Minimalna dolžina fragmentov za uspešno rekonstrukcijo genomov

Ivan Antešić (63130003)

January 17, 2020

1 Uvod

V okviru tretjega mini projekta pri predmetu Uvod v bioinformatiko smo v Pythonu implementirali algoritem, ki za podano krožno zaporedje nukleotidov določi minimalno dolžino fragmentov, ki jih lahko nato uspešno in unikatno sestavimo nazaj v originalno zaporedje. V nadaljevanju je na kratko predstavljen algoritem in pridobljeni rezultati.

2 Podatki

Algoritem smo testirali s petimi geni. Prvi gen (Homo sapiens S100A4) vsebuje 306 nukleotidov, drugi gen (Danio rerio VAMP2) 333, tretji (Bos taurus RPL39) 156, četrti (Canis lupus familiaris RPL35) 372 in zadnji gen (Mus musculus MNS1) 1477 nukleotidov.

3 Algoritem

Na začetku fragmentiramo vhodno zaporedje na podzaporedja dolžine k - t.i. kmere. Vsak kmer se začne z enim znakom zaporedja in obsega naslednjih k znakov. Če prekoračimo dolžino zaporedja vzamemo preostale znake od začetka (zato je zaporedje krožno). Primer ATACGGTC tako fragmentiramo na kmere ATA, TAC, ACG, CGG, GGT, GTC, TCA in CAT.

Iz pridobljenih kmerov sestavimo de Brujinov graf. Vsak kmer predstavlja usmerjeno povezavo od vozlišča pripone $[1 : k]$ do vozlišča predpone $[0 : k - 1]$. Na takšnem grafu lahko iščemo Eulerjev obhod s katerim rekonstruiramo originalno zaporedje.

Nato preiščemo povezave de Brujinovega grafa in odkrijemo, če graf vsebuje vejitve: te lahko povzročijo različne Eulerjeve obhode, kar pomeni, da ne obstaja ena sama unikatna rešitev.

Postopoma povečujemo dolžino kmerov od $k = 2$, dokler ne najdemo de Brujinovega grafa brez vejitev. Toda ni nujno, da vse vejitve povzročijo različne Eulerjeve obhode. V zaporedju se lahko nahajajo območja, ki se večkrat ponovijo (angl. repeats) in povzročijo različne vrstne rede Eulerjeva obhoda, ki pa so na koncu le zamaknjene različice enega samega cikla.

Zato z grobo oceno dolžine k izračunamo de Brujinov graf in poiščemo vse možne (različne) Eulerjeve obhode. V ta namen uporabimo prilagojen Hierholzerjev algoritem, ki spremlja ob-

hode za vse vejitve. Postopoma znižujemo dolžino k dokler ne pridobimo več kot en Eulerjev obhod. To pomeni, da je bila prejšnja dolžina $k-1$ točna minimalna dolžina fragmentov za unikatno rekonstrukcijo zaporedja.

Na koncu še preverimo če se rekonstruirano zaporedje res ujema z vhodnim zaporedjem.

4 Rezultati

Pridobili smo naslednje rezultate. V tabeli 1 so vidne grobe ocene minimalnih dolžin fragmentov v tabeli 2 pa natančne dolžine, ki so končni rezultat algoritma. Opazimo lahko, da v večini primerov rahlo izboljšamo grobo oceno.

a

Table 1: **Približna** minimalna dolžina fragmentov, potrebna za uspešno in unikatno rekonstrukcijo danih genov.

gen	minimalna dolžina fragmentov
Homo sapiens S100A4	10
Danio rerio VAMP2	11
Bos taurus RPL39	9
Canis lupus familiaris RPL35	10
Mus musculus MNS1	17

Table 2: **Točna** minimalna dolžina fragmentov, potrebna za uspešno in unikatno rekonstrukcijo danih genov.

gen	minimalna dolžina fragmentov
Homo sapiens S100A4	9
Danio rerio VAMP2	10
Bos taurus RPL39	8
Canis lupus familiaris RPL35	10
Mus musculus MNS1	14

5 Zaključek

Za dolga zaporedja npr. zadnji testni gen, ki vsebuje 1477 nukleotidov bi bilo prepočasno iskati vse možne cikle za dolžine $k = 2, 3, ...$ S predpostavko, da vejitve povzročijo več možnih rekonstrukcij dobimo že dobro grobo oceno, ki jo nato le malo izboljšamo ter pridobimo točno rešitev. Tako učinkovito pohitrimo naš algoritem.

Honor Code

My answers to homework are my own work. I did not make solutions or code available to anyone else. I did not engage in any other activities that will dishonestly improve my results or dishonestly improve/hurt the results of others.