

# Bios 301 - Assignment 1

Danielle Nemetz

October 7, 2013

## 1 Working with data

### 1.1 Problem 1

Load data set and name it cancer.df setting working directory to access data set

```
setwd("/Users/Danie/Bios301/datasets")  
cancer.df <- read.csv("cancer.csv")
```

### 1.2 Problem 2

Determine number of rows in data frame

```
nrow(cancer.df)  
## [1] 42120
```

Determine number of columns in data frame

```
ncol(cancer.df)  
## [1] 8
```

### 1.3 Problem 3

Extract the names of the columns in 'cancer.df'

```
names(cancer.df)  
## [1] "year"      "site"      "state"     "sex"       "race"  
## [6] "mortality" "incidence" "population"
```

Report value of the 3000th row in column 6

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

```
## [1] 350.7
```

Report the contents of the 172nd row

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

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

Create a new column that is the incidence \*rate\* (per 100,000) for each row

```
cancer.df["incidence rate"] <- cancer.df$incidence/cancer.df$population * 1e+05
```

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

```
nrow(subset(cancer.df, cancer.df$"incidence rate" == "0"))
```

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

Find the subgroup with the highest incidence rate

```
which.max(cancer.df$"incidence rate")
```

```
## [1] 5797
```

## 2 Data types

Create the following vector: 'x' ← c("5", "12", "7")'. Which of the following commands will produce an error message? For each command, Either explain why they should be errors, or explain the non-erroneous result.

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

```
max(x)
```

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

```
sort(x)
```

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

```
sum(x)
```

```
## Error: invalid 'type' (character) of argument
```

The code `max()` and `sort()` operate by looking at the numerical representations of the first character in the string. 5 is observed as 5 in decimal, 12 is observed as 1 in decimal and 7 is observed as 7 in decimal. Therefore, the `max()` function returns 7 and `sort()` returns the order: 12, 5, 7.

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

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

## Error: non-numeric argument to binary operator
```

Gives error because a vector can only have one mode. Because the first input is a character, the following two inputs are converted to characters as well. This is confirmed by the `class` function. Similar to the `sum(x)` function, the second command cannot be performed because two characters cannot be added by the binary operator.

```
class(y)

## [1] "character"
```

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

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

## [1] 19

class(z)

## [1] "data.frame"

# The first calculation adds the first row, second value (7) to the first
# row, third value (12). Because both values are numeric, they can be added.
# 3. 1.
c(1:8, 7:1)

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

# 2.
c(rep(1, 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.
m1 <- matrix(0, 3, 3)
m1[lower.tri(m1)] <- 1
m1[upper.tri(m1)] <- 1
m1[2, 1] = 0
m1

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

# 4.
m2 <- matrix(1:9, 3, 3, byrow = TRUE)
m2[1, 1] = 0
m2[2, 1] = 0
m2[2, 3] = 0
m2[3, 2] = 0
m2[3, 3] = 0
m2

##      [,1] [,2] [,3]
## [1,]    0    2    3
## [2,]    0    5    0
## [3,]    7    0    0

# 4. **Basic programming** Write an R program to calculate  $L_h(x,n)$  using a
# `for` loop.
h(x, n) = function(x, n) {
  for (i in 1:n) {
    x = x^n + x
  }
}

## Error: could not find function "h<-"

```