed. Although the results of the experiments were good (data not shown and available upon request), they were no better than the ones obtained with the use of $H_0$. Therefore, the full set of experiments was done with the use of this measure only, which has also the advantage of being simple and fast to compute.

### 2.2 An *in silico* method to distinguish between nucleosome enriched and depleted genomic regions

Assume one is given two sets of genomic sequences coming from the same genome. The first is composed of NDR, while the second of NER. Moreover, assume one is also given an algorithm $\mathcal{A}$ that takes as input a sequence $x$ and assigns a *score* to $x$. Intuitively, such a score is an assessment of how likely it is for $x$ to be a NER or a NDR. In order to test whether $\mathcal{A}$ can distinguish NDR from NER, one can proceed as follows. All sequences in NDR are assigned a class label of zero and all sequences in NER are assigned a class label of one. We anticipate that, in terms of complexity measures, the meaning of such an assignment is that one expects NDR regions to be much less complex than NER ones. Then, one uses $\mathcal{A}$ to assign a score to each sequence in NER $\cup$ NDR. Now, the well known ROC analysis (Hanley and McNeil, 1982) can be used to establish how well $\mathcal{A}$ distinguishes NDR from NER, based on the produced scores. Indeed, the resulting Area Under the Curve (AUC) takes values in [0, 1] with the following meaning: (i) a value below 0.5 indicates that in order to get a correct classification one needs to invert the class labels; (ii) the further away the AUC is from 0.5, a value indicating random classification, and close to one, a value indicating a perfect classification, the better the ability of $\mathcal{A}$ to distinguish NDR from NER.

Such a mode of operation is referred to as *full scores*, since each sequence $x \in$ NER $\cup$ NDR is assigned a score obtained by processing the entire sequence. It results convenient to define also a score that is normalized with respect to the sequence length. This mode of operation is referred to as *normalized scores* and it is described next. Assume that $x$ has length at least 147 bp. A sliding window of 147 bp sweeps $x$ from left to right, and a score, as computed by $\mathcal{A}$ with input the subsequence 'in the window', is assigned to the leftmost position of $x$ covered by the window. Therefore, one obtains $n - 147$ values. In order to assign a score to $x$, one needs to choose a representative value based on the $n - 147$ available ones. Natural candidates are: the average, the minimum and the median of the given set of values. It is well known that the median of a set of values is a statistically robust synoptic value for the entire set and, in fact, having conducted experiments with all three of them, the median yields the best results. When $n < 147$, one computes its score via $\mathcal{A}$ as a whole and that score is assigned to it. The choice of a window of length 147 is related to the typical sequence length forming a nucleosome core.

### 2.3 *In silico* construction of nucleosome occupancy maps

Intuitively, a nucleosome occupancy map, for a given genome, provides a value that can be seen as an *in silico*, *in vitro* or *in vivo* estimate of the 'probability' that a given genomic position is covered by a nucleosome, i.e. an occupancy value. For the convenience of the reader, we point out that a nucleosome positioning map is much more specific than a corresponding occupancy map, since it provides an estimate of how likely it is for a given genomic position to be the initial position (or, in alternative, the center) of a nucleosome. Methodologically, those terms are made precise by Kaplan *et al.* (2010), although pragmatically for nucleosome occupancy maps one resorts to estimates based on the experimental data rather exact probability distributions. When no ambiguity arises, we refer to a nucleosome occupancy map simply as a *map*. Obviously, the same genome can have different maps, depending on which occupancy value estimation method is used. Here we concentrate on how to obtain a map with the use of the generic algorithm $\mathcal{A}$ described in Section 2.2. Assume that the genome of which the map has to be built is divided into maximal regions of contiguous bases $R = \{[s_1, e_1], [s_2, e_2], \ldots, [s_q, e_q]\}$, where the interval endpoints naturally indicate the start and end genome coordinate of each region, respectively. Each region in $R$ is swept, from left to right, by a window of length 147. The corresponding sequence is given in input to $\mathcal{A}$ and the value returned in output is assigned to the genomic position aligned with the center of the window, e.g. when $[s_1, e_1]$ is swept, the result is a map for the genomic positions in $[s_1 + 73, e_1 - 73]$.

