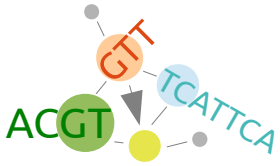
Università degli Studi di Verona

Informational Analysis of Biological Data



Ph.D. Vincenzo Bonnici
University of Verona, Italy

vincenzo.bonnici@univr.it
www.di.univr.it/~bonnici

Informational Analysis of Genomic Sequences

- **Computational biology** and **bioinformatics** are multidisciplinary fields which involve plenty of sciences, such as biology, biotechnology, mathematics, computer science, physics and chemistry.
- In the last decades, bioinformatics approaches, regarding sequence analysis, mainly focused on methodologies based on string **alignment** algorithms.
  - e.g. BLAST Altschul,S.F. et al. (**1990**) **Basic local alignment search tool.** J. Mol. Biol., 215, 403–410
- An alternative perspective is based on **alignment-free** methodologies for genome analysis, where global and local properties of genomes are investigated.
  - Susana Vinga and Jonas Almeida. **Alignment-free sequence comparison - a review**. Bioinformatics, 19(4):513–523, 2003.
- In this context, we focus the attention on those methodology regarding **informational analysis**, which also include information theory, probability, statistics, formal languages and linguistic theory.
- **Information theory** was developed by Shannon to study message transmission over communication systems, and it has later been applied to many other fields of research.
  - Claude Elwood Shannon. **A Mathematical Theory of Communication**. The Bell System Technical Journal, 27:379–423,623–656, 1948**.**
- Links between **informational analysis and biology** are well established, continuously reemerging, and deeply rooted.
  - Shannon's Ph.D. thesis, titled 'An Algebra for Theoretical Genetics' (**1940**), precedes his famous booklet where he notion of information entropy was introduced.
  - LL Gatlin. **The information content of DNA. Journal of theoretical biology**, 10(2):281–300, **1966**.
  - Lila L Gatlin et al. **Information theory and the living system**. **1972**.
  - Edward N Trifonov and Joel L Sussman. **The pitch of chromatin DNA is reflected in its nucleotide sequence.** Proceedings of the National Academy of Sciences, 77(7):3816–3820, **1980**.
  - **Peak 3-periodicity**
    - Manfred Eigen and Ruthild Winkler-Oswatitsch. **Transfer-RNA, an early gene?** Naturwissenschaften, 68(6):282–292, **1981**.
    - John CW Shepherd. **Periodic correlations in DNA sequences and evidence suggesting their evolutionary origin in a comma-less genetic code.** Journal of Molecular Evolution, 17(2):94–102, **1981**.
    - James W Fickett. **Recognition of protein coding regions in DNA sequences.** Nucleic acids research, 10(17):5303–5318, **1982**.

Informational Analysis of Genomic Sequences

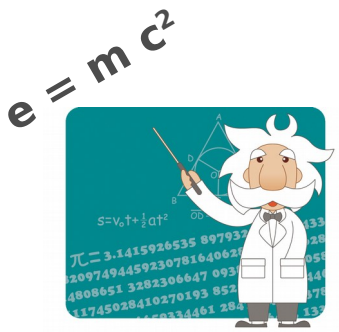
InfoGenomics

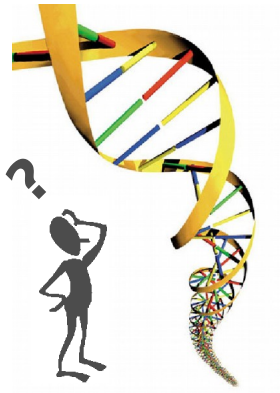
The **InfoGenomics** project aims at proving a systematic approach from an informational point of view by making use of informational analysis and well-characterized genomic features: **indices, distributions, representations (and visualizations).**

The **ENCODE** project created an encyclopedia of DNA elements by annotating the human genome in terms of bio-chemical function. It provided evidence that 80% of the human genome, first considered junk regions, is covered by functional elements.

This scenario has an informational basis linked to DNA fragments related to such functional elements.
An integration between biochemical and informational analysis could provide new possibilities for interpreting data and to discover principles of genome organization and functions.

Maths in action





- Manca, V.: **Infobiotics: information in biotic systems**. Springer (2013)
- Bonnici, V., Manca, V.: **Recurrence Distance Distributions in Computational Genomics.** AJBCB (2015)
- Bonnici, V., Manca, V.: **Infogenomics Tools: A Computational Suite for Informational Analyses of Genomes.** JBPR (2015).
- Castellini, A., Franco, G., Manca, V.: **A dictionary based informational genome analysis.** BMC Genomics, 13, 485 (2012)
- The Encode Project Consortium: **An integrated encyclopedia of DNA elements in the human genome.** Nature 489, 57–72 (2012)

InfoGenomics

Notation and Definitions

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

- DNA alphabet **Γ** = {A,C,G,T}
- Extended alphabet **Γ̄** = {A,C,G,T,N}
- **Γ^k** the set of words of length k over Γ
- **k-mer** a word in **Γ^k**

- Given a string ***G*** = $a_1 a_2 \ldots a_n$
- **G[i,j]**, for 1 ≤ i ≤ j ≤n, the **substring** of G from position i to j
- **G[1,j]**, for 1 ≤ j ≤n, a **prefix** of G
- **G[i,n]**, for 1 ≤ i ≤n, a **suffix** of G

- Given a string **α**, **pos(α,G)** the set of **positions** where α occurs in G
- **mult(α,G)** = |pos(α,G)|, the **multiplicity** of α in G
- **Hapax** a word α with mult(α,G) = 1
- **Repeat** a word α with mult(α,G) > 1

- A dictionary **D** is a set of words
- Given k > 0, **$D_k(G)$** the words in Γ^k that occur in G
- **D_k** is **complete** if it contains all the word in **Γ^k**
- **$T_k(G)$** the multiset of $D_k(G)$ (i.e. words and their multiplicities)

Genomic sequences visualization

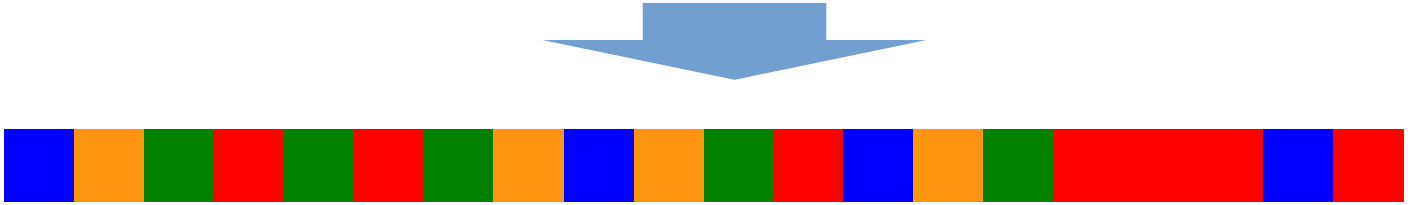
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Color Schema

Sequence

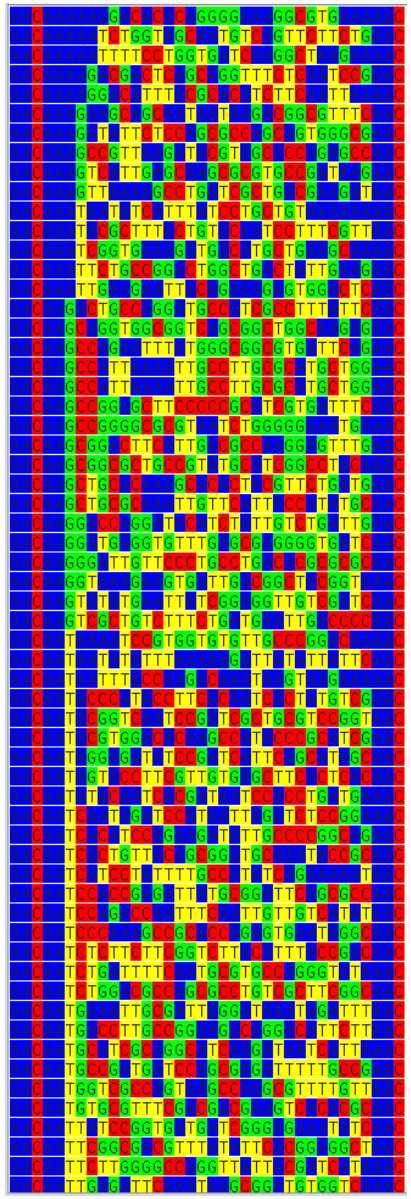
A T G C G C G T A T G C A T G C C C A C

2D Image



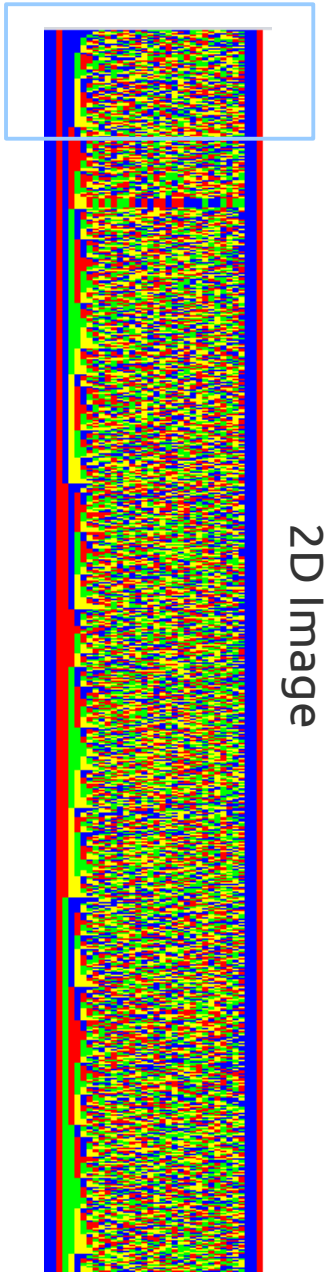
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic sequences visualization



Colored Text

Graphical Compact Representation



2D Image

- From left to right

- One sequence per row
- Lexicographically ordered
- From top to bottom

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions Definition

- The **discrete distribution**, over a domain *A*, is a function which assigns to each element of *A* an element in $\mathcal{R}$, having the property of finite summability over *A*.

$$\sum_{x \in A} f(x) \in \mathbb{R}.$$

- It is a **continuous distribution** over *A* if it is finitely integrable over *A*.

$$\int_{x \in A} f(x)dx \in \mathbb{R}.$$

- When the sum or the integral over A is equal to **1**, then the distribution (discrete or continuous) is a **probability distribution.**

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Definition - Example

There are 10 pencils of different colors

Pencil	Color
1	blue
2	red
3	green
4	blue
5	blue
6	red
7	blue
8	red
9	green
10	blue

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Definition - Example

There are 10 pencils of different colors

Pencil	Color
1	blue
2	red
3	green
4	blue
5	blue
6	red
7	blue
8	red
9	green
10	blue

Domain *A* = set of colors = {blue, red, green}

Distributions
(Assign to each color a related information)

Color	Pencils having that color	Number of pencils having that color (Multiplicity)	Frequency (Probability)
blue	{1, 4, 5, 7, 10}	5	5 / 10
red	{2, 6, 8}	3	3 / 10
green	{3, 9}	2	2 / 10

Genomic Distributions

Taxonomy

Given a genomic sequence **G**, we can define the following genomic distributions:

- **Word position**
  This distribution assigns to any word **α** of **D** the set of positions of **G** where it (its first character) occurs, that constitute what is also called the **spectrum** of **α** in **G**. When the spectrum is given for every word of a dictionary that is complete for **G**, then the whole genome **G** can be easily reconstructed.

