Introduction to Bioinformatics

JTMS-19

Marc-Thorsten Hütt mhuett@constructor.university

Felix Jonas fjonas@constructor.university

session Wed, September 18, 2024

First examples of algorithms, probability models

What is this session about?

Boyer-Moore algorithm and suffix trees are introduced. The concept of a probability model is discussed and key types of probabilities are described: conditional, joint, marginal, posterior, prior probabilities and likelihood.

How can you revise the material after the session?

Read Durbin et al. chapter 1.3

Look at the two algorithms on

https://en.wikipedia.org/wiki/Boyer–Moore_string-search_algorithm

https://en.wikipedia.org/wiki/Suffix_tree

alternative reading: Hütt/Dehnert chapters 2.1 – 2.4

(A few comments on)
Algorithms and data structures

Basic definitions

- **String**
- **Graph**
- **Tree**
- **Path**
- **Algorithm**

Run time of an algorithm

Examples:

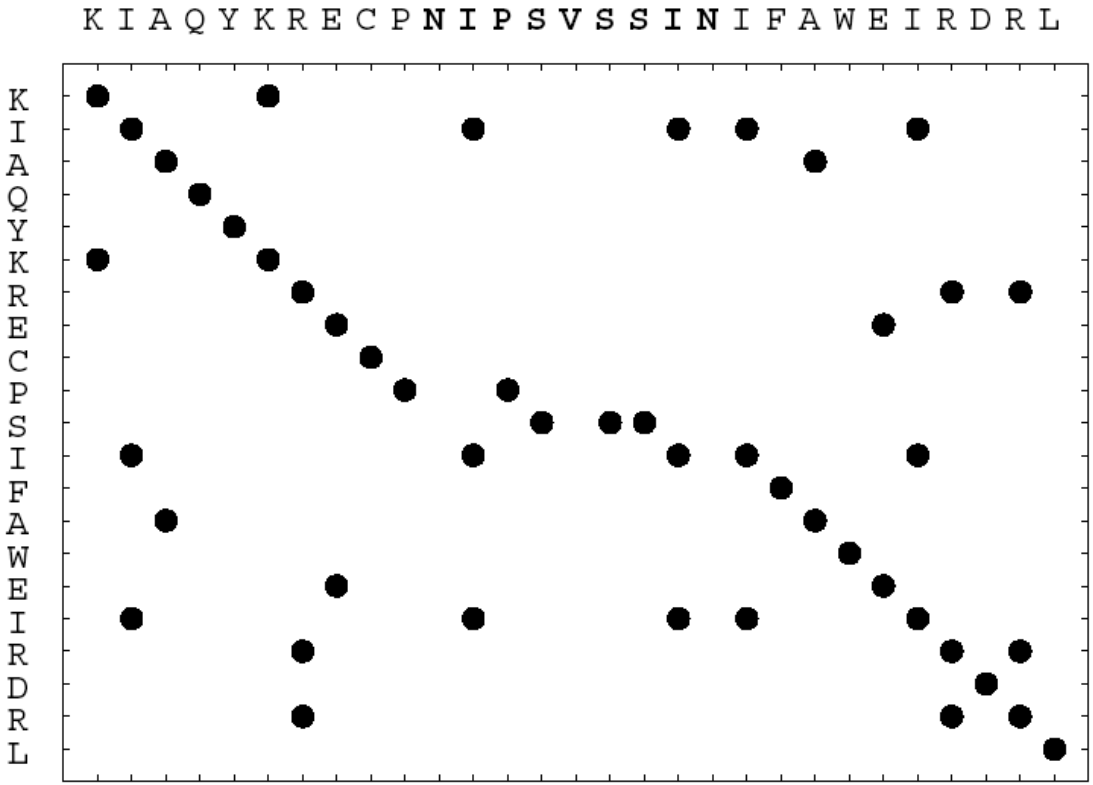
operation on a sequence of numbers

dot plot

paths on graphs

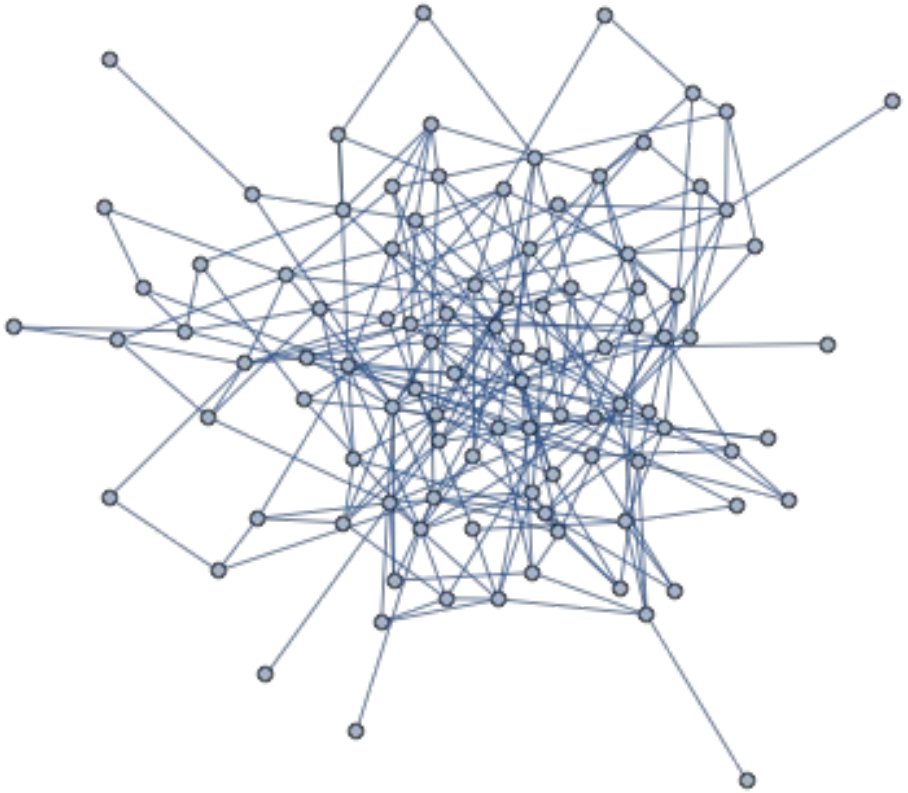
Run time of an algorithm

Example: dot plot



Run time of an algorithm

Example: paths on a graph



n	n!
1	1
2	2
3	6
4	24
5	120
6	720
7	5040
8	40 320
9	362 880
10	3 628 800
11	39 916 800
12	479 001 600
