Elements of Bioinformatics, Autumn 2011, Exercise set 1

Return your answers to Esa Pitkänen (firstname.lastname@cs.helsinki.fi) on writing before the beginning of the review session, Mon 7.11 at 10am at the latest.

1. Examine the following double stranded DNA fragment:

   ```
   GGGGAGGTTGCATCATCACAAACATTTCAACTTCGCTGAGTCTCTGGAGGAGACAGT
   CCCCTCCAACGTAGTAGTGTTTGTAAAGTTGAAGCGACTCAGAGACCTCCTCTGTCA
   ```

   Compute the following statistics

   (a) $G + C$ content

   (b) $G + C$ skew using a window of length 10

   (c) Base frequencies

   (d) Dinucleotide frequencies

   (e) Trinucleotide frequencies

2. Codon adaptation index

   Figure 1 shows the codon usage of Human genome from `http://www.kazusa.or.jp/codon/cgi-bin/showcodon.cgi?species=9606`

   Below, two DNA fragments are given. Using the codon usage table, compute the Codon adaptation index for both sequences.

   (a) `ATGCTGGAATATATGTCAGCTGAAACAAAACTTGCG`

   (b) `ATGGGAATATATGTCAGCTGAAACAAAACTTGCGCT`

   Which of the sequences is more likely to come from a highly expressed gene?

3. Markovian models of sequences

   Below, a transition table for a first order Markov chain model of DNA is given together with base frequences under the same model.

	A	C	G	T
A	0.242	0.278	0.253	0.227
C	0.217	0.222	0.350	0.212
G	0.246	0.240	0.239	0.275
T	0.261	0.241	0.255	0.243

A	C	G	T
0.241	0.245	0.273	0.241

   Below two DNA sequences are given. Compute the probability of both sequences

   - under the above first order Markov chain model
   - under a model assuming each base appears independently with probability given by the base frequency table

***Homo sapiens* [gbpri]: 93487 CDS's (40662582 codons)**

fields: [triplet] [frequency: **per thousand**] ([number])

```
UUU 17.6(714298)  UCU 15.2(618711)  UAU 12.2(495699)  UGU 10.6(430311)
UUC 20.3(824692)  UCC 17.7(718892)  UAC 15.3(622407)  UGC 12.6(513028)
UUA  7.7(311881)  UCA 12.2(496448)  UAA  1.0( 40285)  UGA  1.6( 63237)
UUG 12.9(525688)  UCG  4.4(179419)  UAG  0.8( 32109)  UGG 13.2(535595)

CUU 13.2(536515)  CCU 17.5(713233)  CAU 10.9(441711)  CGU  4.5(184609)
CUC 19.6(796638)  CCC 19.8(804620)  CAC 15.1(613713)  CGC 10.4(423516)
CUA  7.2(290751)  CCA 16.9(688038)  CAA 12.3(501911)  CGA  6.2(250760)
CUG 39.6(1611801)  CCG  6.9(281570)  CAG 34.2(1391973)  CGG 11.4(464485)

AUU 16.0(650473)  ACU 13.1(533609)  AAU 17.0(689701)  AGU 12.1(493429)
AUC 20.8(846466)  ACC 18.9(768147)  AAC 19.1(776603)  AGC 19.5(791383)
AUA  7.5(304565)  ACA 15.1(614523)  AAA 24.4(993621)  AGA 12.2(494682)
AUG 22.0(896005)  ACG  6.1(246105)  AAG 31.9(1295568)  AGG 12.0(486463)

GUU 11.0(448607)  GCU 18.4(750096)  GAU 21.8(885429)  GGU 10.8(437126)
GUC 14.5(588138)  GCC 27.7(1127679)  GAC 25.1(1020595)  GGC 22.2(903565)
GUA  7.1(287712)  GCA 15.8(643471)  GAA 29.0(1177632)  GGA 16.5(669873)
GUG 28.1(1143534)  GCG  7.4(299495)  GAG 39.6(1609975)  GGG 16.5(669768)
```

Coding GC 52.27% 1st letter GC 55.72% 2nd letter GC 42.54% 3rd letter GC 58.55%

Figure 1: Human codon usage table

(a) `ATGGGAATATATGTCAGCTGAAACAAAACTT`

(b) `TAGTCCCGCAATATTGGTTCGGGAAGAAGGA`

Which sequence is more likely to originate from the first order Markov chain model?

4. Tacking unknown exon-intron boundaries RNA-seq based gene finding.

   Read the paper: "TopHat: discovering splice junctions with RNA-Seq" by Cole Trapnell, Lior Pachter, and Steven L. Salzberg `http://bioinformatics.oxfordjournals.org/content/25/9/1105.short`

   How does TopHat tackle the problem with reads that cross the exon-exon boundary in mRNA (correspondingly exon-intron boundary in DNA)?

5. Alignment allowing introns

   In the lectures, a dynamic programming scheme for aligning an amino acid sequence to DNA was mentioned. The idea is to fill in a table of scores S, where the item $S(i, j, k)$ is the score of best alignment of the length-j prefix of the DNA sequence $dna[1...j]$, to a length-i prefix of the amino acid sequence $protein[1...i]$ so that the alignment contains k introns.

   Assume the following scoring model: matching an amino acid to any valid codon (see Figure 1) contributes one unit to the score $Match(AAA, K) = 1$, matching any trinucleotide that is not a valid codon does not contribute to the score: $Match(AAA, K) = 0$. Deleting or inserting codons gives score $-\infty$ (i.e. should not be done).

   (a) Sketch a dynamic programming recurrence that can be used to fill in the score table S?

(b) What is the time complexity of the algorithm?

Hints (Attention! May spoil the fun of finding the solution yourself!):

- *Assume that the values* $S(i, j, k)$ *have already computed for the initial part of the table* $S(p, q, r)$ *where* $1 \leq p \leq i$, $1 \leq q \leq j$ *and* $0 \leq r \leq k$ *and* $(p, q, r) \neq (i, j, k)$
- *The best solution for* $S(i, j, k)$ *should correspond to extending a previously computed solution by (a) extending an existing alignment by matching dna(j,j+1,j+2) to protein(i) (b) adding an intron that has dna(j-1) as the last position and has length that maximizes the alignment score with a prefix of one less introns.*