R- Assignment3

Payton Kim

9/23/2019

1. Read the titanic data set as a tibble. Redo questions 13 to 23 in the Assignment 1 using dplyr

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages ------------------------ tidyverse 1.2.1 --

## v ggplot2 3.2.1     v readr   1.3.1
## v tibble  2.1.3     v purrr   0.3.2
## v tidyr   1.0.0     v stringr 1.4.0
## v ggplot2 3.2.1     v forcats 0.4.0

## -- Conflicts --------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
titanic = read_csv("C:\\Users\\student\\Downloads\\titanic.csv")
```

```
## Parsed with column specification:
## cols(
##   PassengerId = col_double(),
##   Survived = col_double(),
##   Pclass = col_double(),
##   Name = col_character(),
##   Sex = col_character(),
##   Age = col_double(),
##   SibSp = col_double(),
##   Parch = col_double(),
##   Ticket = col_character(),
##   Fare = col_double(),
##   Cabin = col_character(),
##   Embarked = col_character()
## )
```

```
str(titanic)
```

```
## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 891 obs. of  12 variables:
##  $ PassengerId: num  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : num  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : num  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : num  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : num  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  NA "C85" NA "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   PassengerId = col_double(),
##   ..   Survived = col_double(),
##   ..   Pclass = col_double(),
##   ..   Name = col_character(),
##   ..   Sex = col_character(),
##   ..   Age = col_double(),
##   ..   SibSp = col_double(),
##   ..   Parch = col_double(),
##   ..   Ticket = col_character(),
##   ..   Fare = col_double(),
##   ..   Cabin = col_character(),
##   ..   Embarked = col_character()
##   .. )
```

13. Calculate the mean age of female passengers

```
titanic %>%
  filter(Sex == "female") %>%
  summarize(mean(Age, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `mean(Age, na.rm = TRUE)`
##                       <dbl>
## 1                      27.9
```

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

```
titanic %>%
  filter(Pclass == 1) %>%
  summarize(median(Fare, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `median(Fare, na.rm = TRUE)`
##                          <dbl>
## 1                         60.3
```

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

```
titanic %>%
  filter(Sex == "female" & Pclass != 1) %>%
  summarize(median(Fare, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `median(Fare, na.rm = TRUE)`
##                          <dbl>
## 1                         14.5
```

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

```
titanic %>%
  filter(Sex == "female", Pclass != 3, Survived == 1) %>%
  summarize(median(Age, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `median(Age, na.rm = TRUE)`
##                         <dbl>
## 1                          31
```

17. Calculate the mean fare of female teenagers survived passengers

```
titanic %>%
  filter(Sex == "female", Age >= 13, Age <=19, Survived == 1) %>%
  summarize(mean(Fare, na.rm = 1))
```

```
## # A tibble: 1 x 1
##   `mean(Fare, na.rm = 1)`
##                     <dbl>
## 1                    49.2
```

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

```
titanic %>%
  filter(Sex == "female", Age >= 13, Age <=19, Survived == 1, Pclass == 1) %>%
  summarize(mean(Fare, na.rm = 1))
```

```
## # A tibble: 1 x 1
##   `mean(Fare, na.rm = 1)`
##                     <dbl>
## 1                    108.
```

```
titanic %>%
  filter(Sex == "female", Age >= 13, Age <=19, Survived == 1, Pclass == 2) %>%
  summarize(mean(Fare, na.rm = 1))
```

```
## # A tibble: 1 x 1
##   `mean(Fare, na.rm = 1)`
##                     <dbl>
## 1                    20.0
```

```
titanic %>%
  filter(Sex == "female", Age >= 13, Age <=19, Survived == 1, Pclass == 3) %>%
  summarize(mean(Fare, na.rm = 1))
```

```
## # A tibble: 1 x 1
##   `mean(Fare, na.rm = 1)`
##                     <dbl>
## 1                    8.77
```

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

```
titanic %>% filter(Fare>mean(Fare, na.rm = 1)) %>% group_by(Survived) %>% summarise(nn =n()) %>% mutate
```

```
## # A tibble: 2 x 3
##   Survived    nn  freq
##      <dbl> <int> <dbl>
## 1        0    85 0.403
## 2        1   126 0.597
```

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

```
titanic %>%
  mutate(sfare = (Fare - mean(Fare, na.rm = 1))/sd(Fare, na.rm = 1))
```

