SS21 STT 180 Homework 1

Emani Hunter

Jan30-Feb13, 2021

Contents

This homework assignment consists of two sections. The first section deals with data structures and the second section is a small data analysis project. You will use the data wrangling and tidying knowledge for this section.

General Instructions:

- This is an individual assignment. You may consult with others as you work on the assignment, but each student should write up a separate set of solutions.
- Rather than creating a new Rmd file, just add your solutions to the supplied Rmd file. Submit both the Rmd file and the resulting HTML/PDF file to D2L.
- Except for questions, or parts of questions, that ask for your commentary, use R in a code chunk to answer the questions.
- The code chunk option `echo = TRUE` is specified in the setup code chunk at the beginning of the document. Please do not override this in your code chunks.
- A solution will lose points if the Rmd file does not compile. If one of your code chunks is causing your Rmd file to not compile, you can use the `eval = FALSE` option. Another possibility is to use the `error = TRUE` option in the code chunk.
- This Homework is due on **Saturday, February 13, 2021 on or before 11 pm.**
- Kindly submit both the .rmd and the HTML or pdf output files. If you submit the output in html format, zip the files while submitting.

Section 1

This section focuses on some basic manipulations of vectors in R.

Question 1

Create three vectors in R: One called `evennums` which contains the even integers from 1 through 15. One called `charnums` which contains character representations of the numbers 4 through 8, namely, "4", "5", "6", "7", "8". And one called `mixed` which contains the same values as in `charnums` but which also contains the letters "a", "b" and "c". **No commentary or explanations are necessary.**

```
evennums <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)
charnums <- c("4","5","6","7","8")
mixed <- c("4","5","6","7","8","a","b","c")
```

Question 2

Investigate what happens when you try to convert `evennums` to character and to logical. Investigate what happens when you convert `charnums` to numeric. Investigate what happens when you convert `mixed` to numeric. **Comment on each of these conversions.**

```
as.character(evennums)
```

```
 [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10" "11" "12" "13" "14" "15"
```

```
as.logical(evennums)
```

```
 [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

```
as.numeric(charnums)
```

```
[1] 4 5 6 7 8
```

```r
as.numeric(mixed)
```

```
[1]  4  5  6  7  8 NA NA NA
```

Converting 'evennums' to character: - When calling as.character(evennums), a vector of type character is created, each value in the evennums vector is turned into a character.

Converting 'evennums' to logical: - When calling as.logical(evennums), a vector of type logical is created, each value in the evennums vector is turned into a logical 'true' value.

Converting 'charnums' to numeric: - When calling as.numeric(charnums), a vector of type numeric is created, each character in the charnums vector is turned into an actual numeric value.

Converting 'mixed' to numeric: - When calling as.numeric(mixed), a vector of type numeric is created, each character that is a "number" is turned into a numeric value, but, each character that is a letter cannot be turned into a numeric value and so, we get an NA position in the vector. ### Question 3

No commentary is necessary on this part.

a. Show how to extract the first element of `evennums`.

```r
evennums[1]
```

```
[1] 1
```

b. Show how to extract the last element of `evennums`. In this case you are NOT allowed to use the fact that `evennums` has seven elements, rather, you must give code which would work no matter how many elements `evennums` has.

```r
evennums[length(evennums)]
```

```
[1] 15
```

c. Show how to extract all but the first element of `evennums`.

```r
evennums[-1]
```

```
 [1]  2  3  4  5  6  7  8  9 10 11 12 13 14 15
```

d. Show how to extract all but the first two and last two elements of `evennums`.

```r
evennums[-c(1,2,length(evennums)-1,length(evennums))]
```

```
 [1]  3  4  5  6  7  8  9 10 11 12 13
```

Question 4

a. Generate a sequence "y" of 50 evenly spaced values between 0 and 1.

