Information theory for Computational Biology

Michele Caselle – University of Torino and INFN
caselle@to.infn.it

Plan of the lecture

1. Introduction

2. Co-expression network: Mutual information and “Aracne”

3. Information conìtent of binding sequences: KL divergence and “Logos”

Boltzmann Entropy

Let us take an isolated system (Microcanonical Ensemble): all the microstates (microscopic configurations of the system) have the same probability

$$p = 1/\Omega$$

Where Ω is the total number of microstates.

Then following Boltzmann we can identify the **Entropy** of the system with:

$$S = k_B \log(\Omega)$$

where k_B is called the "Boltzmann constant".

Gibbs Entropy

If the probability distribution is non-trivial (non-isolated system) then The Boltzmann entropy is not the right choice. One must use the "Gibbs entropy"

$$S = -k_B \sum_{i=1}^{\Omega} p_i \log p_i$$

which reduces to the Boltzmann definition if one has an uniform probability distribution

$$p = 1/\Omega$$

Shannon Entropy

The Shannon entropy H(X) is similar the Gibbs entropy:

$$S = -k_B \sum_{i=1}^{\Omega} p_i \log p_i$$

$$H(X) = -\sum_{x} p(x) \log p(x)$$

where x is one of the possible values of the random variable X and the sum is on all the possible values of x. The Gibbs/Shannon entropy measures the "asymmetry" of the probability distribution. Hence it measures the "uncertainty" of the microstate or, equivalently the amount of information needed to fix it

Shannon Entropy

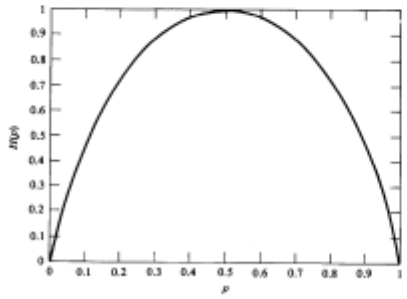
Example: one bit

$$X = 0 \quad 1-p$$

$$X = 1 \quad p$$

$$H(X) = -p\log(p) - (1-p)\log(1-p)$$

Maximum at p=0.5 (equiprobability)



Shannon Entropy

Example: four values

$X = a \quad p = 1/2$
$X = b \quad p = 1/4$
$X = c \quad p = 1/8$
$X = d \quad p = 1/8$

H(X)=7/4

We wish to determine X with the minimum number of binary questions

First question “Is X=a?”. This splits the probabilities in half. If the answer is no, the the second question is “Is X=b?”. The third question can be “Is X=c?”

Expected number of binary questions required is

$$1 \times \frac{1}{2} + 2 \times \frac{1}{4} + 3 \times \frac{1}{4} = \frac{7}{4} = H(X)$$

Shannon Entropy

Another (non-optimal) coding strategy

$$X = a \quad p = 1/2$$
$$X = b \quad p = 1/4$$
$$X = c \quad p = 1/8$$
$$X = d \quad p = 1/8$$

First question “Is X=d?”. If the answer is no, the the second question is “Is X=c?”. The third question can be “Is X=b?”

Expected number of binary questions required is

$$1 \times \frac{1}{8} + 2 \times \frac{1}{8} + 3 \times \frac{3}{4} = \frac{21}{8} \geq \frac{7}{4} = H(X)$$

Shannon Theorem

The minimum expected number of binary questions required to determine X lies between H(X) and H(X)+1

H(X) is a measure of the amount of information needed on average to describe the random variable.

A code is said optimal (with respect to the probability distribution p(x) of X) if saturates the entropy bound.

Joint Entropy

If X and Y are two random variables we can define the "Joint Entropy" as:

$$H(X,Y) = -\sum_x \sum_y p(x,y) \log p(x,y)$$

Relative Entropy (Kullback-Leibler Divergence)

Measure of the "distance" between probability distributions. It measures the inefficacy of using q(x) instead of p(x)

$$D(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

For example, let us consider a coding scheme optimal for $q(x)$, and let us use it to describe a p-distributed variable. On average we will need $H(p) + D(p||q)$ bits (binary questions) to describe the variable, whilst $H(p)$ is the number of bits needed using a code optimized for p.

