Lab 2: Subsetting and Basic Data Summaries
STAT 630, Fall 2021

Remark: This lab borrows from the "Introduction to Data" lab available here:
`https://m.openintro.org/stat/labs.php`.

1 BRFSS Data Set

The Behavioral Risk Factor Surveillance System (BRFSS) is an annual telephone survey of over 400,000 people in the United States. The survey is conducted by the Centers for Disease Control and Prevention (CDC), a government agency focused on public health issues. As its name implies, the BRFSS is designed to identify risk factors in the adult population and report emerging health trends. For example, respondents are asked about their diet and weekly physical activity, their HIV/AIDS status, possible tobacco use, and level of healthcare coverage. The BRFSS web site `http://www.cdc.gov/brfss` contains a complete description of the survey, including the research questions that motivate the study and many interesting results derived from the data.

We will focus on a random sample of 20,000 people from the BRFSS survey conducted in the year 2000. While there are over 200 variables in this data set, we will work with a small subset.

Run the following command to load the data set of 20,000 observations into your R workspace.

```r
cdc <- readRDS(url("https://ericwfox.github.io/data/cdc.rds"))
```

To view the variable names and dimension of the `cdc` data frame type the following commands.

```r
names(cdc)

## [1] "genhlth"  "exerany"  "hlthplan" "smoke100" "height"   "weight"   "wtdesire"
## [8] "age"      "gender"

dim(cdc)

## [1] 20000     9
```

We can see clearly now the the data frame contains 20,000 entries (rows) on 9 variables. Each of the variables corresponds to a question that was asked in the survey. Descriptions of the variables are provided below:

- `genhlth`: a categorical variable indicating general health, with categories excellent, very good, good, fair, and poor
- `exerany`: a categorical variable, 1 if the respondent exercised in the past month and 0 otherwise
- `hlthplan`: a categorical variable, 1 if the respondent has some form of health coverage and 0 otherwise

- `smoke100`: a categorical variable, 1 if the respondent has smoked at least 100 cigarettes in their entire life and 0 otherwise
- `height`: a numerical variable, respondent's height in inches
- `weight`: a numerical variable, respondent's weight in pounds
- `wtdesire`: a numerical variable, respondent's desired weight in pounds
- `age`: a numerical variable, respondent's age in years
- `gender`: a categorical variable, respondent's gender

We can have a look at the first several rows of the data with the command

```
head(cdc)

##     genhlth exerany hlthplan smoke100 height weight wtdesire age gender
## 1      good       0        1        0     70    175      175  77      m
## 2      good       0        1        1     64    125      115  33      f
## 3      good       1        1        1     60    105      105  49      f
## 4      good       1        1        0     66    132      124  42      f
## 5 very good       0        1        0     61    150      130  55      f
## 6 very good       1        1        0     64    114      114  55      f
```

You could also look at all of the data frame at once by typing its name into the console, but that might be unwise here. We know `cdc` has 20,000 rows, so viewing the entire data set would mean flooding your screen. It's better to take small peeks at the data with `head()`, `tail()` or the indexing techniques covered during the last lab.

2 Summaries and Tables

The BRFSS questionnaire is a massive trove of information. A good first step in any analysis is to distill all of that information into a few summary statistics and graphics. As a simple example, the function `summary()` returns a numerical summary: minimum, first quartile, median, mean, third quartile, and maximum. For `weight` this is

```
summary(cdc$weight)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    68.0   140.0   165.0   169.7   190.0   500.0
```

As discussed in the previous lab, R also has built-in functions to compute summary statistics one at a time. For example:

```
mean(cdc$weight)

## [1] 169.683

median(cdc$weight)

## [1] 165

sd(cdc$weight)

## [1] 40.08097
```

While it makes sense to describe a numerical variable like `weight` in terms of these statistics, what about categorical data? We could instead consider the frequency or relative frequency distribution. The function `table()` does this for you by counting the number of times each kind of response was given. For example, to see the number of people who have smoked 100 cigarettes in their lifetime, type

```
table(cdc$smoke100)

##
##     0     1
## 10559  9441
```

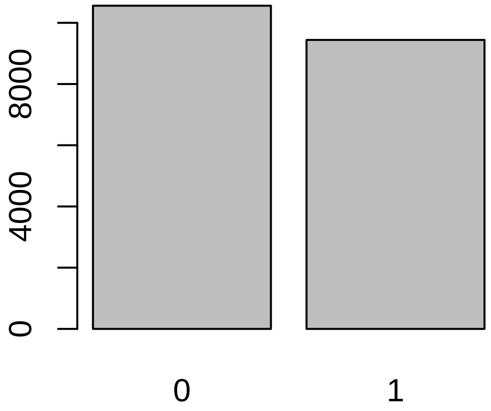
or instead look at the relative frequency distribution by typing

```
table(cdc$smoke100)/20000

##
##       0       1
## 0.52795 0.47205
```

Next, we make a bar plot of the entries in the table by putting the table inside the `barplot()` command.

```
barplot(table(cdc$smoke100))
```



Notice what we've done here! We've computed the table of `cdc$smoke100` and then immediately applied the graphical function, `barplot()`. This is an important idea: R commands can be nested. You could also break this into two steps by typing the following:

```
smoke_tb <- table(cdc$smoke100)
barplot(smoke)
```