```
y <- seq(0,1,length.out = 50)
y
```

```
 [1] 0.00000000 0.02040816 0.04081633 0.06122449 0.08163265 0.10204082
 [7] 0.12244898 0.14285714 0.16326531 0.18367347 0.20408163 0.22448980
[13] 0.24489796 0.26530612 0.28571429 0.30612245 0.32653061 0.34693878
[19] 0.36734694 0.38775510 0.40816327 0.42857143 0.44897959 0.46938776
[25] 0.48979592 0.51020408 0.53061224 0.55102041 0.57142857 0.59183673
[31] 0.61224490 0.63265306 0.65306122 0.67346939 0.69387755 0.71428571
[37] 0.73469388 0.75510204 0.77551020 0.79591837 0.81632653 0.83673469
[43] 0.85714286 0.87755102 0.89795918 0.91836735 0.93877551 0.95918367
[49] 0.97959184 1.00000000
```

b. Calculate the mean of the sequence.

```
mean(y)
```

```
[1] 0.5
```

Section 2

The dataset contains information about births in the United States. The full data set is from the Centers for Disease Control. The data for this homework assignment is a "small" sample (chosen at random) of slightly over one million records from the full data set. The data for this homework assignment also only contain a subset of the variables in the full data set.

Setting up

Load `tidyverse`, which includes `dplyr`, `tidyr`, and other packages, and the load 'knitr.

```
library(tidyverse)
library(knitr)
```

Read in the data and convert the data frame to a tibble.

```
birth_data <- read.csv("BirthData.csv", header = TRUE)
birth_data <- as_tibble(birth_data)
```

A glimpse of the data:

```
glimpse(birth_data)
```

```
Rows: 1,103,629
Columns: 8
$ year          <int> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969,...
$ month         <int> 9, 8, 9, 2, 3, 5, 5, 5, 6, 8, 8, 11, 11, 11, 1, 12, 3...
$ state         <chr> "AL", "AZ", "AZ", "CA", "CA", "CA", "CA", "CA", "CA",...
$ is_male       <lgl> FALSE, TRUE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE, TR...
```

```
$ weight_pounds <dbl> 1.624807, 7.500126, 8.937540, 6.999677, 6.876218, 7.1...
$ mother_age    <int> 20, 35, 17, 20, 25, 30, 17, 22, 26, 26, 19, 25, 26, 2...
$ child_race    <int> 2, 1, 1, 1, 2, 1, 1, 4, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1,...
$ plurality     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
```

The variables in the data set are:

Variable	Description
`year`	the year of the birth
`month`	the month of the birth
`state`	the state where the birth occurred, including "DC" for Washington D.C.
`is_male`	which is `TRUE` if the child is male, `FALSE` otherwise
`weight_pounds`	the child's birth weight in pounds
`mother_age`	the age of the mother
`child_race`	race of the child.
`plurality`	the number of children born as a result of the pregnancy, with 1 representing a single birth, 2 representing twins, etc.

For both of Questions 1 and 2 you should show the R code used and the output of the `str` and`glimpse` functions applied to the data frame. Use of dplyr functions and the pipe operator is highly recommended.

Question 1

Create a variable called `region` in the data frame `birth_data` which takes the values `Northeast`, `Midwest`, `South`, and `West`. The first two Steps have been done for you.

Here are the states in each region:

Northeast Region: Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island and Vermont, New Jersey, New York, and Pennsylvania

Midwest Region: Illinois, Indiana, Michigan, Ohio and Wisconsin, Iowa, Kansas, Minnesota, Missouri, Nebraska, North Dakota, and South Dakota

South Region: Delaware, District of Columbia, Florida, Georgia, Maryland, North Carolina, South Carolina, Virginia, and West Virginia, Alabama, Kentucky, Mississippi, and Tennessee, Arkansas, Louisiana, Oklahoma, and Texas

West Region: Arizona, Colorado, Idaho, Montana, Nevada, New Mexico, Utah and Wyoming, Alaska, California, Hawaii, Oregon and Washington

```r
#Step 1: Assign the regions.
NE <- c("CT", "ME", "MA", "NH", "RI", "VT", "NJ", "NY", "PA")
MW <- c("IL", "IN", "MI", "OH", "WI", "IA", "KS", "MN", "MO", "NE", "ND", "SD")
SO <- c("DE", "DC", "FL", "GA", "MD", "NC", "SC", "VA", "WV", "AL", "KY", "MS", "TN", "AR", "LA", "OK",
WE <- c("AZ", "CO", "ID", "MT", "NV", "NM", "UT", "WY", "AK", "CA", "HI", "OR", "WA")
## Step 2 Create a blank vector
```