Relative Entropy: Properties

- $D(p||q)$ is not exactly a distance because it is not symmetric and does not satisfy the triangle inequality, but it has two properties which are typical of well defined distances
  - it is always non-negative
  - it is zero if and only if $p = q$
- If there is a value of x for which $p(x) > 0$ and $q(x) = 0$ then $D(p||q) = \infty$

Mutual Information

The Mutual Information $I(X;Y)$ is the Relative entropy (the "distance") between the joint distribution and the product distribution

$$I(X;Y) = D(p(x,y)||p(x)p(y))$$

$$I(X;Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

$$I(X;Y) = H(X) + H(Y) - H(X,Y)$$

$I(X;Y)$ measures the amount of information that one random variable contains about another random variable. If X and Y are scorrelated then $H(X,Y) = H(X) + H(Y)$ and $I(X;Y) = 0$

Application to sequences of symbols

Goal: use Entropy–like quantities as tools to identify sequence structures.

Definitions: In order to describe the structure of a given string of length L using an alphabet of λ letters $\{A_1 A_2 \dots A_\lambda\}$ we introduce the following notations:

Let $A_1 A_2 \dots A_n$ be the letters of a given substring of length $n \leq L$. We define the probability to find in a string a block of length n (subword of length n) with the letters $A_1 \dots A_n$ as

$$p^{(n)}(A_1 \dots A_n).$$

Entropies: We shall use the following quantities:

1. Shannon Entropy:

$$H_1 = -\sum_{i=1}^{\lambda} p^{(1)}(A_i) \log p^{(1)}(A_i) \ ,$$

a few results: for white noise $H_1 = log(\lambda)$, which is a maximum. If all the symbols are the same, $H_1 = 0$.

2. Shannon word entropy

the entropy per word of length n is given by

$$H_n = -\sum p^{(n)}(A_1 \dots A_n) \log p^{(n)}(A_1 \dots A_n) \ ,$$

The sum is over λ^n entries. The limit

$$H_{met} = \lim_{n \to \infty} \frac{H_n}{n}$$

is usually called metric Entropy. In case of white noise it should be $\frac{H_n}{n} = log(\lambda)$ See figures below

3. The mutual information between far apart symbols. Let us define the probability of having a pair with $(n-2)$ arbitrary letters in between as

$$p^{(n)}(A_1, A_n) = p^{(n)}(A_1 \; ? \; ? \; ? \; ? \; A_n) \; .$$

Then we may define the mutual information as:

$$I(n) = \sum_{A_i A_j} p^{(n)}(A_i, A_j) \log \left(\frac{p^{(n)}(A_i, A_j)}{p^{(1)}(A_i) \cdot p^{(1)}(A_j)} \right) \; ,$$

We can use these quantities to study properties of the DNA sequence.

Mutual Information

The mutual information between different regions of DNA can be used to detect structural or functional properties of the DNA chain.
For instance distinguish coding from non coding regions by using MI to detect the period 3 in the coding region (see [Grosse et al., 2000])