- **n-Word Count**
  This distribution assigns to any value of **$n < |G|$** the **cardinality** of **$D_n(G)$**.

- **n-Repeat Count**
  This distribution assigns to any value of **n**, from 0 to some maximum value, the **cardinality** of the set of the **repeats** of **G** having **length n**.

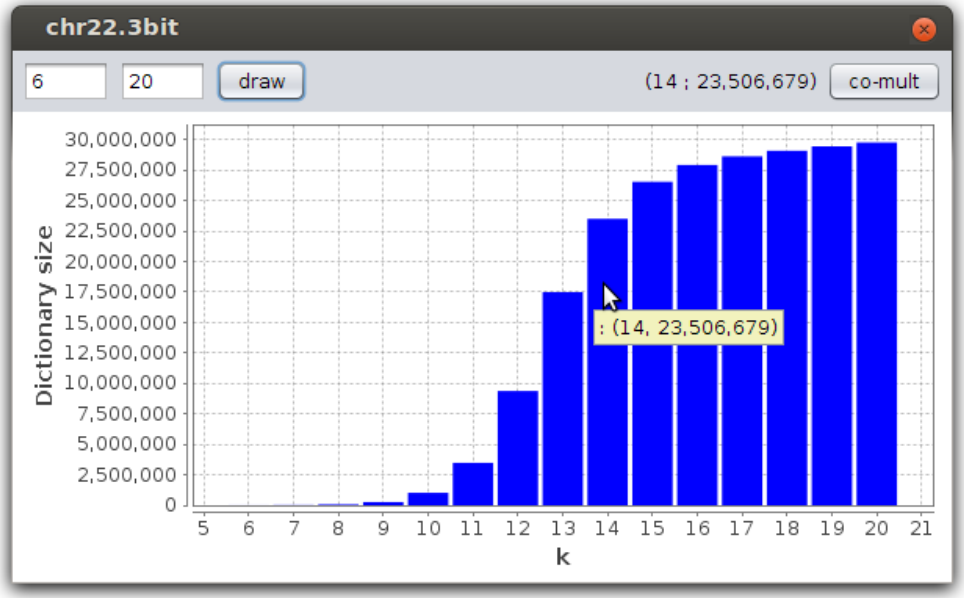
- **Rank Frequency**
  If we **order** the words, of a complete dictionary for **G**, according to their **frequencies** in **G** (in decreasing order) we say that the most frequent words have **rank** 1, the most frequent words, after words of rank 1, have rank 2, and so on. Therefore, this distribution assigns to each rank the vale corresponding its frequency. This distribution, also called **Zipf distribution**, after the scholar who introduced it, was extensively studied in natural languages.

Genomic Distributions

Examples

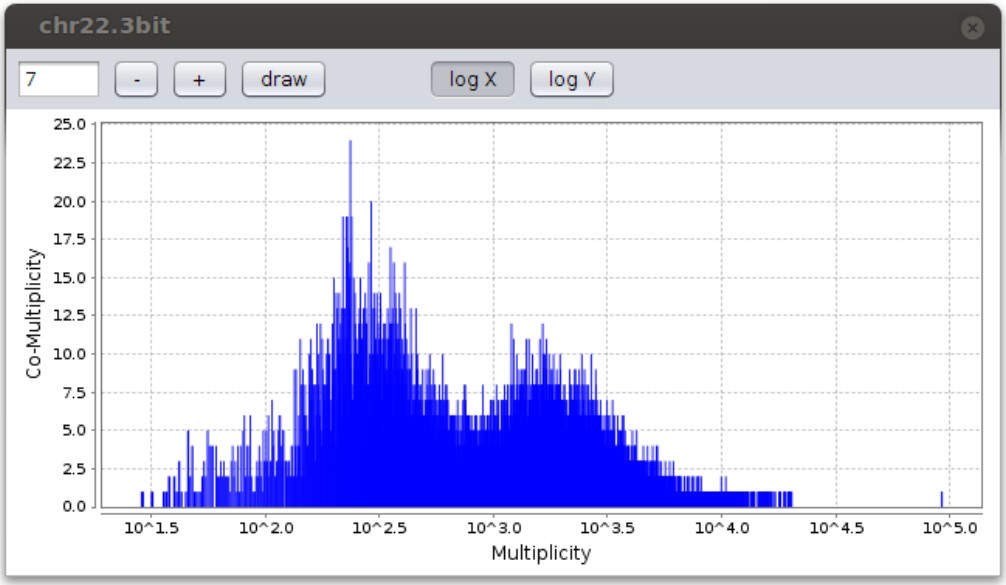
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

n-Word Count Distribution



Chromosome 22 of
Homo sapiens

Word Co-multiplicity Distribution

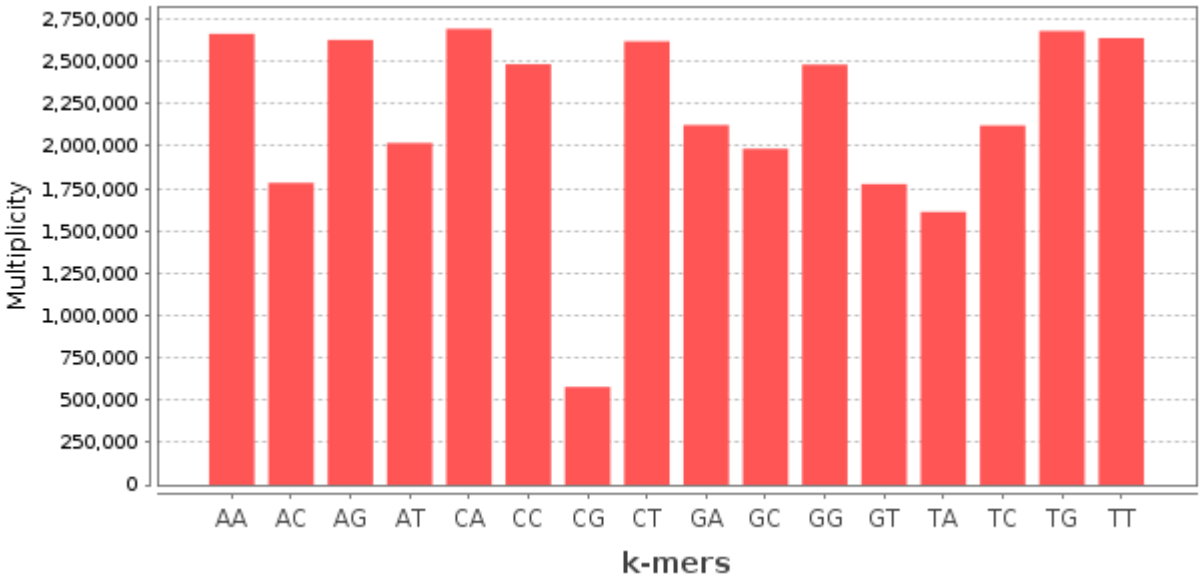


GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Examples (cont.)

Word Multiplicity Distribution



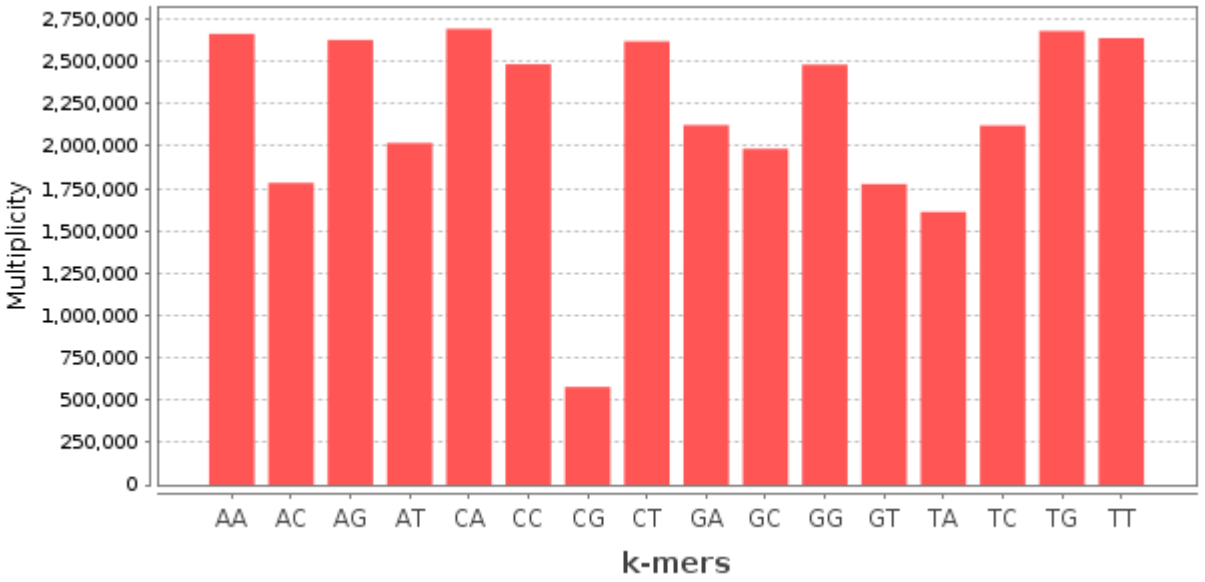
Chromosome 22
of Homo sapiens

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Word Multiplicity Distribution and HeatMaps