```r
birth_data$region <- rep(NA, length(birth_data$state))

## Hint use if-else and %in% to create the regions.


birth_data$region[birth_data$state %in% NE] <-  "Northeast"

birth_data$region[birth_data$state %in% MW] <-  "Midwest"

birth_data$region[birth_data$state %in% SO] <- "South"

birth_data$region[birth_data$state %in% WE] <-  "West"

glimpse(birth_data)
```

```
Rows: 1,103,629
Columns: 9
$ year          <int> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969,...
$ month         <int> 9, 8, 9, 2, 3, 5, 5, 5, 6, 8, 8, 11, 11, 11, 1, 12, 3...
$ state         <chr> "AL", "AZ", "AZ", "CA", "CA", "CA", "CA", "CA", "CA",...
$ is_male       <lgl> FALSE, TRUE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE, TR...
$ weight_pounds <dbl> 1.624807, 7.500126, 8.937540, 6.999677, 6.876218, 7.1...
$ mother_age    <int> 20, 35, 17, 20, 25, 30, 17, 22, 26, 26, 19, 25, 26, 2...
$ child_race    <int> 2, 1, 1, 1, 2, 1, 1, 4, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1,...
$ plurality     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
$ region        <chr> "South", "West", "West", "West", "West", "West", "Wes...
```

Question 2

Create a variable in `birth_data` called `state_color` which takes the values `red`, `blue`, and `purple`, using the following divisions.

Red: Alaska, Idaho, Kansas, Nebraska, North Dakota, Oklahoma, South Dakota, Utah, Wyoming, Texas, Alabama, Mississippi, South Carolina, Montana, Georgia, Missouri, Louisiana, Tennessee, Arkansas, Kentucky, Arizona, West Virginia.

Purple: North Carolina, Virginia, Florida, Ohio, Colorado, Nevada, Indiana, Iowa, New Mexico.

Blue: New Hampshire, Pennsylvania, California, Michigan, Illinois, Maryland, Delaware, New Jersey, Connecticut, Vermont, Maine, Washington, Oregon, Wisconsin, New York, Massachusetts, Rhode Island, Hawaii, Minnesota, District of Columbia.

```r
RED <- c("AK", "ID", "KS", "NE", "ND", "OK", "SD", "UT", "WY", "TX", "AL", "MS", "SC", "MT", "GA", "MO"
PURPLE <- c("NC", "VA", "FL", "OH", "CO", "NV", "IN", "IA", "NM")
BLUE <- c("NH", "PA", "CA", "MI", "IL", "MD", "DE", "NJ", "CT", "VT", "ME", "WA", "OR", "WI", "NY", "MA

## try using mutate
birth_data$state_color <- rep(NA,length(birth_data$state))

#using if-else to check if state color is red, blue or purple
```

```r
birth_data$state_color <- ifelse(birth_data$state%in% RED, "red",ifelse(birth_data$state%in% PURPLE, "pu

glimpse(birth_data)
```

```
Rows: 1,103,629
Columns: 10
$ year          <int> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969,...
$ month         <int> 9, 8, 9, 2, 3, 5, 5, 5, 6, 8, 8, 11, 11, 11, 1, 12, 3...
$ state         <chr> "AL", "AZ", "AZ", "CA", "CA", "CA", "CA", "CA", "CA",...
$ is_male       <lgl> FALSE, TRUE, TRUE, TRUE, FALSE, TRUE, FALSE, TRUE, TR...
$ weight_pounds <dbl> 1.624807, 7.500126, 8.937540, 6.999677, 6.876218, 7.1...
$ mother_age    <int> 20, 35, 17, 20, 25, 30, 17, 22, 26, 26, 19, 25, 26, 2...
$ child_race    <int> 2, 1, 1, 1, 2, 1, 1, 4, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1,...
$ plurality     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
$ region        <chr> "South", "West", "West", "West", "West", "West", "Wes...
$ state_color   <chr> "red", "red", "red", "blue", "blue", "blue", "blue", ...
```

Question 3

Create two new objects `perc_male` and `perc_female` that caluclates the percentile ranking of a baby's weight with respect to the baby's sex.