### 2.4 Statistics on the level of agreement between two maps via binning

Because maps can be seen as numeric vectors, the degree of agreement between two different maps of the same organism can be assessed by computing standard correlation coefficients. However, care must be exercised. Indeed, since the comparison is made on a genome-wide scale, it implies estimating the correlation of a very large number of points. In those circumstances, some correlation measures, e.g. Pearson coefficient, may suffer from what is known as "*the most influential point effect*". That is, relatively few points may be responsible for a good correlation. Such a potential problem has been specifically mentioned by Stein *et al.* (2010) as a criticism

to the way in which Kaplan *et al.* (2008) assessed the correlation between maps in their study. The approach described below accounts for this criticism.

Let $R_1, R_2, \ldots, R_k$ be a collection of maps, given in the format of sets of intervals, of the same genome. Each of them can be obtained either via the methods described in Section 2.3 or by other techniques, even *in vivo* and *in vitro*. Assume that we are interested in establishing the level of relatedness between each pair of maps. In that case, all of them must refer to the same set of genomic positions, i.e. genomic areas covered by one map but not present or with an undefined occupancy value in another must be eliminated. Intuitively, one needs a simple 'superimposition' of all maps, from which a projection is taken of the genomic coordinates having a value assigned to them in all maps. From such a projection, deriving the set of intervals common to all maps is straightforward and left to the reader.

Now, the level of agreement between $R_i$ and $R_j$ is established according to the following heuristic procedure in which, intuitively, each interval is divided in small pieces for comparison. The length of the pieces should be such that the correlation between two pieces must be statistically significant, avoid the most influential point effect and guarantee that the maps are not divided into too many pieces, resulting in a slowdown of the computational process associated to the comparison of maps. In this study, given the data being used, a length of 1000 seems reasonable. However, we choose 1029, i.e. the smallest integer greater that 1000 and divisible by 147, since that would allow to have an 'integral' number of nucleosome centers in each piece. More formally, each interval $[s, e]$ obtained as outlined above is processed with the use of a sliding window of size 1029. It sweeps the interval from left to right. The correlation between portions of the maps $R_i$ and $R_j$ in the window is computed and assigned to the corresponding genomic region. Once that each interval has been processed, the following binning procedure is used, in order to get a synoptic rendering of the correlation of the entire two maps, assuming that the correlation has been computed with a method that returns a value in $[-1, 1]$. That interval is partitioned into subintervals, referred to as bins, and each genomic region is assigned to the bin comprising the correlation value computed for that region. If the bins are picked in such a way to represent qualitatively increasing levels of correlation, i.e. bad, poor, fair, good, very good, excellent, the end result of this procedure is the production of accurate statistics of how many relatively short genomic regions fall into the qualitative levels of correlation represented by the bins.

## 3 Results and discussion

Kaplan *et al.* (2008) intended to establish to what extent the DNA sequence determines nucleosome positions in eukaryotes. Towards that end, the main part of their strategy was to show that an *in vitro* nucleosome occupancy map was correlated with an *in vivo* one. Here we want to establish that the DNA sequence, considered as a combinatorial object, has enough information about nucleosome occupancy in the corresponding genome. Therefore, our strategy is as in (Kaplan *et al.*, 2008), with the use of the complexity formulas presented in Section 2.1 that have the role of quantifying the sequence information. In particular the same maps are used, with the addition of two (see Section 3.1). Two computational experiments, nearly *verbatim* replicas of the ones described in the mentioned article, are performed: correlation analysis between the map of an organism and the corresponding one obtained by computational methods (see Section 3.2) and ROC analysis to assess how well NERs and NDRs can be classified by a computational method (see Section 3.3).

### 3.1 Maps

Five nucleosome occupancy maps are used in this study, the first three common to the study by Kaplan *et al.* (2008). Namely, the normalized *in vitro* and *in vivo Saccharomyces cerevisiae* maps, the adjusted occupancy *Caenorhabditis elegans* map by Valouev *et al.* (2008) (chromosome 2), and the *Drosophila melanogaster* maps by Mavrich *et al.* (2008a) (only chromosome 2). In particular, from this latter map, two have been extracted, one for the left and the second for the right arm of the chromosome. Individually, they are referred to as `SCvitro`, `SCvivo`, `CE`, `DM-2L` and `DM-2R`, respectively. Additional details regarding them are given in Supplementary Section S1. The inclusion of a map for *D.melanogaster* is justified by our interest in checking our hypothesis on a eukaryotic organism more complex than *S.cerevisiae* and *C.elegans*. It is also worth pointing out that `DM-2R` and `DM-2L` are much more challenging maps for this study than the ones being used here for the other two organisms. Indeed, as opposed to them, where each genomic position is given an estimate of the chance of that position being covered by a nucleosome, whereas `DM-2R` and `DM-2L` are binary maps. That is, each genomic position is assigned a value of one, when that position is covered by a nucleosome, or zero, when it is not.