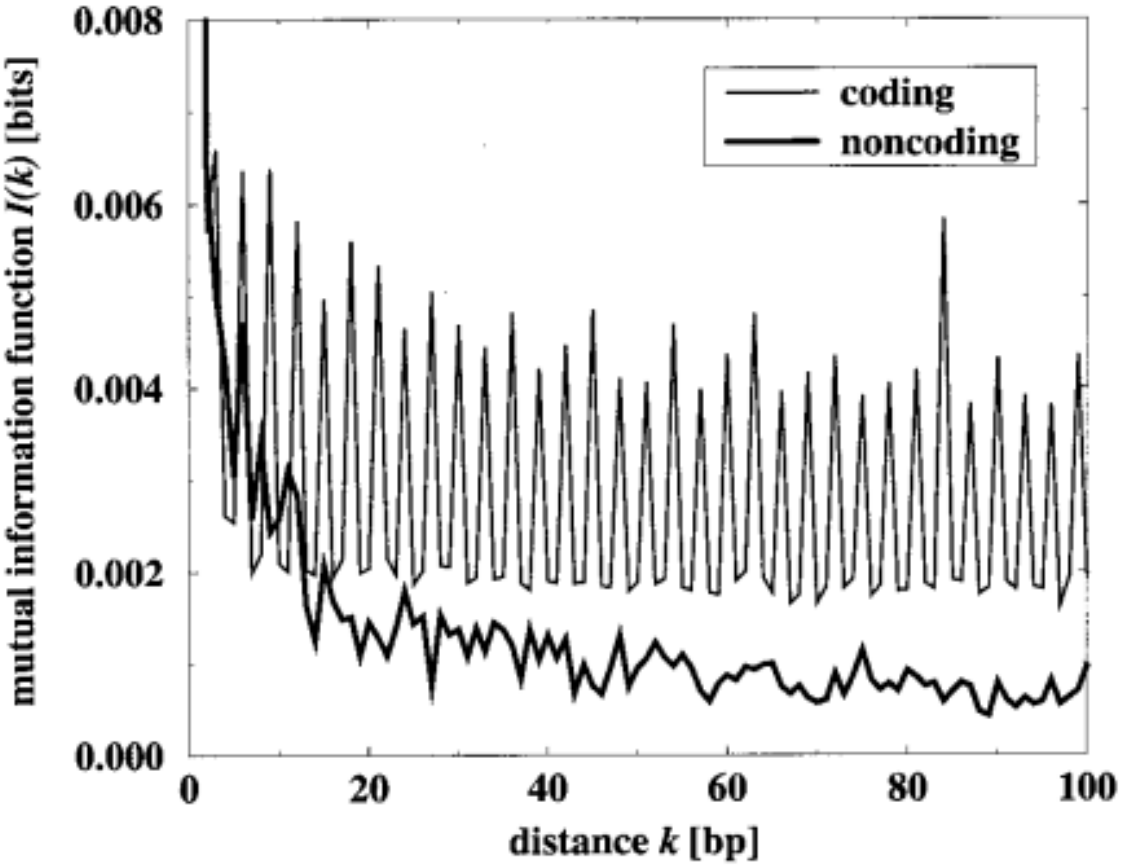
We choose $P_{ij}(k)$ to denote the joint probability of finding the pair of nucleotides n_i and n_j $(n_i, n_j \in \{A, C, G, T\})$ spaced by a gap of $k-1$ nucleotides, and we define $p_i \equiv \Sigma_j P_{ij}(k)$ and $q_j \equiv \Sigma_i P_{ij}(k)$. Then

$$\mathcal{I}(k) \equiv \sum_{i,j=1}^{4} P_{ij}(k) \log_2 \frac{P_{ij}(k)}{p_i q_j} \qquad (1)$$

quantifies the degree of statistical dependence between the nucleotides X and Y spaced by a gap of $k-1$ nucleotides,

Mutual Information

The mutual information between different regions of DNA can be used to detect structural or functional properties of the DNA chain.



Combining three stochastic variables: Synergy and Redundance

Information Theory allows to give a rigorous definition of intuitive concepts like Synergy and Redundancy. The natural extension to three variables of the mutal information is the "Interaction Information": $R(X, Y, Z)$ defined as

$$R(x, y, z) = I(x; y) - I(x; y|z)$$

where $I(x; y|z)$ is the "Conditional Mutual Information" defined as:

$$I(x; y|z) = \sum_x \sum_y \sum_z p(x, y, x) \log \frac{p(x, y|z)}{p(x|z)p(y|z)}$$

