R Assingment 1

Code segments provided for each question, full R script for the assignment at the bottom

Importing Data

> What is the age of the participant in the third row of the data set?

```
lhs$age[3]
```

The age for the participant in the third row of the data set is 44.

> The participant in the third row of the data set was alive at the end of the study, and therefore, doesn't have values for dstatus, days2ded, or deadcode. How does your software indicate data is missing?

The missing data is indicated as `NA` (not available)

> What data type did R put for the race variable, race? For the clinical center code variable, ACLINIC? For the BMI variable, bmi?

```
str(lhs$race)      # int
str(lhs$ACLINIC)   # chr
str(lhs$bmi)       # num
```

R put the race variable as integer, the clinical center code as character (string literals or string vectors) and the bmi variable as numeric.

> Using the LHS data documentation and the output from the str() function, do the data types of the variables in R match the actual type for the variables? If they do not match for a variable, think about why R incorrectly defined a data type for the variable. That is, what is it about the way the data was recorded that made R misclassify the data type?

```
str(lhs)
```

compare output for each variable with assumed data type from data dictionary. The following data types do not match what I expect them to be:

variable	assumed data type	data type in R	reason for misclassification
dstatus	categorical	integer	death status is classified as 1, 2 or NA, so R interprets it as integers
deadcode	categorical	integer	cause of death is classified as 1-33
race, marital0	categorical	integer	each race/ marital status is given a number and coded as that, R only sees the number
f31pipe, f31snuff, f31snuff, f060snuff, f260pipe, f260snuff, f360pipe, f360snuff, f460pipe, f460snuff, f560pipe, f560snuff, BLCIGAR, A1CIGAR, A2CIGAR, A3CIGAR, A4CIGAR, A5CIGAR, AV1GUM, AV2GUM, AV3GUM, AV4GUM, AV5GUM	logical	integer	options are yes and no, but are coded as 1 and 2

For each of these variables, the categories are displayed as numbers which is why R interprets it as integer. This might cause problems if the values are interpreted to have an order and a set distance between them, when neither is actually there.

Modifying Categorical Variables

> The closest census to the time period of our LHS dataset is the 1990 census. According to the 1990 census, the population of the United States was 80.3% White, 12.1% Black/African American, 2.9% Asian/Pacific Islander, and 0.8% American Indian/Eskimo/Aleutian.
>
> Which races are over-represented and which are underrepresented in our study?

```
table(lhs$race1)
prop.table(table(lhs$race1))
```

Race	1990 Census	represented in study
White	80.3%	95.8%
Black/ African American	12.1%	3.8%
Asian/ Pacific Islander	2.9%	0.1%
Native American/ Other	0.8%	0.3%

White is over-represented in the study, while all other races are under-represented by varying degree. On a percentage basis, the most under-represented race is Asian/ Pacific Islander.

> Create a table of the new intervention variable. Which group is larger, the intervention group (1) or the control group (0)?

```
lhs$intervention<-rep("NA", length(lhs$alphagroup))
lhs$intervention[which(lhs$alphagroup == "SI-A")]<-1
lhs$intervention[which(lhs$alphagroup == "SI-P")]<-1
lhs$intervention[which(lhs$alphagroup == "UC")]<-0
table(lhs$intervention)

0    1
1964 3923
```

Group 1 is nearly twice the size of group 0, so there larger group is group 1 with some form of intervention.

> What is the name of the new variable in the LHS data set when the categories of the variable f060pipe was re-coded?

```
###Access the package using the library() function
library(plyr)
###Recode categories using the mapvalues() function
lhs$pipe1 <- mapvalues(lhs$f060pipe, from=c(1, 2), to = c(1, 0))
```

The name of the new variable in the LHS data set is `pipe1`.