### 3.2 Correlation between complexity-based maps and the corresponding experimental ones of *S.cerevisiae* and *C.elegans*

We carry out a set of correlation experiments between maps, via the approach outlined in Section 2.4, and with the use of the Spearman rank correlation coefficient. Such a choice is motivated by the fact that it provides a statistically robust non-parametric estimate of correlation, which is largely immune to the presence of outliers. For completeness, we also carry out the same experiment with the use of the Pearson correlation coefficient. As for the measures to be tested, we use $H_0$, LCS($k$), at $k = 7$, and LC($k$), at $k = 17$, (those are the values of $k$ that maximise the AUC value of the corresponding measure in the ROC analysis—see Section 3.3). Moreover, we also use the probabilistic model devised by (Kaplan *et al.*, 2008), here referred to as `KModel`. As opposed to the measures proposed here, that consider a sliding window of 147 bp, `KModel` is allowed to use the entire genome as input for its prediction. Finally, we exclude both `DM-2L` and `DM-2R` because their binary form does not lend itself to meaningful correlation studies, based on monotonicity or linear relations between numeric vectors.

The results are reported in Tables 1–3 for the Spearman rank correlation coefficient. The analogous results for the Pearson correlation coefficient are reported in Supplementary Tables S2–S4. As for the statistical significance of those correlations (both Spearman and Pearson), since we are in a case of repeated tests of hypothesis, we have computed a $p$-value with a standard Bonferroni correction for the cumulative number of tests performed in the interval [0.3, 1.0]. The remaining intervals are somewhat irrelevant. The significance level is below $0.80264 \times 10^{-19}$.

In order to fully highlight the implications of those results, the discussion of an example is helpful. From the Spearman rank correlation in the interval [0.6, 0.9) between $H_0$ and `SCvivo`, we have randomly selected one of the genomic regions of 1029 bp falling into it. For that region, Figure 1 displays the plots of the occupancy values of `SCvivo` and `SCvitro`. As for the computational methods, it displays the plots obtained by applying them to that region,

**Table 1.** Spearman rank correlation SCvivo

| | [−1, 0) | [0,0.3) | [0.3, 0.6) | [0.6, 0.75) | [0.75, 0.9) | [0.9, 1] |
|---|---|---|---|---|---|---|
| (SCvivo, $H_0$) | 2.77% | 9.48% | 31.77% | 28.28% | 25.38% | 2.34% |
| (SCvivo, LCS(7)) | 14.29% | 48.21% | 25.22% | 9.02% | 3.20% | 0.06% |
| (SCvivo, LC(17)) | 9.38% | 21.66% | 39.27% | 20.11% | 9.40% | 0.27% |
| (SCvivo, SCvitro) | 1.22% | 6.72% | 26.54% | 27.69% | 32.09% | 5.74% |
| (SCvivo, KModel) | 0.12% | 0.88% | 11.47% | 23.26% | 48.77% | 15.50% |

The interval [−1, 1] has been divided into six intervals, reported on top of the table, each corresponding to a qualitative 'value' of the Spearman rank correlation, as follows (from left to right): bad, poor, fair, good, very good, excellent. The first column in the table indicates the experiment whose result is reported, in the form (map, predictor). The predictor, i.e. the method used to assign scores to NER and NDR is either $H_0$, LCS(7), LC(17), SCvitro or KModel. For each row, the percentage of regions that fall within that level of correlation is given in the corresponding column position.

**Table 2.** Spearman rank correlation SCvitro

| | [−1, 0) | [0, 0.3) | [0.3, 0.6) | [0.6, 0.75) | [0.75, 0.9) | [0.9, 1] |
|---|---|---|---|---|---|---|
| (SCvitro, $H_0$) | 5.21% | 15.48% | 35.13% | 25.01% | 18.10% | 1.07% |
| (SCvitro, $LCS(7)$) | 15.58% | 49.74% | 24.19% | 7.50% | 2.14% | 0.08% |
| (SCvitro, $LC(17)$) | 12.06% | 24.91% | 39.91% | 17.01% | 5.99% | 0.12% |
| (SCvitro, KModel) | 1.27% | 7.23% | 29.99% | 29.57% | 28.28% | 3.66% |

