July 05, 2021

Assignment 12
Algorithms for Sequence Analysis, Summer 2021

Algorithmic Bioinformatics · Prof. Dr. Sven Rahmann

Hand in date: Monday, July 12, before 20:00

Exercise 1: Choice of q for a q-gram index (4 Theory)

When indexing a collection of DNA sequences of total length n with a q-gram index, we typically want to choose $q \approx \mathrm{round}(\log_4(n/8))$. Why is this a good choice?

Exercise 2: Parameters of QUASAR/SWIFT (4 Theory)

DNA Database search tools (or read mappers) based on q-gram filtration tend to have many parameters. For QUASAR/SWIFT, important parameters are

- the block size b (or parallelogram width) in the database or genome,
- the window length w in the query or read.

Explain advantages and disadvantages of increasing b and w (for fixed distance threshold d and choice of q).

Exercise 3: Hash functions (4 Theory)

Consider strings of (fixed) length n over the DNA alphabet (of size 4). Pick a random hash function as follows: Randomly select k out of the n positions of the string, and concatenate the characters, yielding a string of length k, i.e. a k-mer x. The hash value is the integer encoding of x (see Exercise 4, $enc(x)$).

Assume that two sequences s, t of length n have Hamming distance d. What is the probability that their hash values are equal?

Notes: This is in fact an exercise in combinatorics. Randomness/probability is only over choice of hash function, i.e., we assume nothing about the generation of the strings.

Exercise 4: q-gram or k-mer index (4 Programming)

A DNA q-gram (or k-mer) index consists of the suffix array `pos` and a table `start` of size $4^q + 1$ that contains the starting ranks in `pos` of every q-gram x (the final entry contains the rank n, which does not exist, as a sentinel). A q-gram is base-4 integer encoded ($\mathtt{A} \mapsto 0, \mathtt{C} \mapsto 1, \mathtt{G} \mapsto 2, \mathtt{T} \mapsto 3$), so $enc(\mathtt{TAC}) = (301)_4 = 49$. So $\mathtt{pos}[\mathtt{start}[enc(x)] : \mathtt{start}[enc(x)+1]]$ are all text positions where the q-gram x occurs.

Use the provided code (that computes the suffix array and lcp array) and write a function that computes the q-gram index for given q. Apply your program to the provided *E. coli* genome (gzipped FASTA format, needs to be gunzipped). Show a (textual) length histogram of the 9-gram buckets, i.e., output text of the form

```
length_of_qgram_bucket  number_of_buckets
```

Omit rows where the second column is zero.