13	6 227 020 800
14	87 178 291 200
15	1 307 674 368 000

Exact matching of two strings:

- **target (text) sequence t**
- **pattern sequence p**

Example of a naive algorithm:

Input: pattern $p = p_1\ p_2\ ...\ p_m$
text $t = t_1\ t_2\ ...\ t_n$

Algorithm:

```
I = {},
For j = 0 to n–m do
        i = 1
        While p_i = t_j+1 and i <= m do
                i := i+1
        If i = m+1 then
                I := I U {j+1}
End(for)
```

Exact matching of two strings:

- target (text) sequence t
- pattern sequence p

Naive algorithm:

- shifts p along t from left to right by single symbols
- compares p with t from left to right

Boyer-Moore algorithm:

- shifts by more than one symbol
- compares p with t from right to left

- preprocessing of p is needed for that!

- two rules for determining the shift size

    - bad character rule (BCR)
    - good suffix rule (GSR)

Exact matching of two strings:

- **target (text) sequence t**
- **pattern sequence p**

Programming Techniques

G. Manacher, S.L. Graham Editors

A Fast String Searching Algorithm

Robert S. Boyer
Stanford Research Institute

J Strother Moore
Xerox Palo Alto Research Center

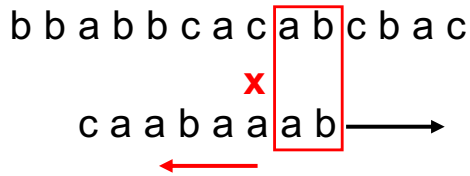
Communications of the ACM

October 1977
Volume 20
Number 10

Exact matching of two strings:

- target (text) sequence t
- pattern sequence p

Boyer-Moore algorithm: small example



- bad character rule (BCR)

$t_{j+m} \leftrightarrow p_m$

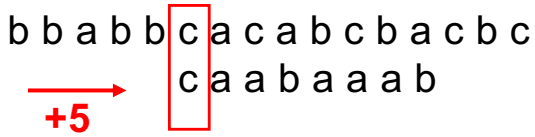
$t_{j+m-1} \leftrightarrow p_{m-1}$

$t_{j+m-2} \leftrightarrow p_{m-2}$

→ i = m – 2

→ store t_{j+m-2} = c

→ shift to the rightmost c in p



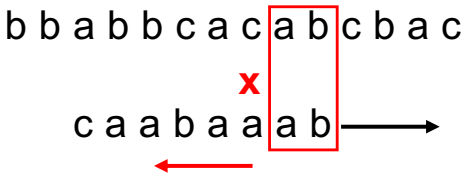
preprocessing of p:

for all s from the alphabet
compute b(s), the rightmost
position of s

Exact matching of two strings:

- target (text) sequence t
- pattern sequence p

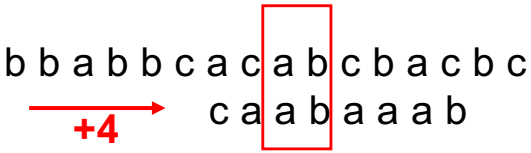
Boyer-Moore algorithm: small example



- good suffix rule (GSR)

  $p_{m-1}p_m$: good suffix

  → shift to the rightmost occurrence of $p_{m-1}p_m$ in p



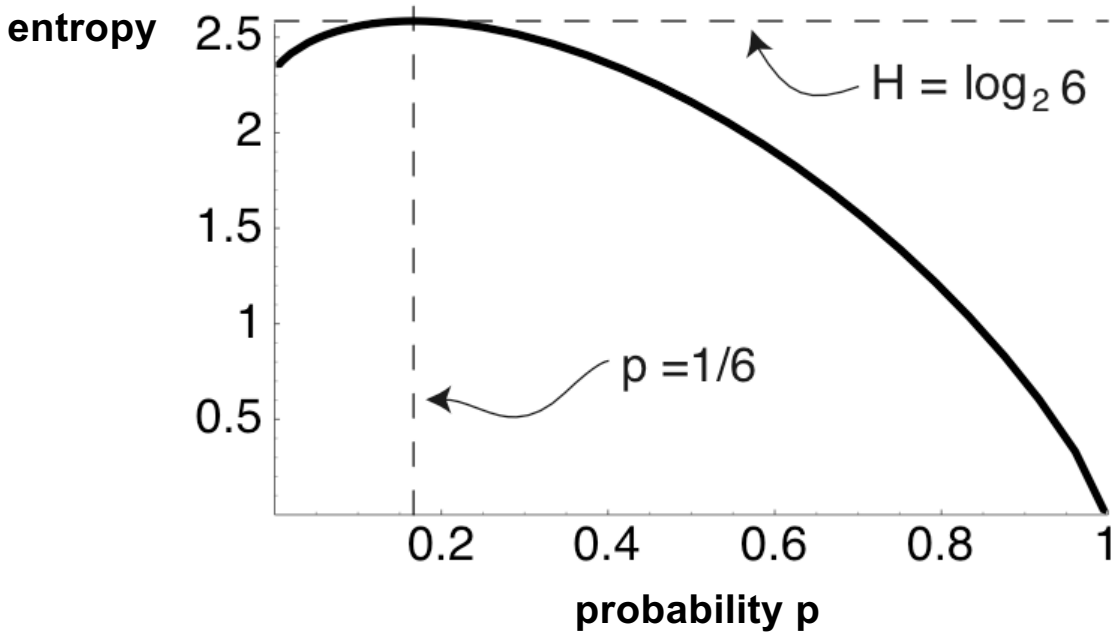
Brief digression:

What determines, if the Boyer -Moore algorithm runs fast?