**Table 3.** Spearman rank correlation CE

| | [−1, 0) | [0, 0.3) | [0.3, 0.6) | [0.6, 0.75) | [0.75, 0.9) | [0.9, 1] |
|---|---|---|---|---|---|---|
| (CE, $H_0$) | 16.99% | 22.94% | 33.22% | 15.59% | 10.51% | 0.75% |
| (CE, LCS(7)) | 18.22% | 45.49% | 24.19% | 8.09% | 3.87% | 0.14% |
| (CE, LC(17)) | 16.92% | 25.90% | 34.04% | 15.15% | 7.69% | 0.30% |
| (CE, KModel) | 12.74% | 21.53% | 33.52% | 17.67% | 12.99% | 1.55% |

according to the methodology already discussed. We take as reference the *in vivo* curve (top of the Figure). As it is evident, the corresponding curve of the *in vitro* map closely follows the *in vivo* one. An analogous observation can be made regarding the curves generated by the computational procedures. The example seems to suggest that, given in input a sequence, the output of the mathematical formulas and the computational procedure are able to generate curves for that sequence that are in agreement with the corresponding ones obtained from *in vivo* and *in vitro* maps. Tables 1–3 here and Supplementary Tables S2–S4 quantify to what extend such an agreement hold for the maps involved in this study.

With the previous example in mind, an immediate conclusion that can be drawn from those results is that the good correlation between SCvivo and SCvitro established by Kaplan *et al.* (2008) is also found here, with a method and a correlation coefficient that does not suffer from the most influential point effect. The comparison between KModel and the formulas proposed here has important methodological implications, discussed next, regarding the nature of those formulas and what can be learned from a sequence about its nucleosome occupancy.

KModel has been trained on SCvitro. Therefore, it is not surprising that it performs better than the methods proposed here on that particular map and on the strongly related one SCvivo. However, for CE, such an advantage for KModel disappears and its performance is in line with that of the measures proposed here. Those observations seem to suggest that those formulas have a 'foundational nature' in quantifying nucleosome occupancy from sequence, providing some sort of 'baseline' common to at least two organisms. Machine Learning procedures can certainly profitably extract relevant features when trained on a specific genome. However, such training may not always be possible, and in those cases, they may not do much better than the 'baseline' procedures proposed here. That is, training on one genome in the hope to perform well on another does not seem to be so easy, in view of the results of this Section.

Although numerically different, the experiments with the Pearson correlation coefficient, reported in the Supplementary Material, also support the above analysis.

### 3.3 ROC curve analysis

For conciseness, it results convenient to describe in detail first the ROC curve analysis for one map only: SCvivo.

For each of the thresholds $t = 0, 0.25, 0.50, 0.75$, the corresponding $\mathrm{NER}_t$ and $\mathrm{NDR}_t$ are extracted according to the method described in Supplementary Section S2. With reference to the procedure outlined in Section 2.2, $\mathrm{NER}_t \cup \mathrm{NDR}_t$ is processed in full and normalized score modes of operation, respectively, with $\mathcal{A} = H_0$; $\mathcal{A} = LCS(k), k$ in [2, 36]; $\mathcal{A} = LC(k), k$ in [2, 36], respectively. For the normalized score mode of operation, only the results obtained by using the median to assign a score to the sequences in $\mathrm{NER}_t \cup \mathrm{NDR}_t$ are reported (see Section 2.2 again). As for the minimum and the average, we limit ourselves to mention that they also ensure a good classification (data not shown and available upon request), however inferior with respect to the median.

The same experiments have been performed for all of the other maps mentioned in Section 3.1. No thresholding has been applied for *D.melanogaster* since it is meaningless for that map. The complete results are reported in Table 4 for $H_0$. As for LC and LCS, Figure 2 provides the relevant AUC plots, as a function of $k$, but for the case of threshold zero. The corresponding plots for all the thresholds are reported in Supplementary Figures S1–S3. For LCS, Table 5 provides the value of both the maximum AUC and of the $k$ for which it is achieved in the mentioned plots. The corresponding data for LC is reported in Supplementary Table S1. In all Tables, the analogous results from the study by Kaplan *et al.* (2008) are also reported for comparison.

