# Titanic dataset analasys

Jetze Luyten, David Silva Troya, Axel Van Gestel

## Setup the enviroment

### install required packages and load required libraries

```r
library(tidyverse) # Contains all tidyverse packages (ggplot2, dplyr, ...)
library(ggcorrplot) # Used for generating correlation heatmaps (uses ggplot2)
```

### Setup enviroment settings

```r
Sys.setenv(LANG = "en") # Set language to English
setwd(getwd()) # Set the working directory to the script directory
rm(list = ls()) # Clears the Global Env
theme_update(plot.title = element_text(hjust = 0.5)) # Center all plot titles
```

## Read and import the data set

### Read the data set (uses readr)

```r
column_types <- cols(
  Survived = col_factor(),
  Pclass = col_factor(include_na = TRUE, ordered = TRUE),
  Sex = col_factor(),
  Embarked = col_factor(include_na = TRUE, ordered = TRUE)
)
train <- read_csv("./kaggle/titanic/train.csv", col_types = column_types)
```

### Rename the factors to be human readable (uses dplyr)

```r
train$Survived <- recode_factor(train$Survived,
                                "0" = "No",
                                "1" = "Yes")

train$Pclass <- recode_factor(train$Pclass,
                              "1" = "1st",
                              "2" = "2nd",
                              "3" = "3rd",
                              .default = "Unknown", # NA -> Unknown
                              .ordered = TRUE)

train$Embarked <- recode_factor(train$Embarked,
                                "S" = "Southampton (England)",
                                "C" = "Cherbourg (France)",
                                "Q" = "Queenstown (Ireland)",
                                .default = "Unknown", # NA -> Unknown
                                .ordered = TRUE)
```

## Filtering and cleaning

### Check for the number of NA's in each column

```
sanity_check <- function(my_df) {
  for (j in 1:ncol(my_df)) {
    print(paste(names(my_df[j]), ":", sum(is.na(my_df[, j]))))
  }
}

sanity_check(train)
```

```
## [1] "PassengerId : 0"
## [1] "Survived : 0"
## [1] "Pclass : 0"
## [1] "Name : 0"
## [1] "Sex : 0"
## [1] "Age : 177"
## [1] "SibSp : 0"
## [1] "Parch : 0"
## [1] "Ticket : 0"
## [1] "Fare : 0"
## [1] "Cabin : 687"
## [1] "Embarked : 0"
```

### View 'train' tibble

```
train
```

```
## # A tibble: 891 x 12
##    PassengerId Survived Pclass Name   Sex     Age SibSp Parch Ticket  Fare Cabin
##          <dbl> <fct>    <ord>  <chr>  <fct> <dbl> <dbl> <dbl> <chr>  <dbl> <chr>
##  1           1 No       3rd    Braun~ male     22     1     0 A/5 2~  7.25 <NA>
##  2           2 Yes      1st    Cumin~ fema~    38     1     0 PC 17~ 71.3  C85
##  3           3 Yes      3rd    Heikk~ fema~    26     0     0 STON/~  7.92 <NA>
##  4           4 Yes      1st    Futre~ fema~    35     1     0 113803 53.1  C123
##  5           5 No       3rd    Allen~ male     35     0     0 373450  8.05 <NA>
##  6           6 No       3rd    Moran~ male     NA     0     0 330877  8.46 <NA>
##  7           7 No       1st    McCar~ male     54     0     0 17463  51.9  E46
##  8           8 No       3rd    Palss~ male      2     3     1 349909 21.1  <NA>
##  9           9 Yes      3rd    Johns~ fema~    27     0     2 347742 11.1  <NA>
## 10          10 Yes      2nd    Nasse~ fema~    14     1     0 237736 30.1  <NA>
## # ... with 881 more rows, and 1 more variable: Embarked <ord>
```

## Adding useful columns

### Add a total Family size column

```
train <- mutate(train, FamilySize = SibSp + Parch)
```

### Group the cabin label into has cabin and has no cabin

```
train <- mutate(train, CabinGroups = ifelse(is.na(train$Cabin),
                                            "No cabin",
                                            "Cabin"))
```

