1 Problem 1

We will introduce R in this lab. You should have installed R and Rstudio already.

Problem 1a: Manually Enter Data Into R.

The following data set represents the ages (in years) of a small class of students:

$$\{18, 19, 23, 19, 24, 20, 18, 21, 22, 23, 18\}$$

Enter the following command into the console:

```
age <- c(18, 19, 23,19, 24, 20, 18, 21, 22, 23,18)
```

where `<-` is an assignment operator and `c()` is the notation used in R to represent a vector of datapoints.

```
> age <- c(18, 19, 23,19, 24, 20, 18, 21, 22, 23,18)
```

Problem 1b: R has functions that can quickly calculate summary statistics for us. Explore the following functions:

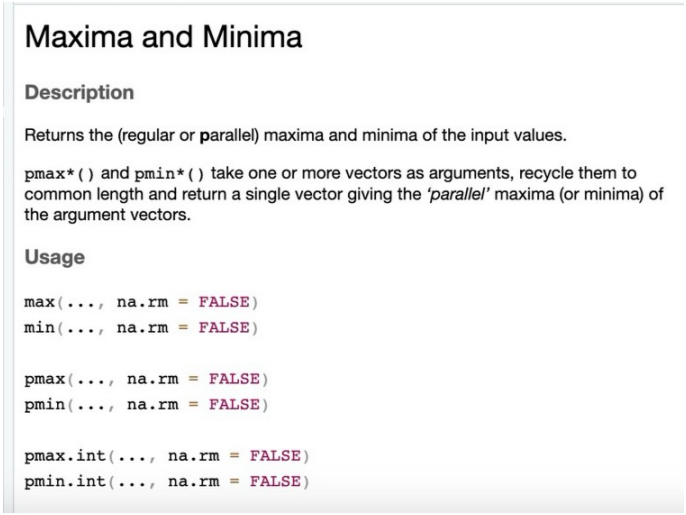
- `mean()`
- `median()`
- `min()`
- `max()`
- `range()`
- `quantile()`
- `sd()`
- `var()`
- `summary()`
- `length()`
- `sum()`
- `IQR()`
- `table()`

```
> median(age)
[1] 20
> min(age)
[1] 18
> max(age)
[1] 24
> range(age)
[1] 18 24
> quantile(age)
  0%  25%  50%  75% 100%
18.0 18.5 20.0 22.5 24.0
> sd(age)
[1] 2.252272
> var(age)
[1] 5.072727
> summary(age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  18.00   18.50   20.00   20.45   22.50   24.00
> length(age)
[1] 11
> sum(age)
[1] 225
> IQR(age)
[1] 4
> table(age)
age
18 19 20 21 22 23 24
 3  2  1  1  1  2  1
```

Problem 1c: To view documentation for a function, we place a question mark in front of the function. For example, `?mean`

Choose one of the functions above that's not `sd()` and take a screenshot of its documentation. Explain how it works.

The max() function gives the maximum number in the age list.



Maxima and Minima

Description

Returns the (regular or **p**arallel) maxima and minima of the input values.

`pmax*()` and `pmin*()` take one or more vectors as arguments, recycle them to common length and return a single vector giving the *'parallel'* maxima (or minima) of the argument vectors.

Usage

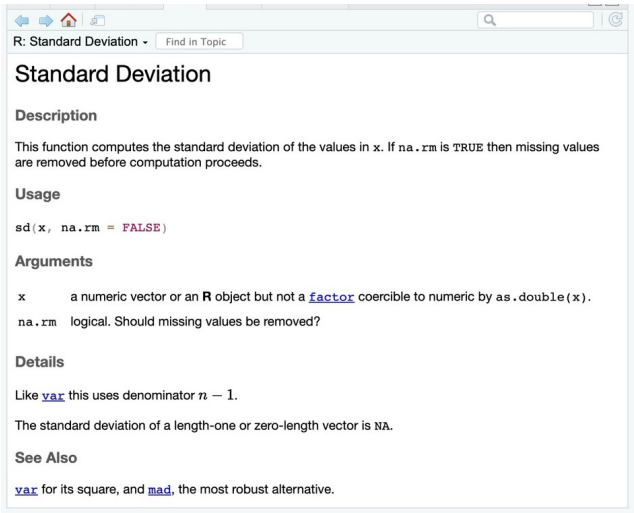
```
max(..., na.rm = FALSE)
min(..., na.rm = FALSE)

pmax(..., na.rm = FALSE)
pmin(..., na.rm = FALSE)

pmax.int(..., na.rm = FALSE)
pmin.int(..., na.rm = FALSE)
```