```r
## The dataset to find the male percentiles
birth_data1<-birth_data%>%
              filter(is_male== TRUE)#%>%
               # select(is_male, weight_pounds, plurality)
glimpse(birth_data1)
```

```
Rows: 566,380
Columns: 10
$ year          <int> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969,...
$ month         <int> 8, 9, 2, 5, 5, 6, 1, 3, 6, 7, 10, 2, 3, 5, 7, 8, 10, ...
$ state         <chr> "AZ", "AZ", "CA", "CA", "CA", "CA", "CO", "CT", "CT",...
$ is_male       <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,...
$ weight_pounds <dbl> 7.500126, 8.937540, 6.999677, 7.187070, 7.374463, 9.6...
$ mother_age    <int> 35, 17, 20, 30, 22, 26, 27, 28, 24, 20, 23, 26, 23, 1...
$ child_race    <int> 1, 1, 1, 1, 4, 2, 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 1, 9,...
$ plurality     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
$ region        <chr> "West", "West", "West", "West", "West", "West", "West...
$ state_color   <chr> "red", "red", "blue", "blue", "blue", "blue", "purple...
```

```r
## Hint: use the quantile function to find the percentiles.
#Male percentiles
perc_male <- quantile(birth_data1$weight_pounds, probs = seq(0,1,0.25), na.rm = TRUE, names = TRUE)
perc_male
```

```
        0%        25%        50%        75%       100%
 0.5004493  6.7505545  7.5001262  8.2717441 17.9897206
```

```r
birth_data_female <- birth_data%>%
                     filter(is_male == FALSE)

glimpse(birth_data_female)
```

```
Rows: 537,249
Columns: 10
$ year          <int> 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969, 1969,...
$ month         <int> 9, 3, 5, 8, 8, 11, 11, 11, 12, 3, 3, 8, 11, 11, 9, 8,...
$ state         <chr> "AL", "CA", "CA", "CA", "CA", "CA", "CA", "CA", "CO",...
$ is_male       <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALSE, FALS...
$ weight_pounds <dbl> 1.624807, 6.876218, 7.749249, 7.936641, 6.499227, 9.0...
$ mother_age    <int> 20, 25, 17, 26, 19, 25, 26, 26, 36, 31, 23, 19, 21, 2...
$ child_race    <int> 2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 2, 2, 1, 2, 1, 2,...
$ plurality     <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
$ region        <chr> "South", "West", "West", "West", "West", "West", "Wes...
$ state_color   <chr> "red", "blue", "blue", "blue", "blue", "blue", "blue"...
```

```r
#Female percentiles
perc_female <- quantile(birth_data_female$weight_pounds, probs = seq(0,1,0.25), na.rm = TRUE, names = T
perc_female
```

```
        0%        25%        50%        75%       100%
 0.5004493  6.4992275  7.2510038  7.9983709 17.1453501
```

Question 4

Create another new variable that records the percentile ranking of a baby's weight with respect to the baby's plurality (i.e., whether it was a single child, twin, triplet, etc.). [i.e., if a baby is a twin (plurality = 2), the variable should record the percentile ranking of the baby's weight relative only to all other twins.]

```r
## The dataset for plurality = 1 ; do the same for the other pluralities
birth_data1<-birth_data%>%
             filter(plurality == 1)#%>%
glimpse(birth_data1)
```