Main Properties

- R is invariant under permutations of the three variables
- R is related to the mutual information by:

$$R(x, y, z) = I(x; z) + I(y; z) - I(\{x, y\}; z)$$

- R describes the informational character of triplets of variables:
  - $R = 0$ Information Independence: $I(\{x, y\}; z) = I(x; z) + I(y; z)$
  - $R < 0$ Synergy: $I(\{x, y\}; z) > I(x; z) + I(y; z)$
  - $R > 0$ Redundancy: $I(\{x, y\}; z) < I(x; z) + I(y; z)$

Example

Suppose that we apply two input perturbations s_1 and s_2 to a system and measure the system response r, then

- if $R = 0$ this means that $I(\{s_1, s_2\}; r) = I(s_1; r) + I(s_2; r)$ i.e. the system is sensitives to completely independent features of the two inputs.
- if $R > 0$ then $I(\{s_1, s_2\}; r) < I(s_1; r) + I(s_2; r)$ i.e. the knowledge of both inputs conveys more information than treating them separately: r is a function of both s_1 and s_2 which act in a synergic way.
- if $R < 0$ then $I(\{s_1, s_2\}; r) > I(s_1; r) + I(s_2; r)$ i.e. one input is almost enough to understand the behaviour of the system and the second one is redundant.

Kolmogorov-Chaitin Entropy

.

The Kolmogorov-Chaitin entropy of a string of characters is the length (in bits) of the smallest program which produces as output the string.

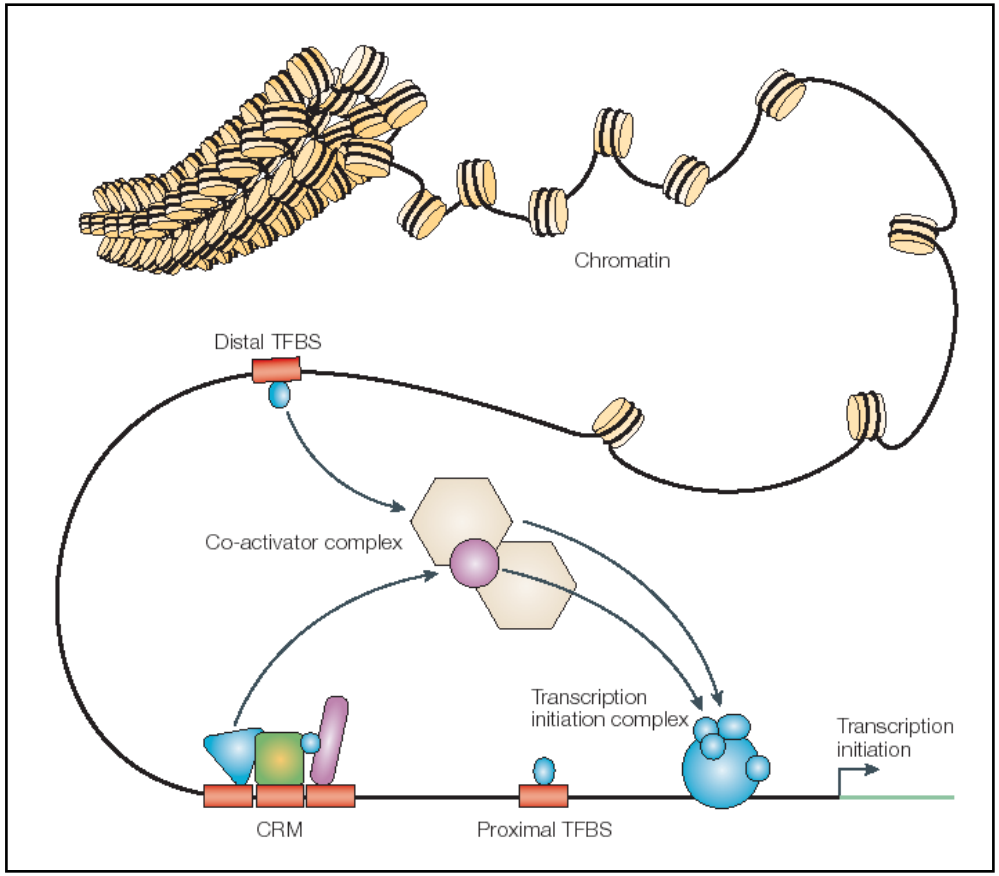
This idea is at the basis of the file compressors or zippers. A zipper takes a file and tries to transform it in the shortest possible file. The most popular compression algorithm is the Lempel and Ziv algorithm (LZ77) used by gzip, zip....