### Quick sanity check of the ‘train’ tibble

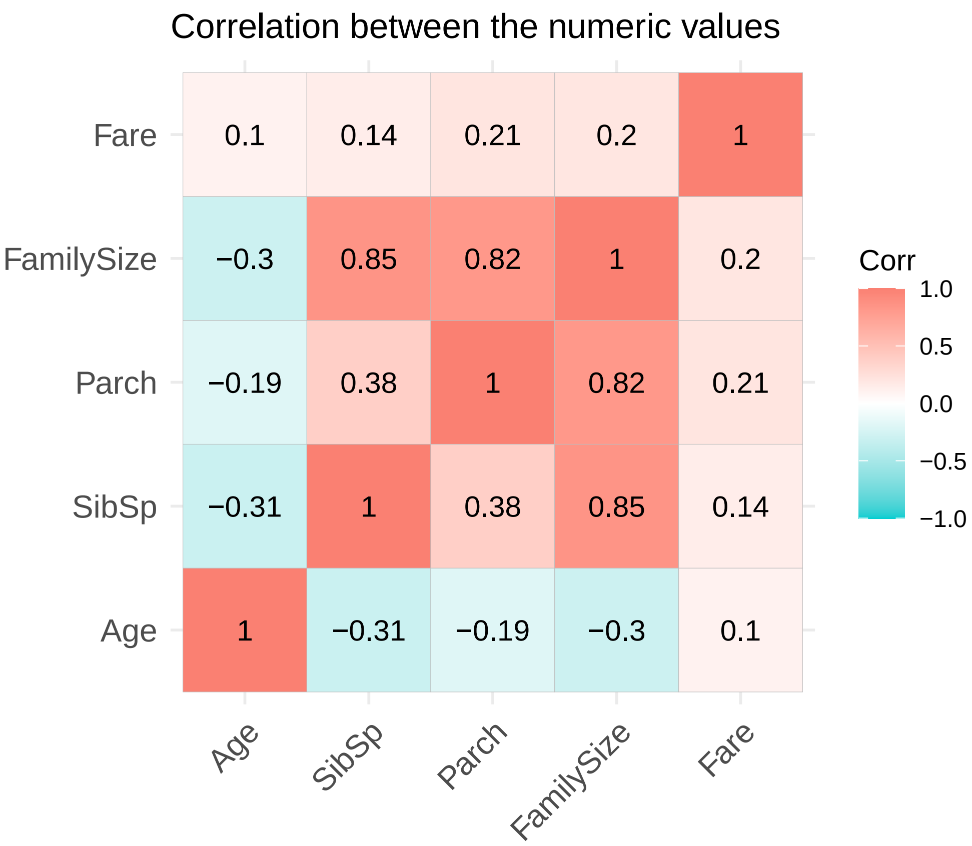
```
tail(train)
```

```
## # A tibble: 6 x 14
##   PassengerId Survived Pclass Name    Sex     Age SibSp Parch Ticket  Fare Cabin
##         <dbl> <fct>    <ord>  <chr>   <fct> <dbl> <dbl> <dbl> <chr>  <dbl> <chr>
## 1         886 No       3rd    "Rice,~ fema~    39     0     5 382652 29.1  <NA>
## 2         887 No       2nd    "Montv~ male     27     0     0 211536 13    <NA>
## 3         888 Yes      1st    "Graha~ fema~    19     0     0 112053 30    B42
## 4         889 No       3rd    "Johns~ fema~    NA     1     2 W./C.~ 23.4  <NA>
## 5         890 Yes      1st    "Behr,~ male     26     0     0 111369 30    C148
## 6         891 No       3rd    "Doole~ male     32     0     0 370376  7.75 <NA>
## # ... with 3 more variables: Embarked <ord>, FamilySize <dbl>,
## #   CabinGroups <chr>
```