```
Rows: 1,046,856
Columns: 10
$ year          <int> 1971, 1971, 1971, 1971, 1971, 1971, 1971, 1971, 1971,...
$ month         <int> 5, 8, 9, 10, 12, 1, 1, 1, 1, 1, 2, 2, 3, 3, 4, 5, 6, ...
$ state         <chr> "AL", "AL", "AL", "AL", "AL", "CA", "CA", "CA", "CA",...
$ is_male       <lgl> FALSE, TRUE, FALSE, TRUE, FALSE, TRUE, FALSE, FALSE, ...
$ weight_pounds <dbl> 6.000983, 8.313632, 5.500533, 6.437498, 6.499227, 7.3...
$ mother_age    <int> 32, 38, 20, 25, 38, 24, 38, 20, 20, 24, 28, 25, 20, 2...
$ child_race    <int> 2, 1, 2, 2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 1,...
$ plurality     <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
$ region        <chr> "South", "South", "South", "South", "South", "West", ...
$ state_color   <chr> "red", "red", "red", "red", "red", "blue", "blue", "b...
```

```r
## Hint: use the quantile function to find the percentiles.
birth_data2 <- birth_data%>%
                filter(plurality == 2)
birth_data3 <- birth_data%>%
                filter(plurality == 3)
birth_data4 <- birth_data%>%
                filter(plurality == 4)
birth_data5 <- birth_data%>%
                filter(plurality == 5)
# percentile

#Percentile for plurality 1
percentile_plu1 <- quantile(birth_data1$weight_pounds,probs = seq(0,1,0.25),na.rm = TRUE, names = TRUE)
percentile_plu1
```

```
        0%        25%        50%        75%       100%
 0.5004493  6.6866204  7.4383967  8.1791499 17.9897206
```

```r
#Percentile for plurality 2
percentile_plu2 <- quantile(birth_data2$weight_pounds,probs = seq(0,1,0.25),na.rm = TRUE, names = TRUE)
percentile_plu2
```

```
        0%        25%        50%        75%       100%
 0.5004493  4.4379053  5.4079393  6.1883757 13.4658350
```

```r
#Percentile for plurality 3
percentile_plu3 <- quantile(birth_data3$weight_pounds,probs = seq(0,1,0.25),na.rm = TRUE, names = TRUE)
percentile_plu3
```

```
       0%       25%       50%       75%      100%
0.5004493 2.8748279 3.8117925 4.6567141 8.9882464
```

```r
#Percentile for plurality 4
percentile_plu4 <- quantile(birth_data4$weight_pounds,probs = seq(0,1,0.25),na.rm = TRUE, names = TRUE)
percentile_plu4
```

```
        0%        25%        50%        75%       100%
 0.6613868  1.8088929  2.6874350  3.5455843 10.3749540
```

```r
#Percentile for plurality 5
percentile_plu5 <- quantile(birth_data5$weight_pounds,probs = seq(0,1,0.25),na.rm = TRUE, names = TRUE)
percentile_plu5
```

```
       0%       25%       50%       75%      100%
0.6856376 1.0262518 2.1076192 3.0396234 3.6243996
```

Question 5

Provide an example case in which these two percentile rankings in Question 3 and Question 4 (gender vs plurality) would be quite similar. Provide another example case in which these two percentile rankings would be quite different.

```r
perc_male
```

```
        0%        25%        50%        75%       100% 
 0.5004493  6.7505545  7.5001262  8.2717441 17.9897206 
```

```r
perc_female
```

```
        0%        25%        50%        75%       100% 
 0.5004493  6.4992275  7.2510038  7.9983709 17.1453501 
```

```r
percentile_plu1 #similar rankings
```

```
        0%        25%        50%        75%       100% 
 0.5004493  6.6866204  7.4383967  8.1791499 17.9897206 
```

```r
percentile_plu4 #different rankings
```

```
        0%        25%        50%        75%       100% 
 0.6613868  1.8088929  2.6874350  3.5455843 10.3749540 
```

One example for when the two percentile rankings in Question 3 and 4 would be quite similar is the percentile ranking of male/female versus the percentile ranking for plurality of 1. There is a lot of commonality between percentile values, the numbers a extremely similar.

One example for when the two percentile rankings in Question 3 and 4 would be quite different is the percentile ranking of male/female versus the percentile ranking for plurality of 4. The values are extremely different, with the percentile values for plurality of 4 being extremely low, especially in the median range.

Question 6

Agree or disagree with this claim. If you agree, provide a rationale for why it is correct. If you disagree, provide a counter-example that reveals the error in its thinking:

"If these two percentile rankings are very different from one another, we should suspect that the baby in question is more likely to be a twin/triplet/etc., than a single-birth."

