Lab 1 - Part B

The following is your first chunk to start with. Remember, you can add chunks using the menu above (Insert -> R) or using the keyboard shortcut Ctrl+Alt+I. A good practice is to use different code chunks to answer different questions. You can delete this comment if you like.

Other useful keyboard shortcuts include Alt- for the assignment operator, and Ctrl+Shift+M for the pipe operator. You can delete these reminders if you don't want them in your report.

```
#setwd("C:/...")

library("tidyverse")

## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

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

library("tidymodels")

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

## -- Attaching packages -------------------------------------- tidymodels
0.0.3 --

## v broom     0.5.3     v recipes   0.1.9
## v dials     0.0.4     v rsample   0.0.5
## v infer     0.5.1     v yardstick 0.0.4
## v parsnip   0.0.5

## -- Conflicts -----------------------------------------
tidymodels_conflicts() --
## x scales::discard()   masks purrr::discard()
## x dplyr::filter()     masks stats::filter()
## x recipes::fixed()    masks stringr::fixed()
## x dplyr::lag()        masks stats::lag()
## x dials::margin()     masks ggplot2::margin()
## x yardstick::spec()   masks readr::spec()
## x recipes::step()     masks stats::step()
## x recipes::yj_trans() masks scales::yj_trans()
```

```
library("plotly")
```

```
## 
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
## 
##     last_plot
```

```
## The following object is masked from 'package:stats':
## 
##     filter
```

```
## The following object is masked from 'package:graphics':
## 
##     layout
```

```
library("skimr")
```

Load the Titanic dataset

```
dfTit <-
  read_csv("titanic.csv") %>%
  rename_all(tolower)
```

```
## 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()
## )
```

What was in the titanic dataset?

Variable	Definition	Key
survived	Survival	0 = No, 1 = Yes
class	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd

name	Name
sex	Gender
age	Age in years
sibsp	# of siblings / spouses aboard the Titanic
parch	# of parents / children aboard the Titanic
ticket	Ticket number
fare	Passenger fare
cabin	Cabin number
embarked	Port of Embarkation C = Cherbourg, Q = Queenstown, S = Southampton

Practice the Tidyverse functions

Part 1: Arrange

Q&A: Sort the Titanic dataset by age from high to low.

```
dfTit %>%
  arrange(desc(age))

## # A tibble: 891 x 12
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1         631        1      1 Bark~ male   80       0     0 27042   30
A23
##  2         852        0      3 Sven~ male   74       0     0 347060   7.78
<NA>
##  3          97        0      1 Gold~ male   71       0     0 PC 17~ 34.7
A5
##  4         494        0      1 Arta~ male   71       0     0 PC 17~ 49.5
<NA>
##  5         117        0      3 Conn~ male   70.5     0     0 370369   7.75
<NA>
##  6         673        0      2 Mitc~ male   70       0     0 C.A. ~ 10.5
<NA>
##  7         746        0      1 Cros~ male   70       1     1 WE/P ~ 71
B22
##  8          34        0      2 Whea~ male   66       0     0 C.A. ~ 10.5
<NA>
##  9          55        0      1 Ostb~ male   65       0     1 113509 62.0
B30
```

```
## 10         281        0      3 Duan~ male   65       0     0 336439  7.75
<NA>
## # ... with 881 more rows, and 1 more variable: embarked <chr>
```