→ **entropy of a string**

aaabaaaaaaaaabaaaaaaabbbaaaaaa → **low entropy**
abaababbababaababababababababaab → **high entropy**

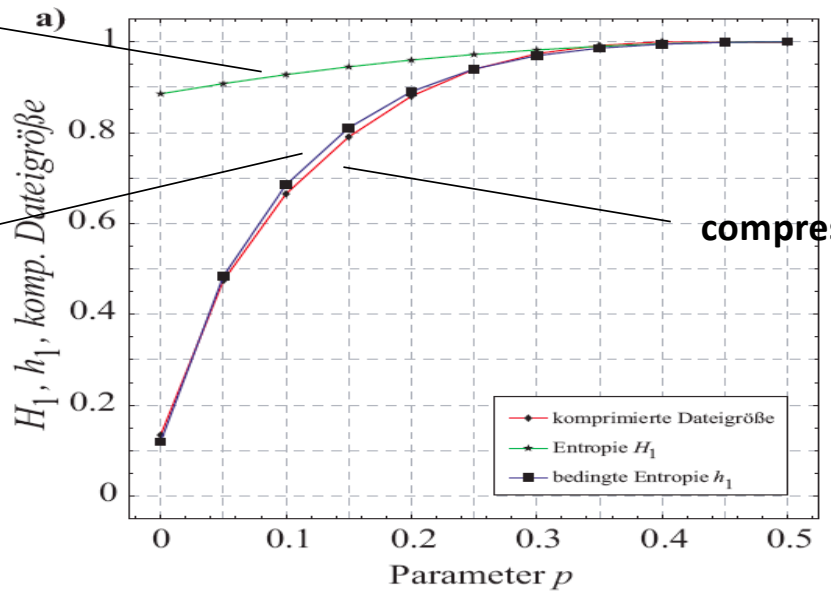
entropy $H = -\sum_{i \in \Sigma} p_i \log_2 p_i.$



p=0.1 *p=0.2* *p=0.3* *p=0.4* *p=0.5*

entropy H_1

conditional entropy h_1

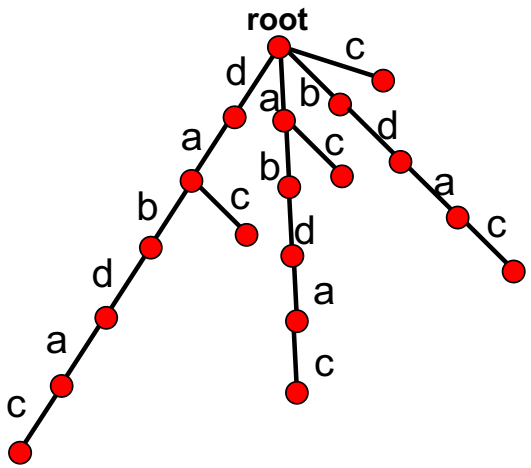


compressed file size

Suffix trees

- **preprocessing of text t**
- **efficient storing of the suffix information of t**

(1) d a b d a c



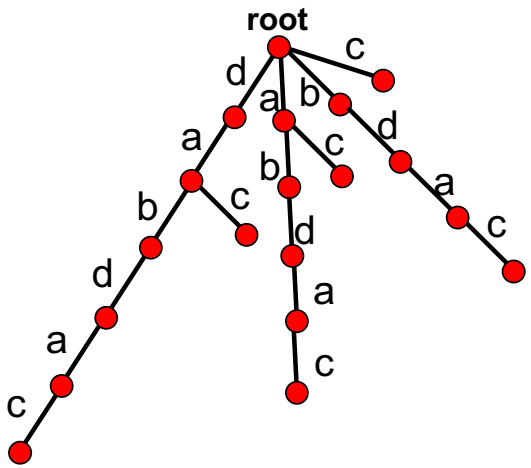
Suffix tree for $t = t_1 t_2 \ldots t_n$

- n leaves
- edges in the tree are labeled from the alphabet
- all edges leaving a node carry different symbols
- path from root to leaf i has labels $t_i\ t_{i+1} \ldots t_n$