> There is a variable that indicates the marital status of a participant at the start of the study (marital0). Re-code this variable so we have three categories: Never Married, Separated/Divorced, and Married/Widowed.

```
lhs$marital1<-mapvalues(lhs$marital0, from=c(1, 2, 3, 4, 5), to = c(1, 3, 3, 2,
2))
```

With *Never Married* as category 1, *Separated/ Divorced* as category 2 and *Married/ Widowed* as category 3.

Reclassifying Variables

> Why are quotes placed around the labels abnormal and normal in the cut() function?

The quotes around the labels indicate, that all characters in the words *abnormal* and *normal* belong together and should be interpreted as a string.

> What is the name of the variable in the LHS data set where marital status is a factor? An integer?

```
# make a numerical variable categorical
str(lhs$marital0)
lhs$marital<-as.factor(lhs$marital0)
str(lhs$marital)
```

The variable where marital status is a factor is the newly created variable `marital`. In the already existing variable `marital0` as well as the in part **Modifying Categorical Variables** created variable `marital1`, the marital status is an integer.

> Reclassify the numerical variable for weight change (kg) from baseline to first annual visit, wgtchg01, as a categorical variable with three categories: lost wt, no change, and gained wt.

```
# re-code weight change into categorical variable with three categories. Need to
have unique breaks, so cannot use [0 0] as breaks for no change
lhs$wgtchg01c <- cut(lhs$wgtchg01,
                     breaks = c(min(lhs$wgtchg01, na.rm = T),
                                -0.00001, 0.000001,
                                max(lhs$wgtchg01, na.rm = T)),
                     labels = c("lost wt", "no change", "gained wt"))
```

Complete Script for R Assignment 1

```
###Access the package using the library() function
library(plyr)

lhs <- read.csv(file = file.choose(), header = TRUE)


head(lhs)    # view first six rows of data
str(lhs)    # show data type for each column

lhs$age[3]    # 44

str(lhs$race)      # int
str(lhs$ACLINIC)   # chr
str(lhs$bmi)       # num

# new variable for race with names instead of category numbers
lhs$race1<-factor(lhs$race,
                  levels=c(1,2,3,4,5),
```

```
                  labels=c("White","Black","Asian","Native American","Other"))

# table: creates table for each category with number of elements in that
category
table(lhs$race1)
# prop.table: table with percentage of elements in that category
prop.table(table(lhs$race1))

# new variable, initially everything "NA", groups depending on previous value
lhs$intervention<-rep("NA", length(lhs$alphagroup))
lhs$intervention[which(lhs$alphagroup == "SI-A")]<-1
lhs$intervention[which(lhs$alphagroup == "SI-P")]<-1
lhs$intervention[which(lhs$alphagroup == "UC")]<-0
table(lhs$intervention)

# Re-code categories using the mapvalues() function
lhs$pipe1<-mapvalues(lhs$f060pipe, from=c(1, 2), to = c(1, 0))

# re-code categories in new variable
lhs$marital1<-mapvalues(lhs$marital0, from=c(1, 2, 3, 4, 5), to = c(1, 3, 3, 2,
2))

# re-code numerical variable in new categorical variable with threshold defined
for cut-function
lhs$lung.fxn<-cut(lhs$FEVFVC3,
                  breaks=c(0, 69.99999999, 100),
                  labels=c("abnormal","normal"))

# make a numerical variable categorical
str(lhs$marital0)
lhs$marital<-as.factor(lhs$marital0)
str(lhs$marital)
# change base for categorical variable
lhs$marital<-relevel(lhs$marital, ref=2)
str(lhs$marital)

# re-code weight change into categorical variable with three categories. Need to
have unique breaks, so cannot use [0 0] as breaks for no change
lhs$wgtchg01c <- cut(lhs$wgtchg01,
                     breaks = c(min(lhs$wgtchg01, na.rm = T),
                                -0.00001, 0.000001,
                                max(lhs$wgtchg01, na.rm = T)),
                     labels = c("lost wt", "no change", "gained wt"))
```