STA 444 Exercise 4

Richard McCormick

2023-09-11

1. The dataset ChickWeight tracks the weights of 48 baby chickens (chicks) feed four different diets. Feel free to complete all parts of the exercise in a single R pipeline at the end of the problem.

a. Load the dataset using

```
data("ChickWeight")
```

b. Look at the help files for the description of the columns.

```
?ChickWeight
```

```
## starting httpd help server ... done
```

c. Remove all the observations except for observations from day 10 or day 20. The tough part in this instruction is distinguishing between "and" and "or." Obviously there are no observations that occur from both day 10 AND day 20. Google 'R logical operators' to get an introduction to those, but the short answer is that and is & and or is |.

```
observedChickWeight = ChickWeight[ChickWeight$Time==10 | ChickWeight$Time==20,]

summary(observedChickWeight)
```

```
##      weight           Time           Chick    Diet
##  Min.   : 51.0   Min.   :10.00   13     : 2   1:36
##  1st Qu.:104.5   1st Qu.:10.00   9      : 2   2:20
##  Median :128.0   Median :10.00   20     : 2   3:20
##  Mean   :157.2   Mean   :14.84   10     : 2   4:19
##  3rd Qu.:199.0   3rd Qu.:20.00   8      : 2
##  Max.   :361.0   Max.   :20.00   17     : 2
##                                  (Other):83
```

d. Calculate the mean and standard deviation of the chick weights for each diet group on days 10 and 20.

```
observedChickWeight %>%
  group_by(Time,Diet) %>%
  summarize(mean.Weight=mean(weight),
            stddev.Weight=sd(weight))
```

```
## `summarise()` has grouped output by 'Time'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 8 x 4
## # Groups:   Time [2]
##    Time Diet  mean.Weight stddev.Weight
##   <dbl> <fct>       <dbl>         <dbl>
## 1    10 1            93.1          22.5
## 2    10 2           108.           24.3
## 3    10 3           117.           20.2
## 4    10 4           126            11.4
## 5    20 1           170.           55.4
## 6    20 2           206.           70.3
## 7    20 3           259.           65.2
## 8    20 4           234.           37.6
```

2. The OpenIntro textbook on statistics includes a data set on body dimensions. Instead of creating an R chunk for each step of this problem, create a single R pipeline that performs each of the following tasks.

a. Load the file using

```
Body <- read.csv('http://www.openintro.org/stat/data/bdims.csv')
```

b. The column sex is coded as a 1 if the individual is male and 0 if female. This is a non-intuitive labeling system. Create a new column sex.MF that uses labels Male and Female. Use this column for the rest of the problem. Hint: The ifelse() command will be very convenient here. It functions similarly to the same command in Excel.

```
Body <- Body %>%
  mutate(
    MF = if_else(sex == '1', 'M', 'F')
  )
```

c. The columns wgt and hgt measure weight and height in kilograms and centimeters (respectively). Use these to calculate the Body Mass Index (BMI) for each individual where

$$BMI = \frac{Weight(kg)}{[Height(m)]^2}$$

```
Body <- Body %>%
  mutate(
    BMI = wgt / ((hgt/100)^2)
  )
```