```
## # A tibble: 891 x 13
##    PassengerId Survived Pclass Name  Sex     Age SibSp Parch Ticket  Fare
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
##  1           1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
##  2           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
##  3           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
##  4           4        1      1 Futr~ fema~    35     1     0 113803 53.1
##  5           5        0      3 Alle~ male     35     0     0 373450  8.05
##  6           6        0      3 Mora~ male     NA     0     0 330877  8.46
##  7           7        0      1 McCa~ male     54     0     0 17463  51.9
##  8           8        0      3 Pals~ male      2     3     1 349909 21.1
##  9           9        1      3 John~ fema~    27     0     2 347742 11.1
## 10          10        1      2 Nass~ fema~    14     1     0 237736 30.1
## # ... with 881 more rows, and 3 more variables: Cabin <chr>,
## #   Embarked <chr>, sfare <dbl>
```

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.

```
titanic %>% mutate(cfare = ifelse(Fare>mean(Fare, na.rm = 1), "expensive", "cheap"))
```

```
## # A tibble: 891 x 13
##    PassengerId Survived Pclass Name  Sex     Age SibSp Parch Ticket  Fare
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
##  1           1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
##  2           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
##  3           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
##  4           4        1      1 Futr~ fema~    35     1     0 113803 53.1
##  5           5        0      3 Alle~ male     35     0     0 373450  8.05
##  6           6        0      3 Mora~ male     NA     0     0 330877  8.46
##  7           7        0      1 McCa~ male     54     0     0 17463  51.9
##  8           8        0      3 Pals~ male      2     3     1 349909 21.1
##  9           9        1      3 John~ fema~    27     0     2 347742 11.1
## 10          10        1      2 Nass~ fema~    14     1     0 237736 30.1
## # ... with 881 more rows, and 3 more variables: Cabin <chr>,
## #   Embarked <chr>, cfare <chr>
```

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

```
titanic %>% mutate(cage = cut(Age, breaks = c(0,10,20,30,40,50,60,70,80,90,Inf), labels = c(0,1,2,3,4,5
```

```
## # A tibble: 891 x 13
##    PassengerId Survived Pclass Name  Sex     Age SibSp Parch Ticket  Fare
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
##  1           1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
##  2           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
##  3           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
##  4           4        1      1 Futr~ fema~    35     1     0 113803 53.1
##  5           5        0      3 Alle~ male     35     0     0 373450  8.05
##  6           6        0      3 Mora~ male     NA     0     0 330877  8.46
##  7           7        0      1 McCa~ male     54     0     0 17463  51.9
##  8           8        0      3 Pals~ male      2     3     1 349909 21.1
##  9           9        1      3 John~ fema~    27     0     2 347742 11.1
## 10          10        1      2 Nass~ fema~    14     1     0 237736 30.1
## # ... with 881 more rows, and 3 more variables: Cabin <chr>,
## #   Embarked <chr>, cage <fct>
```

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.

```
titanic %>% group_by(Embarked) %>% count(Embarked)
```

```
## # A tibble: 4 x 2
## # Groups:   Embarked [4]
##   Embarked     n
##   <chr>    <int>
## 1 C          168
## 2 Q           77
## 3 S          644
## 4 <NA>         2
```

```
titanic %>% mutate(Embarked = replace_na(Embarked, "S")) %>% count(Embarked)
```

```
## # A tibble: 3 x 2
##   Embarked     n
##   <chr>    <int>
## 1 C          168
## 2 Q           77
## 3 S          646
```

2. Using Dplyr and in Assignment 2, redo 4 using sample_n function, redo 5 using glimpse, redo 11, 12 and 13. For 11, 12 and 13, you may want to use the combo group_by and summarise

3. Use dim function to check the dimension of the data. Since this data is quite big, a common practice is to randomly subset the data to analyze. Use sample function to create a new dataset that has a random 1000 observations from the original data. Use set.seed(2019) before using the sample function to set the seed for the randomness so that everyone in class is working with the same random subset of the data.

```
library(readxl)
c2015 <- read_excel("C:/Users/student/Downloads/c2015.xlsx")
set.seed(2019)
c <-  c2015 %>% sample_n(1000)
```

5. Use summary function to have a quick look at the data. You will notice there is one variable is actually a constant. Remove that variable from the data.

```
glimpse(c)
```

```
## 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", ...
```

```
cc = c %>%
  select(-"YEAR")
```

11. Compare the average speed of those who had “No Apprent Injury” and the rest. What do you observe?