Q1: You're looking for a passenger with a last name "Zimmerman." Sort the data in a way to spot her visually in the table.

```
dfTit %>%
  arrange(desc(name))

## # A tibble: 891 x 12
##    passengerid survived pclass name  sex     age sibsp parch ticket   fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>   <dbl>
<chr>
##  1         423        0      3 "Zim~ male   29       0     0 315082   7.88
<NA>
##  2         241        0      3 "Zab~ fema~  NA       1     0 2665    14.5
<NA>
##  3         112        0      3 "Zab~ fema~  14.5     1     0 2665    14.5
<NA>
##  4         200        0      2 "Yro~ fema~  24       0     0 248747  13
<NA>
##  5         496        0      3 "You~ male   NA       0     0 2627    14.5
<NA>
##  6         355        0      3 "You~ male   NA       0     0 2647     7.22
<NA>
##  7         204        0      3 "You~ male   45.5     0     0 2628     7.22
<NA>
##  8         326        1      1 "You~ fema~  36       0     0 PC 17~ 136.
C32
##  9         831        1      3 "Yas~ fema~  15       1     0 2659    14.5
<NA>
## 10         621        0      3 "Yas~ male   27       1     0 2659    14.5
<NA>
## # ... with 881 more rows, and 1 more variable: embarked <chr>
```

Q2: You're looking for the infant twins who boarded the Titanic together. Sort the data in a way to spot them visually in the table.

```
dfTit %>%
  arrange((age))

## # A tibble: 891 x 12
##    passengerid survived pclass name  sex     age sibsp parch ticket   fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>   <dbl>
<chr>
##  1         804        1      3 Thom~ male   0.42     0     1 2625     8.52
<NA>
##  2         756        1      2 Hama~ male   0.67     1     1 250649  14.5
```

```
<NA>
##  3         470        1      3 Bacl~ fema~  0.75     2     1 2666    19.3
<NA>
##  4         645        1      3 Bacl~ fema~  0.75     2     1 2666    19.3
<NA>
##  5          79        1      2 Cald~ male   0.83     0     2 248738  29
<NA>
##  6         832        1      2 Rich~ male   0.83     1     1 29106   18.8
<NA>
##  7         306        1      1 Alli~ male   0.92     1     2 113781 152.
C22 ~
##  8         165        0      3 Panu~ male   1        4     1 31012~  39.7
<NA>
##  9         173        1      3 John~ fema~  1        1     1 347742  11.1
<NA>
## 10         184        1      2 Beck~ male   1        2     1 230136  39
F4
## # ... with 881 more rows, and 1 more variable: embarked <chr>
```

Part 2: Select

Q&A: Select only the name, age, and survived columns.

```
dfTit %>%
  select(name, age, survived)

## # A tibble: 891 x 3
##    name                                                  age survived
##    <chr>                                               <dbl>    <dbl>
##  1 Braund, Mr. Owen Harris                                22        0
##  2 Cumings, Mrs. John Bradley (Florence Briggs Thayer)    38        1
##  3 Heikkinen, Miss. Laina                                 26        1
##  4 Futrelle, Mrs. Jacques Heath (Lily May Peel)           35        1
##  5 Allen, Mr. William Henry                               35        0
##  6 Moran, Mr. James                                       NA        0
##  7 McCarthy, Mr. Timothy J                                54        0
##  8 Palsson, Master. Gosta Leonard                          2        0
##  9 Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)      27        1
## 10 Nasser, Mrs. Nicholas (Adele Achem)                    14        1
## # ... with 881 more rows
```

Q1: Select all of the columns except the sex column [Hint: Simply use the negative sign!].

```
dfTit %>%
  select(-sex)

## # A tibble: 891 x 11
##    passengerid survived pclass name    age sibsp parch ticket  fare cabin
##          <dbl>    <dbl>  <dbl> <chr> <dbl> <dbl> <dbl> <chr>  <dbl> <chr>
##  1           1        0      3 Brau~    22     1     0 A/5 2~  7.25 <NA>
##  2           2        1      1 Cumi~    38     1     0 PC 17~ 71.3  C85
##  3           3        1      3 Heik~    26     0     0 STON/~  7.92 <NA>
##  4           4        1      1 Futr~    35     1     0 113803 53.1  C123
##  5           5        0      3 Alle~    35     0     0 373450  8.05 <NA>
##  6           6        0      3 Mora~    NA     0     0 330877  8.46 <NA>
##  7           7        0      1 McCa~    54     0     0 17463  51.9  E46
##  8           8        0      3 Pals~     2     3     1 349909 21.1  <NA>
##  9           9        1      3 John~    27     0     2 347742 11.1  <NA>
## 10          10        1      2 Nass~    14     1     0 237736 30.1  <NA>
## # ... with 881 more rows, and 1 more variable: embarked <chr>
```