## Correlation heatmap (uses ggcorrplot)

### Generate a correlation heatmap of the numeric values

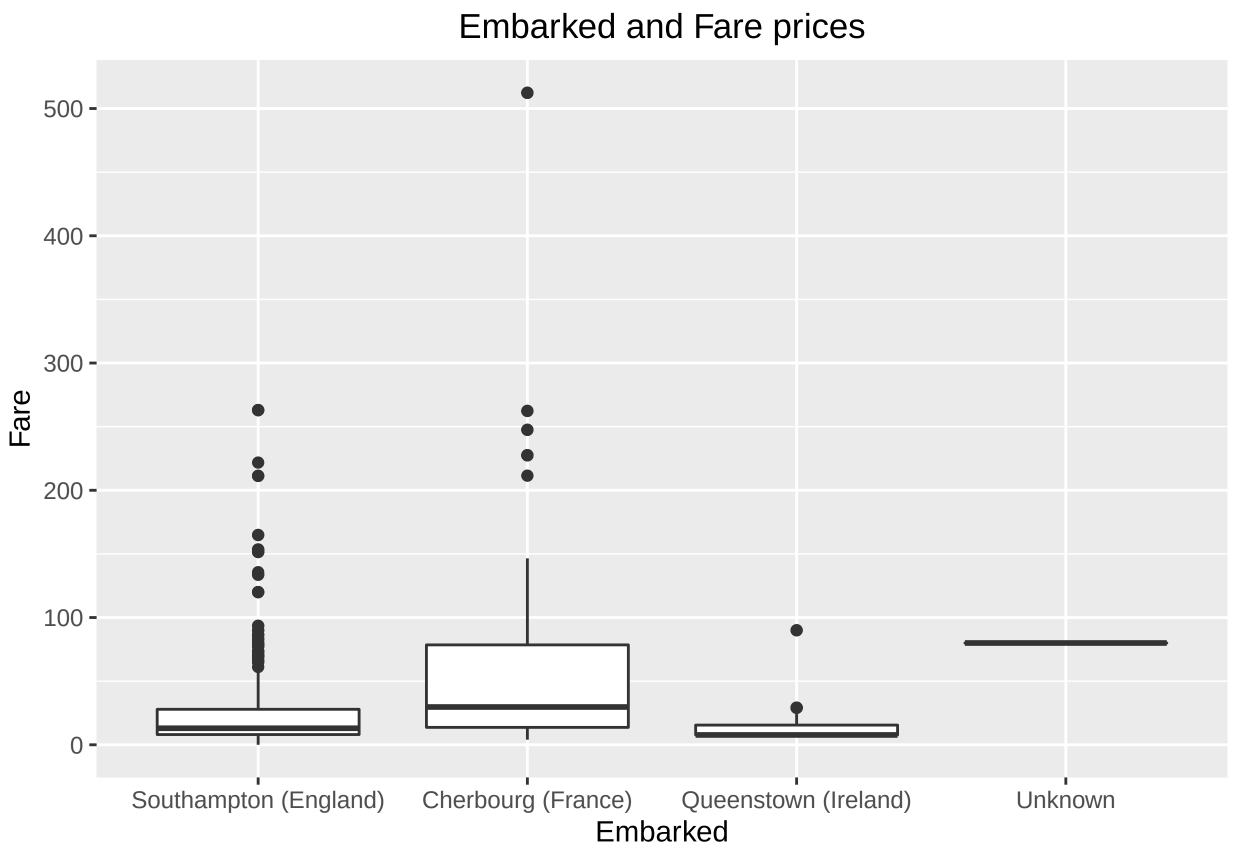
```
train_numeric <- select(train, Age, SibSp, Parch, FamilySize, Fare)
train_numeric_corr <- cor(train_numeric, use = "complete.obs") # Use only non NA
ggcorrplot::ggcorrplot(train_numeric_corr,
                       lab = TRUE, # Show correlation coefficients
                       colors = c("darkturquoise", "white", "salmon"),
                       title = "Correlation between the numeric values")
```