d. Double check that your calculated BMI column is correct by examining the summary statistics of the column (e.g. summary(Body)). BMI values should be between 18 to 40 or so. Did you make an error in your calculation?

```
summary(Body)
```

```
##      bia.di          bii.di          bit.di          che.de
##  Min.   :32.40   Min.   :18.70   Min.   :24.70   Min.   :14.30
##  1st Qu.:36.20   1st Qu.:26.50   1st Qu.:30.60   1st Qu.:17.30
##  Median :38.70   Median :28.00   Median :32.00   Median :19.00
##  Mean   :38.81   Mean   :27.83   Mean   :31.98   Mean   :19.23
##  3rd Qu.:41.15   3rd Qu.:29.25   3rd Qu.:33.35   3rd Qu.:20.90
```

```
##  Max.   :47.40   Max.   :34.70   Max.   :38.00   Max.   :27.50
##      che.di          elb.di          wri.di          kne.di
##  Min.   :22.20   Min.   : 9.90   Min.   : 8.10   Min.   :15.70
##  1st Qu.:25.65   1st Qu.:12.40   1st Qu.: 9.80   1st Qu.:17.90
##  Median :27.80   Median :13.30   Median :10.50   Median :18.70
##  Mean   :27.97   Mean   :13.39   Mean   :10.54   Mean   :18.81
##  3rd Qu.:29.95   3rd Qu.:14.40   3rd Qu.:11.20   3rd Qu.:19.60
##  Max.   :35.60   Max.   :16.70   Max.   :13.30   Max.   :24.30
##      ank.di          sho.gi          che.gi          wai.gi
##  Min.   : 9.90   Min.   : 85.90   Min.   : 72.60   Min.   : 57.90
##  1st Qu.:13.00   1st Qu.: 99.45   1st Qu.: 85.30   1st Qu.: 68.00
##  Median :13.80   Median :108.20   Median : 91.60   Median : 75.80
##  Mean   :13.86   Mean   :108.20   Mean   : 93.33   Mean   : 76.98
##  3rd Qu.:14.80   3rd Qu.:116.55   3rd Qu.:101.15   3rd Qu.: 84.50
##  Max.   :17.20   Max.   :134.80   Max.   :118.70   Max.   :113.20
##      nav.gi          hip.gi          thi.gi          bic.gi
##  Min.   : 64.00   Min.   : 78.80   Min.   :46.30   Min.   :22.40
##  1st Qu.: 78.85   1st Qu.: 92.00   1st Qu.:53.70   1st Qu.:27.60
##  Median : 84.60   Median : 96.00   Median :56.30   Median :31.00
##  Mean   : 85.65   Mean   : 96.68   Mean   :56.86   Mean   :31.17
##  3rd Qu.: 91.60   3rd Qu.:101.00   3rd Qu.:59.50   3rd Qu.:34.45
##  Max.   :121.10   Max.   :128.30   Max.   :75.70   Max.   :42.40
##      for.gi          kne.gi          cal.gi          ank.gi          wri.gi
##  Min.   :19.60   Min.   :29.00   Min.   :28.40   Min.   :16.40   Min.   :13.0
##  1st Qu.:23.60   1st Qu.:34.40   1st Qu.:34.10   1st Qu.:21.00   1st Qu.:15.0
##  Median :25.80   Median :36.00   Median :36.00   Median :22.00   Median :16.1
##  Mean   :25.94   Mean   :36.20   Mean   :36.08   Mean   :22.16   Mean   :16.1
##  3rd Qu.:28.40   3rd Qu.:37.95   3rd Qu.:38.00   3rd Qu.:23.30   3rd Qu.:17.1
##  Max.   :32.50   Max.   :49.00   Max.   :47.70   Max.   :29.30   Max.   :19.6
##       age             wgt             hgt             sex
##  Min.   :18.00   Min.   : 42.00   Min.   :147.2   Min.   :0.0000
##  1st Qu.:23.00   1st Qu.: 58.40   1st Qu.:163.8   1st Qu.:0.0000
##  Median :27.00   Median : 68.20   Median :170.3   Median :0.0000
##  Mean   :30.18   Mean   : 69.15   Mean   :171.1   Mean   :0.4872
##  3rd Qu.:36.00   3rd Qu.: 78.85   3rd Qu.:177.8   3rd Qu.:1.0000
##  Max.   :67.00   Max.   :116.40   Max.   :198.1   Max.   :1.0000
##       MF                BMI
##  Length:507         Min.   :16.88
##  Class :character   1st Qu.:20.96
##  Mode  :character   Median :23.16
##                     Mean   :23.46
##                     3rd Qu.:25.47
##                     Max.   :38.19
```

e. The function cut takes a vector of continuous numerical data and creates a factor based on your given cut-points.

```
# Define a continuous vector to convert to a factor
x <- 1:10

# divide range of x into three groups of equal length
cut(x, breaks=3)
```

```
##  [1] (0.991,4] (0.991,4] (0.991,4] (0.991,4] (4,7]     (4,7]     (4,7]
```

```
##  [8] (7,10]    (7,10]    (7,10]
## Levels: (0.991,4] (4,7] (7,10]
```

```
# divide x into four groups, where I specify all 5 break points
cut(x, breaks = c(0, 2.5, 5.0, 7.5, 10))
```

```
##  [1] (0,2.5]  (0,2.5]  (2.5,5]  (2.5,5]  (2.5,5]  (5,7.5]  (5,7.5]  (7.5,10]
##  [9] (7.5,10] (7.5,10]
## Levels: (0,2.5] (2.5,5] (5,7.5] (7.5,10]
```

```
# (0,2.5] (2.5,5] means 2.5 is included in first group
# right=FALSE changes this to make 2.5 included in the second

# divide x into 3 groups, but give them a nicer
# set of group names
cut(x, breaks=3, labels=c('Low','Medium','High'))
```

```
##  [1] Low    Low    Low    Low    Medium Medium Medium High   High   High
## Levels: Low Medium High
```

Create a new column of in the data frame that divides the age into decades (10-19, 20-29, 30-39, etc). Notice the oldest person in the study is 67.

```
Body <- Body %>%
  mutate( Age.Grp = cut(age,
                        breaks=c(10,20,30,40,50,60,70),
                        right=FALSE))
```

f. Find the average BMI for each Sex.MF by Age.Grp combination.

```
Body %>%
  group_by(MF, Age.Grp) %>%
  summarize(mean.BMI = mean(BMI))
```

```
## `summarise()` has grouped output by 'MF'. You can override using the `.groups`
## argument.
```

```
## # A tibble: 12 x 3
## # Groups:   MF [2]
##    MF    Age.Grp mean.BMI
##    <chr> <fct>      <dbl>
##  1 F     [10,20)     21.8
##  2 F     [20,30)     21.8
##  3 F     [30,40)     22.5
##  4 F     [40,50)     24.3
##  5 F     [50,60)     22.7
##  6 F     [60,70)     23.7
##  7 M     [10,20)     25.5
##  8 M     [20,30)     24.2
##  9 M     [30,40)     24.9
## 10 M     [40,50)     26.4
## 11 M     [50,60)     24.8
## 12 M     [60,70)     23.9
```