Lab 2 - Gr. 14 - Bioinformatics (732A93)

Andreas Stasinakis (andst745), Hector Plata (hecpl268), Julius Kittler (julki092), Mim Kemal Tekin (mimte666), Stefano Toffol (steto820)

Assignment 1

```
## 33 DNA sequences in binary format stored in a list.
##
## Mean sequence length: 1982.879
##    Shortest sequence: 931
##     Longest sequence: 2920
##
## Labels:
## JF806202
## HM161150
## FJ356743
## JF806205
## JQ073190
## GU457971
## ...
##
## Base composition:
##     a     c     g     t
## 0.312 0.205 0.231 0.252
## (Total: 65.44 kb)
```

Base	Original frequency	Simulated frequency
a	0.3121	0.3120
c	0.2052	0.2045
g	0.2307	0.2329
t	0.2519	0.2505

1.1

Some text and equation here:

$$S = \sqrt{\frac{\sum_{r,s}(d_{rs} - \hat{d}_{rs})^2}{\sum_{r,s} d_{rs}^2}}. \tag{1}$$

1.2

Assignment 2

Note that, by convention, a *coding strand* is used when displaying a DNA sequence. "A coding strand, is the segment within double-stranded DNA that runs from 5' to 3', and which is complementary to the antisense strand of DNA, or template strand, which runs from 3' to 5"' (https://en.wikipedia.org/wiki/Sense_strand).

https://www.ncbi.nlm.nih.gov/nuccore/CU329670

2.1

2.2

2.3

Assignment 3

3.1

3.2

Appendix

```
knitr::opts_chunk$set(fig.width = 7, fig.height = 3, echo = FALSE)

library(dplyr)
library(tidyr)
library(magrittr)
library(ape)      # This is a general R-package for phylogenetics and comparative methods
library(seqinr)   # This is an specialized package for nucleotide sequence management
library(kableExtra)

source("732A51_BioinformaticsHT2018_Lab02_GenBankGetCode.R")

lizards_format_sequences <- read.fasta(file = "lizard_seqs.fasta")  # Alternative version of the file
# Useful in some ways?

n <- length(lizards_accession_numbers)  # Number of sequences to reproduce
p <- base.freq(lizards_sequences)  # Probability of the base sequences
simulated_lizards <- list()  # Object that will contain our simulated data

# The names of the simulated data are the original names + "_sim"
# NOTE: it does not follow the format from GenBank
simulated_names <- paste(lizards_accession_numbers, "_sim", sep = "")

set.seed(1535)  # Set seed in order to reproduce the experiment
for(i in 1:n) {  # Cycle through every single object of the lizard_sequences
  len_seq <- length(lizards_sequences[[i]])  # Lenght of each sequence
  simulated_lizards[[ simulated_names[i] ]] <-
    sample(c("a", "c", "g", "t"), len_seq, replace = T, prob = p)
  # Creating the artificial sequence sampling with probabilities p equal to the original ones
}

write.dna(simulated_lizards, file = "simulated_lizards.fasta", format = "fasta", append = F,
          nbcol = 6, colsep = " ", colw = 10)

# Function to print DNA sequence
```

```r
# sapply(test, paste, collapse="")

df_table <- data.frame("Base" = c("a", "c", "g", "t"),
                       "Original\nfrequency" = p,
                       "Simulated\nfrequency" = base.freq(as.DNAbin(simulated_lizards)),
                       row.names = NULL)
kable(df_table, booktabs = T, align = c("r", "l", "l"), digits = c(NA, 4, 4),
      col.names = c("Base", "Original\nfrequency", "Simulated\nfrequency"), format = "latex")
# ------------------------------------------------------------------------------
# Question 1.1
# ------------------------------------------------------------------------------

# ------------------------------------------------------------------------------
# Question 1.2
# ------------------------------------------------------------------------------

# ------------------------------------------------------------------------------
# Question 2.1
# ------------------------------------------------------------------------------

# ------------------------------------------------------------------------------
# Question 2.2
# ------------------------------------------------------------------------------

# ------------------------------------------------------------------------------
# Question 2.2
# ------------------------------------------------------------------------------

# ------------------------------------------------------------------------------
# Question 3.1
# ------------------------------------------------------------------------------

# ------------------------------------------------------------------------------
# Question 3.2
# ------------------------------------------------------------------------------
```