```r
perc_male
```

```
        0%        25%        50%        75%       100% 
 0.5004493  6.7505545  7.5001262  8.2717441 17.9897206 
```

```r
perc_female
```

```
        0%        25%        50%        75%       100% 
 0.5004493  6.4992275  7.2510038  7.9983709 17.1453501 
```

```r
#comparing percentile rankings
percentile_plu1
```

```
        0%        25%        50%        75%       100% 
 0.5004493  6.6866204  7.4383967  8.1791499 17.9897206 
```

```
percentile_plu2
```

```
        0%        25%        50%        75%       100% 
 0.5004493  4.4379053  5.4079393  6.1883757 13.4658350 
```

```
percentile_plu3
```

```
       0%       25%       50%       75%      100% 
0.5004493 2.8748279 3.8117925 4.6567141 8.9882464 
```

I disagree with this claim. Seeing as the plurality 1 percentile values are more closely related to the male/female percentile values, it is apparent that the baby in question is more likely to be a single-birth rather than a twin/triplet/etc. Whether the baby is male or female, a single birth plurality percentile values will always be within the same range since one baby can be either gender.

Some of the variables have missing values, and these may be related to different data collection choices during different years. For example, possibly plurality wasn't recorded during some years, or state of birth wasn't recorded during some years. In this exercise we investigate using some `dplyr` functions. Hint: The `group_by` and `summarize` functions will help.

Question 7

Count the number of missing values in each variable in the data frame.

Is this question asking for the total missing values in each variable or missing values for each variable per year?

```r
#Total missing values in each variable
sum(is.na(birth_data$state))
```

```
[1] 135937
```

```r
sum(is.na(birth_data$month))
```

```
[1] 0
```

```r
sum(is.na(birth_data$year))
```

```
[1] 0
```

```r
sum(is.na(birth_data$weight_pounds))
```

```
[1] 1660
```

```r
sum(is.na(birth_data$is_male))
```

```
[1] 0
```

```r
sum(is.na(birth_data$mother_age))
```

```
[1] 0
```

```r
sum(is.na(birth_data$child_race))
```

```
[1] 201636
```

```r
sum(is.na(birth_data$plurality))
```

```
[1] 29088
```

```r
#Missing values for each variable per year
birth_data %>%
  group_by(year) %>%
  summarize(states_na = sum(is.na(state)),child_race_na = sum(is.na(child_race)), month_na = sum(is.na(m
```

```
# A tibble: 40 x 8
    year states_na child_race_na month_na weight_na is_male_na age_na plu_na
 * <int>     <int>         <int>    <int>     <int>      <int>  <int>  <int>
 1  1969         0             0        0        96          0      0  14280
 2  1970         0             0        0        73          0      0  14808
 3  1971         0             0        0        40          0      0      0
 4  1972         0             0        0        40          0      0      0
 5  1973         0             0        0        35          0      0      0
 6  1974         0             0        0        48          0      0      0
 7  1975         0             0        0        56          0      0      0
 8  1976         0             0        0        48          0      0      0
 9  1977         0             0        0        44          0      0      0
10  1978         0             0        0        50          0      0      0
# ... with 30 more rows
```

Question 8

Use `group_by` and `summarize` to count the number of missing values of the two variables, `state` and `child_race`, for each year, and to also count the total number of observations per year.

Are there particular years when these two variables are either not available, or of limited availability?

```r
birth_data %>%
  group_by(year) %>%
  summarize(states_na = sum(is.na(state)),child_race_na = sum(is.na(child_race)), total_obs = length(sta
```

```
# A tibble: 40 x 4
    year states_na child_race_na total_obs
 * <int>     <int>         <int>     <int>
 1  1969         0             0     14280
 2  1970         0             0     14808
 3  1971         0             0     14209
 4  1972         0             0     14106
```

```
 5  1973         0             0     14840
 6  1974         0             0     16432
 7  1975         0             0     18194
 8  1976         0             0     19537
 9  1977         0             0     22036
10  1978         0             0     23064
# ... with 30 more rows
```

It can be seen that in the years between 1969 and 2002, the values for state and child_race are not available. The later years, 2005 and beyond seem to have values for each state and child_race. Due to the earlier years not having values, it could be assumed that birth data may have been lost.

Question 9

Create the following data frame which gives the counts, the mean weight of babies and the mean age of mothers for the six levels of `plurality`. Comment on what you notice about the relationship of plurality and birth weight, and the relationship of plurality and age of the mother.

