Homework 2

NAME

DATE

Introduction

In this homework assignment, you will be working with data and managing factors. The data sets you will use are, `ncbirths` and `smoking` data set which come part of the `openintro` package.

The code chunk below sets some code chunk options (using `opts_chunk` from the `knitr` package) to make your knitted report output more readable. I encourage you to play around with these options to learn how they work.

```
knitr::opts_chunk$set(warning=FALSE, message=FALSE, fig.height=4, fig.width=5, fig.align='center')
library(dplyr)
library(forcats)
ncbirths <- openintro::ncbirths
smoking <- openintro::smoking
```

Working with Data

This section we will use `ncbirths` data set.

1. Calculate the mean age of the mothers (`mage`).

```
mean(ncbirths$mage)
```

```
## [1] 27
```

2. Pregnancies last on average 38 weeks. Recode the `weeks` variable to change all records where `weeks` is greater than 38, to equal 38. That is, for all record where `weeks>38`, change the value of `weeks` to `<- 38`.

```
ncbirths$weeks[ncbirths$weeks > 38] <- 38
```

3. Use the `summary` function to calculate summary statistics on the fathers age (`fage`). Round to 3 digits using the `digits=` argument. Don't forget that you can look at the bottom of the help for summary (`?summary`) file for examples on how to use this function.

```
summary(ncbirths$fage,digits=3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    14.0    25.0    30.0    30.3    35.0    55.0     171
```

4. Use the `is.na()` function to create a new variable called `missing_gained` on the `ncbirths` data set that identifies if data on the `gained` variable is missing.

```
ncbirths$missing_gained <- is.na(ncbirths$gained)
```

5. What percent of records are missing data on weight gained during pregnancy? There are several ways you can calculate this! Explain what your process, and answer the question in a full sentence.

```
table(ncbirths$missing_gained)
```

```
##
## FALSE  TRUE
##   973    27
```

```
mean(ncbirths$missing_gained)
```

```
## [1] 0.027
```

We could use the table function and do the calculations by hand, or be clever about it and let the function `mean()` do the math for us. Either way, `.027*100`% is missing.

6. Use the `ifelse()` function to dichotomize the `weeks` variable at it's mean where records with values over the mean are labeled `AboveAve` and records with values below the mean are labeled `UnderAve`. Call this new variable `week_ave`. (*Hint: Calculate the mean value for the variable* `weeks`*, then use that number in the logical statement part of the* `ifelse` *function.*)

```
week_ave <- mean(ncbirths$weeks, na.rm=TRUE)
ncbirths$week_ave <- ifelse(ncbirths$weeks <= week_ave, "UnderAve", "AboveAve")
```

7. Create a frequency table for your new variable (`week_ave`) in the previous question. Then use the pipe operator `%>%` to add on the function `prop.table()` at the end. What does the `prop.table()` function do?

```
ncbirths$week_ave %>% table() %>% prop.table()
```

```
## .
## AboveAve UnderAve
## 0.742485 0.257515
```

- `prop.table()` function gives you,instead of frequency, the percentage of a value or category in a variable.

Wrangling Factors

In this section, you will use `smoking` data set.

1. Examine the variable `ethnicity` using `fct_count`. Then, collapse levels of that variable into a smaller number of factors using `fct_collapse`. *Hint: Create a new factor variable as part of the* ***`smoking`*** *data set (i.e smoking$ethnicity_new).*

- "NA" = c("Refused","Unknown")
- Asian = c("Chinese", "Asian")

```
fct_count(smoking$ethnicity)
```

```
## # A tibble: 7 x 2
##   f           n
##   <fct>   <int>
## 1 Asian      41
## 2 Black      34
## 3 Chinese    27
## 4 Mixed      14
## 5 Refused    13
## 6 Unknown     2
## 7 White    1560
```

```
smoking$ethnicity_new <- fct_collapse(smoking$ethnicity
,"NA"=c("Refused","Unknown")
,Asian=c("Asian","Chinese"))
```

2. Create a two-way `table` of `ethnicity` against `ethnicity_new` to confirm that this new factor variable was created correctly.

```
table(smoking$ethnicity, smoking$ethnicity_new)
```

```
##
##           Asian Black Mixed   NA White
##   Asian      41     0     0    0     0
##   Black       0    34     0    0     0
##   Chinese    27     0     0    0     0
##   Mixed       0     0    14    0     0
##   Refused     0     0     0   13     0
##   Unknown     0     0     0    2     0
##   White       0     0     0    0  1560
```

3. Using `fct_recode`, create a new factor variable `recode_ethnicity` from `ethnicity_new` with labels "A"(Asian), "B"(Black), "M"(Mixed), "W"(White). Make sure you create this new variable as part of the `smoking` data set. (i.e. `smoking$recode_ethnicity <-`)

```
smoking$recode_ethnicity <- smoking$ethnicity_new %>%
        fct_recode("A"="Asian", "B"="Black","M"="Mixed", "W"="White")
table(smoking$recode_ethnicity)
```

```
##
##    A    B    M   NA    W
##   68   34   14   15 1560
```

4. Manually reorder the level of `ethnicity_new` variable in an increasing order using `fct_relevel`.

```
smoking$ethnicity_new %>% fct_relevel("Mixed","NA","Black","Asian","White") %>% table()
```

```
## .
## Mixed    NA Black Asian White
##    14    15    34    68  1560
```