## Plots and stuff (uses ggplot2)

### Embarked and Fare prices

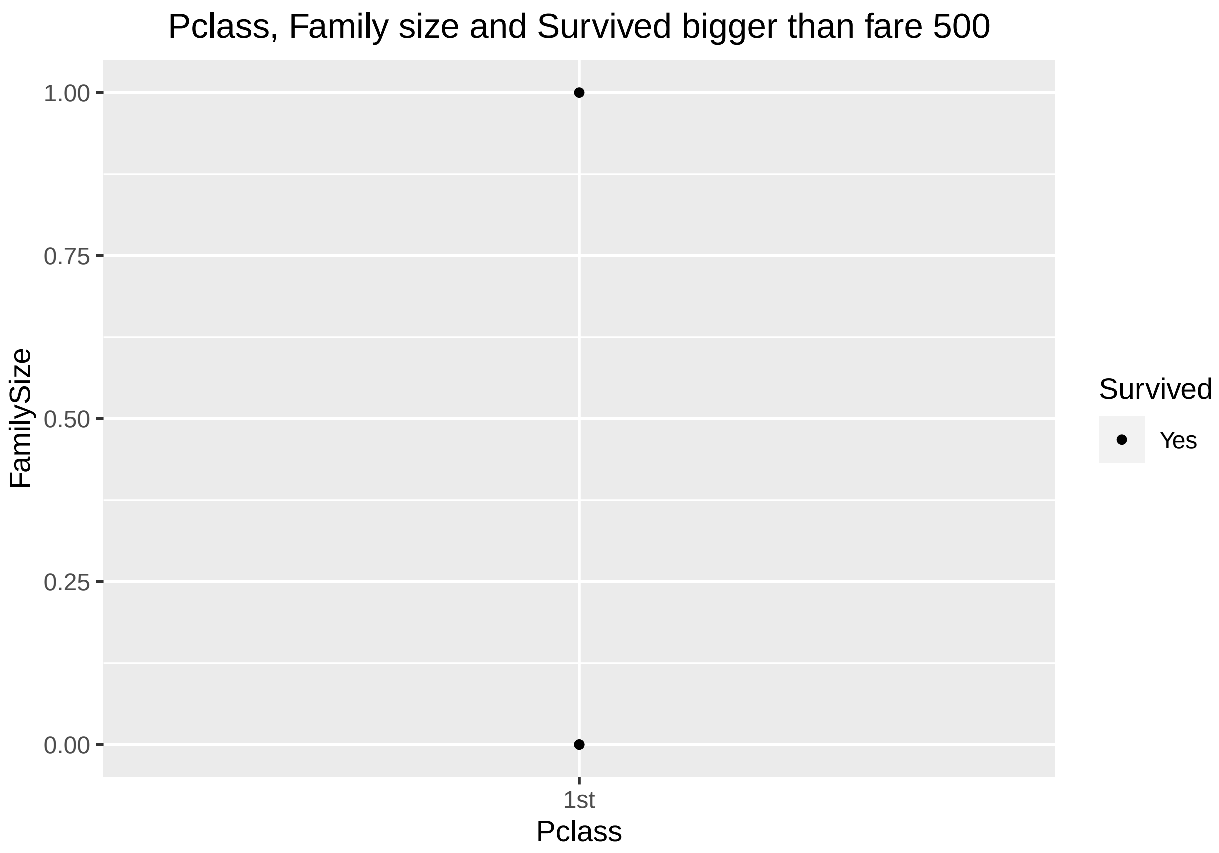
```
ggplot(data = train, mapping = aes(x = Embarked, y = Fare)) +
  geom_boxplot() +
  ggtitle("Embarked and Fare prices")
```