```
birth_data %>%
  group_by(plurality) %>%
  summarize(count = n(),mean_weight = mean(weight_pounds,na.rm = TRUE),mean_age = mean(mother_age,na.rm
```

```
# A tibble: 6 x 4
  plurality   count mean_weight mean_age
*     <int>   <int>       <dbl>    <dbl>
1         1 1046856        7.37     26.3
2         2   26582        5.22     28.1
3         3    1018        3.74     30.7
4         4      75        2.81     31.3
5         5      10        2.05     30.9
6        NA   29088        7.21     24.6
```

The relationship between plurality and birth weight is inversely proportional. As plurality increases, birth weight decreases. Whereas, the relationship between plurality and age of the mother is somewhat directly proportional. As plurality increases, the mother's age seems to be within the same range or a bit older. It can be said the older aged mothers tend to have more kids.

Question 10

Create a data frame which gives the counts, the mean weight of babies and the mean age of mothers for each combination of the four levels of `state_color` and the two levels of `is_male`.

```
birth_data %>%
  group_by(state_color) %>%
  summarize(count = n(),mean_weight = mean(weight_pounds,na.rm = TRUE),mean_age = mean(mother_age,na.rm
```

```
# A tibble: 4 x 4
  state_color  count mean_weight mean_age
* <chr>        <int>       <dbl>    <dbl>
1 blue        469298        7.37     26.8
2 purple      193152        7.30     25.9
3 red         305242        7.28     25.3
4 <NA>        135937        7.19     27.4
```

```r
birth_data %>%
  group_by(is_male) %>%
  summarize(count = n(),mean_weight = mean(weight_pounds,na.rm = TRUE),mean_age = mean(mother_age,na.rm
```

```
# A tibble: 2 x 4
  is_male  count mean_weight mean_age
* <lgl>    <int>       <dbl>    <dbl>
1 FALSE   537249        7.18     26.3
2 TRUE    566380        7.44     26.3
```

Essential details

Deadline and submission

The deadline to submit Homework 1 is **11:00pm on Saturday, February 13th.** This is a individual assignment. Submit your work by uploading your RMD and HTML/PDF files through D2L. Kindly double check your submission to note whether the everything is displayed in the uploaded version of the output in D2L or not. If submitting HTML outputs, please zip the files for submission. Late work will not be accepted except under certain extraordinary circumstances.

Help

- Post general questions in the Teams HW 1 channel. If you are trying to get help on a code error, explain your error in detail
- Feel free to visit us in during our virtual office hours or make an appointment.
- Communicate with your classmates, but do not share snippets of code.
- The instructional team will not answer any questions within the first 24 hours of this homework being assigned, and we will not answer any questions after 6 P.M of the due date.

Academic integrity

This is an individual assignment.You may discuss ideas, how to debug code, and how to approach a problem with your classmates in the discussion board forum. You may not copy-and-paste another's code from this class. As a reminder, below is the policy on sharing and using other's code.

> Similar reproducible examples (reprex) exist online that will help you answer many of the questions posed on group assignments, and homework assignments. Use of these resources is allowed unless it is written explicitly on the assignment. You must always cite any code you copy or use as inspiration. Copied code without citation is plagiarism and will result in a 0 for the assignment.

Grading

You must use R Markdown. Formatting is at your discretion but is graded. Use the in-class assignments and resources available online for inspiration. Another useful resource for R Markdown formatting is available at: https://holtzy.github.io/Pimp-my-rmd/

Topic	**Points**
Questions 1-4 (Sec 1) and 1-10 (Sec 2)	70
R Markdown formatting	5
Communication of results	10
Rmd file compilation	5
Code style and named code chunks	10

Total|100