As it is evident, all three complexity measures distinguish quite well NER from NDR for all three organisms studied here. It is also to be appreciated that the AUC values go up in accordance with the

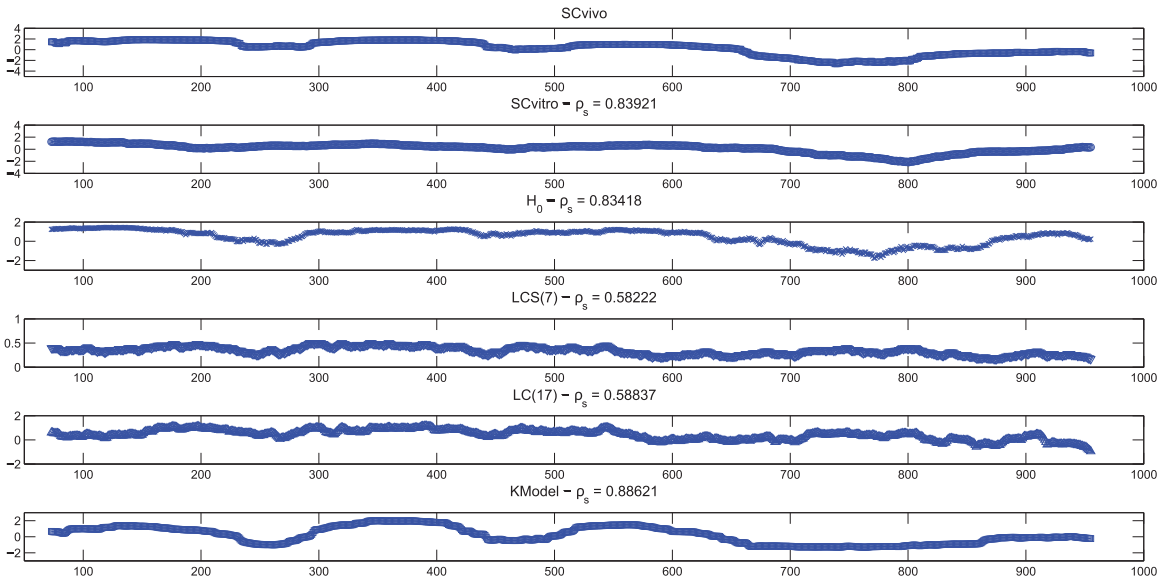


**Fig. 1.** For a genomic region selected as specified in the main text, the plots of the occupancy values for SCvivo and SCvitro (top two). The abscissa indicates a position in the sequence and the corresponding ordinate provides the occupancy value. The remaining plots provide curves analogous to the ones just described. However, the 'occupancy' value of a position has been determined by one of the computational methods considered in this article. Taking as reference SCvivo, the value of the Spearman rank correlation coefficient is also indicated for each curve

**Table 4.** $H_0$ separates well NER from NDR in three model organisms and an *in vitro* map for *S.cerevisiae*

| | 0 | 0.25 | 0.5 | 0.75 |
|---|---|---|---|---|
| (SCvivo, $H_0$) | 0.904 (0.844) | 0.942 (0.887) | 0.967 (0.916) | 0.979 (0.934) |
| (SCvitro, $H_0$) | 0.938 (0.889) | 0.964 (0.925) | 0.980 (0.947) | 0.987 (0.962) |
| (CE, $H_0$) | 0.818 (0.744) | 0.866 (0.787) | 0.900 (0.822) | 0.910 (0.838) |
| (DM-2L, $H_0$) | 0.626 (0.578) | — | — | — |
| (DM-2R, $H_0$) | 0.638 (0.591) | — | — | — |
| (SCvivo, SCvitro) | 0.871 | 0.923 | 0.956 | 0.974 |
| (SCvitro, KModel) | 0.953 | 0.983 | 0.996 | 0.999 |
| (CE, KModel) | 0.763 | 0.825 | 0.870 | 0.890 |

The legend for the first column of the table is as in Table 1. The remaining columns correspond to the four thresholds. The AUC value obtained via ROC analysis corresponding to an experiment is reported as a numeric value in the entry summarizing the experiment. For the first five rows, the value refers to the AUC of that experiment in normalized scores mode, while that in full scores mode is provided in parenthesis. For comparison, the last three rows report the results of the experiments conducted in Kaplan *et al.* (2008) (full scores mode only).

threshold value. That is, as NER and NDR become more and more representative, the complexity measures distinguish them better and better. Moreover, the AUC values obtained with the use of the complexity measures are competitive with respect to the computational model designed by Kaplan *et al.* (2008), given the simplicity of those measures and the total absence of a learning phase on a training set of sequences. Indeed, the results obtained here confirm, from a different point of view, what has been stated at the end of Section 3.2.