Word Multiplicity Distribution



heatmap

legend

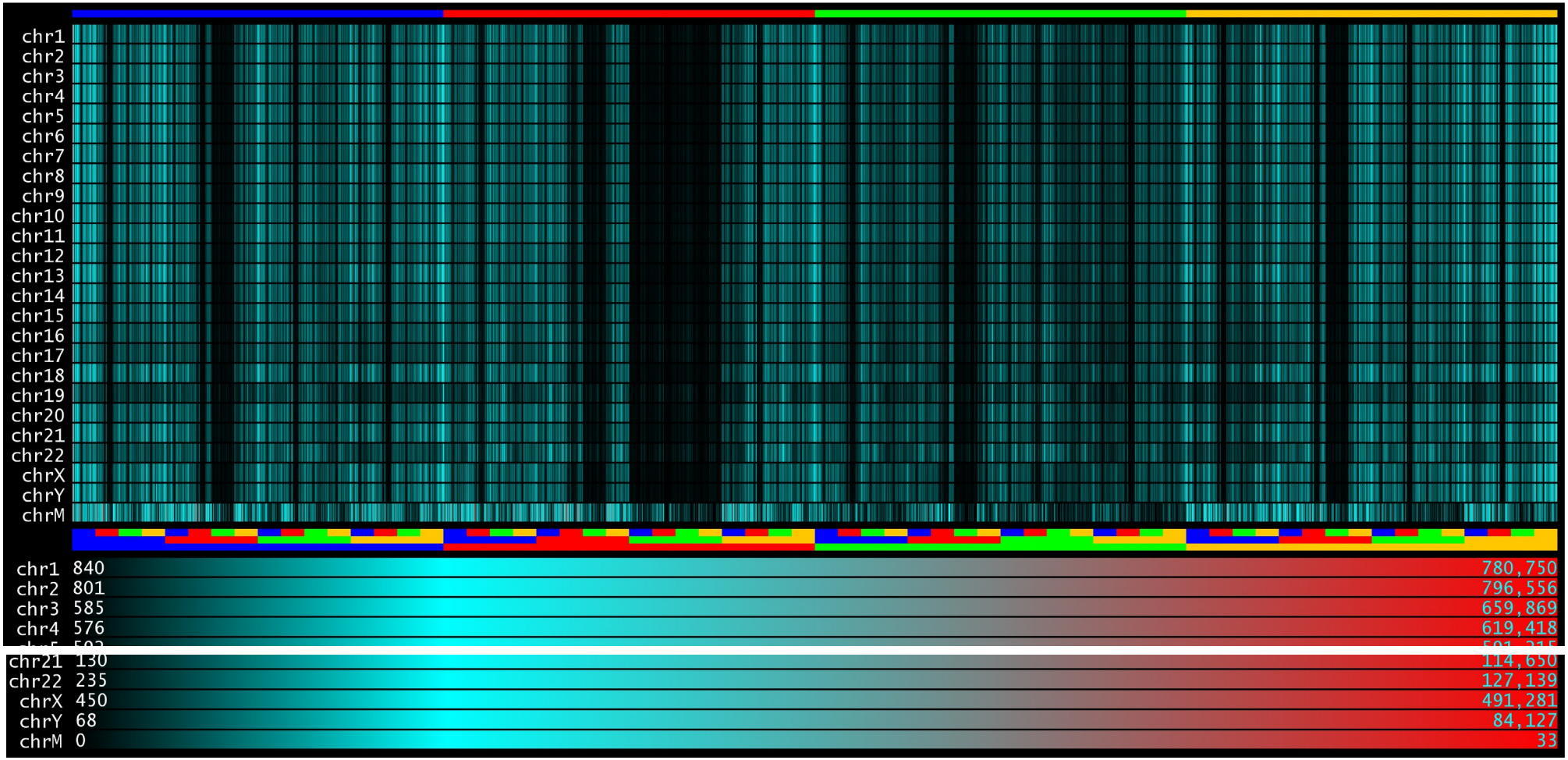
Chromosome 22
of Homo sapiens

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Word Multiplicity Distribution and HeatMaps

The **language similarity** among human chromosomes by their **6-mer** multiplicity distributions.



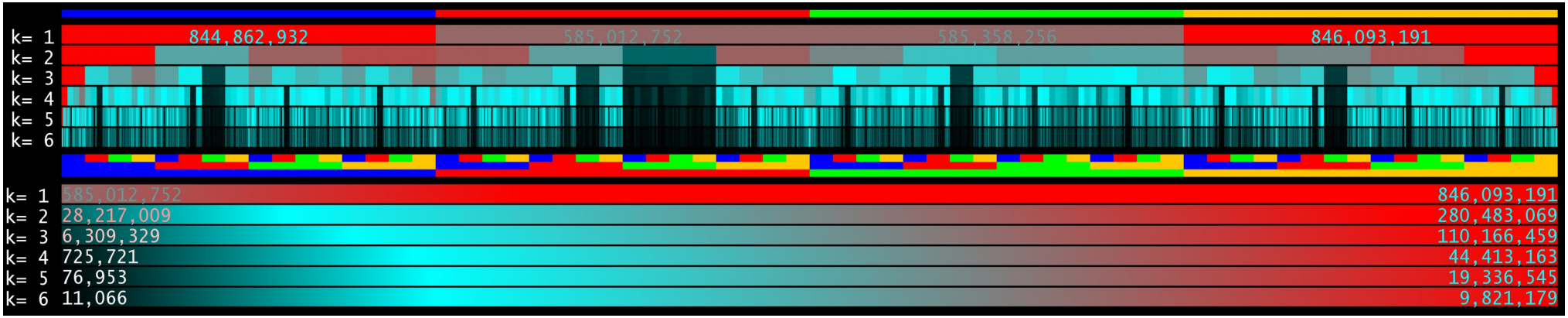
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

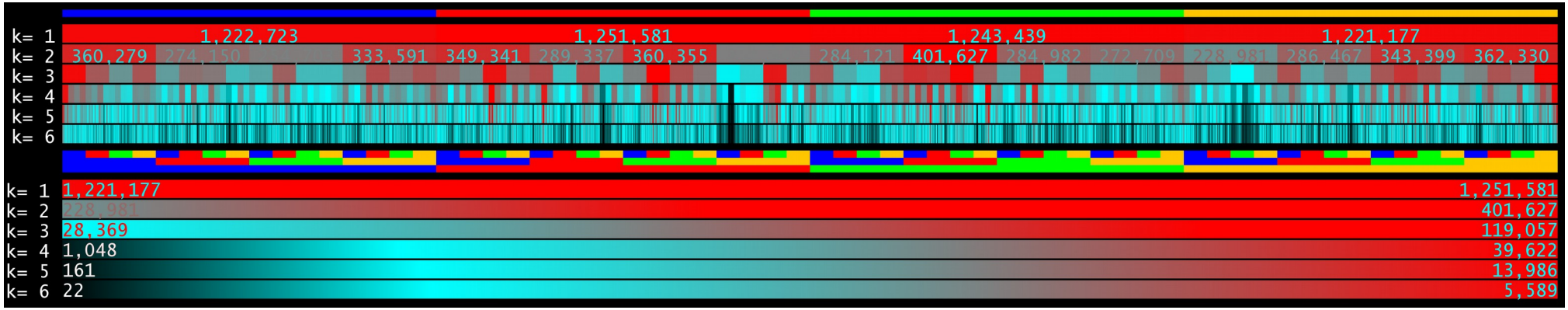
Word Multiplicity Distribution and HeatMaps

Visualization of species diversity.

Homo sapiens



Escherichia coli

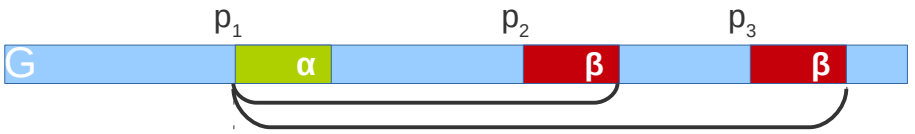


GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

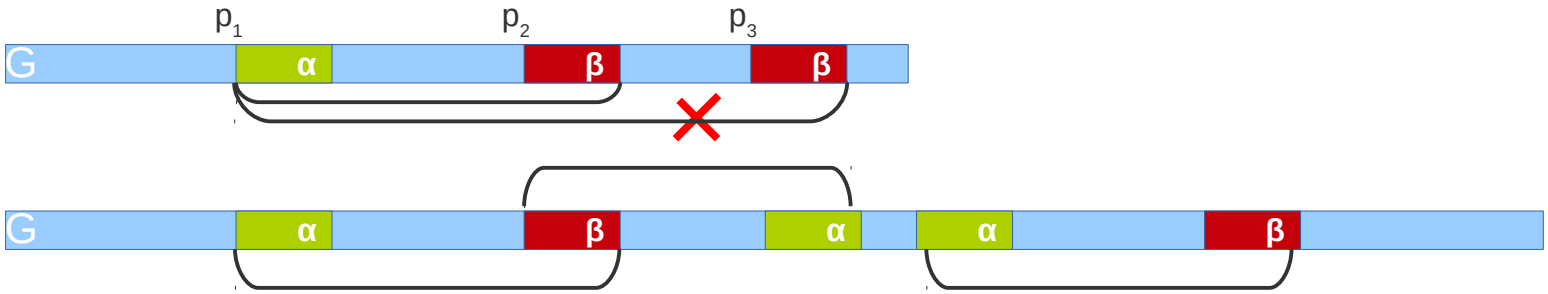
Genomic Distributions

Recurrence and Recurrence Distance

- Let $\alpha \in D_{k'}(G)$, $\beta \in D_{k''}(G)$
- **Global co-occurrence** (p_1,p_2)
  - $p_1 \in pos(\alpha,G)$ AND $p_2 \in pos(\beta,G)$ or vice versa
  - $p_1 < p_2$



- **Minimal co-occurrence** (p_1,p_2)
  - $\nexists\, p' : p_1 < p' < p_2$ AND $p' \in \{ pos(\alpha,G) \cup pos(\beta,G)\}$