Kolmogorov-Chaitin Entropy

.

The LZ77 algorithm finds duplicated strings in the input data. More precisely it looks for the longest match with the beginning of the lookahead buffer and outputs a pointer to that match given by two numbers: a distance, representing how far back the match is in the sequence and the length of the match.

Transcription Factors



Coexpression Networks: Aracne

BMC Bioinformatics

BioMed Central

Proceedings

Open Access

ARACNE: An Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context

Adam A Margolin[1,2], Ilya Nemenman[2], Katia Basso[3], Chris Wiggins[2,4], Gustavo Stolovitzky[5], Riccardo Dalla Favera[3] and Andrea Califano*[1,2]

Mutual Information as a coexpression weight

MI Estimation

We estimate MI using a computationally efficient Gaussian Kernel estimator [12]. Given a set of two-dimensional measurements, $\vec{z}_i \equiv \{x_i, y_i\}$, $i = 1 \ldots M$, the JPD is approximated as $f(\vec{z}) = 1/M \sum_i h^{-2} G\left(h^{-1} \left| \vec{z} - \vec{z}_i \right|\right)$, where $G(\ldots)$ is the bivariate standard normal density. With $f(x)$ and $f(y)$ being the marginals of $f(\vec{z})$, the MI is:

$$I\left(\{x_i\},\{y_i\}\right) = \frac{1}{M} \sum_i \log \frac{f(x_i, y_i)}{f(x_i) f(y_i)} \qquad (2)$$