Q2: Keep all of the columns but rearrange them so that class and fare are the first two columns [Hint: There is a shortcut for that!].

```
dfTit %>%
  select(3,10,1:12)
```

```
## # A tibble: 891 x 12
##    pclass  fare passengerid survived name  sex     age sibsp parch ticket
cabin
##     <dbl> <dbl>       <dbl>    <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>
<chr>
##  1      3  7.25           1        0 Brau~ male     22     1     0 A/5 2~
<NA>
##  2      1 71.3            2        1 Cumi~ fema~    38     1     0 PC 17~
C85
##  3      3  7.92           3        1 Heik~ fema~    26     0     0 STON/~
<NA>
##  4      1 53.1            4        1 Futr~ fema~    35     1     0 113803
C123
##  5      3  8.05           5        0 Alle~ male     35     0     0 373450
<NA>
##  6      3  8.46           6        0 Mora~ male     NA     0     0 330877
<NA>
##  7      1 51.9            7        0 McCa~ male     54     0     0 17463
E46
##  8      3 21.1            8        0 Pals~ male      2     3     1 349909
<NA>
##  9      3 11.1            9        1 John~ fema~    27     0     2 347742
<NA>
## 10      2 30.1           10        1 Nass~ fema~    14     1     0 237736
<NA>
## # ... with 881 more rows, and 1 more variable: embarked <chr>
```

Part 3: Filter

Q&A: Filter the dataset to the male passengers who have survived.

```
dfTit %>%
  filter(sex == 'male', survived == 1)
```

```
## # A tibble: 109 x 12
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1          18        1      2 Will~ male  NA        0     0 244373 13
<NA>
##  2          22        1      2 Bees~ male  34        0     0 248698 13
D56
##  3          24        1      1 Slop~ male  28        0     0 113788 35.5
A6
##  4          37        1      3 Mame~ male  NA        0     0 2677    7.23
<NA>
##  5          56        1      1 Wool~ male  NA        0     0 19947  35.5
C52
##  6          66        1      3 Moub~ male  NA        1     1 2661   15.2
<NA>
##  7          75        1      3 Bing~ male  32        0     0 1601   56.5
<NA>
##  8          79        1      2 Cald~ male   0.83     0     2 248738 29
<NA>
##  9          82        1      3 Shee~ male  29        0     0 345779  9.5
<NA>
## 10          98        1      1 Gree~ male  23        0     1 PC 17~ 63.4
D10 ~
## # ... with 99 more rows, and 1 more variable: embarked <chr>
```

Q1: How many of the survived passengers are older than 35? [Hint: Yes, you can see the number of rows at the bottom, but you can also pipe into nrow() function]

```
dfTit %>%
  filter(age>35)
```

```
## # A tibble: 217 x 12
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
C85
##  2           7        0      1 McCa~ male     54     0     0 17463  51.9
E46
##  3          12        1      1 Bonn~ fema~    58     0     0 113783 26.6
C103
##  4          14        0      3 Ande~ male     39     1     5 347082 31.3
<NA>
##  5          16        1      2 Hewl~ fema~    55     0     0 248706 16
<NA>
##  6          26        1      3 Aspl~ fema~    38     1     5 347077 31.4
<NA>
```

```
##  7          31        0      1 Uruc~ male     40     0     0 PC 17~ 27.7
<NA>
##  8          34        0      2 Whea~ male     66     0     0 C.A. ~ 10.5
<NA>
##  9          36        0      1 Holv~ male     42     1     0 113789 52
<NA>
## 10          41        0      3 Ahli~ fema~    40     1     0 7546    9.48
<NA>
## # ... with 207 more rows, and 1 more variable: embarked <chr>
```