Problem 1d: View the documentation for the `sd()` function. Does it divide by $n-1$ or n? As a result, does it calculate population standard deviation, or sample standard deviation?

It divides by n-1. As a result, it calculates sample standard deviation.



R: Standard Deviation Find in Topic

Standard Deviation

Description

This function computes the standard deviation of the values in `x`. If `na.rm` is `TRUE` then missing values are removed before computation proceeds.

Usage

```
sd(x, na.rm = FALSE)
```

Arguments

`x`	a numeric vector or an **R** object but not a `factor` coercible to numeric by `as.double(x)`.
`na.rm`	logical. Should missing values be removed?

Details

Like `var` this uses denominator $n-1$.

The standard deviation of a length-one or zero-length vector is `NA`.

See Also

`var` for its square, and `mad`, the most robust alternative.

2 Problem 2

What if you have more than one variable recorded for each individual? For example, suppose the results of the first exam are recorded as follows:

$$\{75, 77, 94, 75, 79, 80, 66, 82, 86, 80, 78\}$$

Problem 2a: Enter this information into R by typing the following command into the R console (which stores the data into a variable named 'score'):

```
score <- c(75, 77, 94, 75, 79, 80, 66, 82, 86, 80, 78)
```

Problem 2b: To identify the data with each student, a student ID is assigned to each student. Type the following into the R console (which stores the identification numbers into a variable named 'id'):

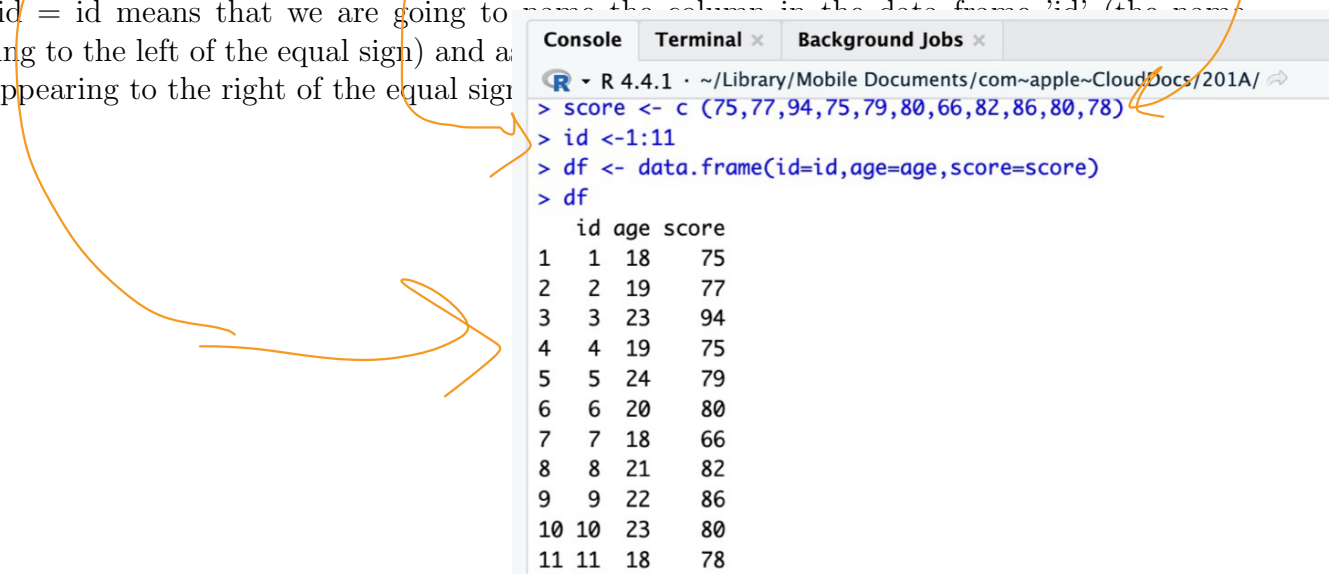
```
id <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11)
```