### Pclass, Family size and Survived bigger than fare 500

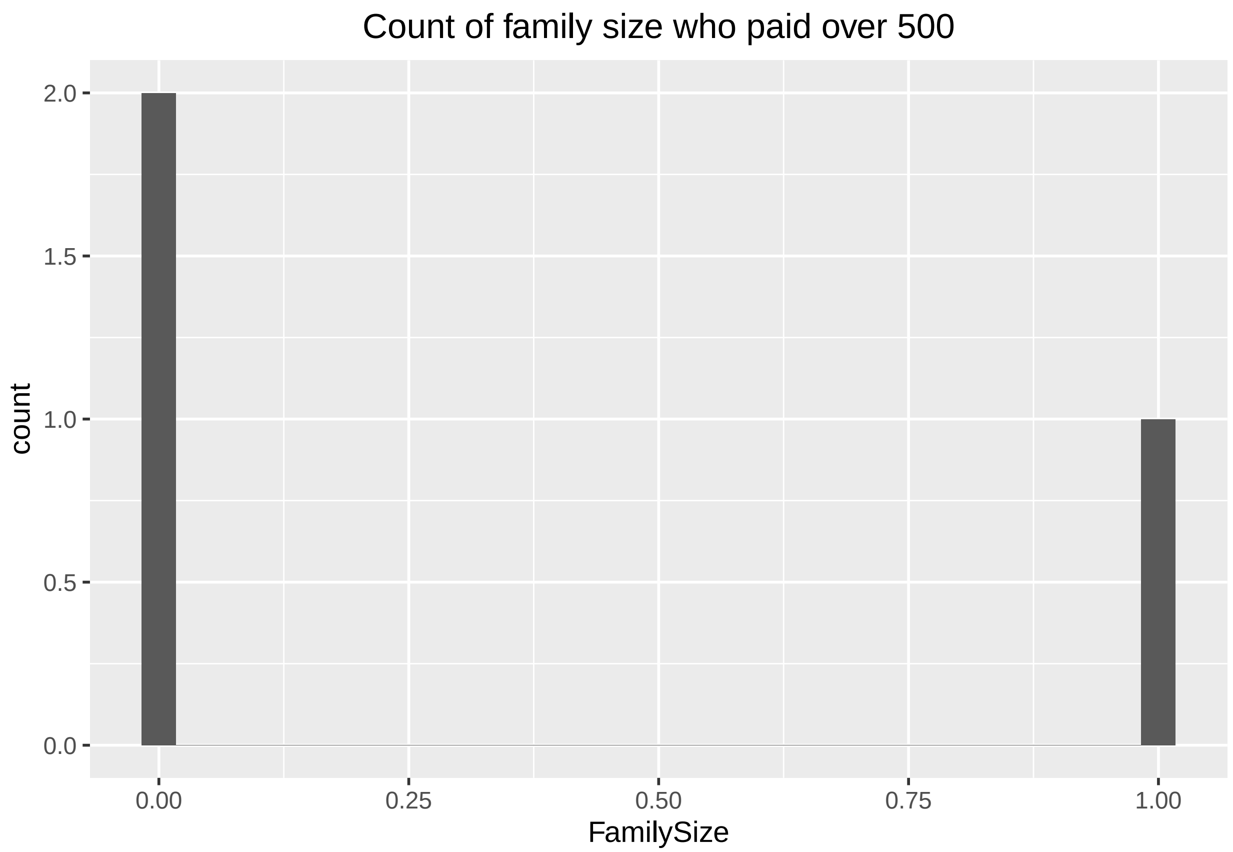
```
FareEnough <- filter(train, Fare > 500) # Fare bigger than 500

ggplot(data = FareEnough, mapping = aes(x = Pclass, y = FamilySize)) +
  geom_point(aes(shape=Survived)) +
  ggtitle("Pclass, Family size and Survived bigger than fare 500")
```



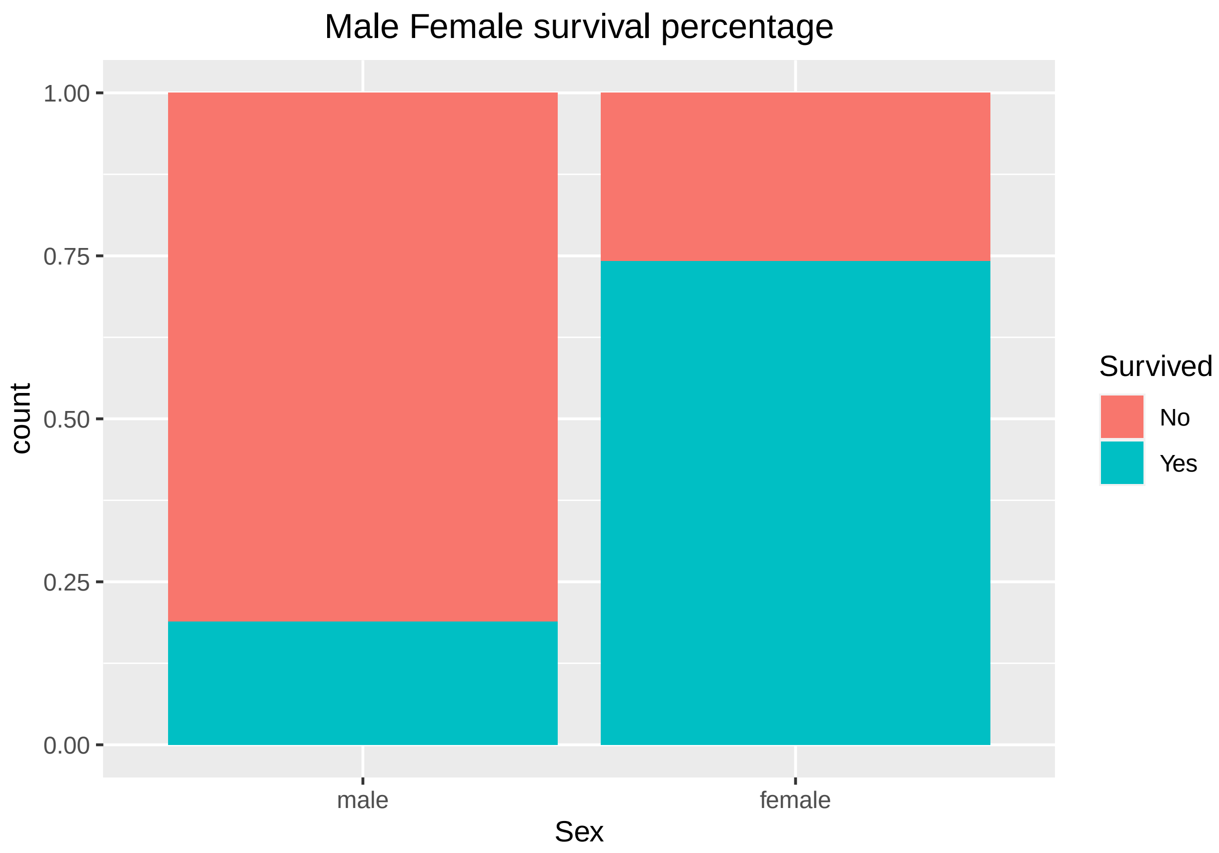
### Count of family size who paid over 500

```
ggplot(data = FareEnough, mapping = aes(x = FamilySize)) +
  geom_histogram() +
  ggtitle("Count of family size who paid over 500")
```