Data Processing Inequality: Pruning of the network

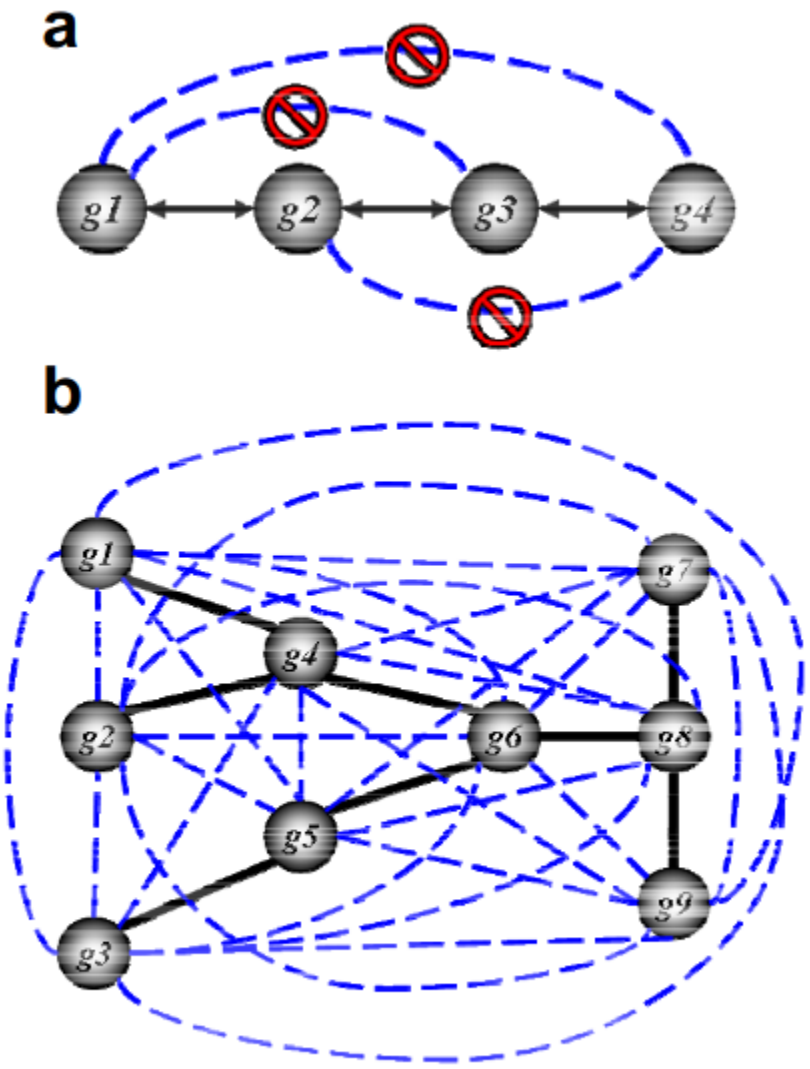
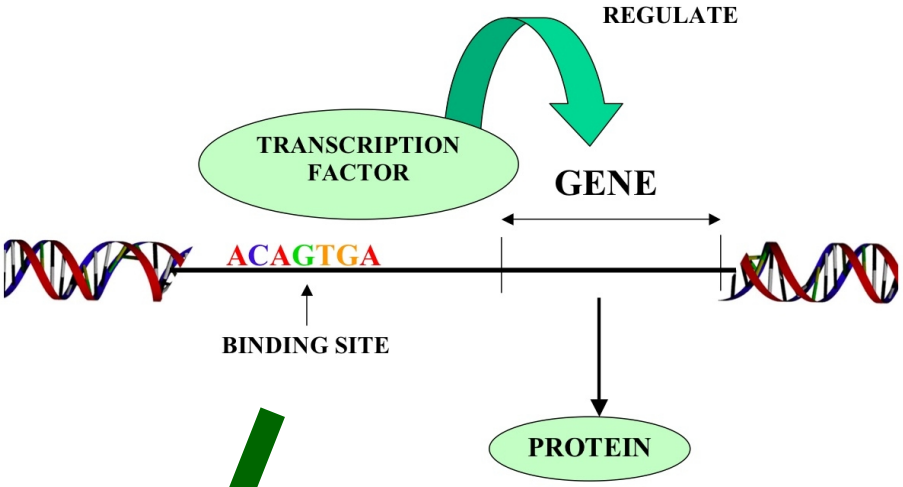
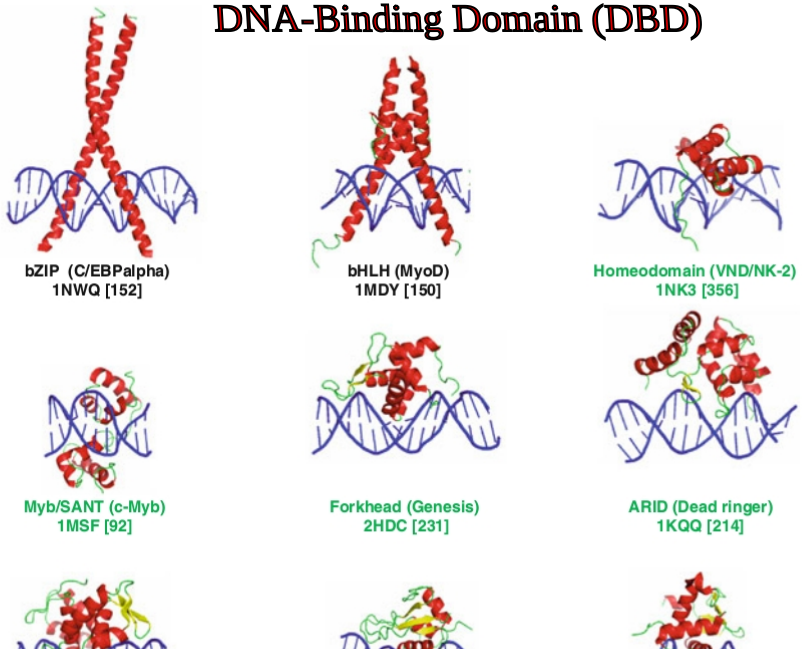