In terms of structural organization of sequence features for nucleosome positioning, we also get brand new insights. When one considers the ROC analysis curves for LC and LCS, it is evident that the *k*-mers organization of the underlying sequences follows the same trend in favoring/disfavoring nucleosome positioning. Moreover, the ROC analysis together with the correlation analysis bring to light that the DNA-encoded organization of eukaryotic genomes is much easier to grasp in *S.cerevisiae* than in *C.elegans*. Such an 'encoding' is present in *D.melanogaster* but it either has a much less prominent role with respect to the other two, much simpler, organisms, or it is more complex to describe by a closed form formula. Quite remarkably, at least for *S.cerevisiae* and *C.elegans*, those results are in agreement with a recent information-theoretic study by Giancarlo *et al.* (2015). Using the same data as here, the study showed that NER in the mentioned organisms have minor differences in their information-theoretic content, yet those differences are statistically very significant. The same holds for NDR. It is also of interest to note that the best AUC values of the various methods on the various organisms decrease in accordance with the genetic density of the latter, supporting the fact that more complex organisms may hide more complex mathematical rules with respect to the ones proposed here.

## 3.4 Time performance of our methods

Although not the major objective of this research, the simplicity of the formulas proposed here has also the remarkable side effect to provide efficient procedures for the computation of nucleosome density maps. The algorithmic engineering details, as well as the full results of an experiment conducted on CE, are reported in Supplementary Section S3. Here we limit ourselves to report that the timing for $H_0$, LCS(7) and LC(17) (the values of $k$ for which those measures perform best), executed in normalized scores mode, are $0.490 \times 10^2$, $0.239 \times 10^3$ and $0.108 \times 10^4$ seconds, respectively.

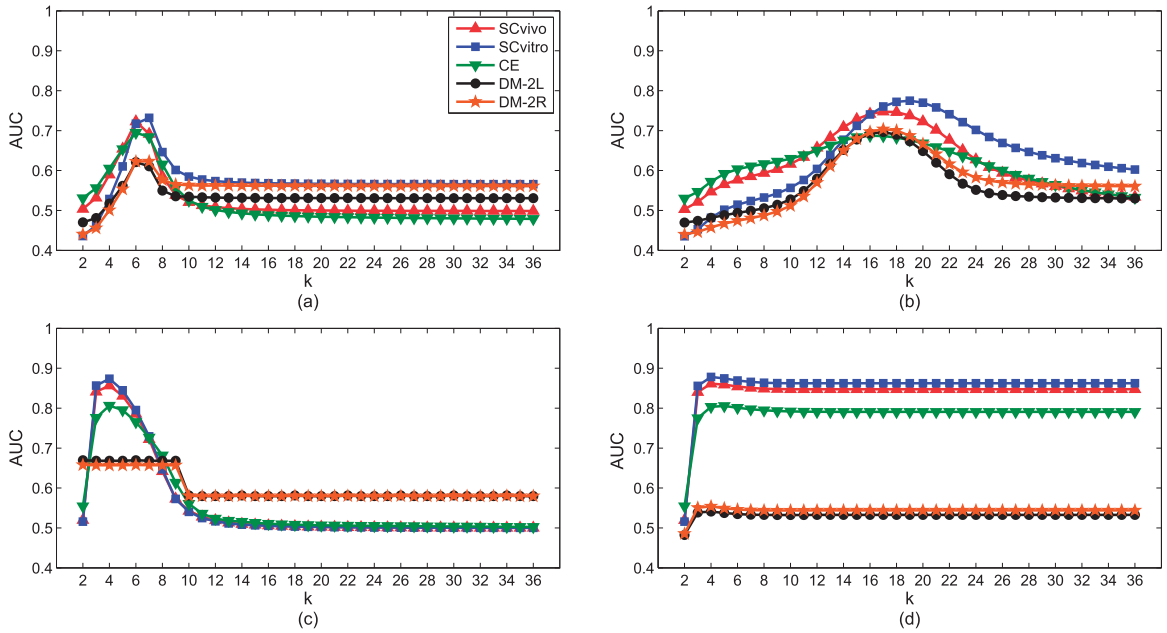


**Fig. 2.** The plots of the AUC values obtained when LCS(*k*) and LC(*k*) are used as predictors to separate NER from NDR, for *k* in [2,36]. The threshold is zero. The correspondence between the plots and the maps is given in (a). **(a)** LCS in full scores mode; **(b)** LC in full scores mode; **(c)** LCS in normalized scores mode; **(d)** LC in normalized scores mode