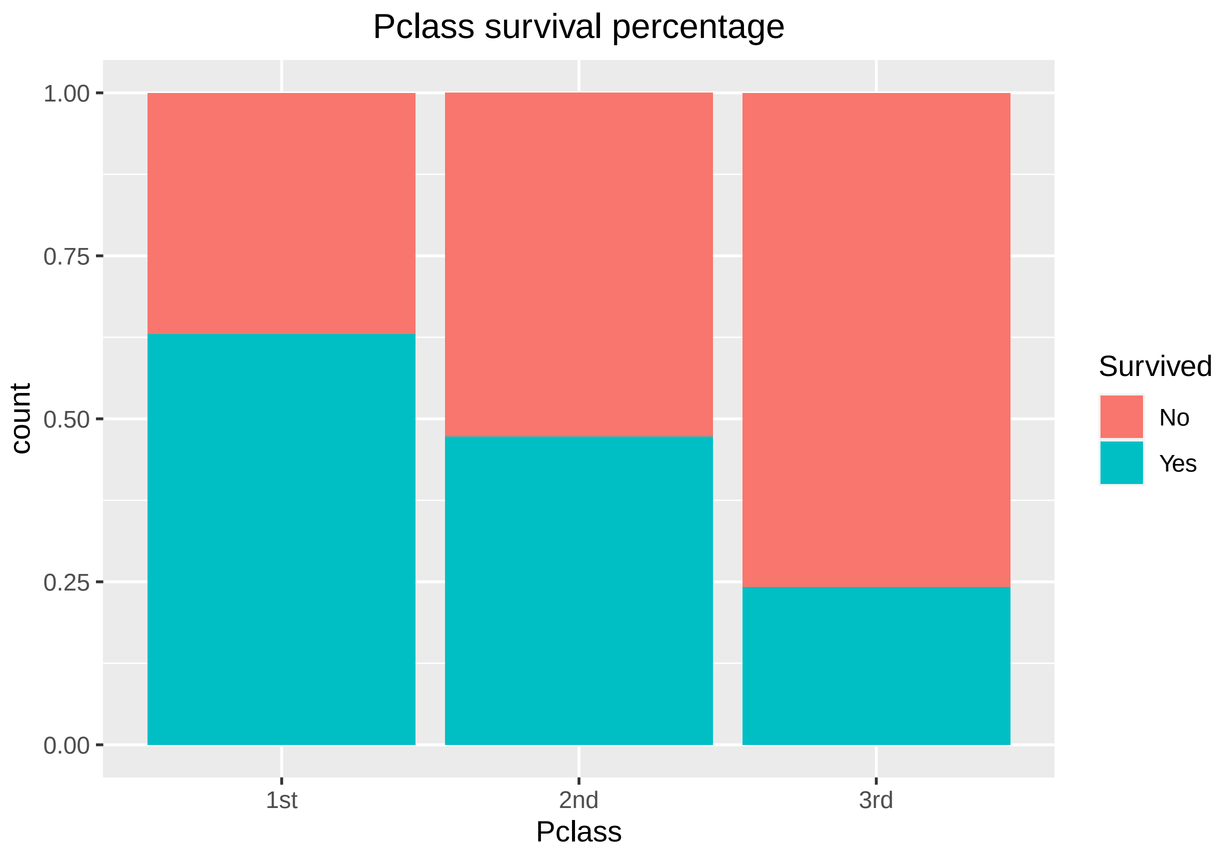
### Male Female survival percentage

```
ggplot(data = train, mapping = aes(x = Sex, fill = Survived)) +
  geom_bar(position = "fill") +
  ggtitle("Male Female survival percentage")
```