You can also do: `id <- 1:11`

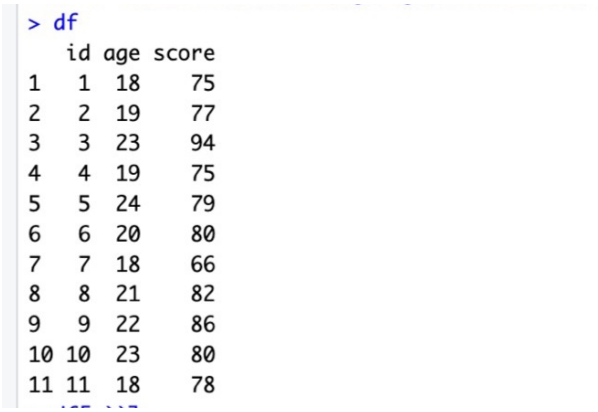
Problem 2c: It may be more convenient to create one object in R that contains all the information in one place. We use what is called a 'data frame' to do this. Type the following command into the R console:

```
df <- data.frame(id = id, age = age, score = score)
```

Note: id = id means that we are going to name the column in the data frame 'id' (the name appearing to the left of the equal sign) and a name appearing to the right of the equal sign



Take a screenshot of your dataframe.



```
> df
   id age score
1   1  18    75
2   2  19    77
3   3  23    94
4   4  19    75
5   5  24    79
6   6  20    80
7   7  18    66
8   8  21    82
9   9  22    86
10 10  23    80
11 11  18    78
```

Problem 2d: With the dataframe we just created, here are some ways we can extract relevant information from it. Explain what each command does.

- `df[,1]`

It gives out the first column of the data frame

- `df[,2]`

It gives out the second column of the data frame

- `df[1,1]`

It gives out the first element of the first column in the data frame

- `df[2,2]`

It gives out the second element of the second column in the data frame

- `df[1,]`

It gives out the first row of the data frame

- `df[2,]`

It gives out the second row of the data frame

- `df$id`

It gives out the id column as a list from the data frame

- `df["id"]`

It extract the column called “id” from the data frame

- `df$id[3]`

It gives out the 3rd element of the column called “id” in the datafra

- `df$age`

It gives out the age column as a list from the data frame

- `df$score`

It gives out the score column as a list from the data frame

- `df["score"]`

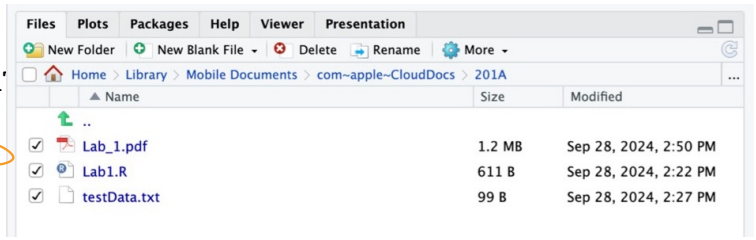
It extract the column called “id” from the data frame

3 Problem 3

Now we learn how to read data from a Text File into R:

Problem 3a: First you need to set the working directory to where the file is located.

```
setwd("ENTER PA
```



Problem 3b:

To read the data file into R and construct a data frame as done above, type the following command into the R console:

```
df <- read.table("testData.txt", sep=",")
  colnames(df) <- c("id","age","score")
        df <- as.data.frame(df)
```

