https://github.com/klynch416 (https://github.com/klynch416)

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(BiocManager)
install(c("sangerseqR", "annotate"))
```

```
## Bioconductor version 3.16 (BiocManager 1.30.19), R 4.2.2 (2022-10-31 ucrt)
```

```
## Warning: package(s) not installed when version(s) same as or greater than current; use
##   `force = TRUE` to re-install: 'sangerseqR' 'annotate'
```

```
## Installation paths not writeable, unable to update packages
##   path: C:/Program Files/R/R-4.2.2/library
##   packages:
##     boot, class, codetools, foreign, MASS, Matrix, nlme, spatial, survival
```

```
## Old packages: 'httpuv', 'utf8', 'xfun'
```

```
library(sangerseqR)
```

```
## Loading required package: Biostrings
```

```
## Loading required package: BiocGenerics
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min
```

```
## Loading required package: S4Vectors
```

```
## Loading required package: stats4
```

```
##
## Attaching package: 'S4Vectors'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, rename
```

```
## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname
```

```
## Loading required package: IRanges
```

```
##
## Attaching package: 'IRanges'
```

```
## The following objects are masked from 'package:dplyr':
##
##     collapse, desc, slice
```

```
## The following object is masked from 'package:grDevices':
##
##     windows
```

```
## Loading required package: XVector
```

```
## Loading required package: GenomeInfoDb
```

```
##
## Attaching package: 'Biostrings'
```

```
## The following object is masked from 'package:base':
##
##     strsplit
```

```
library(annotate)
```

```
## Loading required package: AnnotationDbi
```

```
## Loading required package: Biobase
```

```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
##
## Attaching package: 'AnnotationDbi'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## Loading required package: XML
```

```
library(ggplot2)
```

Import CSV

```
Sequences <- read.csv("./Sequences.csv")
```

Print out each sequence. *Sequences*

```
Sequences$Sequence[1]
```

```
## [1] "AGCATGCAAGTCAAACGAGATGTAGCAATACATCTAGTGGCGAACGGGTGAGTAACGCGTGGATGATCTACCTATGAGATGGGGATAA
CTATTAGAAATAGTAGCTAATACCGAATAAGGTCAATTAATTTGTTAATTGATGAAAGGAAGCCTTTAAAGCTTCGCTTGTAGATGAGTCTGCGTC
TTATTAGTTAGTTGGTAGGGTAAATGCCTACCAAGGCGATGATAAGTAACCGGCCTGAGAGGGTGAACGGTCACACTGGAACTGAGACACGGTCCA
GACTCCTACGGGAGGCAGCAGCTAAGAATCTTCCGCAATGGGCGAAAGCCTGACGGAGCGACACTGCGTGAATGAAGAAGGTCGAAAGATTGTAAA
ATTCTTTTATAAATGAGGAATAAGCTTTGTAGGAAATGACGAAGTGATGACGTTAATTTATGAATAAGCCCCGGCTAATTACGTGCCAGCAGCCGC
GGTAATACG"
```

```
Sequences$Sequence[2]
```

```
## [1] "AGCATGCAAGTCAAACGGGATGTAGCAATACATTCAGTGGCGAACGGGTGAGTAACGCGTGGATGATCTACCTATGAGATGGGGATAA
CTATTAGAAATAGTAGCTAATACCGAATAAGGTCAGTTAATTTGTTAATTGATGAAAGGAAGCCTTTAAAGCTTCGCTTGTAGATGAGTCTGCGTC
TTATTAGCTAGTTGGTAGGGTAAATGCCTACCAAGGCAATGATAAGTAACCGGCCTGAGAGGGTGAACGGTCACACTGGAACTGAGATACGGTCCA
GACTCCTACGGGAGGCAGCAGCTAAGAATCTTCCGCAATGGGCGAAAGCCTGACGGAGCGACACTGCGTGAATGAAGAAGGTCGAAAGATTGTAAA
ATTCTTTTATAAATGAGGAATAAGCTTTGTAGGAAATGACAAAGTGATGACGTTAATTTATGAATAAGCCCCGGCTAATTACGTGCCAGCAGCAGC
GGTAATACG"
```

```
Sequences$Sequence[3]
```

```
## [1] "AGCATGCAAGTCAAACGAGATGTAGTAATACATCTAGTGGCGAACGGGTGAGTAACGCGTGGATGATCTACCTATGAGATGGGGATAA
CTATTAGAAATAGTAGCTAATACCGAATAAGGTCAATTAATTTGTTAATTGATGAAAGGAAGCCTTTAAAGCTTCGCTTGTAGATGAGTCTGCGTC
TTATTAGTTAGTTGGTAGGGTAAATGCCTACCAAGGCGATGATAAGTAACCGGCCTGAGAGGGTGAACGGTCACACTGGAACTGAGACACGGTCCA
GACTCCTACGGGAGGCAGCAGCTAAGAATCTTCCGCAATGGGCGAAAGCCTGACGGAGCGACACTGCGTGAATGAAGAAGGTCGAAAGATTGTAAA
ATTCTTTTATAAATGAGGAATAAGCTTTGTAGGAAATGACGAAGTGATGACGTTAATTTATGAATAAGCCCCGGCTAATTACGTGCCAGCAGCCGC
GGTAATACG"
```

Number of each nucleotide per sequence