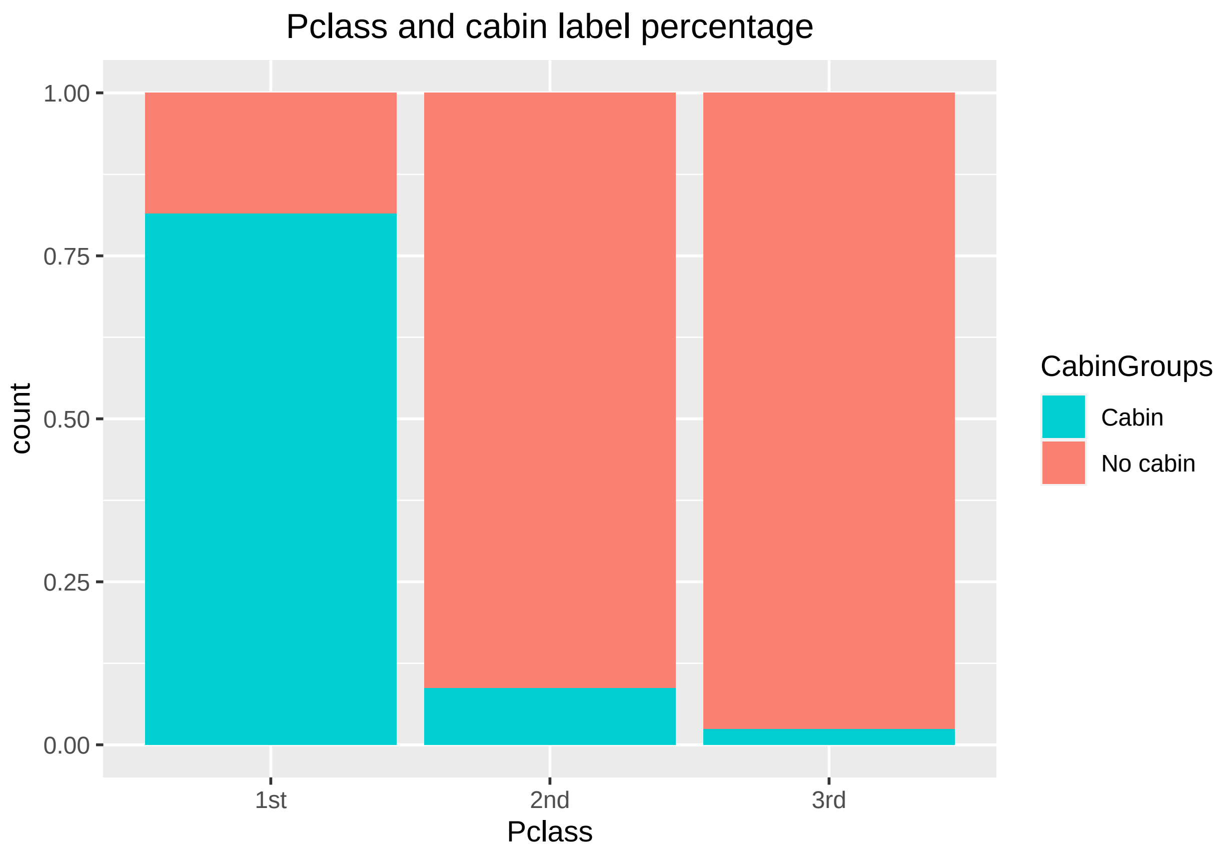
### Pclass survival percentage

```
ggplot(data = train, mapping = aes(x = Pclass, fill = Survived)) +
  geom_bar(position = "fill") +
  ggtitle("Pclass survival percentage")
```



