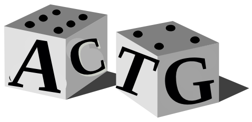
DNA & PROBABILITY



P(2 bp subsequence in n bp DNA sequence)

The probability for a 2-nucleotide-base subsequence occurring within an n-nucleotide base DNA sequence:

S = XY,

$$P(n) = 1 - (\frac{1}{2} + \frac{1}{\sqrt{3}})(\frac{1}{2} + \frac{\sqrt{3}}{4})^n - (\frac{1}{2} - \frac{1}{\sqrt{3}})(\frac{1}{2} - \frac{\sqrt{3}}{4})^n$$

S = XX,

$$P(n) = 1 - (\frac{1}{2} + \frac{5}{2\sqrt{21}})(\frac{3}{8} + \frac{\sqrt{21}}{8})^n - (\frac{1}{2} - \frac{5}{2\sqrt{21}})(\frac{3}{8} - \frac{\sqrt{21}}{8})^n$$

PROBABILITY & DNA

alphabet $\mathbb{A}$ containing L > 1 symbols
DNA: $\mathbb{A}$ = {A, C, G, T} , L = 4

subsequence S, length k
sequence, length n ≥ k

P(n) for S as subsequence in sequence

$S = s_1 \, s_2 \ldots s_k$

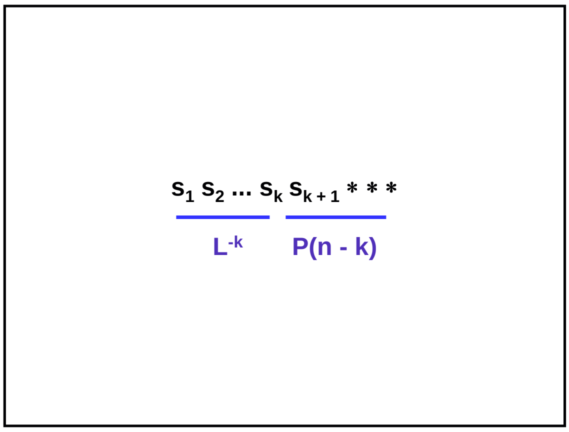
S self-overlapping with shift w

$s_1 = s_{1+w}, \; s_2 = s_{2+w}, \; \ldots, \; s_{k-w} = s_k$

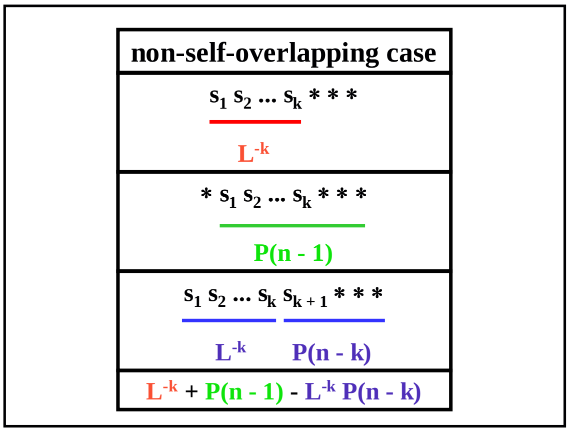
non-self-overlapping case

$\underbrace{s_1 \, s_2 \ldots s_k}_{L^{-k}} * * *$

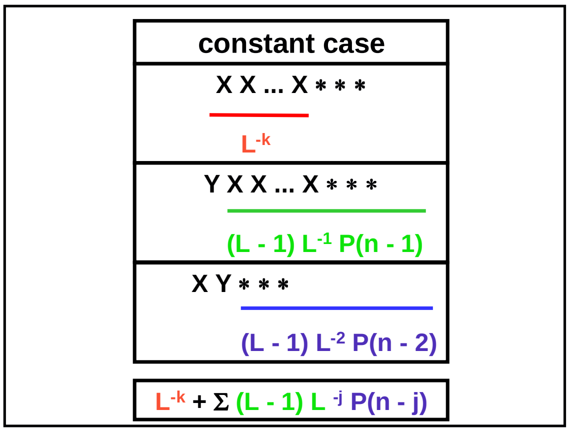
$* \underbrace{s_1 \, s_2 \ldots s_k}_{P(n-1)} * * *$



$$\underbrace{s_1\ s_2\ \dots\ s_k}_{L^{-k}}\ \underbrace{s_{k+1}\ *\ *\ *}_{P(n\ -\ k)}$$



non-self-overlapping case
$\underbrace{s_1\ s_2\ \dots\ s_k}_{L^{-k}}\ *\ *\ *$
$*\ \underbrace{s_1\ s_2\ \dots\ s_k\ *\ *\ *}_{P(n\ -\ 1)}$
$\underbrace{s_1\ s_2\ \dots\ s_k}_{L^{-k}}\ \underbrace{s_{k+1}\ *\ *\ *}_{P(n\ -\ k)}$
$L^{-k} + P(n - 1) - L^{-k}\ P(n - k)$



