Kaggle: Titanic

Suman

June 26, 2017

Executive Summary:

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew. This sensational tragedy shocked the international community and led to better safety regulations for ships.

One of the reasons that the shipwreck led to such loss of life was that there were not enough lifeboats for the passengers and crew. Although there was some element of luck involved in surviving the sinking, some groups of people were more likely to survive than others, such as women, children, and the upper-class.

In this challenge, we ask you to complete the analysis of what sorts of people were likely to survive. In particular, we ask you to apply the tools of machine learning to predict which passengers survived the tragedy.

Libaries and Data

```r
#### Load libraries
library(caret)
library(randomForest)
library(rpart)
library(rpart.plot)
library(corrplot)
library(dplyr)
genderSub <- read.csv('/R/workspace/kaggle/titanic/gender_submission.csv',sep=',',header=T)
testData <- read.csv('/R/workspace/kaggle/titanic/test.csv',sep=',',header=T)
trainData <- read.csv('/R/workspace/kaggle/titanic/train.csv',sep=',',header=T)

str(trainData)
```

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 58
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
```

```r
summary(trainData)
```

```
##   PassengerId       Survived          Pclass
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000
##  Median :446.0   Median :0.0000   Median :3.000
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##                                     Name         Sex           Age
##  Abbing, Mr. Anthony                  :  1   female:314   Min.   : 0.42
##  Abbott, Mr. Rossmore Edward          :  1   male  :577   1st Qu.:20.12
##  Abbott, Mrs. Stanton (Rosa Hunt)     :  1                Median :28.00
##  Abelson, Mr. Samuel                  :  1                Mean   :29.70
##  Abelson, Mrs. Samuel (Hannah Wizosky):  1                3rd Qu.:38.00
##  Adahl, Mr. Mauritz Nils Martin       :  1                Max.   :80.00
##  (Other)                              :885                NA's   :177
##      SibSp           Parch             Ticket          Fare
##  Min.   :0.000   Min.   :0.0000   1601    :  7   Min.   :  0.00
##  1st Qu.:0.000   1st Qu.:0.0000   347082  :  7   1st Qu.:  7.91
##  Median :0.000   Median :0.0000   CA. 2343:  7   Median : 14.45
##  Mean   :0.523   Mean   :0.3816   3101295 :  6   Mean   : 32.20
##  3rd Qu.:1.000   3rd Qu.:0.0000   347088  :  6   3rd Qu.: 31.00
##  Max.   :8.000   Max.   :6.0000   CA 2144 :  6   Max.   :512.33
##                                   (Other) :852
##          Cabin     Embarked
##             :687    :  2
##  B96 B98    :  4   C:168
##  C23 C25 C27:  4   Q: 77
##  G6         :  4   S:644
##  C22 C26    :  3
##  D          :  3
##  (Other)    :186
```

Variables and their description:

Variable	Definition	Key
survival	Survival	0 if Survival = No, 1 if Survival = Yes
pclass	Ticket class accorinding to socio-economic status	1 = Upper, 2 = Middle, 3 = Lower
sex	Sex	
Age	Age of passengers in years	
sibsp	No. of siblings / spouses aboard the Titanic	
parch	No. 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

Cleaning Data and Imputation

From the summary of the data above we got our data have some missing values. In this section we are going to fix and imputate the missing values.

Cleaning and imputation of AGE

From Summary of the data we can observe that the Age variable has some missing values. We will be developing and implementing workaround to deal with missing values.

```
sum(is.na(trainData$Age))
```

```
## [1] 177
```

```
trailData <- trainData
```

To impute the age, lets introduce a new variable Age categorey using below rule:

1) If (number of spouses or siblings > 1) then it's probably a child and represented by 0
2) If (number of parents or children > 2) then it's probably an adult and represented by 1
3) If all above conditions are not met lets call it is unteremined and represented by 2

```
trailData <- mutate(trailData, AgeCat = ifelse(SibSp > 1 , 0, ifelse(Parch > 2, 1, 2)))
```

Building model to predict age.