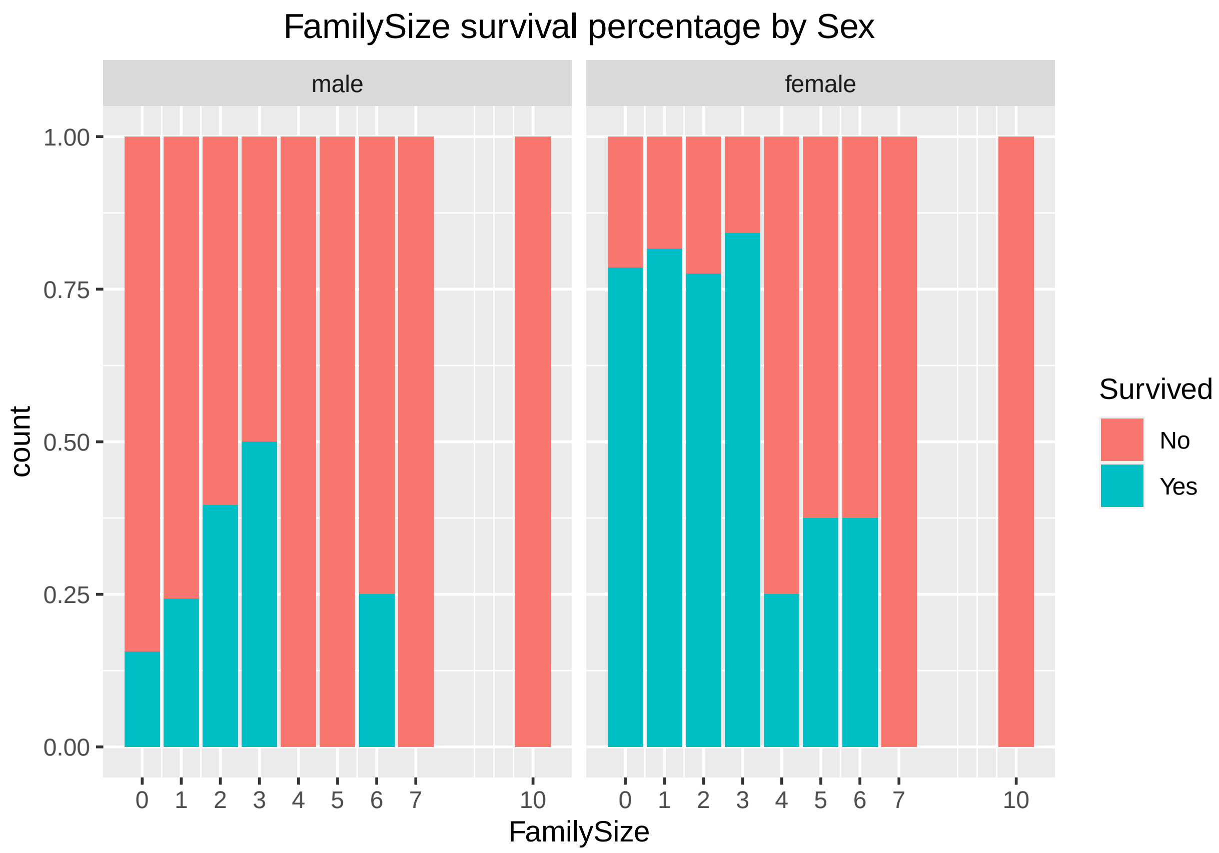
### Pclass and cabin label percentage

```
ggplot(data = train, mapping = aes(x = Pclass, fill = CabinGroups)) +
  geom_bar(position = position_fill(reverse = TRUE)) +
  scale_fill_manual(values = c("darkturquoise",
                               "salmon")) +
  ggtitle("Pclass and cabin label percentage")
```



### FamilySize survival percentage by Sex

```
ggplot(data = train, mapping = aes(x = FamilySize, fill = Survived)) +
  geom_bar(position = "fill") +
  facet_wrap(~ Sex) +
  scale_x_continuous(breaks = unique(train$FamilySize)) +
  ggtitle("FamilySize survival percentage by Sex")
```


FamilySize survival percentage by Sex
male
female
count
1.00
0.75
0.50
0.25
0.00
0
1
2
3
4
5
6
7
10
FamilySize
Survived
No
Yes