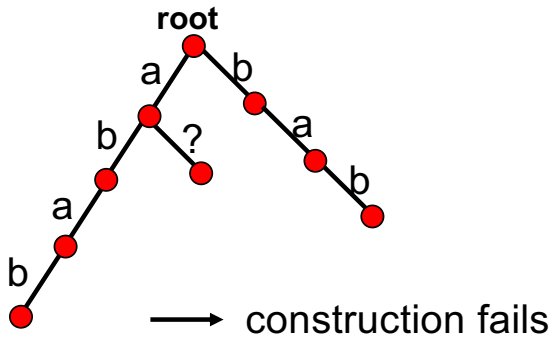
Suffix trees

- **preprocessing of text t**
- **efficient storing of the suffix information of t**

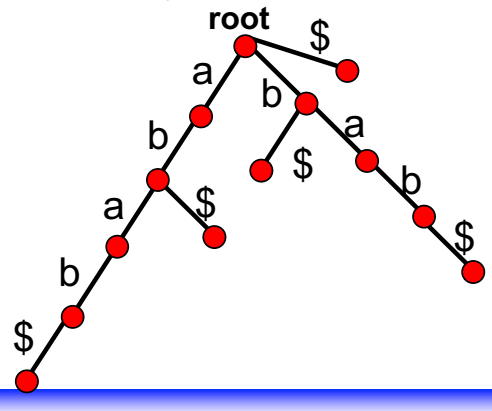
(1) d a b d a c



(2) a b a b



(3) a b a b $



From a research article to data sets – a sample path

Comprehensive analysis of CpG islands in human chromosomes 21 and 22

Daiya Takai* and Peter A. Jones

Department of Biochemistry and Molecular Biology, University of Southern California/Norris Comprehensive Cancer Center, Keck School of Medicine of the University of Southern California, 1441 Eastlake Avenue, Los Angeles, CA 90033

CpG islands are useful markers for genes in organisms containing 5-methylcytosine in their genomes. In addition, CpG islands located in the promoter regions of genes can play important roles in gene silencing during processes such as X-chromosome inactivation, imprinting, and silencing of intragenomic parasites. The generally accepted definition of what constitutes a CpG island was proposed in 1987 by Gardiner-Garden and Frommer [Gardiner-Garden, M. & Frommer, M. (1987) *J. Mol. Biol.* 196, 261–282] as being a 200-bp stretch of DNA with a C+G content of 50% and an observed CpG/expected CpG in excess of 0.6. Any definition of a CpG island is somewhat arbitrary, and this one, which was derived before the sequencing of mammalian genomes, will include many sequences that are not necessarily associated with controlling regions of genes but rather are associated with intragenomic parasites. We have therefore used the complete genomic sequences of human chromosomes 21 and 22 to examine the properties of CpG islands in different sequence classes by using a search algorithm that we have developed. Regions of DNA of greater than 500 bp with a G+C equal to or greater than 55% and observed CpG/expected CpG of 0.65 were more likely to be associated with the 5′ regions of genes and this definition excluded most *Alu*-repetitive elements.

3740–3745 | PNAS | March 19, 2002 | vol. 99

Comprehensive analysis of CpG islands in human chromosomes 21 and 22

Daiya Takai* and Peter A. Jones

Department of Biochemistry and Molecular Biology, University of Southern California/Norris Comprehensive Cancer Center, Keck School of Medicine of the University of Southern California, 1441 Eastlake Avenue, Los Angeles, CA 90033