```
library(stringr)
cc$TRAV_SP <- str_replace(cc$TRAV_SP, " MPH","")
cc$TRAV_SP <- str_replace(cc$TRAV_SP, "No Rep","")
cc$TRAV_SP <- str_replace(cc$TRAV_SP, "Unknown","")
cc$TRAV_SP <- as.numeric(cc$TRAV_SP)
```

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

```
cc = cc[!(is.na(cc$TRAV_SP)),]

cc %>%
  group_by(INJ_SEV) %>%
  summarize(mean(TRAV_SP))
```

```
## # A tibble: 7 x 2
##   INJ_SEV                     `mean(TRAV_SP)`
##   <chr>                                 <dbl>
## 1 Fatal Injury (K)                       55.6
## 2 Injured, Severity Unknown              35
## 3 No Apparent Injury (O)                 44.6
## 4 Possible Injury (C)                    43.1
## 5 Suspected Minor Injury(B)              52.3
## 6 Suspected Serious Injury(A)            55.4
## 7 Unknown                                35
```

```
  #No apparent injury had the lowest tavel speed
```

12. Use the SEAT_POS variable to filter the data so that there is only drivers in the dataset. Compare the average speed of man drivers and woman drivers. Comment on the results.

```
cc %>%
  filter(SEAT_POS == "Front Seat, Left Side") %>%
  group_by(SEX) %>%
  summarise(mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 3 x 2
##   SEX     `mean(TRAV_SP, na.rm = TRUE)`
##   <chr>                           <dbl>
## 1 Female                           46.1
## 2 Male                             51.7
## 3 Unknown                          36.7
```

```
#Female drivers were driving slower than the men drivers
```

13. Compare the average speed of drivers who drink and those who do not. Comment on the results.

```
cc %>%
  group_by(DRINKING) %>%
  summarize(mean(TRAV_SP, na.rm = 1))
```

```
## # A tibble: 4 x 2
##   DRINKING                   `mean(TRAV_SP, na.rm = 1)`
##   <chr>                                           <dbl>
## 1 No (Alcohol Not Involved)                        44.8
## 2 Not Reported                                     52.6
## 3 Unknown (Police Reported)                        53.6
## 4 Yes (Alcohol Involved)                           68.6
```

```
#Drivers with no alcohol involved were driving, on average, slower than those who did drink
```

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.

```
cc %>%
  group_by(DAY) %>%
  summarize(mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 31 x 2
##      DAY `mean(TRAV_SP, na.rm = TRUE)`
##    <dbl>                         <dbl>
##  1     1                          59.1
##  2     2                          55
##  3     3                          60.8
##  4     4                          40.9
##  5     5                          46.2
##  6     6                          47.3
##  7     7                          45.9
##  8     8                          52.5
##  9     9                          50.8
## 10    10                          50.4
## # ... with 21 more rows
```

```
cc %>%
  filter(DAY <= 5) %>%
  summarize(mean(TRAV_SP, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `mean(TRAV_SP, na.rm = TRUE)`
##                           <dbl>
## 1                          50.7
```

```r
cc %>% filter(DAY >= 26) %>%
  summarize(mean(TRAV_SP, na.rm = T))
```

```
## # A tibble: 1 x 1
##   `mean(TRAV_SP, na.rm = T)`
##                        <dbl>
## 1                       53.4
```

```r
# the travel speed of the first five days of the month is, on average, slower than the last five days
```

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

```r
cc %>%
  group_by(DAY_WEEK) %>%
  summarize(mean(TRAV_SP, na.rm = 1))
```

```
## # A tibble: 7 x 2
##   DAY_WEEK  `mean(TRAV_SP, na.rm = 1)`
##   <chr>                          <dbl>
## 1 Friday                          50.7
## 2 Monday                          48.6
## 3 Saturday                        53.3
## 4 Sunday                          55.8
## 5 Thursday                        50.8
## 6 Tuesday                         47.2
## 7 Wednesday                       44.7
```

```r
cc %>%
  filter(DAY_WEEK == c("Saturday", "Sunday")) %>%
  summarize(mean(TRAV_SP,na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `mean(TRAV_SP, na.rm = TRUE)`
##                           <dbl>
## 1                          52.3
```

```r
cc %>%
  filter(DAY_WEEK != c("Saturday", "Sunday")) %>%
  summarize(mean(TRAV_SP,na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `mean(TRAV_SP, na.rm = TRUE)`
##                           <dbl>
## 1                          50.5
```

```r
#The travel speed on weekends is, on average, higher than on weekdays
```

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

```
cc %>%
  group_by(STATE) %>%
  summarize(mean(TRAV_SP, na.rm = TRUE)) %>%
  top_n(5)
```

```
## Selecting by mean(TRAV_SP, na.rm = TRUE)

## # A tibble: 5 x 2
##   STATE        `mean(TRAV_SP, na.rm = TRUE)`
##   <chr>                                <dbl>
## 1 Kentucky                              65.4
## 2 Nevada                                73.5
## 3 North Dakota                          85
## 4 South Dakota                         107
## 5 Wyoming                               66.5
```

6. Rank the travel speed by MONTH.