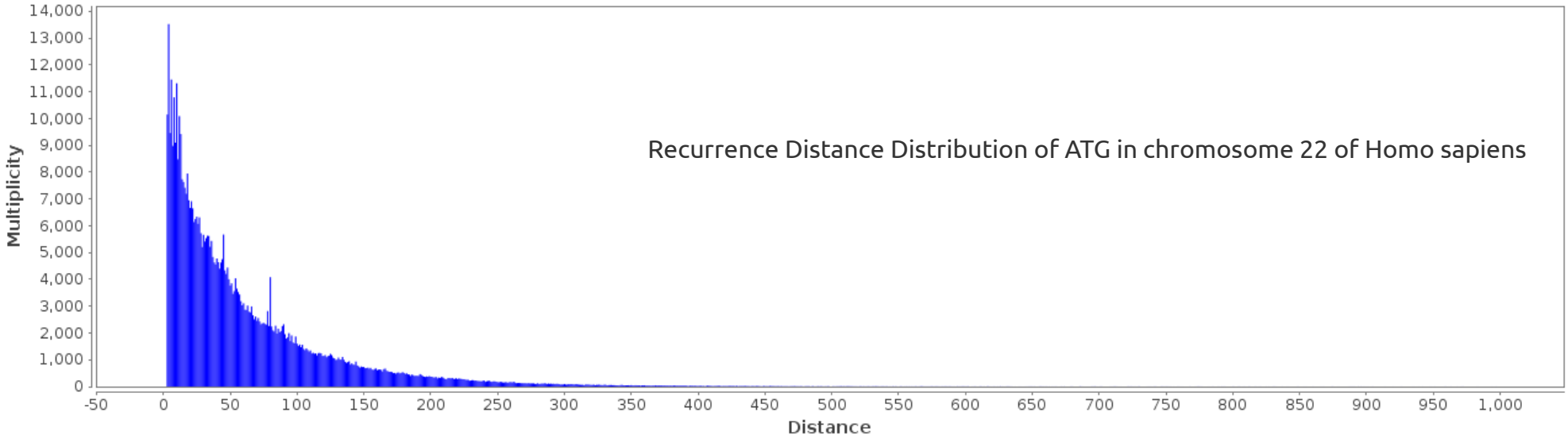
- It is a **recurrence** if $\alpha = \beta$
- The **recurrence distance** is given by $p_2 - p_1$

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Recurrence Distance Distribution

- Given a word **α**, this distribution assigns to any **distance n**, going from 1 to some maximum value, the number of times it occurs at distance **n** from its previous occurrence.



- Given a set of words **D**, the **average recurrence distance** assigns to any distance **n** the value

$$\frac{\sum_{\alpha \in D} |R(G, \alpha, n)|}{|D|}$$

where **|R(G, α, n)|** is the number of times **α** occurs in **G** at distance **n** (**|D|** is the cardinality of **D**).

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Word Probability and Waiting Time

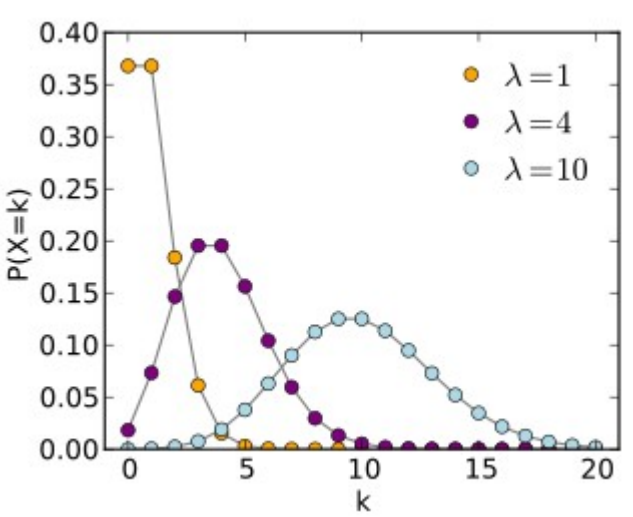
- The **probability**, with which words appear in a sequence, follow a specific distribution.
- In our case, it is described by the **Word Frequency Distribution,** namely the Word Multiplicity Distribution normalized by the total count of word occurrences and ordered by the multiplicity value
- For us: number of occurrences = multiplicity = frequency = probability

- But in general it can follow such us
  - **Uniform distribution**: all the words have the same multiplicity (frequency, probability of appearance)
  - **Gaussian distribution:** word frequencies can be described by a Gaussian distribution

  - **Poisson distribution**: word frequencies can be described by a Poisson distribution, where the probability that a word has a multiplicity equal to **k**, **Pr(X = k)**, is given by

$$\Pr(X = k) = \frac{\lambda^k e^{-\lambda}}{k!},$$

where **e** is the **Euler's number**
and **λ** is the **variance** (the mean value of the distribution)



GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

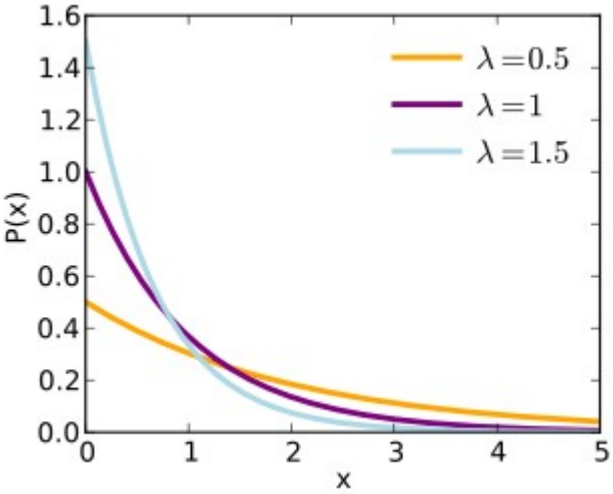
Genomic Distributions

Word Probability and Waiting Time

- The **Word Frequency Distribution** describes how many times a word occurs.
- It does not describe where such occurrences are, neither the distances between them.
- The **Waiting Time** of a **Frequency Distribution** describes the amount of **time** we need to attend between two occurrences of an event.
- In our case, events are words and the amount of time is given by the **number of symbols** (characters)
- The **Waiting Time** of a **Poisson Distribution** is the **Exponential Distribution**

$$f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0, \\ 0 & x < 0. \end{cases}$$

- In our case
  - **x** is the **distance** between two occurrences
  - **P(x)** is the **number of times** two occurrences appear at distance **x**
  - It is exactly the **Recurrence Distance Distribution !!!**

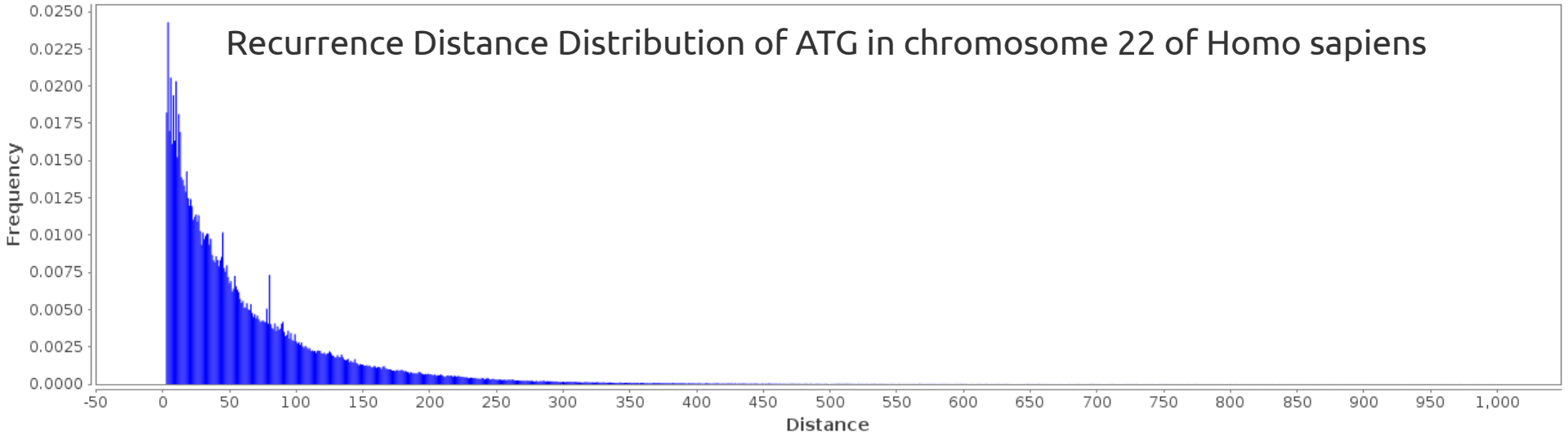


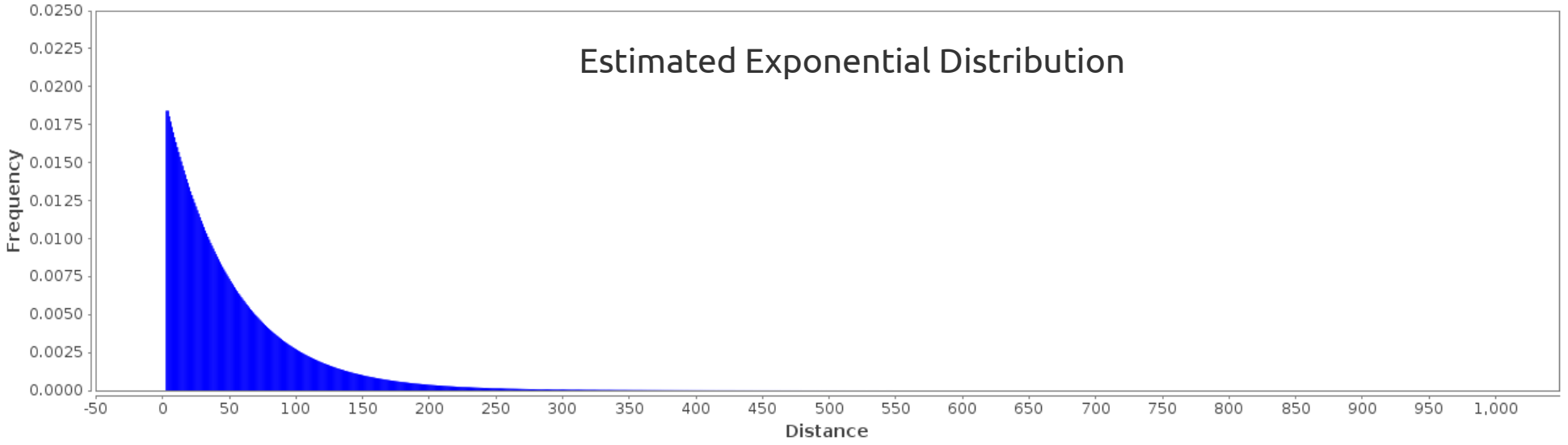
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Recurrence Distance Distribution (RDD)