**Table 5.** LCS separates well NER from NDR in three model organisms and an *in vitro* map

| | 0 | 0.25 | 0.5 | 0.75 |
|---|---|---|---|---|
| (SCvivo, LCS) | 0.856, 4 (0.723, 6) | 0.896, 4 (0.769, 6) | 0.926, 4 (0.802, 6) | 0.945, 4 (0.824, 6) |
| (SCvitro, LCS) | 0.874, 4 (0.732, 7) | 0.905, 4 (0.760, 6) | 0.928, 4 (0.795, 6) | 0.944, 4 (0.819, 6) |
| (CE, LCS) | 0.806, 4 (0.695, 6) | 0.853, 4 (0.752, 6) | 0.879, 4 (0.786, 6) | 0.874, 4 (0.815, 5) |
| (DM-2L, LCS) | 0.670, 2 (0.621, 6) | — | — | — |
| (DM-2R, LCS) | 0.658, 2 (0.623, 7) | — | — | — |
| (SCvivo, SCvitro) | 0.871 | 0.923 | 0.956 | 0.974 |
| (SCvitro, KModel) | 0.953 | 0.983 | 0.996 | 0.999 |
| (CE, KModel) | 0.763 | 0.825 | 0.870 | 0.890 |

The first column in the table indicates the experiment whose result is reported, in the form (map, predictor). The predictor, i.e. the method used to assign scores to NER and NDR is either SCvitro or KModel or LC. The remaining columns correspond to the four thresholds. The AUC value obtained via ROC analysis corresponding to an experiment is reported as a numeric value in the entry summarizing the experiment, in the form $(\text{AUC}, k_{\max})$, together with a value of *k* where it is maximum. For the first three rows, the reported values refer to the experiment in normalized scores mode, while those in full scores mode are provided in parenthesis. For comparison, the last three rows report the results of the experiments conducted in Kaplan *et al.* (2008) (full scores mode only).

For comparison, KModel takes $0.289 \times 10^4$ seconds. Therefore, the measures proposed here are, in their best predictive setting, always faster than KModel and the best performing one by two orders of magnitude. It is worth pointing out that in full score mode, the performances of our measures are $0.384 \times 10$, $0.188 \times 10^2$ and $0.650 \times 10^2$ seconds, respectively. However, a comparison between the performance of our measures in that setting and KModel is inappropriate because our methods produce a single value for each input sequence while KModel produces a value for each position in an input sequence.

## 4 Conclusions

When the findings of Sections 3.2 and 3.3 are placed in the context of the current state of the art on the identification of what plays a role in determining nucleosome organization in eukaryotes, we have the following fundamental advances.

We provide a set of formulas that have the same nice properties of the ones available for the Barrier Model (Kornberg and Stryer, 1988; Möbius and Gerland, 2010) and that apply to the sequence-specific model of nucleosome positioning. It is worth recalling that, prior to this work, the sequence-specific model was populated by a few sequence rules and Machine Learning procedures and that the existence of universal sequence dictated rules for nucleosome positioning was even dismissed (Valouev *et al.*, 2008). The fact that the formulas, proposed here, are well known in the Literature, and have not been previously considered in this context, adds value to our contribution.

The intent of the study by Kaplan *et al.* (2008) was to show that the sequence itself has enough 'information' to influence nucleosome positioning and that such a genomic organization is 'DNA encoded' in eukaryotes. The term 'information' in that setting has to do with Biology and the term 'encoding' is semantically clear. Our results establish that the purely information theoretic content and combinatorial richness of subsequences within a sequence, i.e. two measures

of information and complexity in Computer Science, are both correlated to the biological information alluded to by Kaplan *et al.* (2008). Moreover, our formulas are the first that provide an interpretation of the mathematical nature of the 'encoding' discovered by the mentioned authors.

## Acknowledgements

The authors would like to thank Noam Kaplan for providing part of the nucleosome occupancy maps used in this study, Simona Panni and Simona Ester Rombo for comments and discussion based on an earlier version of this manuscript and to Chiara Romualdi and the reviewers for comments that greatly helped in revising the manuscript. Finally, R.G. is very grateful for having been so fortunate to have many pleasant conversations on this topic with the late Jonathan Widom.

## Funding

FIRB Project 'Bioinfomatica per la Genomica e la Proteomica', and FIRB Project 'Algoritmi per la Scoperta ed il Ritrovamento di Patterns in Strutture Discrete, con Applicazioni alla Bioinformatica', both financed by Italian Ministry of Education, Universities and Research (to R.G.). Additional support to R.G. has been provided by Progetto di Ateneo dell' Universitá degli Studi di Palermo (2012-ATE-0298) 'Metodi Formali e Algoritmici per la Bioinformatica su Scala Genomica'. Fondazione Telethon Grant (TCR09002), Giovanni Armenise Foundation CDA Grant, AIRC Grant (IG12764), MIUR-CRN EPIGEN Grant (to D.F.V.C.).