```
ageModel = rpart(Age ~ Fare + Pclass + SibSp + Parch + AgeCat, data = trailData)
trailData$predictAge = predict(ageModel,trailData)
trailData$Age <- ifelse(is.na(trailData$Age), trailData$predictAge, trailData$Age)
numOfNa <- sum(is.na(trailData$Age))
```

With successful imputation of age variables, there are 0 NA's in AGE variables.

Cleaning and imputation of Embarked

There are smome empty values for variable Embarked. In this section we will expore and clean the missing data.

```
## Check if Embarked is missing & if missing show respective index
which(trailData$Embarked=="")
```

```
## [1]  62 830
```

From above we can see two values for embarked missing. We will use the existing data to predict the empty data.

```
trailData[trailData$Embarked=="",]
```

```
##     PassengerId Survived Pclass                                      Name
## 62           62        1      1                       Icard, Miss. Amelie
## 830         830        1      1 Stone, Mrs. George Nelson (Martha Evelyn)
##        Sex Age SibSp Parch Ticket Fare Cabin Embarked AgeCat predictAge
## 62  female  38     0     0 113572   80   B28                2   37.72566
## 830 female  62     0     0 113572   80   B28                2   37.72566
```

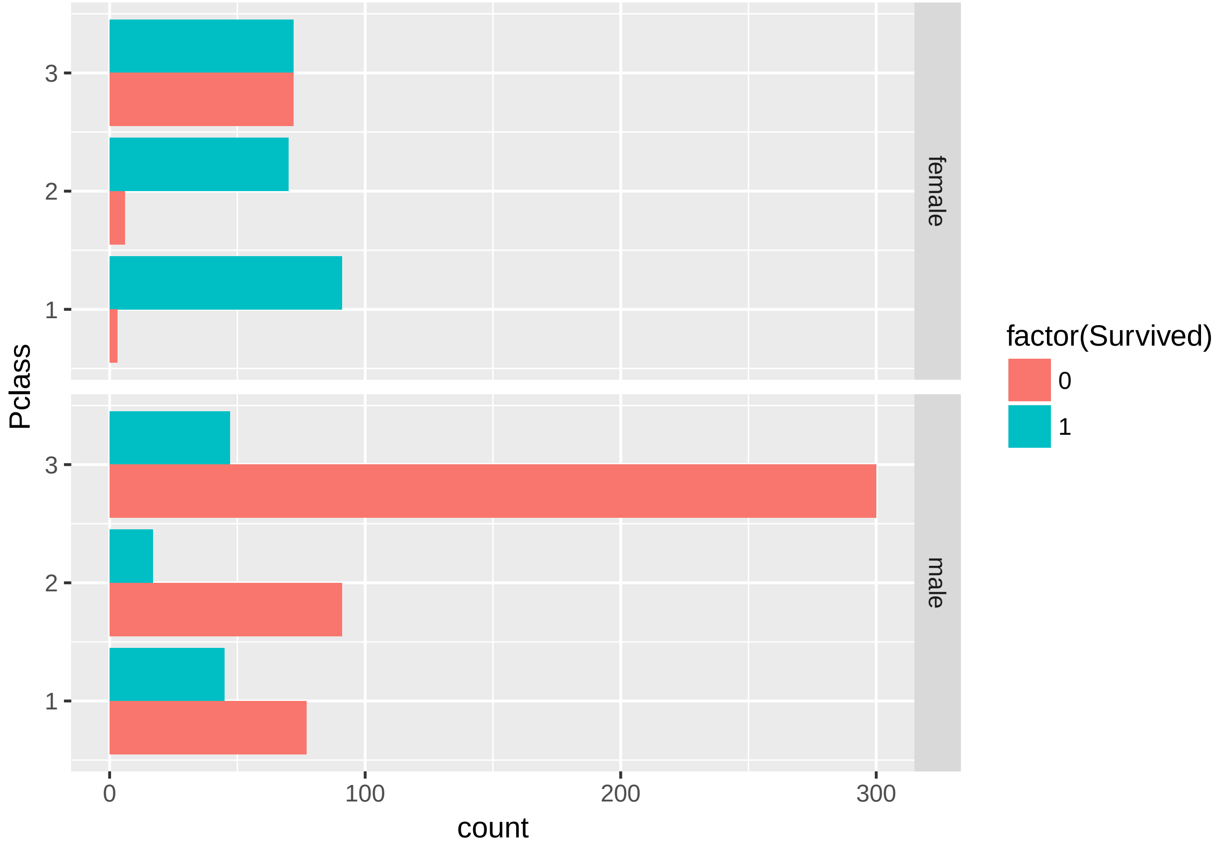
Going in detail we find that the data with missing embarked have same ticked num and same cabin are from same passenger class. Hence lets build a model based on fare and passenger class.