- It is known that some **real genomic sequences** follow a quasi **Poisson Distribution**
- Then, their **Recurrence Distance Distribution** must follow a quasi **Exponential Distribution**



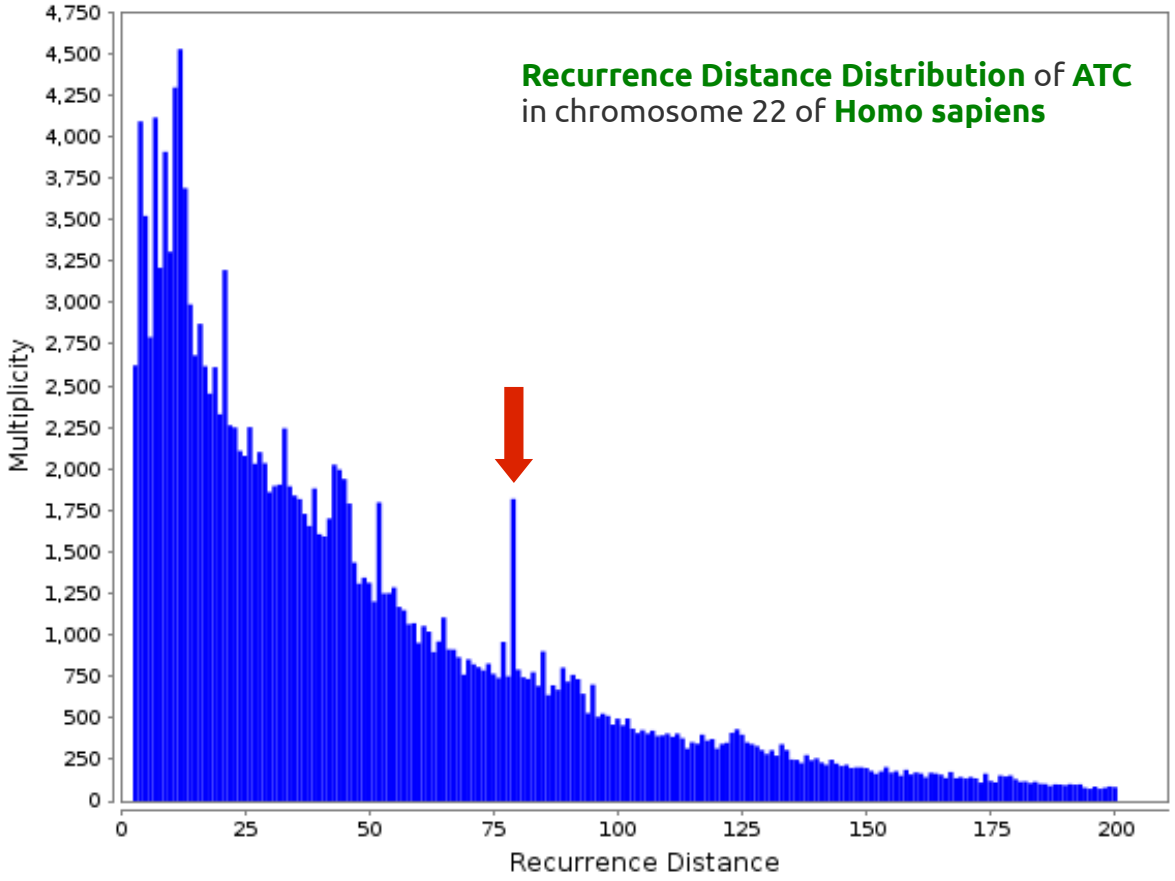


GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

Peaks in Recurrence Distance Distribution

- Peaks in Recurrence Distance Distributions identify **repetitive elements**
- Their quantification can be obtained by subtracting the theoretical distribution.



Sequences **enclosed** between the occurrences of ATC ad distance **81**

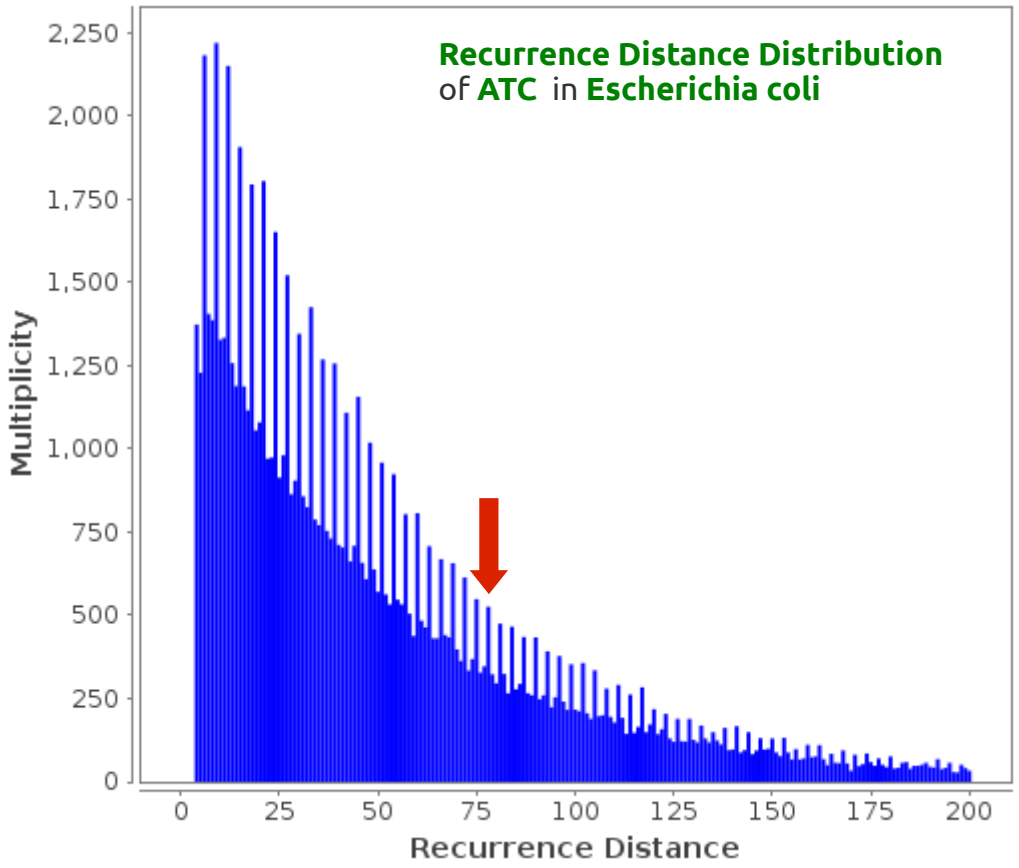
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Genomic Distributions

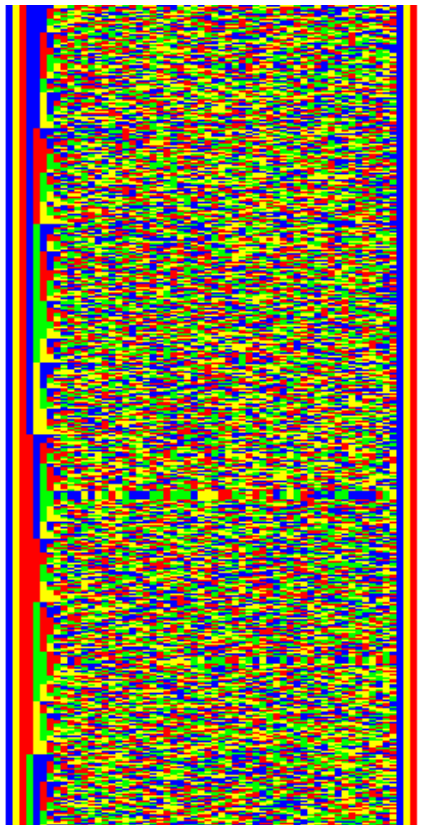
Peaks in Recurrence Distance Distribution

The Escherichia coli case

- **Peaks** are at distances multiples of **3** and they do not represent repetitive elements.



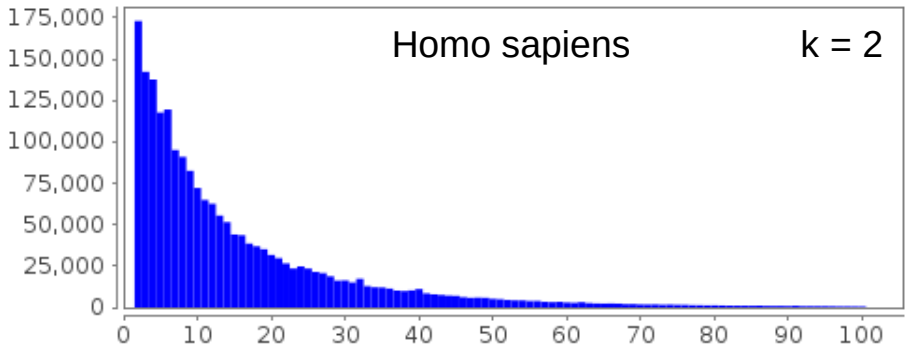
Sequences **enclosed** between the occurrences of ATC ad distance **81**

