Assignment 3

McKenzie Kozma

9/18/2019

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## -- Conflicts ------------------------------------------------------------------------------------ ti
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
data <- read.csv("titanic.csv")
```

REDO OF ASSIGNMENT 1 #13 - 23 (i used dplyr in A1 so the following will be done in base r)

13. Calculate the mean age of female passengers

```
mean(data$Age[data$Sex == "female"], na.rm = TRUE)
```

```
## [1] 27.91571
```

14. Calculate the median fare of the passengers in Class 1

```
median(data$Fare[data$Pclass == 1], na.rm = TRUE)
```

```
## [1] 60.2875
```

15. Calculate the median fare of the female passengers that are not in Class 1

```
median(data$Fare[data$Sex == 'female' & data$Pclass != 1], na.rm = TRUE)
```

```
## [1] 14.45625
```

16. Calculate the median age of survived passengers who are female and Class 1 or Class 2

```
median(data$Age[data$Survived == 1 & data$Sex == 'female' & data$Pclass %in% c(1,2)], na.rm = TRUE)
```

```
## [1] 31
```

17. Calculate the mean fare of female teenagers survived passengers

```r
mean(data$Fare[data$Sex == 'female' & data$Survived == 1 & data$Age < 20 & data$Age > 12], na.rm = TRUE)
```

```
## [1] 49.17966
```

18. Calculate the mean fare of female teenagers survived passengers for each class

```r
by(data$Fare[data$Sex == 'female' & data$Survived == 1 & data$Age > 12 & data$Age < 20],
   data$Pclass[data$Sex == 'female' & data$Survived == 1 & data$Age > 12 & data$Age < 20], FUN = mean)
```

```
## data$Pclass[data$Sex == "female" & data$Survived == 1 & data$Age > : 1
## [1] 107.5407
## ------------------------------------------------------------
## data$Pclass[data$Sex == "female" & data$Survived == 1 & data$Age > : 2
## [1] 20.00885
## ------------------------------------------------------------
## data$Pclass[data$Sex == "female" & data$Survived == 1 & data$Age > : 3
## [1] 8.769885
```

19. Calculate the ratio of Survived and not Survived for passengers who are who pays more than the average fare

```r
nobs <- nrow(data[data$Fare > mean(data$Fare),])
survive <- sum(data$Survived[data$Fare > mean(data$Fare)])

survive/(nobs - survive)
```

```
## [1] 1.482353
```

20. Add column that standardizes the fare (subtract the mean and divide by standard deviation) and name it sfare

```r
sfare = (data$Fare - mean(data$Fare)) / sd(data$Fare)
data2 <- cbind(data, sfare)
```

21. Add categorical variable named cfare that takes value cheap for passengers paying less the average fare and takes value expensive for passengers paying more than the average fare.

```r
cfare = ifelse(data2$Fare < mean(data2$Fare), "cheap", "expensive")
data3 <- cbind(data, cfare)
```

22. Add categorical variable named cage that takes value 0 for age 0-10, 1 for age 10-20, 2 for age 20-30, and so on

```r
cage = trunc(data3$Age / 10) * 10
data4 <- cbind(data3, cage)
```

23. Show the frequency of Ports of Embarkation. It appears that there are two missing values in the Embarked variable. Assign the most frequent port to the missing ports. Hint: Use the levels function to modify the categories of categorical variables.

```r
summary(data$Embarked)
```

```
##       C   Q   S
##   2 168  77 644
```

```r
levels(data$Embarked) <- c("S", "C", "Q", "S")
```

REDO OF ASSIGNMENT 2 #4, 5, 11, 12, AND 13

```r
library(readxl)
c2015 <- read_xlsx("c2015.xlsx")