```
emptyEmbarked <-which(trailData$Embarked=="")
embarkedModel <- train( Embarked ~ Pclass + Fare, data = trailData, method="rpart", na.action = na.pass
trailData$Embarked[emptyEmbarked] <- predict(embarkedModel, trailData[emptyEmbarked, ])
```

Exploratory Data analysis

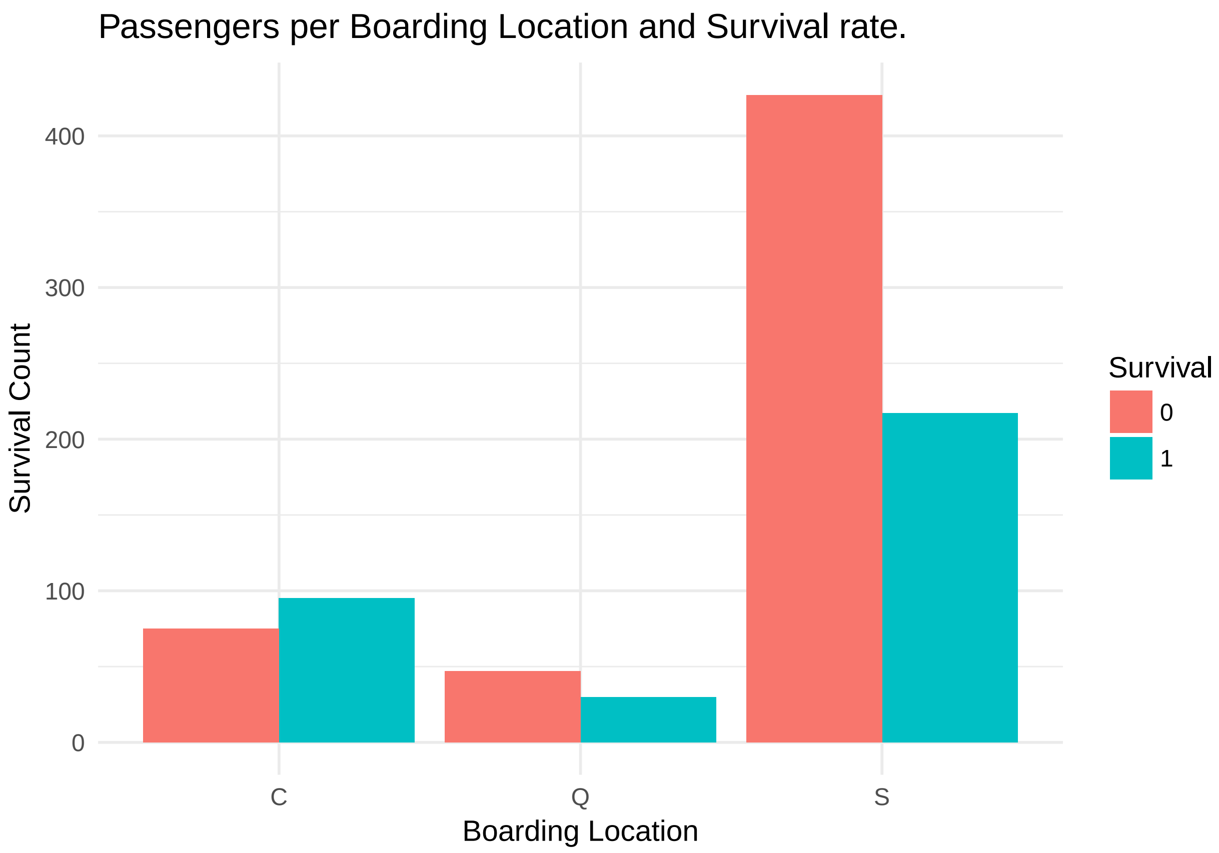
Men and Women Survived by Passenger Class