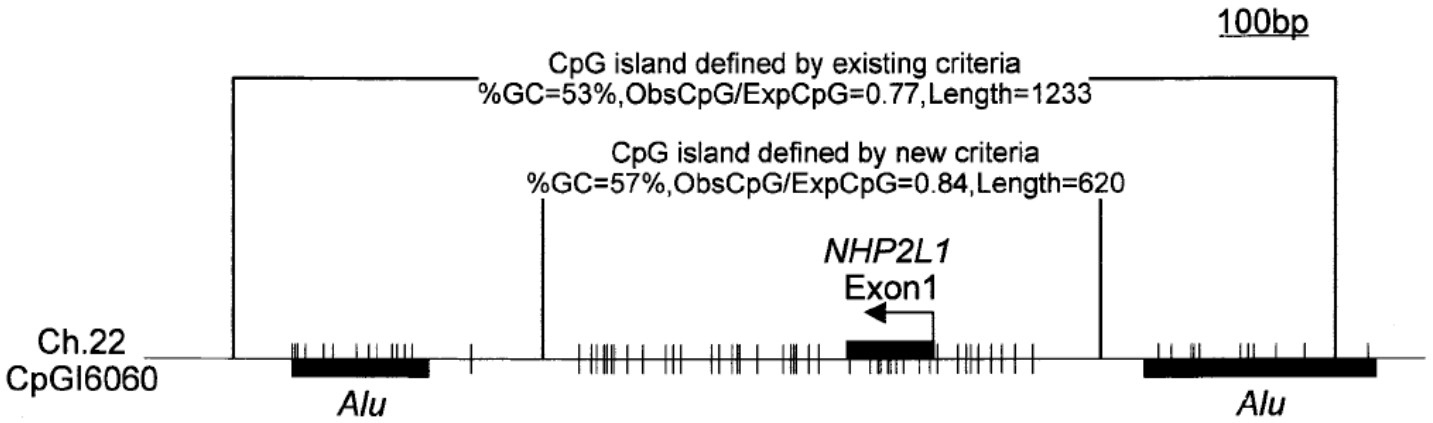
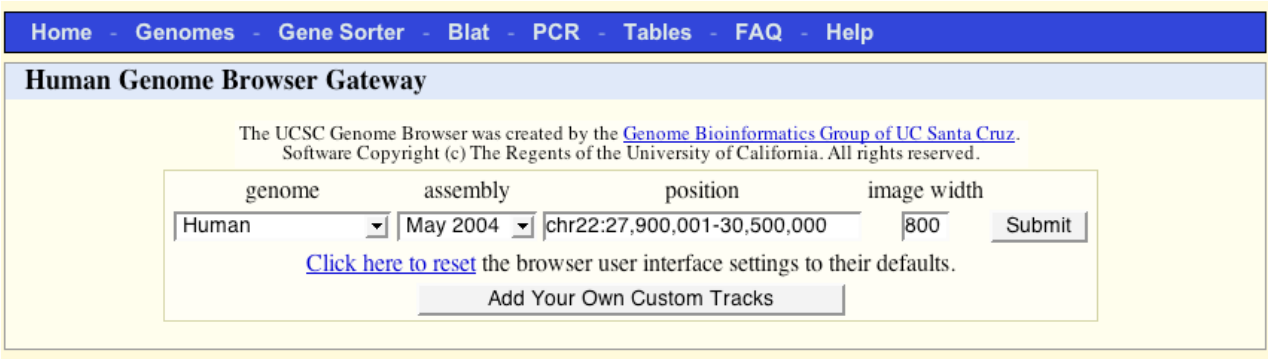