dim(c2015)
```

```
## [1] 80587    28
```

```r
set.seed(2019)
samp <- sample_n(c2015, 1000)
glimpse(samp)
```

```
## Observations: 1,000
## Variables: 28
## $ STATE    <chr> "New Jersey", "Arizona", "Tennessee", "Minnesota", "M...
## $ ST_CASE  <dbl> 340336, 40327, 470789, 270119, 290576, 62865, 330095,...
## $ VEH_NO   <dbl> 1, 1, 1, 2, 1, 1, 0, 0, 2, 5, 1, 2, 1, 0, 1, 1, 2, 1,...
## $ PER_NO   <dbl> 1, 1, 1, 4, 1, 1, 1, 1, 4, 1, 1, 1, 5, 1, 1, 2, 1, 1,...
## $ COUNTY   <dbl> 27, 13, 163, 59, 201, 19, 15, 127, 13, 115, 29, 141, ...
## $ DAY      <dbl> 19, 7, 2, 16, 2, 6, 3, 30, 17, 30, 19, 12, 9, 30, 9, ...
## $ MONTH    <chr> "September", "May", "December", "May", "October", "Ju...
## $ HOUR     <dbl> 3, 22, 8, 21, 15, 15, 14, 20, 7, 14, 14, 17, 18, 6, 4...
## $ MINUTE   <dbl> 17, 15, 26, 59, 38, 20, 32, 20, 41, 36, 15, 50, 55, 4...
## $ AGE      <chr> "Unknown", "47", "23", "15", "55", "56", "26", "63", ...
## $ SEX      <chr> "Unknown", "Female", "Male", "Female", "Male", "Male"...
## $ PER_TYP  <chr> "Driver of a Motor Vehicle In-Transport", "Driver of ...
## $ INJ_SEV  <chr> "Unknown", "No Apparent Injury (O)", "Unknown", "Susp...
## $ SEAT_POS <chr> "Front Seat, Left Side", "Front Seat, Left Side", "Fr...
## $ DRINKING <chr> "Not Reported", "No (Alcohol Not Involved)", "Unknown...
## $ YEAR     <dbl> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015,...
## $ MAN_COLL <chr> "Not a Collision with Motor Vehicle In-Transport", "N...
## $ OWNER    <chr> "Unknown", "Driver (in this crash) Not Registered Own...
## $ MOD_YEAR <chr> "Unknown", "2003", "1994", "2011", "2000", "2013", NA...
## $ TRAV_SP  <chr> "Unknown", "048 MPH", "Not Rep", "055 MPH", "055 MPH"...
## $ DEFORMED <chr> "Unknown", "Functional Damage", "Minor Damage", "Disa...
## $ DAY_WEEK <chr> "Saturday", "Thursday", "Wednesday", "Saturday", "Fri...
## $ ROUTE    <chr> "State Highway", "Local Street", "County Road", "Stat...
## $ LATITUDE <dbl> 40.95270, 33.41048, 36.57834, 45.42841, 37.13481, 36....
## $ LONGITUD <dbl> -74.59644, -112.06459, -82.27889, -93.36788, -89.5946...
## $ HARM_EV  <chr> "Pedestrian", "Pedestrian", "Pedalcyclist", "Motor Ve...
## $ LGT_COND <chr> "Dark - Not Lighted", "Dark - Lighted", "Dark - Not L...
## $ WEATHER  <chr> "Clear", "Clear", "Clear", "Rain", "Cloud", "Clear", ...
```

```
samp$YEAR <- NULL
samp$TRAV_SP <- as.numeric(str_remove(samp$TRAV_SP, "MPH"))
```

```
## Warning: NAs introduced by coercion
```

```
samp %>%
  group_by(INJ_SEV == "No Apparent Injury (O)") %>%
  summarise(speed = mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 2 x 2
##   `INJ_SEV == "No Apparent Injury (O)"` speed
##   <lgl>                                 <dbl>
## 1 FALSE                                  53.1
## 2 TRUE                                   44.6
```

```
samp2 <- samp %>%
  filter(SEAT_POS != "Front Seat, Left Side")

samp2 %>%
  group_by(SEX) %>%
  summarise(speed = mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 4 x 2
##   SEX     speed
##   <chr>   <dbl>
## 1 Female   52.1
## 2 Male     52.3
## 3 Not Rep NaN
## 4 Unknown NaN
```

```
samp2 %>%
  group_by(DRINKING) %>%
  summarise(speed = mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 4 x 2
##   DRINKING                    speed
##   <chr>                       <dbl>
## 1 No (Alcohol Not Involved)    42.6
## 2 Not Reported                 52.6
## 3 Unknown (Police Reported)    45
## 4 Yes (Alcohol Involved)       73.5
```

3. Calculate the travel speed (TRAV_SP variable) by day. Compare the travel speed of the first 5 days and the last 5 days of months.