```
Nucleotide <- matrix(nrow = 3, ncol = 5)

for(i in 1:3){
Nucleotide[i,1] <- i
Nucleotide[i,2] <- lengths(regmatches(Sequences$Sequence[i], gregexpr("A", Sequences$Sequence
[i])))
Nucleotide[i,3] <- lengths(regmatches(Sequences$Sequence[i], gregexpr("T", Sequences$Sequence
[i])))
Nucleotide[i,4] <- lengths(regmatches(Sequences$Sequence[i], gregexpr("G", Sequences$Sequence
[i])))
Nucleotide[i,5] <- lengths(regmatches(Sequences$Sequence[i], gregexpr("C", Sequences$Sequence
[i])))
}

colnames(Nucleotide) <- c("Sequence", "A", "T", "G", "C")
Nucleotide <- as.data.frame(Nucleotide)

print(Nucleotide)
```

```
##   Sequence   A   T   G  C
## 1        1 154 114 131 82
## 2        2 155 114 131 81
## 3        3 154 115 131 81
```

GC content

```
Nucleotide <- Nucleotide %>% mutate(GC_Content = round(((G+C)/(A+T+G+C))*100))

Sequences$Name <- c("HQ433692.1","HQ433694.1","HQ433691.1")

GC_Content <- data.frame("Sequence ID" = Sequences$Name, "GC Content" = paste0(Nucleotide$GC_Con
tent,"%"))

print(GC_Content)
```

```
##   Sequence.ID GC.Content
## 1  HQ433692.1        44%
## 2  HQ433694.1        44%
## 3  HQ433691.1        44%
```

Image and wikipedia page for Borrelia burgdorferi

https://en.wikipedia.org/wiki/Borrelia_burgdorferi (https://en.wikipedia.org/wiki/Borrelia_burgdorferi)

Borrelia burgdorferi

Part 2

Write reproducible R code to search for the closest matching sequence on Genbank and generate an alignment to confirm the degree of similarity.

Blast and alignment score

```
Unkseq <- "GCCTGATGGAGGGGGATAACTACTGGAAACGGTAGCTAATACCGCATGACCTCGCAAGAGCAAAGTGGGGGACCTTAGGGCCTCA
CGCCATCGGATGAACCCAGATGGGATTAGCTAGTAGGTGGGGTAATGGCTCACCTAGGCGACGATCCCTAGCTGGTCTGAGAGGATGACCAGCCAC
ACTGGAACTGAGACACGGTCCAGACTCCTACGGGAGGCAGCAGTGGGGAATATTGCACAATGGGCGCAA"

dataseq <- blastSequences(x = Unkseq, timeout = 200, hitListSize = 10, as = 'data.frame')
```

```
## estimated response time 53 seconds
```

```
## elapsed time 54 seconds
```

```
## elapsed time 64 seconds
```

```
## elapsed time 75 seconds
```

```
## elapsed time 86 seconds
```

```
## elapsed time 97 seconds
```

```
## elapsed time 107 seconds
```

```
## elapsed time 118 seconds
```

```
## elapsed time 129 seconds
```

```
## elapsed time 139 seconds
```

```
## elapsed time 150 seconds
```

```
## elapsed time 161 seconds
```

```
## elapsed time 172 seconds
```

```
## elapsed time 182 seconds
```

```
## elapsed time 193 seconds
```

```
## elapsed time 204 seconds
```

```
## elapsed time 214 seconds
```

```
## elapsed time 225 seconds
```

```
## elapsed time 236 seconds
```

```
## elapsed time 247 seconds
```

```
## elapsed time 257 seconds
```

```
## elapsed time 269 seconds
```

```
## elapsed time 279 seconds
```

```
## elapsed time 290 seconds
```

```
## elapsed time 301 seconds
```

```
## elapsed time 311 seconds
```

```
## elapsed time 322 seconds
```

```
## elapsed time 333 seconds
```

```
## elapsed time 344 seconds
```

```
## elapsed time 354 seconds
```

```
## elapsed time 365 seconds
```

```
## elapsed time 376 seconds
```

```
## elapsed time 386 seconds
```

```
## elapsed time 397 seconds
```

```
## elapsed time 408 seconds
```

```
## elapsed time 418 seconds
```

```
## elapsed time 431 seconds
```

```
## elapsed time 442 seconds
```

```
## elapsed time 452 seconds
```

```
## elapsed time 463 seconds
```

```
## elapsed time 474 seconds
```

```
## elapsed time 485 seconds
```

```
## elapsed time 495 seconds
```

```
uniqdata <- dataseq %>% distinct(Hit_id, .keep_all=TRUE)
print(paste0(uniqdata$Hit_def,": ", uniqdata$Hsp_score))
```

```
##  [1] "Yersinia pestis EV76-CN chromosome, complete genome: 500"
##  [2] "Yersinia pestis strain 20 chromosome, complete genome: 500"
##  [3] "Yersinia pestis strain 94 chromosome, complete genome: 500"
##  [4] "Yersinia pestis strain R chromosome, complete genome: 500"
##  [5] "Yersinia pseudotuberculosis strain 598 chromosome: 500"
##  [6] "Yersinia pestis strain 14D chromosome, complete genome: 500"
##  [7] "Yersinia pestis strain M2085 chromosome, complete genome: 500"
##  [8] "Yersinia pestis strain C-792 chromosome, complete genome: 500"
##  [9] "Yersinia pestis strain M-1770 chromosome, complete genome: 500"
## [10] "Yersinia pestis EV NIIEG chromosome, complete genome: 500"
```

It is another organism, specifically the bacterium *Yersinia pestis*. *Yersinia pestis*, also known as the Black Death, is a gram-negative, non-motile, coccobacillus bacterium without spores and spread through humans via the Oriental rat flea. Patients develop fever, chills, extreme weakness, abdominal pain, shock, and possibly bleeding into the skin and other organs. Skin and other tissues may turn black and die, especially on fingers, toes, and the nose. *Yersinia pestis* can be treated with intravenous or oral antimicrobials for 10 to 14 days.