Figure 2
Examples of the data processing inequality. (a) g_1, g_2,



DNA-Binding Domain (DBD)



Weirauch, M. T., & Hughes, T. R. (2011)

Experiment
ChIp-chip
ChIP-seq
PBM

Collection of binding sites

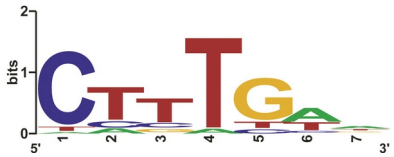
CTTTGTC
CTTTGTC
CCTTGAC
ATCTGAA
CTTTTAA
CTCTGAA

A

	1	2	3	4	5	6	7
A	1	4	1	2	0	17	13
C	28	5	5	0	3	3	2
G	0	0	4	0	25	1	7
T	2	22	21	29	4	10	9

Position Weight Matrix (PWM)

B



TF binding motif

Information content of a PSSM

Prior	Pos	1	2	3	4	5	6	7	8	9	10	11	12
0.325	A	-0.12	0.05	-0.06	-0.08	**0.97**	-0.08	-0.08	-0.08	-0.08	-0.08	-0.12	-0.06
0.175	C	0.08	0.08	**0.25**	**1.50**	-0.04	**1.50**	-0.04	-0.04	-0.04	0.08	-0.04	0.08
0.175	G	-0.04	0.08	**0.25**	-0.04	-0.04	-0.04	**1.50**	-0.04	**0.68**	**0.45**	**0.68**	0.08
0.325	T	0.19	-0.12	-0.08	-0.08	-0.08	-0.08	-0.08	**0.97**	0.05	-0.06	-0.06	-0.06
1.000	Sum	0.11	0.09	**0.36**	**1.29**	**0.80**	**1.29**	**1.29**	**0.80**	**0.61**	**0.39**	**0.47**	0.04

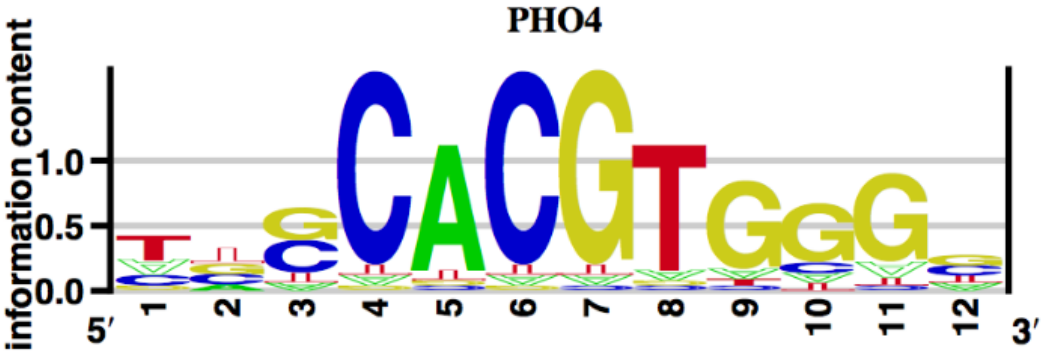
$$f'_{i,j} = \frac{n_{i,j} + p_i k}{\sum_{i=1}^{A} n_{i,j} + k}$$