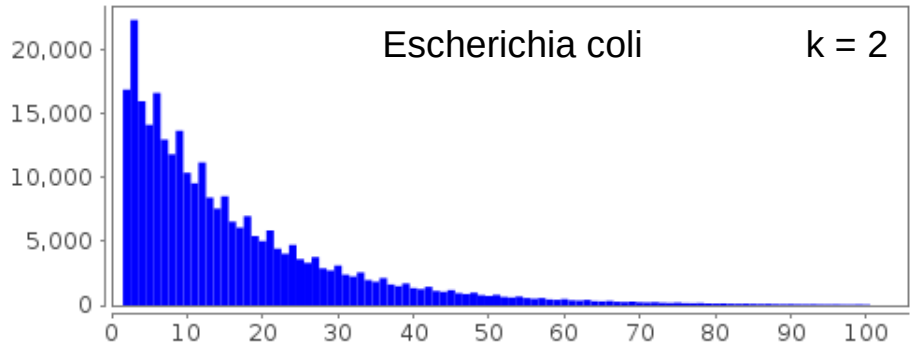


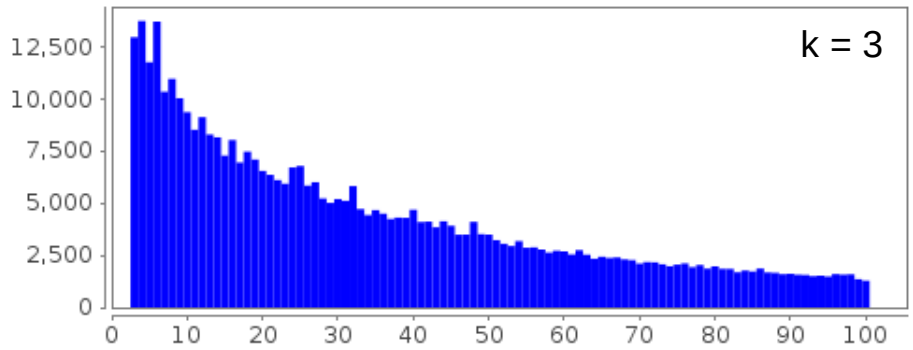
Genomic Distributions

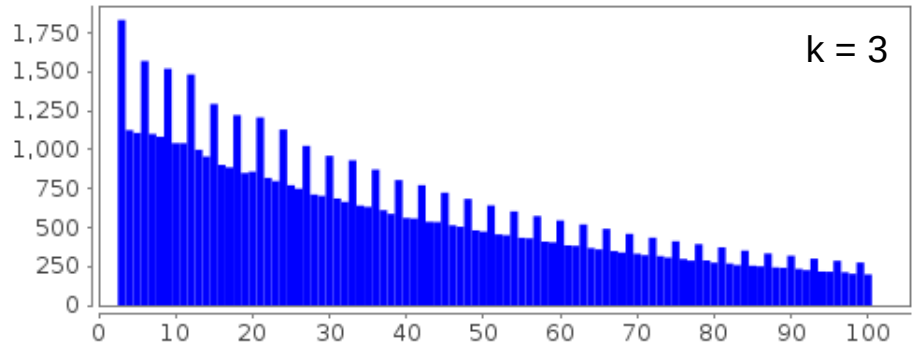
RDD: Averages and peak 3-periodicity

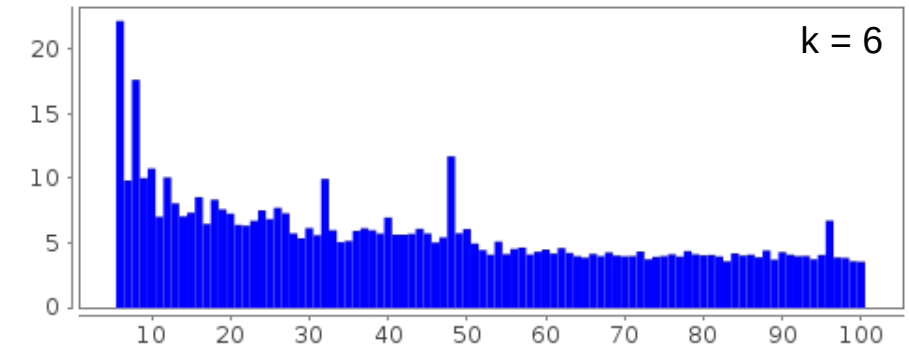
Average RDD in whole sequences

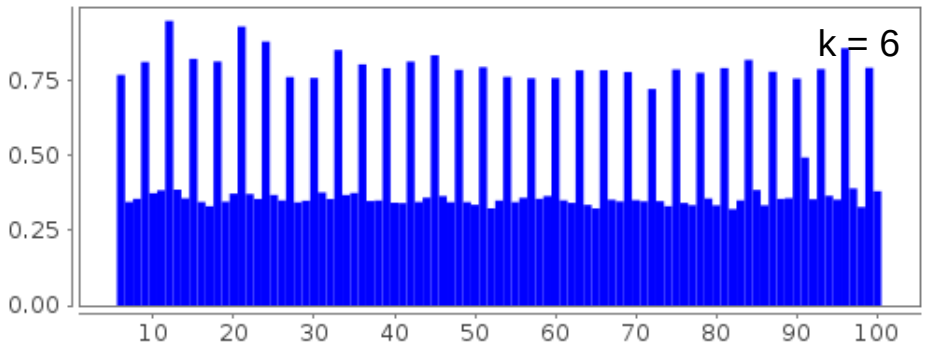








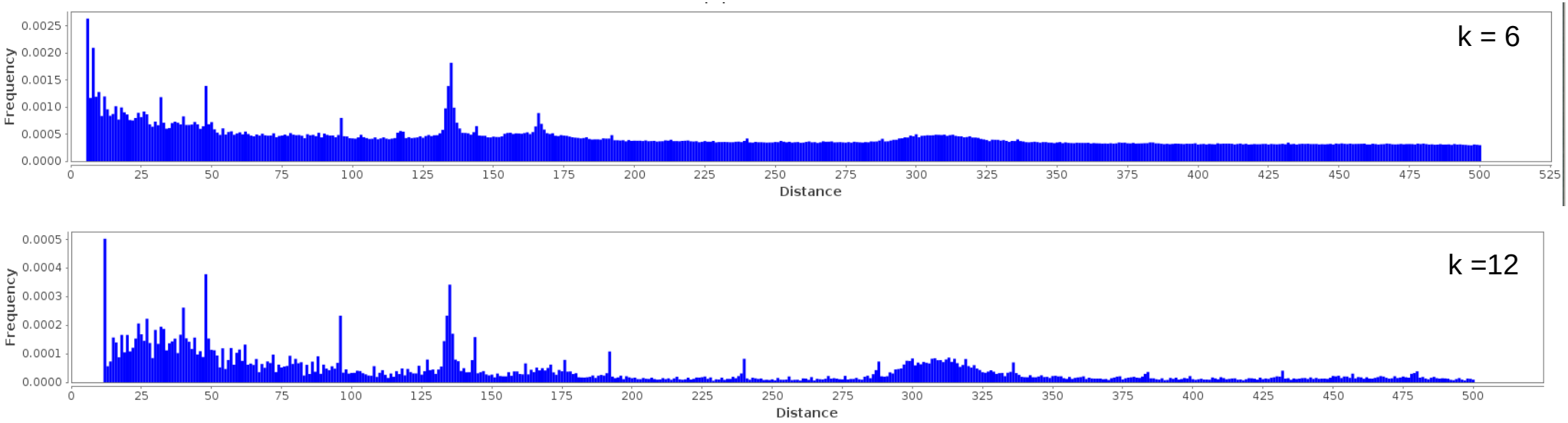




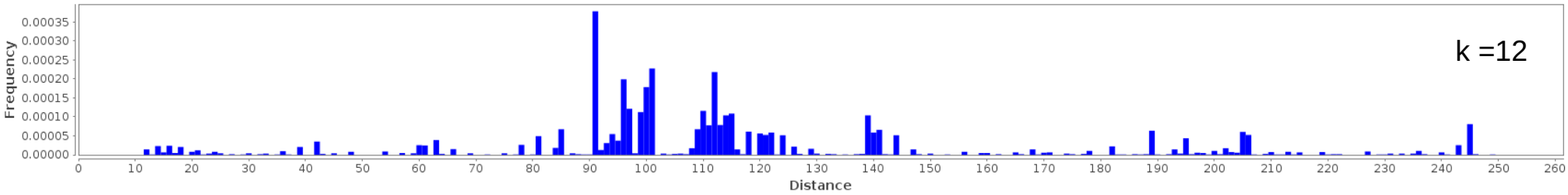
Genomic Distributions

ARDD and recurrence distance preference

Homo sapiens, chromosome 22

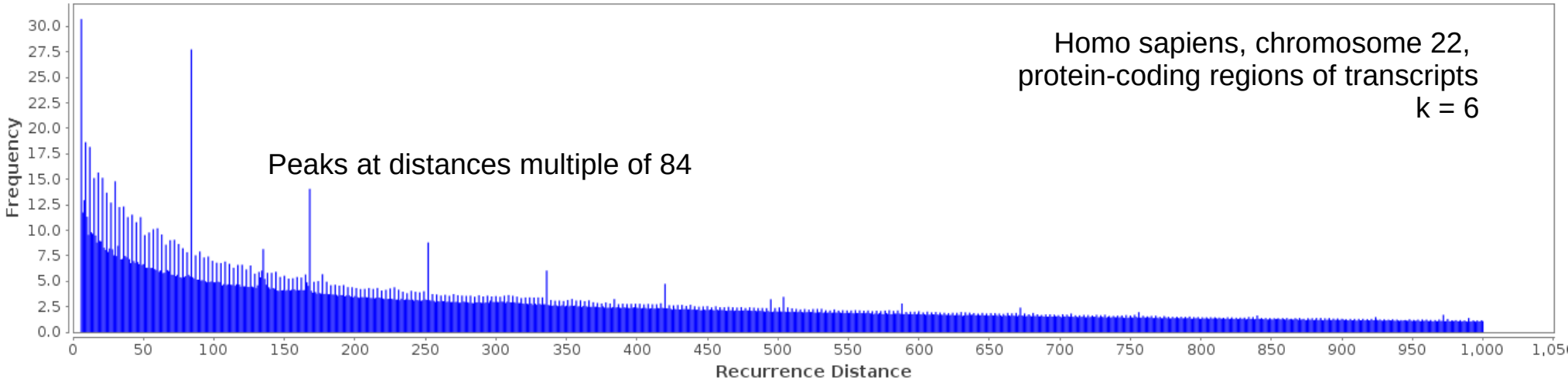


Escherichia coli

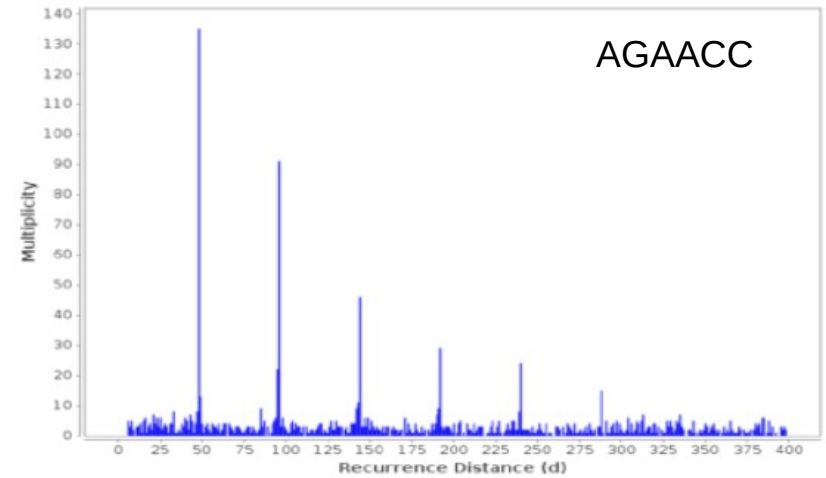


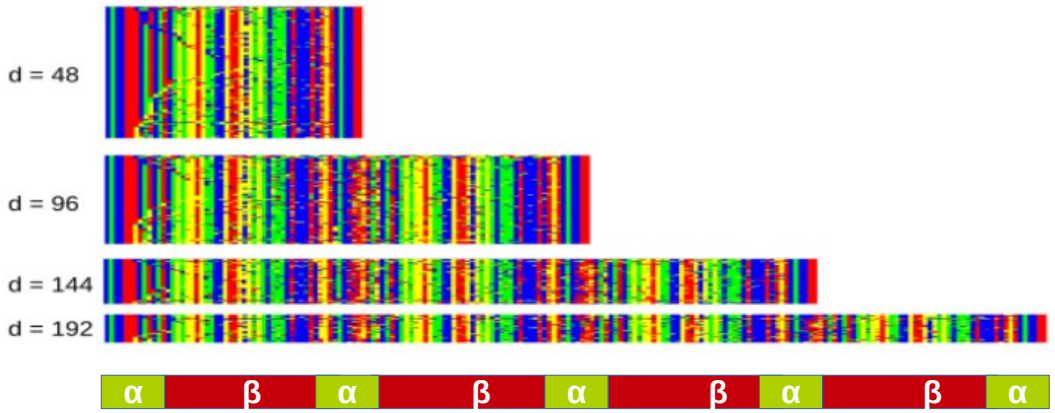
Genomic Distributions

Extra peak periodicity in RDD and ARDD



We extracted the enclosed strings of recurrences at distance 84 (of AAAAAC) and searched them in public databases of protein domains. We found that such strings correspond to the C2H2 zinc finger domain, which often forms tandem sequences.





GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Informational Analysis of Genomic Sequences

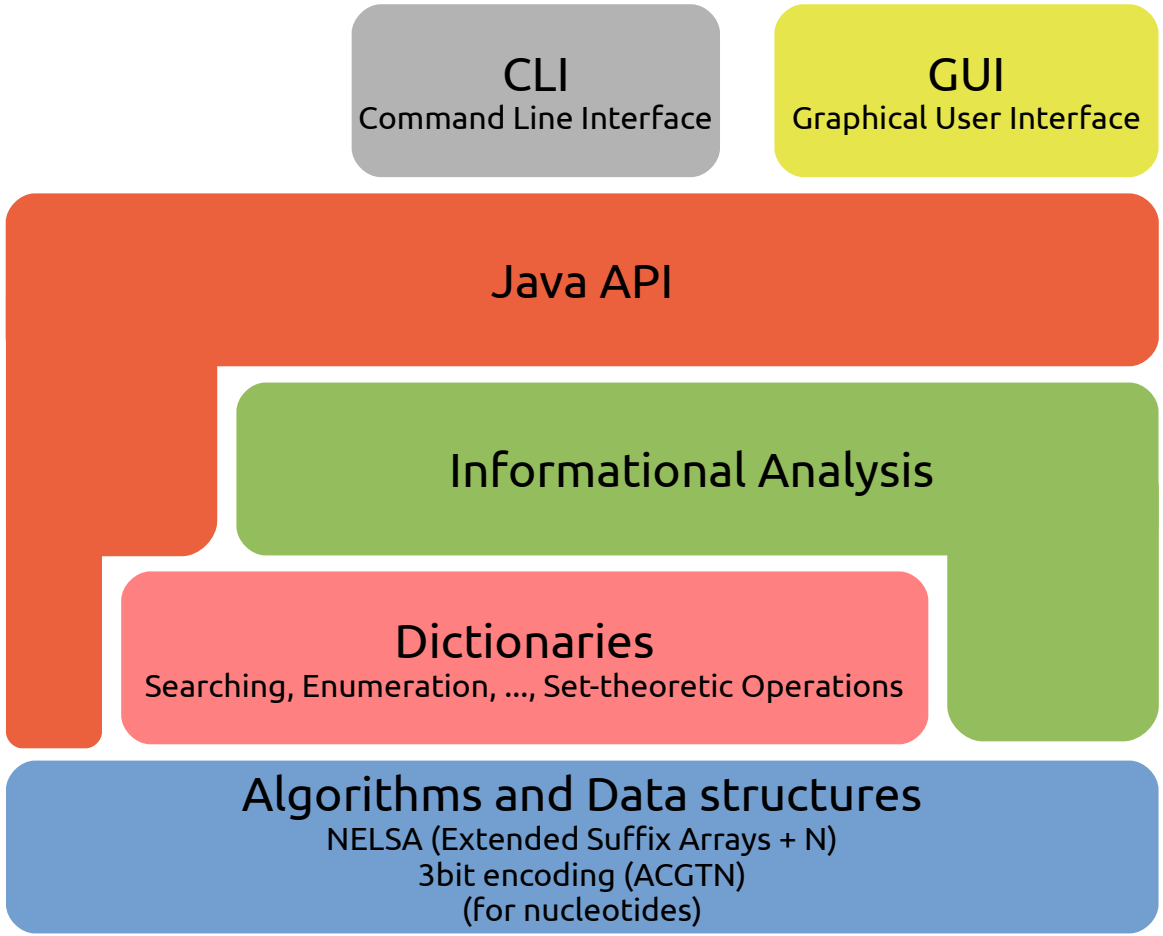
IGTools (InfoGenomics Tools)

An open-source suite for the informational analysis of genomic sequences.

Goals

- **Efficiency**: made on top of well-established data structures and algorithms, adapted for real genomic sequences.
- **Interactive** graphical interfaces and CLI (for batch analyses)
- Also **for developers**: modular Java API ready to be **used and extended**

It aims at providing the **first** framework for informational analysis of genomic sequences.



Bonnici, V., Manca, V.: **InfoGenomics Tools: a computational suite for informational analysis of genomes**. Submitted to Journal of Bioinformatics and Proteomics (2015)

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

Informational Analysis of Genomic Sequences

IGTools (InfoGenomics Tools)

