PHW251 Problem Set 4

Teaching Team

2021

Due date: Monday, September 27

For this problem set you will tidy up a dataset of 500 individuals. We also want to calculate each individual's BMI and appropriately categorize them.

Load your data (500_Person_Gender_Height_Weight.csv):

Question 1

Clean the column headers to be all lower case, have no spaces, and rename "Location information" to location.

```
# find spaces and replace with -
bmi_1 <- rename_with(bmi, ~ tolower(gsub(" ","_", .x, fixed=TRUE)))
# lower case all column names
bmi_1 <- rename(bmi_1, location = location_information)
```

Question 2

Create a new variable that calculates BMI for each individual.

You will need to navigate the different system of measurements (metric vs imperial). Only the United States is using imperial.

- BMI calculation and conversions:
  - metric: $BMI = weight(kg)/[height(m)]^2$
  - imperial: $BMI = 703 * weight(lbs)/[height(in)]^2$
  - 1 foot = 12 inches
  - 1 cm = 0.01 meter

Although there's many ways you can accomplish this task, we want you to use an if_else() to calculate BMI with the appropriate formula based on each person's location.

```
bmi_2 <- bmi_1 %>%
  mutate(bmi = if_else(
    location %in% c("New York", "Colorado", "Hawaii"), # conditional statement
    (703 * weight)/(height * 12)^2, # if true, calculate imperial
    (weight/(height/100)^2))) # if false, calculate metric

head(bmi_2)
```

```
## # A tibble: 6 x 5
##   location       gender height weight   bmi
##   <chr>          <chr>   <dbl>  <dbl> <dbl>
## 1 New York       Male     5.71   212.  31.7
## 2 United Kingdom Male   189       87   24.4
## 3 New York       Female   6.07   243.  32.1
## 4 Taiwan         Female 195      104   27.4
## 5 Taiwan         Male   149       61   27.5
## 6 Taiwan         Male   189      104   29.1
```

Question 3

Create a new variable that categorizes BMI with case_when():

- Underweight: BMI below 18.5
- Normal: 18.5-24.9
- Overweight: 25.0-29.9
- Obese: 30.0 and Above

```
# need to be careful with how we separate these values
# for example: bmi > 30 would leave out 30

bmi_3 <- bmi_2 %>%
  mutate(bmi_cat = case_when(
    bmi >= 30.0    ~ "Obese", # starting from largest value
    bmi >= 25    ~ "Overweight", # step down
    bmi >= 18.5  ~ "Normal", # step again
    TRUE            ~ "Underweight")) # default to underweight, otherwise

# we could also have performed this operation in the reverse

bmi_3 <- bmi_2 %>%
  mutate(bmi_cat = case_when(
    bmi < 18.5 ~ "Underweight",
    bmi < 25 ~ "Normal",
    bmi < 30 ~ "Overweight",
    TRUE ~ "Obese"))
```

Could we have used if_else()?

Yup, we could have! It's a matter of preference and how you are approach the problem. With an if_else() statement we would have to nest many of them, which could become cumbersome.

Question 4

Arrange your data first by location and then by descending order of BMI.

```
bmi_4 <- bmi_3 %>%
  # first arrange by location and then by bmi, descending
  arrange(location, desc(bmi))
```

Question 5

Use a dplyr method to remove the height, weight, and BMI columns from your data.

```
bmi_5 <- bmi_4 %>% select(-c(height, weight, bmi))
str(bmi_5)
```

```
## tibble [500 x 3] (S3: tbl_df/tbl/data.frame)
##  $ location: chr [1:500] "Colorado" "Colorado" "Colorado" "Colorado" ...
##  $ gender  : chr [1:500] "Female" "Female" "Male" "Female" ...
##  $ bmi_cat : chr [1:500] "Obese" "Obese" "Obese" "Obese" ...
```

Optional Challenge

Perform all the actions in this problem set with one dpylr call.

```
# although you can do this, you will want to test each step out individually
# to ensure you aren't making any errors!
bmi_all <- bmi %>%
  rename_with(~ tolower(gsub(" ","_", .x, fixed=TRUE))) %>%
  rename(location = location_information) %>%
  mutate(bmi = if_else(location %in% c("New York", "Colorado", "Hawaii"),
                       (703 * weight)/(height * 12)^2,
                       weight/(height/100)^2),
         bmi_cat = case_when(bmi > 30    ~ "Obese",
                             bmi > 25    ~ "Overweight",
                             bmi > 18.4  ~ "Normal",
                             TRUE        ~ "Underweight")) %>%
  arrange(location, desc(bmi)) %>%
  select(-c(height, weight, bmi))


head(bmi_all)
```

```
## # A tibble: 6 x 3
##   location gender bmi_cat
##   <chr>    <chr>  <chr>
## 1 Colorado Female Obese
## 2 Colorado Female Obese
## 3 Colorado Male   Obese
## 4 Colorado Female Obese
## 5 Colorado Female Obese
## 6 Colorado Male   Obese
```