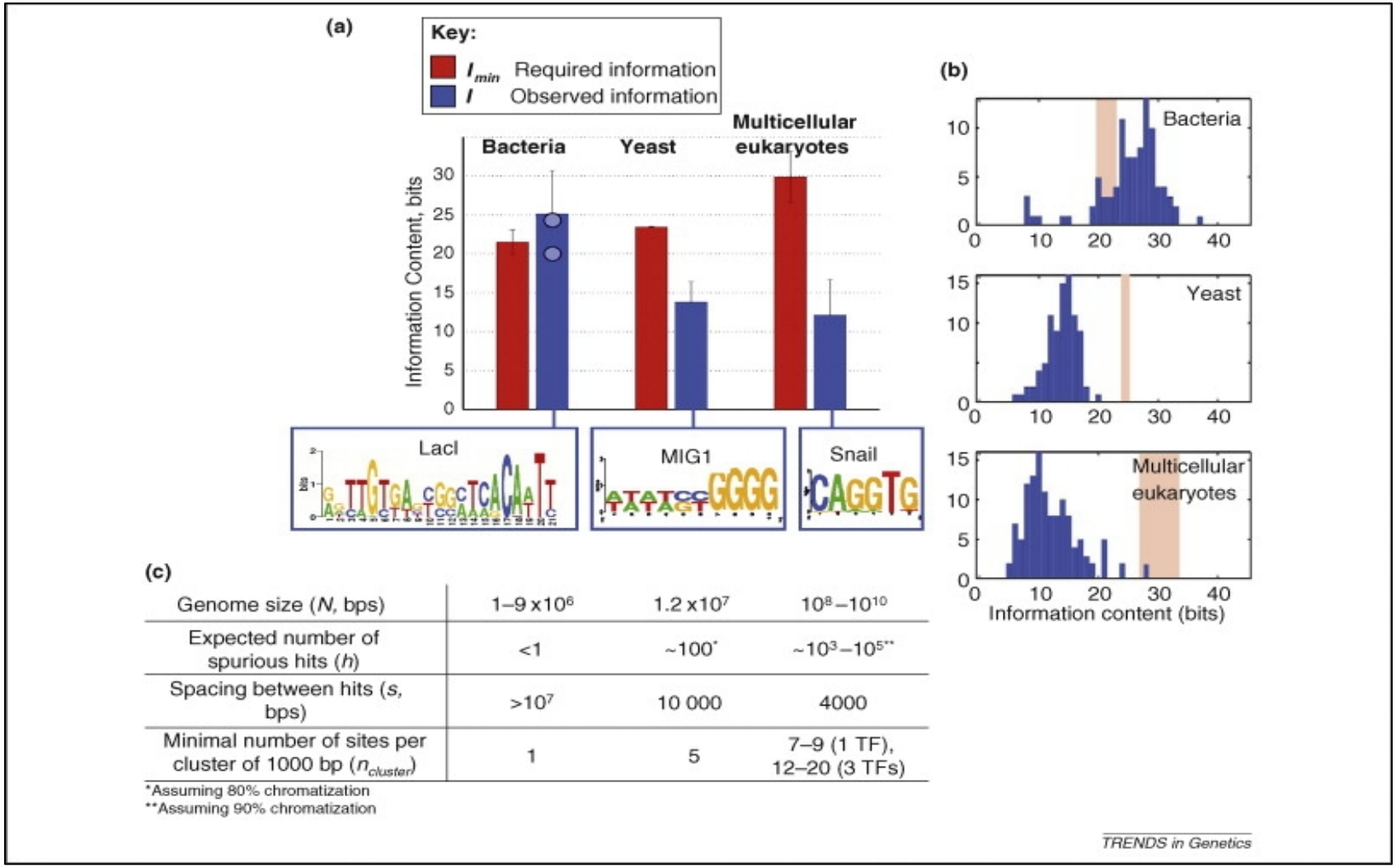
$$I_{i,j} = f'_{i,j} \ln\left(\frac{f'_{i,j}}{p_i}\right)$$

$$I_j = \sum_{i=1}^{A} I_{i,j}$$

$$I_{matrix} = \sum_{j=1}^{w} \sum_{i=1}^{A} I_{i,j}$$

- A *alphabet size (=4)*
- $n_{i,j,}$ *occurrences of residue i at position j*
- w *matrix width (=12)*
- p_i *prior residue probability for residue i*
- $f_{i,j}$ *relative frequency of residue i at position j*
- k *pseudo weight (arbitrary, 1 in this case)*
- $f'_{i,j}$ *corrected frequency of residue i at position j*
- $W_{i,j}$ *weight of residue i at position j*
- $I_{i,j}$ *information of residue i at position j*





(Wunderlich and Mirny , 2009)