Homework Assignment

Pair-wise Sequence Alignment

03-04-2023

Task A

```
# Accession numbers
ac_1 <- "AY884001"
ac_2 <- "MH940245"

# Retrieve the data
q1 <- query("q1", paste("AC=", ac_1))
q2 <- query("q2", paste("AC=", ac_2))

# Get the sequences
seq1 <- getSequence(q1$req[[1]])
seq2 <- getSequence(q2$req[[1]])

# Print the sequences
print(DNAString(c2s(seq1)))
```

```
## 29815-letter DNAString object
## seq: GAGCGATTGACGTTCGTACCGTCTATCAGCTTACGA...TGATTGAAATTAATTATAGCCTTTTGGAGGAATTAC
```

```
print(DNAString(c2s(seq2)))
```

```
## 29811-letter DNAString object
## seq: GATTGACGTTCGTACCGTCTATCAGCTTACGATCTC...TGATTGAAATTAATTATAGCCTTTTGGAGGAATTAC
```

Task B

```
seq1_count <- table(seq1)
seq2_count <- table(seq2)
print(seq1_count)
```

```
## seq1
##     a     c     g     t
##  8261  3847  5701 12006
```

```
print(seq2_count)
```

```
## seq2
##     a     c     g     t
##  8260  3845  5699 12007
```

```
seq1_prop <- proportions(seq1_count)
seq2_prop <- proportions(seq2_count)
print(seq1_prop)
```

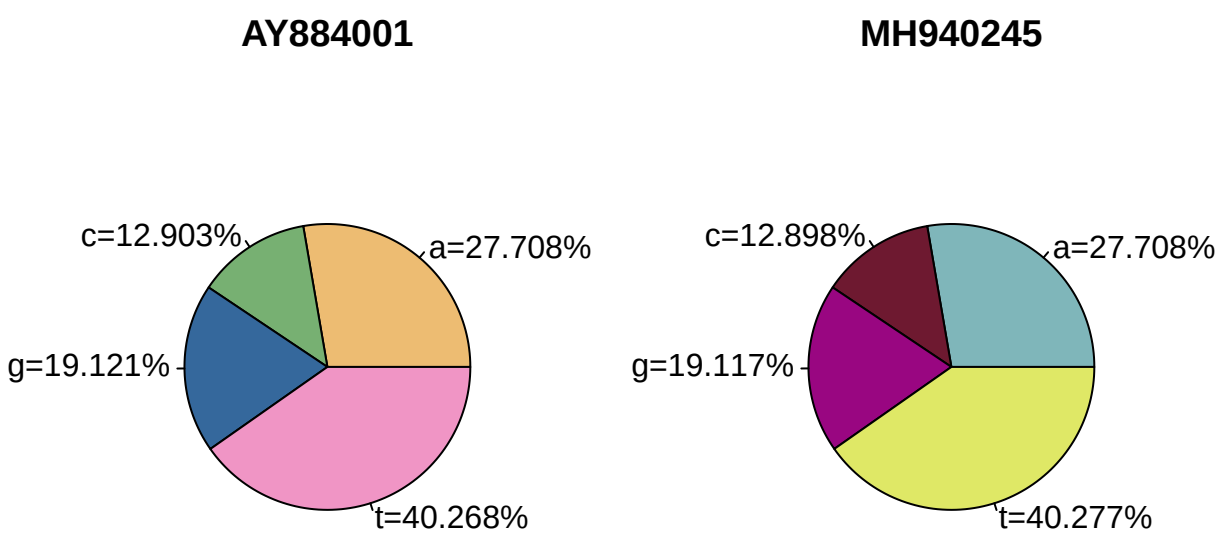
```
## seq1
##         a         c         g         t
## 0.2770753 0.1290290 0.1912125 0.4026832
```

```
print(seq2_prop)
```

```
## seq2
##         a         c         g         t
## 0.2770789 0.1289792 0.1911710 0.4027708
```

```
colors1 <- c("#edbc72", "#77b072", "#35689c", "#f095c5")
colors2 <- c("#7fb6ba", "#6e1930", "#990681", "#dfe866")

par(mfrow = c(1, 2))
pie(seq1_prop, labels = paste(names(seq1_prop), "=", round(seq1_prop *
    100, 3), "%", sep = ""), col = colors1, main = ac_1)
pie(seq2_prop, labels = paste(names(seq2_prop), "=", round(seq2_prop *
    100, 3), "%", sep = ""), col = colors2, main = ac_2)
```