Q2: Remember the twins from Part 1? Can you use the filter function to find their parent?

```
dfTit %>%
  filter(ticket==2666)

## # A tibble: 4 x 12
##   passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##         <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
## 1         449        1      3 Bacl~ fema~  5        2     1 2666    19.3
<NA>
## 2         470        1      3 Bacl~ fema~  0.75     2     1 2666    19.3
<NA>
## 3         645        1      3 Bacl~ fema~  0.75     2     1 2666    19.3
<NA>
## 4         859        1      3 Bacl~ fema~ 24        0     3 2666    19.3
<NA>
## # ... with 1 more variable: embarked <chr>
```

Part 4: Filter within groups

Q&A: Filter to the embarkation ports from which at least 100 passengers survived.

```
dfTit %>%
  group_by(embarked) %>%
  filter(sum(survived) >= 100)

## # A tibble: 644 x 12
## # Groups:   embarked [1]
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1           1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
<NA>
##  2           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
<NA>
##  3           4        1      1 Futr~ fema~    35     1     0 113803 53.1
C123
##  4           5        0      3 Alle~ male     35     0     0 373450  8.05
```

```
<NA>
##  5           7        0      1 McCa~ male     54     0     0 17463  51.9
E46
##  6           8        0      3 Pals~ male      2     3     1 349909 21.1
<NA>
##  7           9        1      3 John~ fema~    27     0     2 347742 11.1
<NA>
##  8          11        1      3 Sand~ fema~     4     1     1 PP 95~ 16.7
G6
##  9          12        1      1 Bonn~ fema~    58     0     0 113783 26.6
C103
## 10          13        0      3 Saun~ male     20     0     0 A/5. ~  8.05
<NA>
## # ... with 634 more rows, and 1 more variable: embarked <chr>
```

Q1: Filter to the passenger classes in which the average fare for the tickets is over $20.

```
dfTit %>%
  group_by(pclass) %>%
  filter(mean(fare)>20)
```

```
## # A tibble: 400 x 12
## # Groups:   pclass [2]
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1           2        1      1 Cumi~ fema~    38     1     0 PC 17~  71.3
C85
##  2           4        1      1 Futr~ fema~    35     1     0 113803  53.1
C123
##  3           7        0      1 McCa~ male     54     0     0 17463   51.9
E46
##  4          10        1      2 Nass~ fema~    14     1     0 237736  30.1
<NA>
##  5          12        1      1 Bonn~ fema~    58     0     0 113783  26.6
C103
##  6          16        1      2 Hewl~ fema~    55     0     0 248706  16
<NA>
##  7          18        1      2 Will~ male     NA     0     0 244373  13
<NA>
##  8          21        0      2 Fynn~ male     35     0     0 239865  26
<NA>
##  9          22        1      2 Bees~ male     34     0     0 248698  13
D56
## 10          24        1      1 Slop~ male     28     0     0 113788  35.5
A6
## # ... with 390 more rows, and 1 more variable: embarked <chr>
```

Part 5: Mutate

Q&A:Create a new column ageGroup: Children (under 15 years old), Working-age (15-64 years) and Elderly (65 years and older)

```
dfTit %>%
  mutate(ageGroup =  ifelse(age<15, "Children", ifelse(age>=15 & age <=64,
"Working-age", "Elderly")))

## # A tibble: 891 x 13
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##   1          1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
<NA>
##   2          2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
C85
##   3          3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
<NA>
##   4          4        1      1 Futr~ fema~    35     1     0 113803 53.1
C123
##   5          5        0      3 Alle~ male     35     0     0 373450  8.05
<NA>
##   6          6        0      3 Mora~ male     NA     0     0 330877  8.46
<NA>
##   7          7        0      1 McCa~ male     54     0     0 17463  51.9
E46
##   8          8        0      3 Pals~ male      2     3     1 349909 21.1
<NA>
##   9          9        1      3 John~ fema~    27     0     2 347742 11.1
<NA>
##  10         10        1      2 Nass~ fema~    14     1     0 237736 30.1
<NA>
## # ... with 881 more rows, and 2 more variables: embarked <chr>, ageGroup
<chr>
```