```
cc %>%
  group_by(MONTH) %>%
  summarize(avgspeed = mean(TRAV_SP, na.rm = 1)) %>%
  arrange(desc(avgspeed))
```

```
## # A tibble: 12 x 2
##    MONTH     avgspeed
##    <chr>        <dbl>
##  1 April         59.3
##  2 December      59.0
##  3 September     54.7
##  4 June          53.4
##  5 October       52.5
##  6 November      52.5
##  7 August        48.9
##  8 May           48.3
##  9 February      46.4
## 10 March         45.4
## 11 January       45.2
## 12 July          44.9
```

7. Find the average speed of teenagers in December.

```
cc %>%
  filter(MONTH == "December", AGE >= 13, AGE <= 19) %>%
  summarize(mean(TRAV_SP, na.rm = 1))
```

```
## # A tibble: 1 x 1
##   `mean(TRAV_SP, na.rm = 1)`
##                        <dbl>
## 1                         80
```

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

```r
cc %>%
  filter(SEX == "Female") %>%
  group_by(MONTH) %>%
  summarize(avgspeed = mean(TRAV_SP, na.rm = TRUE)) %>%
  arrange(desc(avgspeed))
```

```
## # A tibble: 12 x 2
##    MONTH     avgspeed
##    <chr>        <dbl>
##  1 December      60.3
##  2 September     58.3
##  3 May           55.9
##  4 February      53.3
##  5 June          51
##  6 July          47.1
##  7 April         47
##  8 October       44.5
##  9 August        44.4
## 10 January       43.5
## 11 March         43.2
## 12 November      42.4
```

```r
#Women drive the fastest, on average, in December
```

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

```r
cc %>%
  filter(SEX == "Male") %>%
  group_by(MONTH) %>%
  summarize(avgspeed = mean(TRAV_SP, na.rm = 1)) %>%
  arrange(avgspeed)
```

```
## # A tibble: 12 x 2
##    MONTH     avgspeed
##    <chr>        <dbl>
##  1 February      38
##  2 May           43.4
##  3 July          44.8
##  4 March         46
##  5 January       46.1
##  6 September     53.9
##  7 August        54.9
##  8 November      55.4
##  9 June          55.7
## 10 October       56.1
## 11 December      58
## 12 April         63.7
```

```r
#Men drive the slowest, on average, in December.
```

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

```
cc %>%
  mutate(SEASON = ifelse(MONTH == c("December", "January", "February"), "Winter", ifelse(MONTH == c("Ma
  filter(INJ_SEV == "Fatal Injury (K)") %>%
  group_by(SEASON) %>%
  summarize(nn = n()) %>%
  mutate(percentage = nn/sum(nn))
```

```
## Warning in MONTH == c("December", "January", "February"): longer object
## length is not a multiple of shorter object length

## Warning in MONTH == c("March", "April", "May"): longer object length is not
## a multiple of shorter object length

## Warning in MONTH == c("June", "July", "August"): longer object length is
## not a multiple of shorter object length

## # A tibble: 4 x 3
##   SEASON    nn percentage
##   <chr>  <int>      <dbl>
## 1 Fall     101     0.863
## 2 Spring     3     0.0256
## 3 Summer    11     0.0940
## 4 Winter     2     0.0171
```

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

```
cc %>%
  filter(INJ_SEV == "Fatal Injury (K)") %>%
  group_by(DEFORMED) %>%
  summarize(nn = n()) %>%
  mutate(percentage = nn/sum(nn))
```

```
## # A tibble: 4 x 3
##   DEFORMED              nn percentage
##   <chr>              <int>      <dbl>
## 1 Disabling Damage     111    0.949
## 2 Functional Damage      3    0.0256
## 3 Minor Damage           1    0.00855
## 4 Not Reported           2    0.0171
```