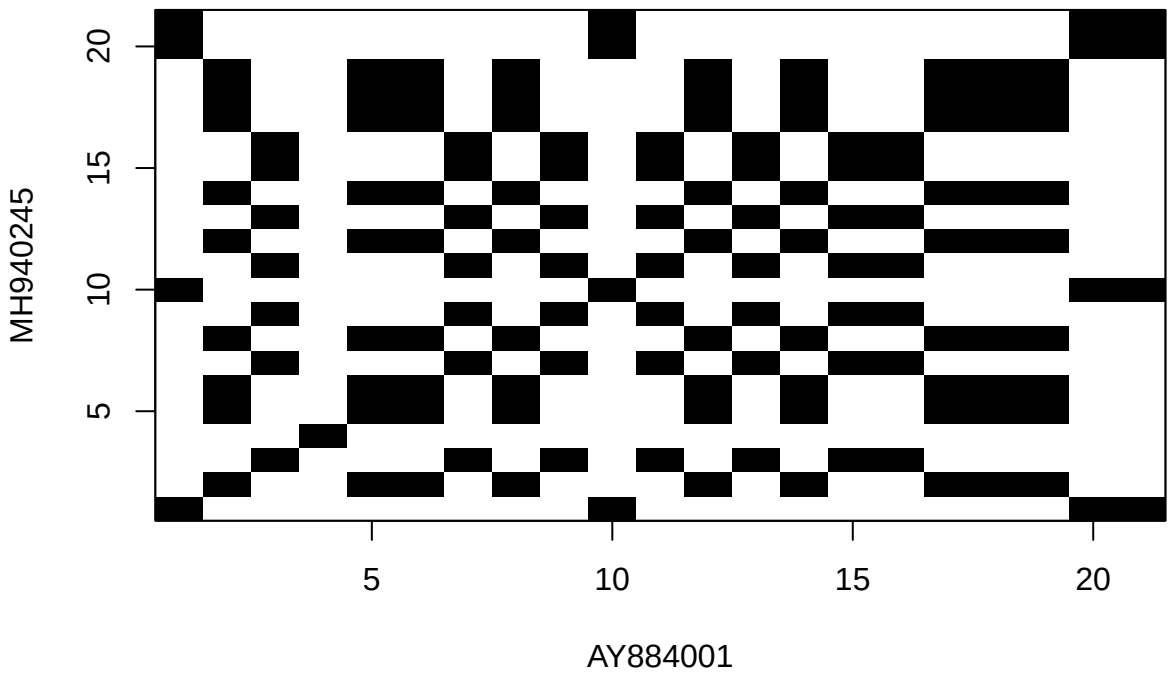


Task C

```
seq1_orfs <- findORFs(c2s(seq1))
seq2_orfs <- findORFs(c2s(seq2))

par(mfrow = c(1, 1))
dotPlot(s2c(seq1_orfs[1, "orf.sequence"]), s2c(seq2_orfs[1, "orf.sequence"]),
```

```
    xlab = ac_1, ylab = ac_2)
```



The symmetric nature of the plot reveals that the two sequences are almost identical. In fact, by doing a direct comparison we can see that they are exactly the same!

```
print(seq1_orfs[1, "orf.sequence"] == seq2_orfs[1, "orf.sequence"])
```

```
## orf.sequence
##         TRUE
```

Task D

```
opt_ga <- pairwiseAlignment(DNAString(c2s(seq1)), DNAString(c2s(seq2)),
    type = "global", substitutionMatrix = nucleotideSubstitutionMatrix(match = 2,
        mismatch = -1, baseOnly = TRUE), gapOpening = 0, gapExtension = -2)
print(opt_ga)
```

```
## Global PairwiseAlignmentsSingleSubject (1 of 1)
## pattern: GAGCGATTGACGTTCGTACCGTCTATCAGCTTAC...ATTGAAATTAATTATAGCCTTTTGGAGGAATTAC
## subject: GA----TTGACGTTCGTACCGTCTATCAGCTTAC...ATTGAAATTAATTATAGCCTTTTGGAGGAATTAC
## score: 59611
```

```
cat("Alignment for first 20 nucleotides:", paste(substr(pattern(opt_ga),
    1, 20), substr(subject(opt_ga), 1, 20), sep = "\n"), sep = "\n")
```

```
## Alignment for first 20 nucleotides:
## GAGCGATTGACGTTCGTACC
```

```
## GA----TTGACGTTCGTACC
```