```
ggplot(trailData, aes(Pclass, fill = factor(Survived))) + geom_bar(stat=’count’, position=’dodge’) +  c
```



Passengers per Boarding Location

```
plot2 <- ggplot(trailData, aes(x = Embarked, fill = factor(Survived)))
plot2 <- plot2 + geom_bar(stat=’count’, position=’dodge’)
plot2 <- plot2 + ggtitle(’Passengers per Boarding Location and Survival rate.’)
plot2 <- plot2 + ylab(’Survival Count’)
plot2 <- plot2 + xlab(’Boarding Location’) + theme_minimal()
plot2 <- plot2 +  scale_fill_discrete(name = "Survival")
plot2
```



Feature Engineering:

Stronger vs Weak

It is obvious that, we can assume that stronger can swim more than weaker ones. So I would like to categorize age into stronger/weak varibles. Categorizing is done:

1. age >=18 || <= 50 => Stronger
2. age<18 && >=50 Weaker

```
trailData <- trainData
```

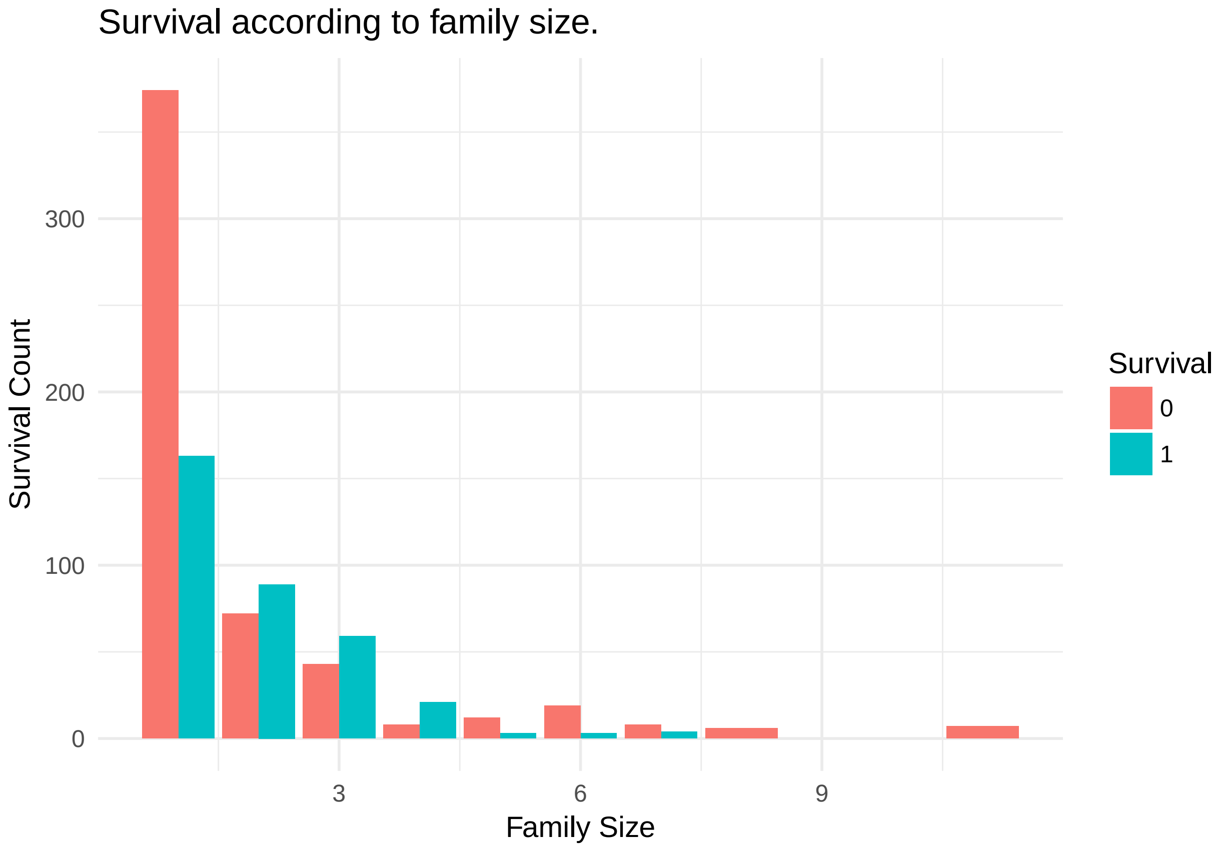
Family Size

Now I am focusing on the passengers if they are travelling in family and their survival rate.