```
samp2 %>% filter(DAY %in% c(1:5, 26:30)) %>%
  group_by(DAY %in% c(1:5)) %>%
  summarise(speed = mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 2 x 2
##   `DAY %in% c(1:5)` speed
##   <lgl>             <dbl>
## 1 FALSE              55.4
## 2 TRUE               47.9
```

```
#drive faster at the end of the month
```

4. Calculate the travel speed (TRAV_SP variable) by day of the week. Compare the travel speed of the weekdays and weekends.

```
samp2 %>% group_by(DAY_WEEK %in% c("Sunday", "Saturday")) %>%
  summarise(speed = mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 2 x 2
##   `DAY_WEEK %in% c("Sunday", "Saturday")` speed
##   <lgl>                                   <dbl>
## 1 FALSE                                    50.5
## 2 TRUE                                     54.5
```

```
#faster on weekends
```

5. Find the top 5 states with greatest travel speed.

```
samp2 %>% group_by(STATE) %>%
  summarise(SPEED = mean(TRAV_SP, na.rm = TRUE)) %>%
  top_n(5, SPEED)
```

```
## # A tibble: 5 x 2
##   STATE     SPEED
##   <chr>     <dbl>
## 1 Kentucky   80.5
## 2 Missouri   70
## 3 Nevada     79
## 4 Texas      80
## 5 Wisconsin  70
```

6. Rank the travel speed by MONTH.

```
samp2 %>% group_by(MONTH) %>%
  summarise(SPEED = mean(TRAV_SP, na.rm = TRUE)) %>%
  arrange(-SPEED)
```

```
## # A tibble: 12 x 2
##    MONTH     SPEED
##    <chr>     <dbl>
##  1 December   66.5
##  2 April      58
##  3 June       55.7
##  4 November   54
```

```
##  5 March      53.9
##  6 September  53.8
##  7 February   52.1
##  8 August     51.1
##  9 October    50.2
## 10 July       49.2
## 11 January    42.6
## 12 May        42.5
```

7. Find the average speed of teenagers in December.

```
samp2 %>% filter(MONTH == "December" & AGE %in% c(13:19)) %>%
  summarise(SPEED = mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   SPEED
##   <dbl>
## 1    80
```

8. Find the month that female drivers drive fastest on average.

```
samp2 %>% filter(SEX == "Female") %>%
  group_by(MONTH) %>%
  summarise(SPEED = mean(TRAV_SP, na.rm = TRUE)) %>%
  top_n(1, SPEED)
```

```
## # A tibble: 1 x 2
##   MONTH    SPEED
##   <chr>    <dbl>
## 1 November    75
```

9. Find the month that male driver drive slowest on average.

```
samp2 %>% filter(SEX == "Male") %>%
  group_by(MONTH) %>%
  summarise(SPEED = mean(TRAV_SP, na.rm = TRUE)) %>%
  top_n(-1, SPEED)
```

```
## # A tibble: 1 x 2
##   MONTH   SPEED
##   <chr>   <dbl>
## 1 January    20
```

10. Create a new column containing information about the season of the accidents. Compare the percentage of Fatal Injury by seasons.

```
samp2 %>%
  mutate(SEASON = ifelse(MONTH %in% c("December", "January", "February"), "Winter",
                         ifelse(MONTH %in% c("March", "April", "May"), "Spring",
                                ifelse(MONTH %in% c("June", "July", "August"), "Summer",
                                       ifelse(MONTH %in% c("September", "October", "November"), "Fall",
  group_by(SEASON) %>%
  summarise(prop_fatal = (sum(INJ_SEV == "Fatal Injury (K)"))/n())
```

```
## # A tibble: 4 x 2
##   SEASON prop_fatal
##   <chr>       <dbl>
## 1 Fall        0.467
## 2 Spring      0.37
## 3 Summer      0.432
## 4 Winter      0.429
```

11. Compare the percentage of fatal injuries for different type of deformations (DEFORMED variable)

```
samp2 %>% group_by(DEFORMED) %>%
  summarise(prop_fatal = (sum(INJ_SEV == "Fatal Injury (K)"))/n()) %>%
  arrange(-prop_fatal)
```

```
## # A tibble: 7 x 2
##   DEFORMED          prop_fatal
##   <chr>                  <dbl>
## 1 <NA>                  0.904
## 2 Disabling Damage      0.348
## 3 No Damage             0.333
## 4 Not Reported          0.190
## 5 Minor Damage          0.107
## 6 Functional Damage     0.0345
## 7 Unknown               0
```