Task E

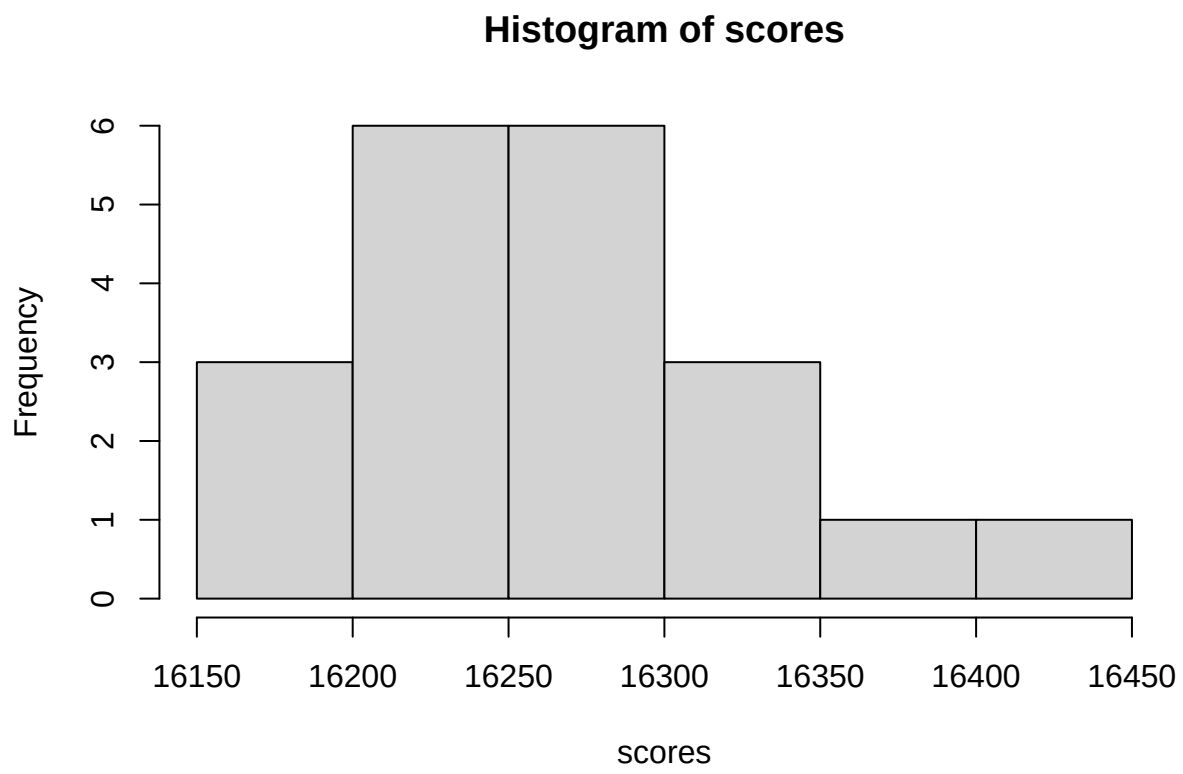
```
seq1_random <- generateSeqsWithMultinomialModel(c2s(seq1), 20)
seq2_random <- generateSeqsWithMultinomialModel(c2s(seq2), 20)

scores <- c()
for (i in 1:20) {
    score <- pairwiseAlignment(DNAString(c2s(seq1_random[[i]])),
        DNAString(c2s(seq2_random[[i]])), type = "global", substitutionMatrix = nucleotideSubstitutionMa
            mismatch = -1, baseOnly = TRUE), gapOpening = 0,
        gapExtension = -2, scoreOnly = TRUE)
    scores <- append(scores, score)
}

print(scores)
```

```
##  [1] 16255 16222 16297 16162 16285 16219 16411 16228 16171 16213 16150 16294
## [13] 16345 16309 16246 16309 16216 16273 16354 16255
```

```
hist(scores)
```




```
p_value <- sum(scores > score(opt_ga))/length(scores)
print(p_value)
```

```
## [1] 0
```

The global alignment is statistically significant as the p-value is below 0.05. This means that the odds of this alignment happening by chance are extremely low.

Task F

```
opt_la <- pairwiseAlignment(DNAString(c2s(seq1)), DNAString(c2s(seq2)),
    type = "local", substitutionMatrix = nucleotideSubstitutionMatrix(match = 3,
        mismatch = -2, baseOnly = TRUE), gapOpening = -4, gapExtension = -2)
print(opt_la)
```

```
## Local PairwiseAlignmentsSingleSubject (1 of 1)
## pattern: [5] GATTGACGTTCGTACCGTCTATCAGCTTACGA...TGAAATTAATTATAGCCTTTTGGAGGAATTAC
## subject: [1] GATTGACGTTCGTACCGTCTATCAGCTTACGA...TGAAATTAATTATAGCCTTTTGGAGGAATTAC
## score: 89428
```

```
print("Length of the alignment:")
```

```
## [1] "Length of the alignment:"
```

```
print(nchar(pattern(opt_la)))
```

```
## [1] 29811
```