

# Computer Assignment 0

781552

9/17/2021

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0    Min.   : 2.00
##  1st Qu.:12.0    1st Qu.: 26.00
##  Median :15.0    Median : 36.00
##  Mean   :15.4    Mean   : 42.98
##  3rd Qu.:19.0    3rd Qu.: 56.00
##  Max.   :25.0    Max.   :120.00
```

## Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

## Question 1a

### Load the packages

```
library("ape", quietly = T)
```

```
## Warning: package 'ape' was built under R version 4.0.5
```

```
library("Biostrings", quietly = T)
```

```
## Warning: package 'BiocGenerics' was built under R version 4.0.5
```

```
##
```

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

```
## The following objects are masked from 'package:parallel':
```

```
##
```

```
##   clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
```

```
##   clusterExport, clusterMap, parApply, parCapply, parLapply,
```

```
##   parLapplyLB, parRapply, parSapply, parSapplyLB
```

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

## The following objects are masked from 'package:base':
##
##   anyDuplicated, 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

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:base':
##
##   expand.grid

##
## Attaching package: 'IRanges'

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

##
## Attaching package: 'Biostrings'

## The following object is masked from 'package:ape':
##
##   complement

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

## Read DNA string

```
seq <- read.GenBank("MF770243.1", as.character = TRUE)[[1]]
seqStr <- paste(seq, sep="", collapse="")
DNAstr <- DNASTring(seqStr)
```

## Subtask 1

```
length(DNAstr)
```

```
## [1] 16851
```

## Subtask 2

```
freq = alphabetFrequency(DNAstr)[DNA_BASES]  
freq['T']
```

```
##      T  
## 4794
```

## Subtask 3

```
freq = alphabetFrequency(DNAstr)[DNA_BASES]  
freq['T']
```

```
##      T  
## 4794
```

## Question 2a

### Subtask 1

```
x <- c(2,3,1)  
y <- c(3,5,2)  
x <- x + 3  
y <- y^2  
Z = matrix(c(x,y), nrow = 2, ncol = 3, byrow=TRUE)  
A = x * t(y)  
c = dim(A)  
B <- c(runif(3))  
dim(B) = c(1,3)  
A * B
```

```
##           [,1]      [,2]      [,3]  
## [1,] 2.692315 26.4328 14.97262
```

```
A%*%t(B)
```

```
##           [,1]  
## [1,] 44.09774
```

## Subtask 2

```
x <- c(1,2,3,4,5)
x[3]
```

```
## [1] 3
```

```
x[1:3]
```

```
## [1] 1 2 3
```

```
x[1:length(x)]
```

```
## [1] 1 2 3 4 5
```

```
x[1:length(x)-1]
```

```
## [1] 1 2 3 4
```

```
x[seq(5,1,-2)]
```

```
## [1] 5 3 1
```

```
x[c(1,3,5)]
```

```
## [1] 1 3 5
```

## Subtask 3

```
A=matrix(c(1,2,3,4,5,6),nrow=2)
A[3]
```

```
## [1] 3
```

```
A[1,]
```

```
## [1] 1 3 5
```

```
A[,2:3]
```

```
##      [,1] [,2]
## [1,]    3    5
## [2,]    4    6
```

```
A[A>3]
```

```
## [1] 4 5 6
```

## Subtask 4

```
# This is a list.
a = list()
# An empty list can be appended as
a[[1]] = 0.0
a[[2]] = c(TRUE,FALSE)
a[['vector']] = c(1,2,3)
print(a)
```

```
## [[1]]
## [1] 0
##
## [[2]]
## [1] TRUE FALSE
##
## $vector
## [1] 1 2 3
```

```
print(a[[1]]) # double brackets select the content of the index in a list
```

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

```
# Three ways to access named entries in a list:
a[[3]]
```

```
## [1] 1 2 3
```

```
a[['vector']]
```

```
## [1] 1 2 3
```

```
a$vector
```

```
## [1] 1 2 3
```

```
# Get names of entries in a list
names(a)
```

```
## [1] "" "" "vector"
```

```
# Print structure of any variable (can be a list, vector,...)
str(a)
```

```
## List of 3
## $      : num 0
## $      : logi [1:2] TRUE FALSE
## $ vector: num [1:3] 1 2 3
```

```
#Question 2b
```

```
x = seq(-10,10)
f = sin(x)
g = cos(x)
plot(x, f, xlim = c(-5, 5), col = "blue");
lines(x,g, col="yellow") ; legend(x = "topright", legend = c("sin(x)", "cos(x)"), col = c(col = "red",
```