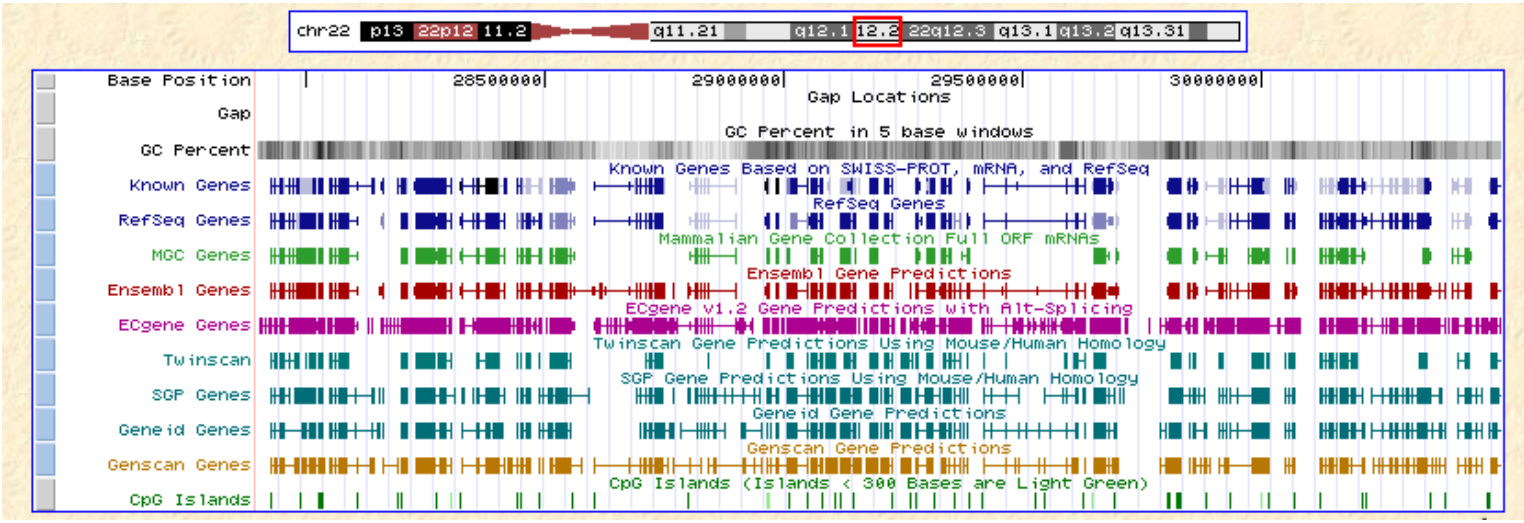
Fig. 3. The modified criteria also helped remove *Alu* sequences previously identified as part of 5′ region CpG islands. In this example, a 1,233-bp fragment originally extracted by the algorithm included two *Alu* sequences with some CpG suppression associated with the nonhistone chromosome protein 2 like 1 (*NHP2L1*). The modified stringent criteria reduced the size of the island to 620 bp and excluded the *Alu* sequences.

3740–3745 | PNAS | March 19, 2002 | vol. 99

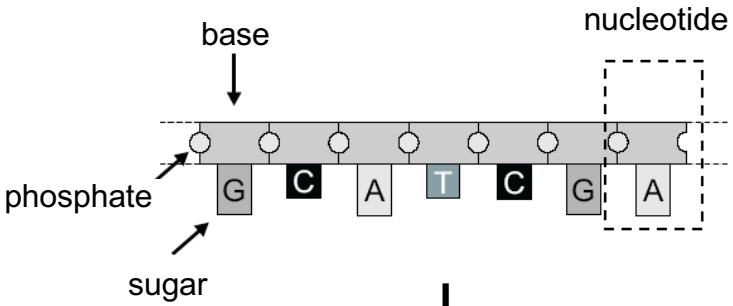
```
CpG island searcher command line version
Ver 1.3 released 05/21/03
by Takai D. & Jones PA.

Selected lower limits: %GC=55, ObsCpG/ExpCpG=0.65, Length=500
human_ch, CpG island 1, start=83409, end=85437, %GC=64.6, ObsCpG/ExpCpG=0.888, Length=2029
human_ch, CpG island 2, start=199701, end=200609, %GC=55, ObsCpG/ExpCpG=0.756, Length=909
human_ch, CpG island 3, start=224154, end=224658, %GC=55.2, ObsCpG/ExpCpG=0.65, Length=505
human_ch, CpG island 4, start=261555, end=262391, %GC=56.8, ObsCpG/ExpCpG=0.65, Length=837
human_ch, CpG island 5, start=262849, end=263596, %GC=60.1, ObsCpG/ExpCpG=0.65, Length=748
human_ch, CpG island 6, start=353407, end=353937, %GC=63.6, ObsCpG/ExpCpG=0.652, Length=531
human_ch, CpG island 7, start=511685, end=512510, %GC=59.2, ObsCpG/ExpCpG=0.663, Length=826
human_ch, CpG island 8, start=924779, end=925798, %GC=61, ObsCpG/ExpCpG=0.672, Length=1020
human_ch, CpG island 9, start=981212, end=981828, %GC=58.1, ObsCpG/ExpCpG=0.651, Length=617
human_ch, CpG island 10, start=996188, end=996688, %GC=68, ObsCpG/ExpCpG=0.656, Length=501
human_ch, CpG island 11, start=1012831, end=1013415, %GC=63.4, ObsCpG/ExpCpG=0.651, Length=585
human_ch, CpG island 12, start=1020111, end=1021323, %GC=64.7, ObsCpG/ExpCpG=0.81, Length=1213
```


