Bios 6301: Assignment 2

Haoyang Yi

1. Working with data In the `datasets` folder on the course GitHub repo, you will find a file called `cancer.csv`, which is a dataset in comma-separated values (csv) format. This is a large cancer incidence dataset that summarizes the incidence of different cancers for various subgroups.

1. Load the data set into R and make it a data frame called `cancer.df`. (2 points)

```
cancer.df = read.csv('cancer.csv')
```

2. Determine the number of rows and columns in the data frame. (2)

```
nrow(cancer.df)
```

```
## [1] 42120
```

```
ncol(cancer.df)
```

```
## [1] 8
```

3. Extract the names of the columns in `cancer.df`. (2)

```
colnames(cancer.df)
```

```
## [1] "year"       "site"       "state"      "sex"        "race"
## [6] "mortality"  "incidence"  "population"
```

4. Report the value of the 3000th row in column 6. (2)

```
cancer.df[3000,6]
```

```
## [1] 350.69
```

5. Report the contents of the 172nd row. (2)

```
cancer.df[172,]
```

```
##     year                          site  state  sex  race mortality incidence
## 172 1999 Brain and Other Nervous System nevada Male Black         0         0
##     population
## 172      73172
```

6. Create a new column that is the incidence *rate* (per 100,000) for each row. The incidence rate is t

```
cancer.df$rate = cancer.df$incidence/cancer.df$population*100000
```

7. How many subgroups (rows) have a zero incidence rate? (2)

```
nrow(cancer.df[cancer.df$rate==0,])
```

```
## [1] 23191
```

8. Find the subgroup with the highest incidence rate.(3)

```r
cancer.df[which.max(cancer.df$rate),]
```

```
##      year     site                state  sex  race mortality incidence
## 5797 1999 Prostate district of columbia Male Black     88.93       420
##      population     rate
## 5797     160821 261.1599
```

2. Data types

1. Create the following vector: `x <- c("5","12","7")`. Which of the following commands will produce an

```
max(x)
sort(x)
sum(x)
```

```r
x <- c("5","12","7")
max(x)
```

```
## [1] "7"
```

```r
sort(x)
```

```
## [1] "12" "5"  "7"
```

```r
#sum(x) # sum() creates error since the input of this function should be numeric for calculation.
```

2. For the next two commands, either explain their results, or why they should produce errors. (3 points

```
y <- c("5",7,12)
y[2] + y[3]
```

```r
y <- c("5",7,12)
#y[2] + y[3]
# It produce error since y is created as character vector and cannot perform numeric calculation.
```

3. For the next two commands, either explain their results, or why they should produce errors. (3 points

```
z <- data.frame(z1="5",z2=7,z3=12)
z[1,2] + z[1,3]
```

```r
z <- data.frame(z1="5",z2=7,z3=12)
z[1,2] + z[1,3] # it provides the product of numeric value z2(7)+ z3(12)=19
```

```
## [1] 19
```

3. Data structures Give R expressions that return the following matrices and vectors (*i.e.* do not construct them manually).

1. $(1,2,3,4,5,6,7,8,7,6,5,4,3,2,1)$

```r
c(1:8,7:1)
```

```
##  [1] 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1
```

2. $(1,2,2,3,3,3,4,4,4,4,5,5,5,5,5)$

```r
c(1,rep(2,2),rep(3,3),rep(4,4),rep(5,5))
```

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

3. $\begin{pmatrix}
  0 & 1 & 1 \\
  1 & 0 & 1 \\
  1 & 1 & 0 \\
\end{pmatrix}$

```
1-diag(3)
```

```
##      [,1] [,2] [,3]
## [1,]    0    1    1
## [2,]    1    0    1
## [3,]    1    1    0
```

4. $\begin{pmatrix}
  1 & 2 & 3 & 4 \\
  1 & 4 & 9 & 16 \\
  1 & 8 & 27 & 64 \\
  1 & 16 & 81 & 256 \\
  1 & 32 & 243 & 1024 \\
\end{pmatrix}$

```
x = c(1,2,3,4)
matrix(c(x,x^2,x^3,x^4,x^5),ncol = 4,byrow = T)
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    2    3    4
## [2,]    1    4    9   16
## [3,]    1    8   27   64
## [4,]    1   16   81  256
## [5,]    1   32  243 1024
```

4. Basic programming

1. Let $h(x,n)=1+x+x^2+\ldots+x^n = \sum_{i=0}^n x^i$. Write an R program to calculate $h(x,n)$ using a

```
res = 0
for (i in 0:2){
  xi = 5^i
  res = res+xi
}
res
```

```
## [1] 31
```

1. If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The s

    1. Find the sum of all the multiples of 3 or 5 below 1,000. (3, [euler1])

```
x = 1:999
a = which(x%%3==0|x%%5==0)
sum(a)
```

```
## [1] 233168
```

    1. Find the sum of all the multiples of 4 or 7 below 1,000,000. (2)

```
x = 1:999999
a = which(x%%4==0|x%%7==0)
sum(a)
```

```
## [1] 178571071431
```

1. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting w

```
res= NULL
for (i in 1:15){
  res = append(res,2*i-1)
}
sum(res)
```

```
## [1] 225
```

Some problems taken or inspired by projecteuler.