Q1: Create a new variable called fareCategory which divides the ticket prices into three bins: Low (<20), Medium (20-60), and High (>60)

```
dfTit %>%
  mutate(fareCategory = ifelse(fare<20, "Low", ifelse(fare>=20 &
fare<=60,"Medium", "High")))

## # A tibble: 891 x 13
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##   1          1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
<NA>
```

```
##  2           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3 
C85  
##  3           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92 
<NA> 
##  4           4        1      1 Futr~ fema~    35     1     0 113803 53.1 
C123 
##  5           5        0      3 Alle~ male     35     0     0 373450  8.05 
<NA> 
##  6           6        0      3 Mora~ male     NA     0     0 330877  8.46 
<NA> 
##  7           7        0      1 McCa~ male     54     0     0 17463  51.9 
E46  
##  8           8        0      3 Pals~ male      2     3     1 349909 21.1 
<NA> 
##  9           9        1      3 John~ fema~    27     0     2 347742 11.1 
<NA> 
## 10          10        1      2 Nass~ fema~    14     1     0 237736 30.1 
<NA> 
## # ... with 881 more rows, and 2 more variables: embarked <chr>,
## #   fareCategory <chr>
```

Q2: Add a new variable called familyOnBoard that adds up the number of passengers from one's family including siblings/spouses, parents/children, and oneself. Also sort by your calculated variable in a descending order to find the most crowded family.

```
dfTit %>%
  group_by(ticket) %>%
  mutate(familyOnBoard = n()) %>%
  arrange(desc(familyOnBoard))
```

```
## # A tibble: 891 x 13
## # Groups:   ticket [681]
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare 
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl> 
<chr>
##  1          14        0      3 Ande~ male     39     1     5 347082   31.3 
<NA> 
##  2          75        1      3 Bing~ male     32     0     0 1601     56.5 
<NA> 
##  3         120        0      3 Ande~ fema~     2     4     2 347082   31.3 
<NA> 
##  4         160        0      3 Sage~ male     NA     8     2 CA. 2~   69.6 
<NA> 
##  5         170        0      3 Ling~ male     28     0     0 1601     56.5 
<NA> 
##  6         181        0      3 Sage~ fema~    NA     8     2 CA. 2~   69.6 
<NA> 
##  7         202        0      3 Sage~ male     NA     8     2 CA. 2~   69.6 
<NA> 
```

```
##  8         325        0      3 Sage~ male     NA     8     2 CA. 2~  69.6
<NA>
##  9         510        1      3 Lang~ male     26     0     0 1601    56.5
<NA>
## 10         542        0      3 Ande~ fema~     9     4     2 347082  31.3
<NA>
## # ... with 881 more rows, and 2 more variables: embarked <chr>,
## #   familyOnBoard <int>
```

Part 6: Mutate with groups

Q&A: Based on whether passengers survived or not, calculate the deviation of the fare from the mean of each group. Save it to fareDeviation variable. Because you are interested in deviation in absolute terms, use take the absolute value.

```
dfTit %>%
  group_by(survived) %>%
  mutate(fareDeviation = abs(fare - mean(fare))) %>%
  ungroup()

## # A tibble: 891 x 13
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>   <dbl>
<chr>
##  1           1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
<NA>
##  2           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
C85
##  3           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
<NA>
##  4           4        1      1 Futr~ fema~    35     1     0 113803 53.1
C123
##  5           5        0      3 Alle~ male     35     0     0 373450  8.05
<NA>
##  6           6        0      3 Mora~ male     NA     0     0 330877  8.46
<NA>
##  7           7        0      1 McCa~ male     54     0     0 17463  51.9
E46
##  8           8        0      3 Pals~ male      2     3     1 349909 21.1
<NA>
##  9           9        1      3 John~ fema~    27     0     2 347742 11.1
<NA>
## 10          10        1      2 Nass~ fema~    14     1     0 237736 30.1
<NA>
## # ... with 881 more rows, and 2 more variables: embarked <chr>,
## #   fareDeviation <dbl>
```