An open-source suite for the informational analysis of genomic sequences.

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools

Sequences

FASTA and Binary Encoding

FASTA format (textual)

- A T G C G

2-bit encoding

- A → 00
- C → 01
- G → 10
- T → 11

```
00 11 01 10 01
A  T  G  C  G
```

3-bit encoding

- A → 000
- C → 001
- G → 010
- T → 011
- N → 100

```
000 011 001 010 001 100
A   T   G   C   G   N
```

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools

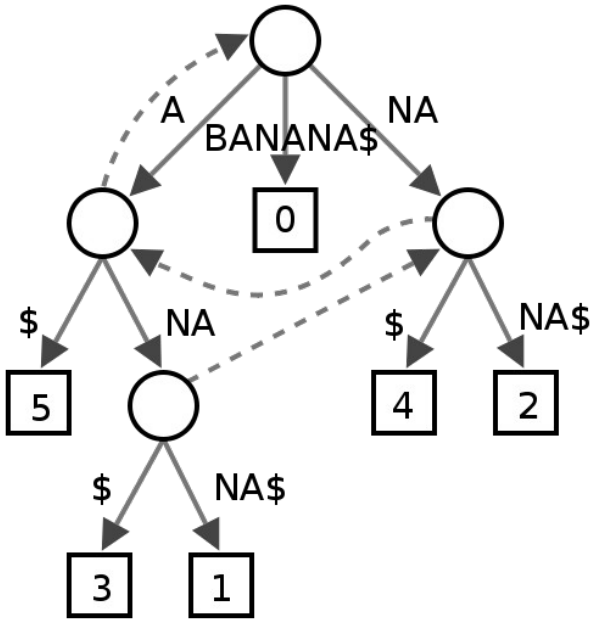
Text self-indexing

for genomic sequences

Indexes (data structures) to efficiently solve operations (i.e. pattern search).

Suffix Trees

is a compressed trie containing all the suffixes of the given text as their keys and positions in the text as their values. Suffix trees allow particularly fast implementations of many important string operations.



Suffix Arrays

is a sorted array of all suffixes of a string. Suffix arrays were introduced by Manber & Myers (1990) as a simple, space efficient alternative to suffix trees.

Suffix	i
$	7
a$	6
ana$	4
anana$	2
banana$	1
na$	5
nana$	3

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools
Text self-indexing
for genomic sequences

Suffix Arrays + LCP
Longest Common Prefix.
It allows for efficient enumeration of k-mers.

i	suffix	SA[i]	LCP[i]
1	$	7	0
2	a$	6	0
3	ana$	4	1
4	anana$	2	3
5	banana$	1	0
6	na$	5	0
7	nana$	3	2

They do not represents the sequence it self.

They are an index to the suffixes of the sequences, plus other regarding data.

SA + LCP + sequence

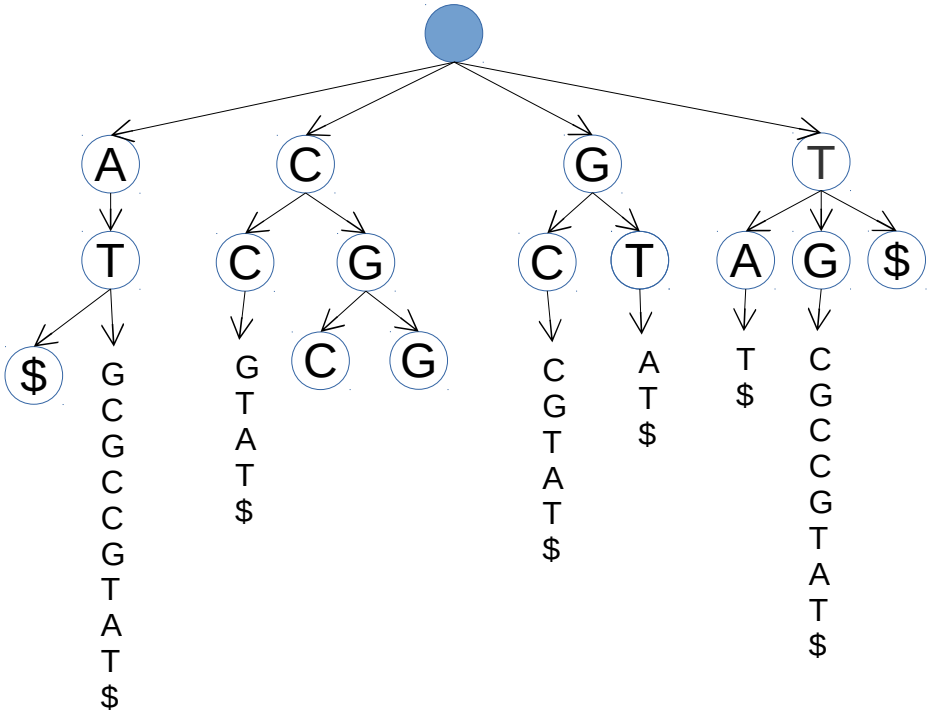
Dictionaries

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools

SA + LCP + Sequence = Dictionary

S = ATGCGCCGTAT



suffix	SA[i]	LCP[i]
ATGCGCCGTAT$	1	0
AT$	9	2
CCGTAT$	5	0
CGCCGTAT$	3	1
CGTAT$	6	2
GCCGTAT$	4	0
GTAT$	7	1
TAT$	8	0
TGCGCCGTAT$	2	1
T$	10	1