Home - Genomes - Gene Sorter - Blat - PCR - Tables - FAQ - Help
Human Genome Browser Gateway
The UCSC Genome Browser was created by the Genome Bioinformatics Group of UC Santa Cruz.
Software Copyright (c) The Regents of the University of California. All rights reserved.
genome
assembly
position
image width
Human
May 2004
chr22:27,900,001-30,500,000
800
Submit
Click here to reset the browser user interface settings to their defaults.
Add Your Own Custom Tracks


chr22 p13 22p12 11.2 q11.21 q12.1 12.2 22q12.3 q13.1 q13.2 q13.31
Base Position
28500000
29000000
29500000
30000000
Gap
Gap Locations
GC Percent
GC Percent in 5 base windows
Known Genes
Known Genes Based on SWISS-PROT, mRNA, and RefSeq
RefSeq Genes
RefSeq Genes
MGC Genes
Mammalian Gene Collection Full ORF mRNAs
Ensembl Genes
Ensembl Gene Predictions
ECgene Genes
ECgene v1.2 Gene Predictions with Alt-Splicing
Twinscan
Twinscan Gene Predictions Using Mouse/Human Homology
SGP Genes
SGP Gene Predictions Using Mouse/Human Homology
Geneid Genes
Geneid Gene Predictions
Genscan Genes
Genscan Gene Predictions
CpG Islands
CpG Islands (Islands < 300 Bases are Light Green)

Probability models



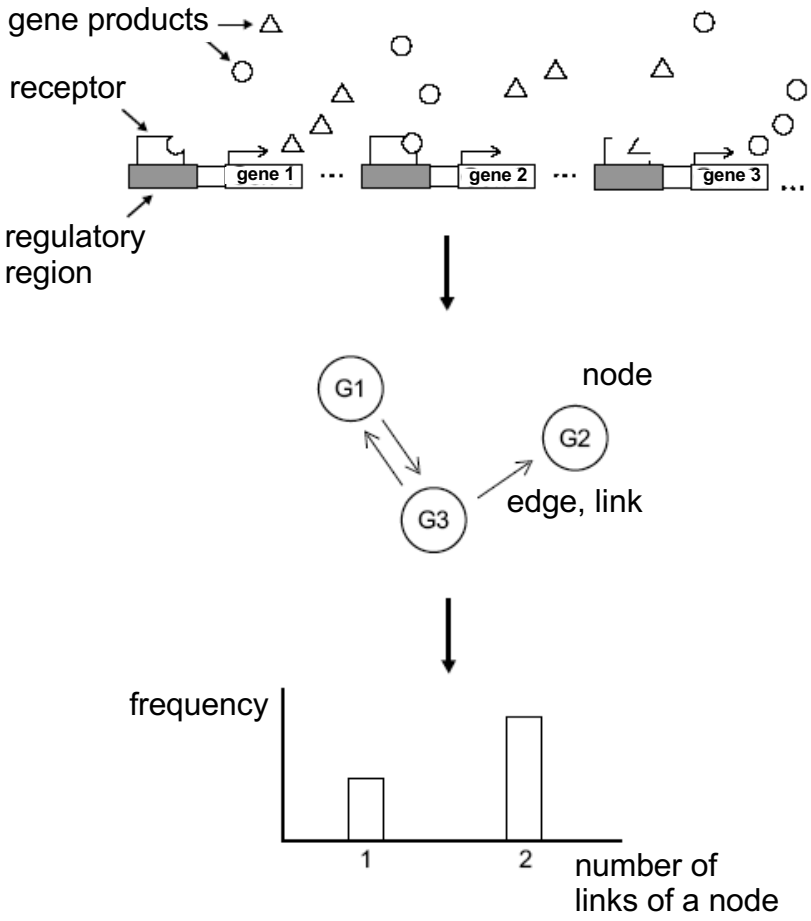
biochemical structure

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

sequence of bases

functional annotation on the symbol sequence

structure → abstraction → analysis



structure → abstraction → analysis

How to quantify the match between data and a probability model?