constant case
$\underbrace{X\ X\ \dots\ X}_{L^{-k}}\ *\ *\ *$
$Y\ \underbrace{X\ X\ \dots\ X\ *\ *\ *}_{(L\ -\ 1)\ L^{-1}\ P(n\ -\ 1)}$
$X\ \underbrace{Y\ *\ *\ *}_{(L\ -\ 1)\ L^{-2}\ P(n\ -\ 2)}$

$$L^{-k} + \Sigma\ (L - 1)\ L^{-j}\ P(n - j)$$

GENERAL CASE

$S = s_1\ s_2\ ...\ s_k$

S self-overlapping with shift w

$S = S_1\ S_2\ \dots\ S_m$

S_i segment, length l_i , $w_i = \Sigma\ l_i$

S = ACAACATACAACATACAACA

S = ACAACAT ACAACAT ACA AC A

$w_1 = 7$ $w_2 = 14$ $w_3 = 17$

General Case

$$P(n) = L^{\ k} + \sum_{j=1}^{m} (L^{\ w_{j-1}} P(n - w_{j\ 1} - 1) - L^{-w_j} P(n - w_j))$$

non-self-overlapping case, m = 1, $w_1 = k$

$$P(n) = L^{-k} + \sum_{j=1}^{m} (L^{-w_{j-1}} P(n - w_{j-1} - 1) - L^{-w_j} P(n - w_j))$$

$$= \quad {}^{-k} + \quad {}^{-w_0} \qquad - \quad {}_0 - \quad - \quad {}^{-w_1} \qquad - w_1))$$

$$= L^{-} \ + (L^0 P(n-1) - L^{-k} P(n-k))$$

$$= L^{-k} + P(n-1) - L^{-k} P(n-k)$$

$w_0 = 0$, $w_m = k$

Recursion formula may be used to set lower and upper bounds for P(n)

For fixed k ≤ n,

P(n) minimal if S is constant

P(n) maximal if S is non-self-overlapping

otherwise strictly between these extremes*

Recursion formula may be implemented as computer algorithm

Eigenvalue eigenvector analysis enables calculation of P(n)

‘Dupcheck’
Windows, Macintosh, Linux
Maple, Mathematica, Matlab

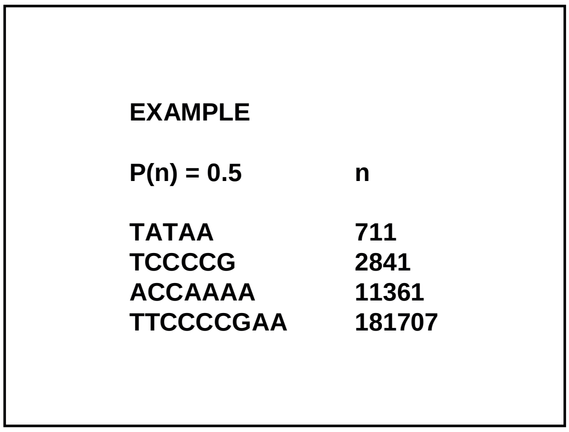
EXAMPLE

TATAA = eukaryotic TATA box

P(500) = 0.385231

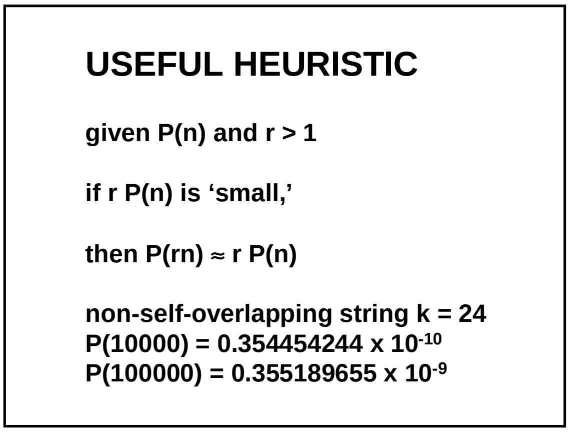
P(711) = 0.5

P(5000) = 0.992557



EXAMPLE

P(n) = 0.5	n
TATAA	711
TCCCCG	2841
ACCAAAA	11361
TTCCCCGAA	181707



USEFUL HEURISTIC

given P(n) and r > 1

if r P(n) is ‘small,’

then P(rn) ≈ r P(n)

non-self-overlapping string k = 24
P(10000) = 0.354454244 x 10^{-10}
P(100000) = 0.355189655 x 10^{-9}