```
## Introduce new variable familySize, (+1 is for the individual him or her self)
trailData <- mutate(trailData, familySize = trailData$SibSp + trailData$Parch + 1)
str(trailData)
```

```
## 'data.frame':    891 obs. of  13 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 58
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ familySize : num  2 2 1 2 1 1 1 5 3 2 ...
```

```
plot1 <- ggplot(trailData, aes(x = familySize, fill = factor(Survived)))
plot1 <- plot1 + geom_bar(stat='count', position='dodge')
plot1 <- plot1 + ggtitle('Survival according to family size.')
plot1 <- plot1 + ylab('Survival Count')
plot1 <- plot1 + xlab('Family Size') + theme_minimal()
plot1 <- plot1 +  scale_fill_discrete(name = "Survival")
plot1
```



Nice !. We can observe from the above barchart that the survival rate is high for fmily travelling alone or with fmaily size greater than 4. Hence we will take a step further and categorize the Family size according to following rule:

1. familySize = 1 => Single
2. familySize > 1 && <=4 => Small
3. familySize > 4 => Big

```
trailData <- mutate(trailData, familyType = ifelse(familySize == 1, "Single", ifelse(familySize > 1 & fa
str(trailData)
```

```
## 'data.frame':    891 obs. of  14 variables:
```

```
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 58
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ familySize : num  2 2 1 2 1 1 1 5 3 2 ...
##  $ familyType : chr  "Small" "Small" "Single" "Small" ...
```

Cleaning Data and Imputation

To impute the age, lets introduce a new variable Age categorey using below rule:

1) If (number of spouses or siblings > 1) then it's probably a child and represented by 0
2) If (number of parents or children > 2) then it's probably an adult and represented by 1
3) If all above conditions are not met lets call it is unterermined and represented 2

```
trailData <- mutate(trailData, AgeCat = ifelse(SibSp > 1 , 0, ifelse(Parch > 2, 1, 2)))
```

Building model to predict age.

```
ageModel = rpart(Age ~ Fare + Pclass + SibSp + Parch + familySize + AgeCat, data = trailData)
trailData$predictAge = predict(ageModel,trailData)
trailData$Age <- ifelse(is.na(trailData$Age), trailData$predictAge, trailData$Age)
sum(is.na(trailData$Age))
```

```
## [1] 0
```

```
trailData$Age <- round(trailData$Age)
trailData <- mutate(trailData, fitness = ifelse(Age > 18 & Age <= 50, "Strong", "Weak"))
trailData$fitness <- as.factor(trailData$fitness)
str(trailData)
```