*Conflict of Interest*: none declared.

## References

Cao,M.D. *et al.* (2007) A simple statistical algorithm for biological sequence compression. In *Proceedings of the IEEE Data Compression Conference (DCC)*. IEEE Computer Society, pp. 43–52.

Cover,T.M. and Thomas,J.A. (1991) *Elements of Information Theory*. Wiley-Interscience, New York City, NY, USA.

De Luca,A. and Varricchio,S. (1999) *Finiteness and Regularity in Semigroups and Formal Languages*. EATCS Monographs on Theoretical Computer Science. Springer, Heidelberg, Germany.

Ferragina,P. *et al.* (2005) Boosting textual compression in optimal linear time. *J. ACM*, **52**, 688–713.

Giancarlo,R. *et al* . (2009) Textual data compression in computational biology: a synopsis. *Bioinformatics*, **25**, 1575–1586.

Giancarlo,R. *et al.* (2012) Textual data compression in computational biology: Algorithmic techniques. *Comput. Sci. Rev.*, **6**, 1–25.

Giancarlo,R. *et al* . (2014) Compressive biological sequence analysis and archival in the era of high-throughput sequencing technologies. *Brief. Bioinformatics*, **15**, 390–406.

Giancarlo,R. *et al.* (2015) Epigenomic k-mer dictionaries: Shedding light on how sequence composition influences nucleosome positioning *in vivo*. *Bioinformatics*, **31**, 2939–2946.

Hanley,J.A. and McNeil,B.J. (1982) The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, **143**, 29–36.

Kaplan,N. *et al.* (2008) The DNA-encoded nucleosome organization of a eukaryotic genome. *Nature*, **458**, 362–366.

Kaplan,N. *et al.* (2010) Contribution of histone sequence preferences to nucleosome organization: proposed definitions and methodology. *Genome Biol.*, **11**, 140.

Kornberg,R. and Stryer,L. (1988) Statistical distributions of nucleosomes: nonrandom locations by a stochastic mechanism. *Nucleic Acids Res.*, **16**, 6677–6690.

Kornberg,R.D. (1981) The locations of nucleosomes in chromatin:specific or statistical?. *Nature*, **292**, 579–580.

Li,M. and Vitányi,P.M.B. (1997) *An introduction to Kolmogorov Complexity and its Application*. Springer-Verlag, New York, NY, USA.

Mavrich,T. *et al.* (2008a) Nucleosome organization in the *Drosophil a* genome. *Nature*, **453**, 358–364.

Mavrich,T.N. *et al.* (2008b) A barrier nucleosome model for statistical positioning of nucleosomes throughout the yeast genome. *Genome Res.*, **18**, 1073.

Minary,P. and Levitt,M. (2014) Training-free atomistic prediction of nucleosome occupancy. *Proc. Natl. Acad. Sci.*, **111**, 6293–6298.

Möbius,W. and Gerland,U. (2010) Quantitative test of the barrier nucleosome model for statistical positioning of nucleosomes up- and downstream of transcription start sites. *PLoS Comput. Biol.*, **6**, e891.

Peckham,H. *et al.* (2007) Nucleosome positioning signals in genomic dna. *Genome Res.*, **17**, 1170–1177.

Segal,E. *et al.* (2006) A genomic code for nucleosome positioning. *Nature*, **442**, 772–778.

Stein,A. *et al.* (2010) Are nucleosome positions in vivo primarily determined by histoneDNA sequence preferences?. *Nucleic Acids Res.*, **38**, 709–719.

Tillo,D. and Hughes,T. (2009) G + C content dominates intrinsic nucleosome occupancy. *BMC Bioinformatics*, **10**, 442.

Trifonov,E. (1990) Making sense of the human genome. In: *Human Genome Initiative and DNA Recombination, volume 1 of Structure and Methods, Proceedings of the Sixth Conversation in the Discipline Biomolecular Stereodynamics*. Academic Press, Albany, NY, pp. 68–78.

Valouev,A. *et al.* (2008) A high-resolution, nucleosome position map of C. *elegans* reveals a lack of universal sequence-dictated positioning. *Genome Res.*, **18**, 1051–1063.

Witten,I.H. *et al.* (1987) Arithmetic coding for data compression. *Commun. ACM*, **30**, 520–540.