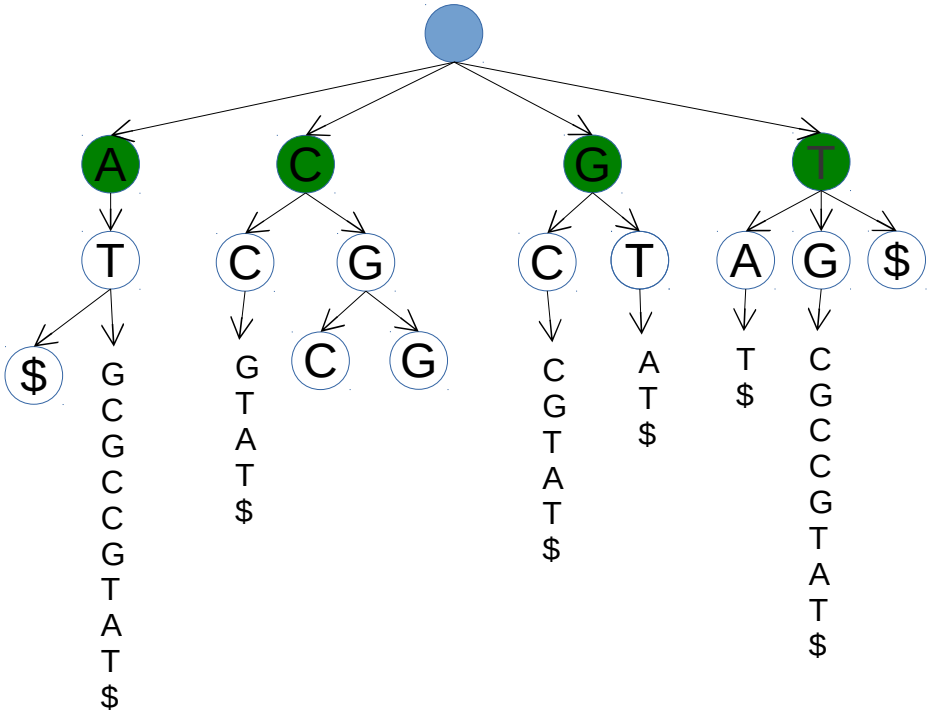
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools

SA + LCP + Sequence = Dictionary

S = ATGCGCCGTAT

1-mers



	suffix	SA[i]	LCP[i]
A	ATGCGCCGTAT$	1	0
	AT$	9	2
C	CCGTAT$	5	0
	CGCCGTAT$	3	1
	CGTAT$	6	2
G	GCCGTAT$	4	0
	GTAT$	7	1
T	TAT$	8	0
	TGCGCCGTAT$	2	1
	T$	10	1

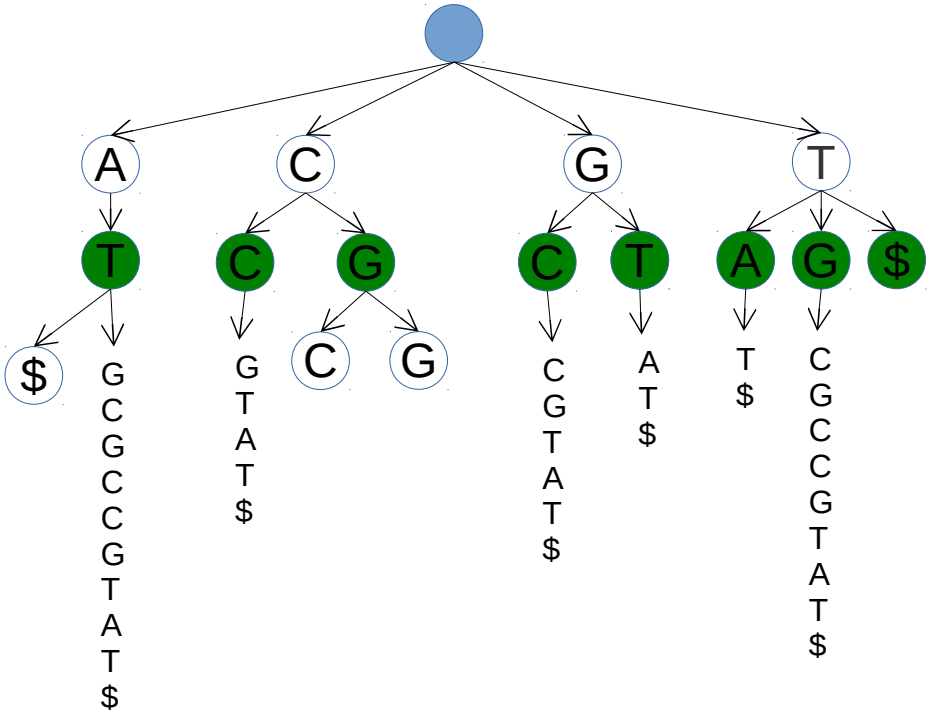
GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools

SA + LCP + Sequence = Dictionary

S = ATGCGCCGTAT

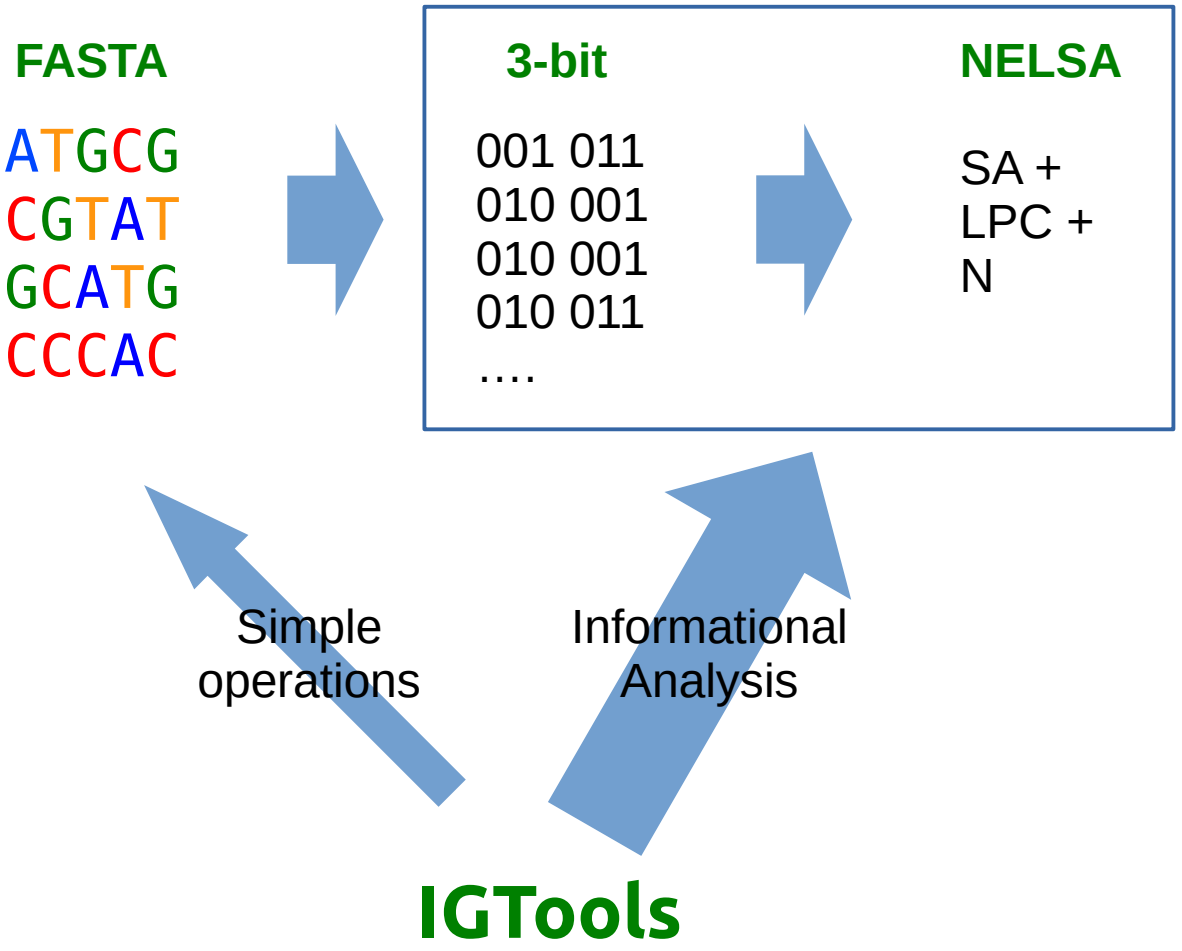
2-mers



	suffix	SA[i]	LCP[i]
AT	ATGCGCCGTAT$	1	0
	AT$	9	2
CC	CCGTAT$	5	0
CG	CGCCGTAT$	3	1
	CGTAT$	6	2
GC	GCCGTAT$	4	0
GT	GTAT$	7	1
TA	TAT$	8	0
TG	TGCGCCGTAT$	2	1
	T$	10	1

GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools
The recipe



GGAGTGAG
ACGTTAC
TCATTCATG
GGAGAGTT

IGTools
Dictionary Iterator

Iterator(S,k) → iterate over $D_k(S)$ following the lexicographic order

Why?
Because, it provides a simple interface to traverse the index, indipendently form the value of k.

Iterator(S,k)


```
NELSA.iterator(k)

while( it.next() ){
  …
}
```

	suffix	SA[i]	LCP[i]
AT	ATGCGCCGTAT$	1	0
	AT$	9	2
CC	CCGTAT$	5	0
CG	CGCCGTAT$	3	1
	CGTAT$	6	2
GC	GCCGTAT$	4	0
GT	GTAT$	7	1
TA	TAT$	8	0
TG	TGCGCCGTAT$	2	1
	T$	10	1