```
## 'data.frame':    891 obs. of  17 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : Factor w/ 891 levels "Abbing, Mr. Anthony",..: 109 191 358 277 16 559 520 629 417 58
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ Age        : num  22 38 26 35 35 28 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : Factor w/ 681 levels "110152","110413",..: 524 597 670 50 473 276 86 396 345 133 ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : Factor w/ 148 levels "","A10","A14",..: 1 83 1 57 1 1 131 1 1 1 ...
##  $ Embarked   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
##  $ familySize : num  2 2 1 2 1 1 1 5 3 2 ...
##  $ familyType : chr  "Small" "Small" "Single" "Small" ...
##  $ AgeCat     : num  2 2 2 2 2 2 2 0 2 2 ...
##  $ predictAge : num  27.8 36.2 27.8 36.2 27.8 ...
##  $ fitness    : Factor w/ 2 levels "Strong","Weak": 1 1 1 1 1 1 2 2 1 2 ...
```

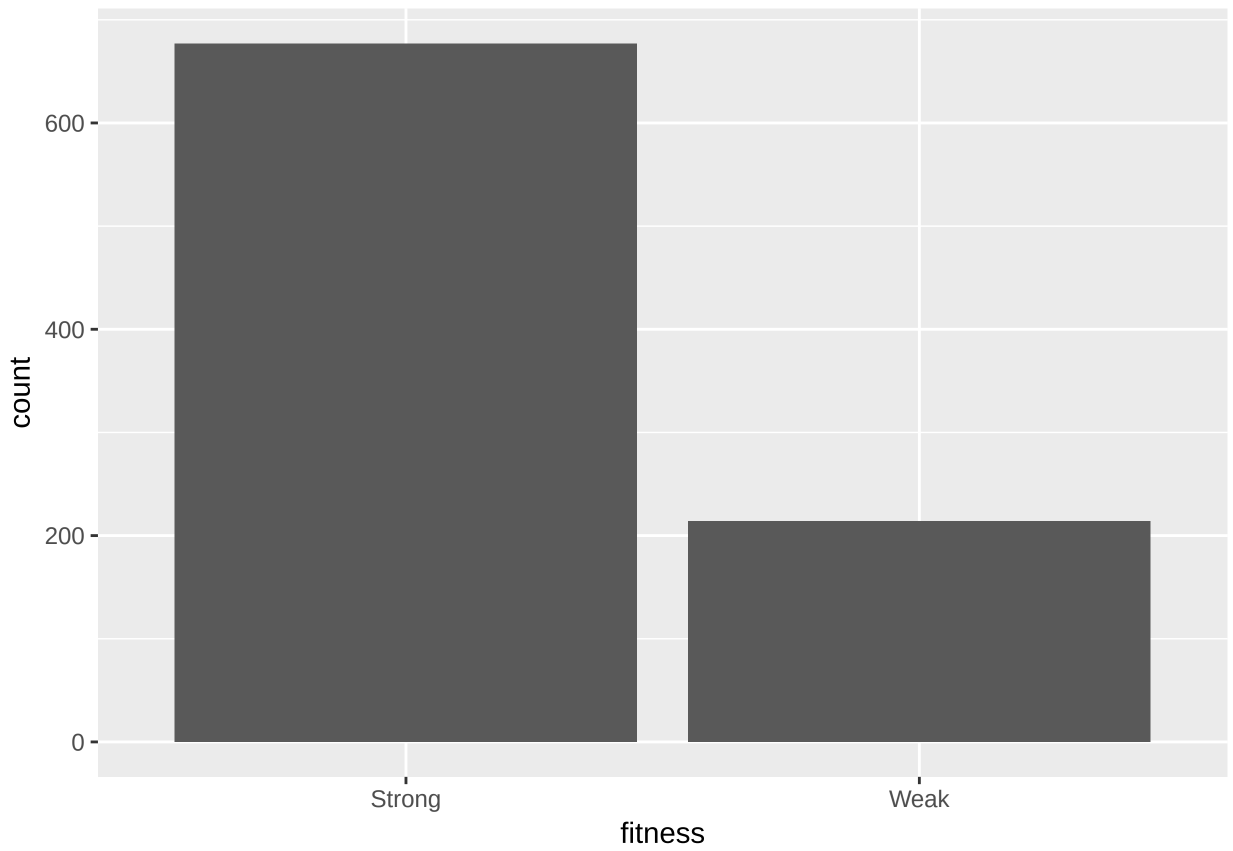
```
trailData$Age
```