Q1: Create a new variable indicating the number of people who are on the same ticket [Hint: Group by the ticket number and use n() function to get the counts].

```
dfTit %>%
  group_by(ticket) %>%
  mutate(number = n())
```

```
## # A tibble: 891 x 13
## # Groups:   ticket [681]
##    passengerid survived pclass name  sex     age sibsp parch ticket  fare
cabin
##          <dbl>    <dbl>  <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr>  <dbl>
<chr>
##  1           1        0      3 Brau~ male     22     1     0 A/5 2~  7.25
<NA>
##  2           2        1      1 Cumi~ fema~    38     1     0 PC 17~ 71.3
C85
##  3           3        1      3 Heik~ fema~    26     0     0 STON/~  7.92
<NA>
##  4           4        1      1 Futr~ fema~    35     1     0 113803 53.1
C123
##  5           5        0      3 Alle~ male     35     0     0 373450  8.05
<NA>
##  6           6        0      3 Mora~ male     NA     0     0 330877  8.46
<NA>
##  7           7        0      1 McCa~ male     54     0     0 17463  51.9
E46
##  8           8        0      3 Pals~ male      2     3     1 349909 21.1
<NA>
##  9           9        1      3 John~ fema~    27     0     2 347742 11.1
<NA>
## 10          10        1      2 Nass~ fema~    14     1     0 237736 30.1
<NA>
## # ... with 881 more rows, and 2 more variables: embarked <chr>, number
<int>
```

Part 7: Summarize

Q&A: Use the summarize command to find the mean age for all passengers.

```
dfTit %>%
  summarize(meanAge = mean(age, na.rm=TRUE)) # na.rm=TRUE is there to exclude
missing values; try removing it and see what happens!
```

```
## # A tibble: 1 x 1
##   meanAge
##     <dbl>
## 1    29.7
```

Q1: Determine the mean fare a passenger paid to get on board the Titanic.

```
dfTit %>%
  summarize(meanFare = mean(fare, na.rm=TRUE))
```

```
## # A tibble: 1 x 1
##   meanFare
##      <dbl>
## 1     32.2
```

Part 8: Summarize with groups

Q&A: Determine the mean fare of the passengers who survived. Compare it with the ones who did not survive.

```
dfTit %>%
  group_by(survived) %>%
  summarize(ageBySurvival = mean(age, na.rm=TRUE)) %>%
  ungroup()
```

```
## # A tibble: 2 x 2
##   survived ageBySurvival
##      <dbl>         <dbl>
## 1        0          30.6
## 2        1          28.3
```

Q1: What is the minimum and maximum age of the passengers based on whether they survived or not?

```
dfTit %>%
  group_by(survived) %>%
  summarize(maximum = max(age, na.rm=TRUE), minimum = min(age, na.rm=TRUE))
%>%
  ungroup()
```

```
## # A tibble: 2 x 3
##   survived maximum minimum
##      <dbl>   <dbl>   <dbl>
## 1        0      74    1
## 2        1      80    0.42
```