The `table()` command can be used to tabulate any number of variables that you provide. For example, to examine which participants have smoked across each gender, we could use the following.

```
table(cdc$gender,cdc$smoke100)

##
##        0    1
##   f 6012 4419
##   m 4547 5022
```

Here, we see column labels of 0 and 1. Recall that 1 indicates a respondent has smoked at least 100 cigarettes. The rows refer to gender. To include the row and column totals use `addmargins()`.

```
addmargins(table(cdc$gender, cdc$smoke100))

##
##            0     1   Sum
##   f     6012  4419 10431
##   m     4547  5022  9569
##   Sum  10559  9441 20000
```

3 Subsetting Data Frames

The first lab went over how to extract rows, columns, and specific elements of a data frame using indexing (i.e., brackets `[]`) or by using `$` to extract columns (variables) by their names. However, it is also useful to extract rows of a data frame that have specific characteristics. For instance, suppose we want the extract the rows of the `cdc` data frame that correspond to a certain gender (male or female), or extract the rows corresponding to respondents who are over 40 years old. To do this we can use logical expressions and subsetting techniques.

To illustrate logical operations in R, lets work with a smaller portion of the `cdc` data frame that consists of the first 10 rows.

```
cdc10 <- cdc[1:10,]
cdc10

##      genhlth exerany hlthplan smoke100 height weight wtdesire age gender
## 1       good       0        1        0     70    175      175  77      m
## 2       good       0        1        1     64    125      115  33      f
## 3       good       1        1        1     60    105      105  49      f
## 4       good       1        1        0     66    132      124  42      f
## 5  very good       0        1        0     61    150      130  55      f
## 6  very good       1        1        0     64    114      114  55      f
## 7  very good       1        1        0     71    194      185  31      m
## 8  very good       0        1        0     67    170      160  45      m
## 9       good       0        1        1     65    150      130  27      f
## 10      good       1        1        0     70    180      170  44      m
```

The following command gives logical values (`TRUE`, `FALSE`) for whether each respondent is male.

```
cdc10$gender

##  [1] "m" "f" "f" "f" "f" "f" "m" "m" "f" "m"

cdc10$gender == "m"

##  [1]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE
```

To extract the rows of the data frame `cdc10` corresponding to the males, use the `subset()` function.

```
subset(cdc10, gender == "m")

##      genhlth exerany hlthplan smoke100 height weight wtdesire age gender
## 1       good       0        1        0     70    175      175  77      m
## 7  very good       1        1        0     71    194      185  31      m
## 8  very good       0        1        0     67    170      160  45      m
## 10      good       1        1        0     70    180      170  44      m
```

Similarly, we can extract the rows of the data frame `cdc10` corresponding to respondents who are over 40 years old.

```
cdc10$age

##  [1] 77 33 49 42 55 55 31 45 27 44

cdc10$age > 40

##  [1]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE

subset(cdc10, age > 40)

##      genhlth exerany hlthplan smoke100 height weight wtdesire age gender
## 1       good       0        1        0     70    175      175  77      m
## 3       good       1        1        1     60    105      105  49      f
## 4       good       1        1        0     66    132      124  42      f
## 5  very good       0        1        0     61    150      130  55      f
## 6  very good       1        1        0     64    114      114  55      f
## 8  very good       0        1        0     67    170      160  45      m
## 10      good       1        1        0     70    180      170  44      m
```

The following command extracts the rows of `cdc10` corresponding to respondents who are males *and* over the age of 40.

```
subset(cdc10, gender == "m" & age > 40)

##      genhlth exerany hlthplan smoke100 height weight wtdesire age gender
## 1       good       0        1        0     70    175      175  77      m
## 8  very good       0        1        0     67    170      160  45      m
## 10      good       1        1        0     70    180      170  44      m
```

The following table summarizes the different logical operators in R:

Operator	Description
`<`	less than
`<=`	less than or equal to
`>`	greater than
`>=`	greater than or equal to
`==`	exactly equal to
`!=`	not equal to
`x | y`	x OR y
`x & y`	x AND y

Note that `=` is used for assignment and is not the same as the `==` logical operator.

Using these new subsetting tools we can explore some interesting aspects of the entire `cdc` data frame. For example, what is the average weight and desired weight for males and females? To answer this question create separate data frames for the males and females. Then use the `summary()` function on each subsetted data frame.

```
cdc_m <- subset(cdc, gender == "m")
cdc_f <- subset(cdc, gender == "f")

summary(cdc_m$weight)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    78.0   165.0   185.0   189.3   210.0   500.0

summary(cdc_m$wtdesire)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    77.0   160.0   175.0   178.6   190.0   680.0

summary(cdc_f$weight)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    68.0   128.0   145.0   151.7   170.0   495.0

summary(cdc_f$wtdesire)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    68.0   120.0   130.0   133.5   145.0   350.0
```

The mean and median desired weight is lower than the actual weight for both genders. The maximum desired weight for males is unusual since someone has a desired weight of 680lbs! This is probably an outlier that we might want to remove.

In-class Exercise: Create a new data frame called `under23_and_smoke` that contains the subset of respondents who are under the age of 23 and have smoked 100 cigarettes in their lifetime. Write the command you used to create the new data frame as the answer to this exercise.