Quantitative Biology / Bioinformatics Homework 1
Fall 2010

Due Date: Email mschatz@cshl.edu by 11:59pm on Nov 5

1. E-values and Exact Matching

a) Plot (in linear and log space) the expected number of occurrences e of a pattern P of length m in a string of length n whose characters are sampled independently and uniformly from an alphabet Σ for m=[1,40], n={1.8Mb, 4.7Mb, 100Mb, 130Mb, 3.1Gb, 670Gb}, and |Σ|={4,20}.

$$e = \frac{(n - m + 1)}{|\Sigma|^m}$$

b) Determine how many bases long P should be to ensure that occurrences of P are unlikely to be chance events in genomes of the following sizes:

i. 1.8Mb (Haemophilus influenzae - the flu)
ii. 4.7Mb (Escherichia coli - intestinal bacteria)
iii. 100Mb (Caenorhabditis elegans - roundworm)
iv. 130Mb (Drosophila melanogaster - fruit fly)
v. 3.1Gb (Homo sapiens - human)
vi. 670Gb (Polychaos dubium – Amoeba)

c) Implement the brute force algorithm in the language (Matlab, Perl, Python, C, C++, etc) of your choice and run it on the E. coli genome:

http://schatzlab.cshl.edu/teaching/2010/Ecoli.fa

Compute the number of occurrences for each of the following queries, and discuss the degree to which the empirical number of matches is consistent with the theoretical model of part (a). Point out, in particular, any particularly significant deviations from the theoretical model.

Gattaca:	GATTACA
Gattaca^2:	GATTACAGATTACA
Gattaca^3:	GATTACAGATTACAGATTACA
GC Content:	A + T vs G + C
Start Codon:	ATG
Stop Codons:	TAG, TAA, TGA
Simple Repeats:	A_1 – A_{15} (A, AA, AAA, …, AAAAAAAAAAAAAAA)

2. Binary Search Analysis

Consider a binary search over the (sorted) numbers 1-100. The number of steps to find a query will be at most lg 100 = 7, but some queries will take fewer than this. For example if the query is 50, it will be found on the first try, if it is 25, it will be found on the second, and so forth.

a) Implement binary search in your language of choice (Matlab, Perl, Python, etc).

b) Using your implementation, plot (in linear space) of the number of steps required for all possible queries in the range [1-100] (x-coordinate = query, y-coordinate = number of steps)

c) Examine the distribution of steps by constructing a histogram (x-coordinate=number of steps, y-coordinate=number of queries requiring that many steps)

3. Practical Limits of Complexity

a) Plot (in linear and log space) the runtime t as a function of n in the range t=[1,10000], n=[1,200] for the following complexities:

t=2^n, t=n^100, t=n^3, t=n^2, t=n lg n, t=n, t=lg n

b) Assume that a computer can sustain executing 1M instructions per second including the time it takes to load the data off disk, process, and write the results back to disk. Compute how many items can be evaluated in the following durations for the following complexities (1M items can be processed in 1 sec with T=n, but only 1000 items can be processed using T=n^2).

T	Items / sec	Items / minute	Items / hour	Items / day	Items / week	Items / year
2^n						
n^100						
n^3						
n^2						
n lg n						
n						
lg n						

4. Challenge Question: Suffix Arrays

In your language of choice, construct the suffix array for (part of) E coli and compare the running time of binary search using the suffix array to the brute force approach. Warning: The builtin sort function in Matlab requires an explicit set of sequences, so you will have to implement a custom quicksort to construct the suffix array for a large genome.