Q2: What is the minimum, maximum, and average fare that passengers of each class paid to get on the ship, based on whether they survived or not?

```
dfTit %>%
  group_by(survived, pclass) %>%
  summarize(minimum = min(fare, na.rm=TRUE), maximum = max(fare, na.rm=TRUE),
average = mean(fare, na.rm=TRUE)) %>%
  ungroup()
```

```
## # A tibble: 6 x 5
##   survived pclass minimum maximum average
##      <dbl>  <dbl>   <dbl>   <dbl>   <dbl>
## 1        0      1     0     263      64.7
## 2        0      2     0      73.5    19.4
## 3        0      3     0      69.6    13.7
## 4        1      1    25.9   512.     95.6
## 5         1       2     10.5     65       22.1
## 6         1       3      0       56.5     13.7
```

Part 9: Combining verbs

Q&A: For the survived passengers who were on a first class ticket, find the mean age and fare by gender.

```
dfTit %>%
  filter(survived == 1 & pclass == 1) %>%
  group_by(sex) %>%
  summarize(avgAge = mean(age, na.rm=TRUE), avgFare = mean(fare, na.rm=TRUE))
%>%
  ungroup()

## # A tibble: 2 x 3
##   sex    avgAge avgFare
##   <chr>   <dbl>   <dbl>
## 1 female   34.9   106.
## 2 male     36.2    74.6
```

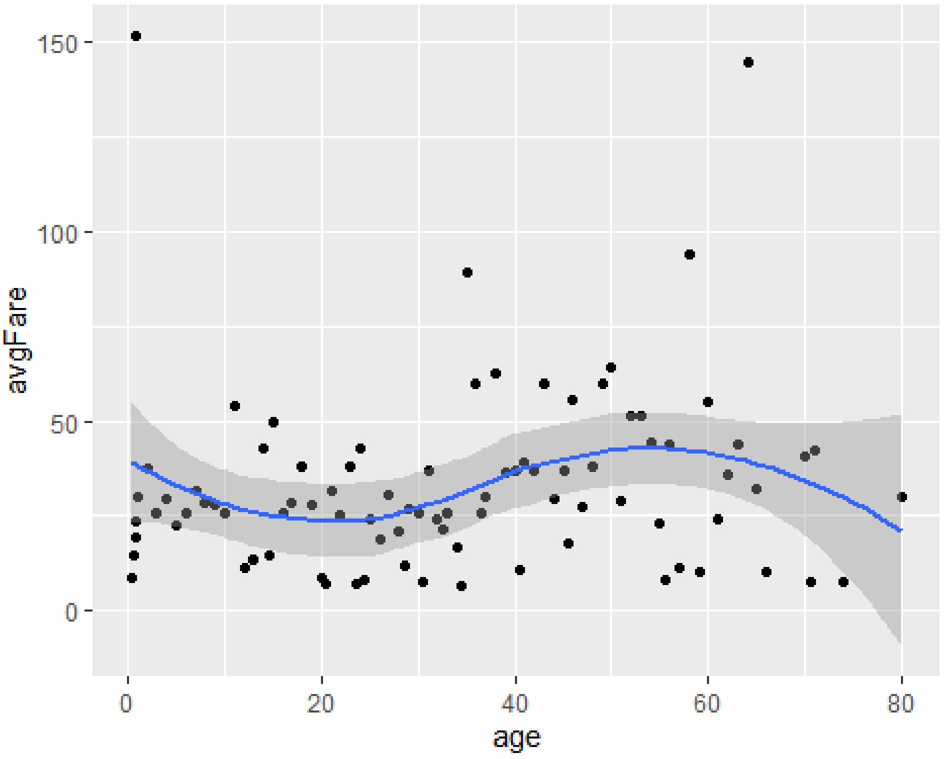
Q1: After excluding individual passengers, calculate (i) the total cost per family (based on whether they are on the same ticket), (ii) the number of family members on the same ticket, and (iii) how many of these family members survived. Then, keep only the ticket number and the three variables you calculated, sort by the total cost descending, and remove the repetitions in the table [Hint: Use the distinct() function with ".keep_all = TRUE" option to display the details for each unique ticket].

```
dfTit %>%
  filter(sibsp>=1|parch>=1) %>%
  group_by(ticket) %>%
  summarize(expense = sum(fare), members = n(), Survive = sum(survived)) %>%
  arrange(desc(expense)) %>%
  ungroup()

## # A tibble: 198 x 4
##    ticket   expense members Survive
##    <chr>      <dbl>   <int>   <dbl>
##  1 19950      1052        4       2
##  2 PC 17608    525.       2       2
##  3 PC 17755    512.       1       1
##  4 PC 17558    495.       2       1
##  5 CA. 2343    487.       7       0
##  6 113760      480        4       4
##  7 113781      455.       3       1
##  8 24160       423.       2       2
##  9 35273       340.       3       2
## 10 17421       333.       3       2
## # ... with 188 more rows
```