Note: If the file is a .csv file, use `df <- read.csv("testData.csv")`

4 Problem 4

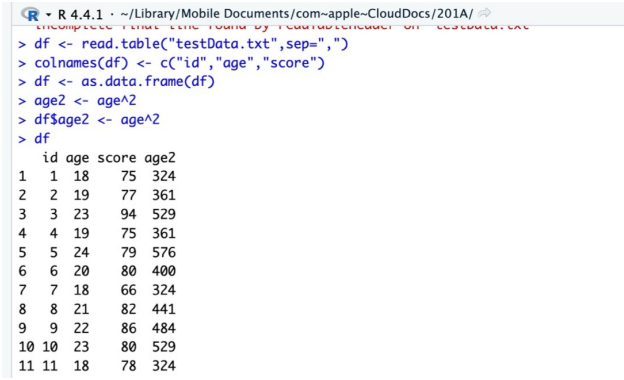
Creating New Variables:

Problem 4a: Suppose we want to create a new variable `age2` where we square each age value. We can do this by typing the following code into the R console:

```
age2 <- age ^ 2
```

If we want to add it to the data frame, we can type:

```
df$age2 <- age^2
```



To view the data frame, enter the following command into the console: `df`

What do the following commands do?

- `age + 1`

  It adds 1 to all the values in the age list

- `age + age`

  It adds each elements in the age list to itself(basically doubled the values)

5 Problem 5

Changing Data Values:

Problem 5a: Suppose the first age value was incorrectly recorded as 18. If the student's real age is 19, we can adjust the variable age using the following code:

```
age[1] <- 19
```

If we want to make the change in the data frame df, the following will do the trick (pick one):

- `df$age[1] <- 19`
- `df[1,2] <- 19`

```
> df[1,2] <-19
> df$age[1] <- 18
> age
 [1] 19 19 23 19 24 20 18 21 22 23 18
```

6 Problem 6

Now, let's report summary statistics of age.

With the originally supplied age data:

$$\{18, 19, 23, 19, 24, 20, 18, 21, 22, 23, 18\}$$

fill out the table:

Statistic	Age
Count (n)	11
Mean	20.45455
St. Dev.	2.252272
Median	20
25%ile	18.5
75%ile	22.5
IQR	4
min	18
max	24
range	18~24

```
> age <- c(18, 19, 23,19, 24, 20, 18, 21, 22, 23,18)
> median(age)
[1] 20
> min(age)
[1] 18
> max(age)
[1] 24
> range(age)
[1] 18 24
> quantile(age)
  0%  25%  50%  75% 100%
18.0 18.5 20.0 22.5 24.0
> sd(age)
[1] 2.252272
> var(age)
[1] 5.072727
> summary(age)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  18.00   18.50   20.00   20.45   22.50   24.00
> length(age)
[1] 11
> sum(age)
[1] 225
> IQR(age)
[1] 4
> table(age)
```

7 Problem 7

Describe the roles biostatistics serves in the discipline of public health.

To successfully complete this competency assessment, you may wish to review some of the following material:

- From Whitlock: Chapter 1
- From Class: Lecture 1

Your written description should be double-space typed and under 250 words.

Biostatistics is a useful discipline helping scientists and health specialists testing hypothesis and making decisions. It uses both quantitative and qualitative data to figure out practical applications such as: if a drug is working on certain group of patients. The professor mentioned a famous clinical trial about VitaminC in class. He mentioned that biostatistics analyzed if Vc works to prevent common cold. The scientist Linus Pauline tested the effect of Vc on volunteers along with a placebo control group. The results showed that his hypothesis was not statistically significant, which means that Vc does not prevents common cold. He also mentioned his involvement in a project collecting human milk to support pre-mature babies' survival, which implies the role of biostatistics includes allocating resources and apply to patients needing the resources. It ensures new treatment/drugs are safe to use and effective on patients.