Part 10: Visualizations

Q&A: Create a plot showing the relationship between age and median fare by age group, and fit a smoothed curve on it (no need to set any parameters, just use the defaults).

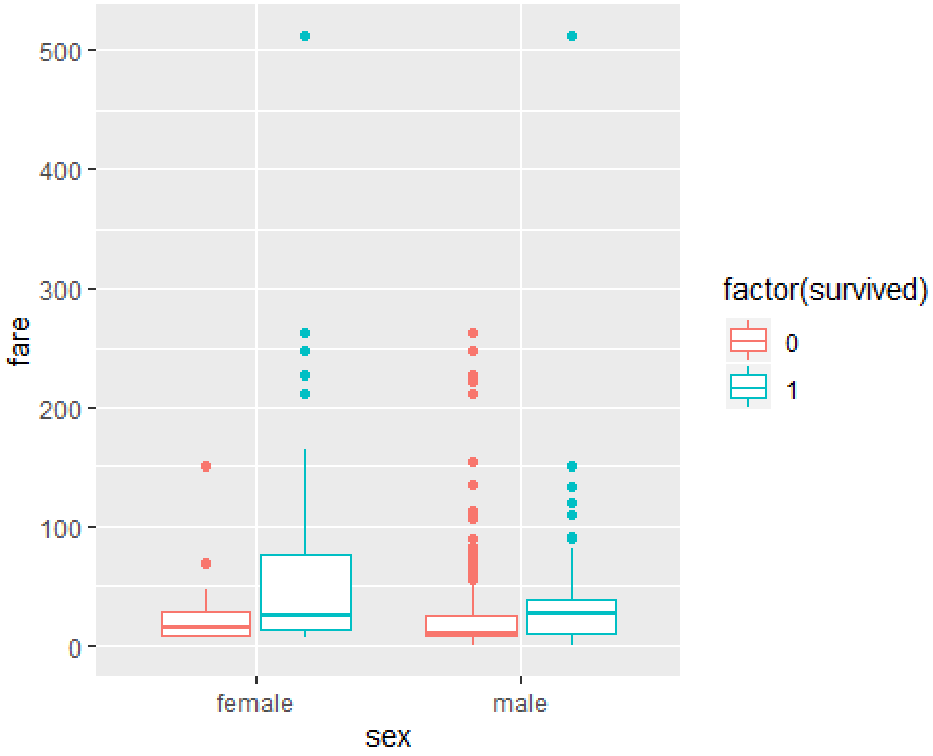
```
pAgeAvgFare <-
  dfTit %>%
  group_by(age) %>%
  summarize(avgFare = mean(fare)) %>%
  ungroup() %>%
  ggplot(aes(x=age,y=avgFare)) + geom_point() + geom_smooth()

pAgeAvgFare

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).
```



Q1: Create a box-plot showing the distribution of fare across genders, and coloring it based on whether a passenger survived or not [Hint: Color will go into the aesthetics of the box plot].

```
genFare <-
  dfTit %>%
  ggplot(aes(x=sex,y=fare, color=factor(survived))) + geom_boxplot()
```

```
genFare
```


fare
female
male
sex
factor(survived)