```
##   [1] 22 38 26 35 35 28 54  2 27 14  4 58 20 39 14 55  2 33 31 28 35 34 15
##  [24] 28  8 38 28 19 28 28 40 36 28 66 28 42 28 21 18 14 40 27 28  3 19 28
##  [47] 28 28 28 18  7 21 49 29 65 46 21 28  5 11 22 38 45  4 46 21 29 19 17
##  [70] 26 32 16 21 26 32 25 28 28  1 30 22 29 28 28 17 33 16 28 23 24 29 20
##  [93] 46 26 59 28 71 23 34 34 28 28 21 33 37 28 21 28 38 28 47 14 22 20 17
## [116] 21 70 29 24  2 21 28 32 32 54 12 28 24 21 45 33 20 47 29 25 23 19 37
## [139] 16 24 21 22 24 19 18 19 27  9 36 42 51 22 56 40 28 51 16 30 28  8 44
## [162] 40 26 17  1  9 36 45 46 28 61  4  1 21 56 18  8 50 30 36  8 33  9  1
## [185]  4 36 28 45 40 36 32 19 19  3 44 58 28 42 28 24 28  8 34 46 18  2 32
## [208] 26 16 40 24 35 22 30 28 31 27 42 32 30 16 27 51 28 38 22 19 20 18  8
## [231] 35 29 59  5 24 28 44  8 19 33 28 28 29 22 30 44 25 24 37 54 28 29 62
## [254] 30 41 29 36 30 35 50 28  3 52 40 28 36 16 25 58 35 46 25 41 37 28 63
## [277] 45 33  7 35 65 28 16 19 46 33 30 22 42 22 26 19 36 24 24 46 24  2 46
## [300] 50 28 28 19 33 28  1 36 17 30 30 24 18 26 28 43 26 24 54 31 40 22 27
## [323] 30 22  8 36 61 36 31 16 28 46 38 16 36 28 29 41 45 45  2 24 28 25 36
## [346] 24 40 28  3 42 23 46 15 25 28 28 22 38 28 28 40 29 45 35 28 30 60 28
## [369] 28 24 25 18 19 22  3 36 22 27 20 19 42  1 32 35 28 18  1 36 28 17 36
## [392] 21 28 23 24 22 31 46 23 28 39 26 21 28 20 34 51  3 21  8 28 28 33 33
## [415] 44 28 34 18 30 10 28 21 29 28 18 28 28 19 28 32 28 28 42 17 50 14 21
## [438] 24 64 31 45 20 25 28 28  4 13 34  5 52 36 28 30 49 28 29 65 36 50 28
## [461] 48 34 47 48 28 38 33 56 28  1 28 38 33 23 22 36 34 29 22  2  9 33 50
## [484] 63 25  8 35 58 30  9 28 21 55 71 21 28 54 28 25 24 17 21 28 37 16 18
## [507] 33 46 28 26 29 28 36 54 24 47 34 28 36 32 30 22 28 44 28 40 50 36 39
## [530] 23  2 28 17 21 30  7 45 30 28 22 36  9 11 32 50 64 19 33 33  8 17 27
## [553] 28 22 22 62 48 36 39 36 28 40 28 28 28 24 19 29 28 32 62 53 36 28 16
## [576] 19 34 39 28 32 25 39 54 36 28 18 47 60 22 28 35 52 47 21 37 36 33 49
## [599] 28 49 24 28 36 44 35 36 30 27 22 40 39 28 28 28 35 24 34 26  4 26 27
## [622] 42 20 21 21 61 57 21 26 28 80 51 32 46  9 28 32 31 41 28 20 24  2 28
## [645]  1 48 19 56 28 23 28 18 21 28 18 24 28 32 23 58 50 40 47 36 20 32 25
## [668] 28 43 36 40 31 70 31 33 18 24 18 43 36 28 27 20 14 60 25 14 19 18 15
## [691] 31  4 28 25 60 52 44 28 49 42 18 35 18 25 26 39 45 42 22 21 24 46 48
## [714] 29 52 19 38 27 28 33  6 17 34 50 27 20 30 28 25 25 29 11 33 23 23 28
## [737] 48 35 28 28 46 36 21 24 31 70 16 30 19 31  4  6 33 23 48  1 28 18 34
## [760] 33 28 41 20 36 16 51 36 30 28 32 24 48 57 28 54 18 28  5 28 43 13 17
## [783] 29 21 25 25 18  8  1 46 28 16  8 46 25 39 49 31 30 30 34 31 11  0 27
## [806] 31 39 18 39 33 26 39 35  6 30 46 23 31 43 10 52 27 38 27  2 28 28  1
## [829] 28 62 15  1 28 23 18 39 21 28 32 46 20 16 30 34 17 42  8 35 28 36  4
## [852] 74  9 16 44 18 45 51 24 28 41 21 48  8 24 42 27 31 28  4 26 47 33 47
## [875] 28 15 20 19 28 56 25 33 22 28 25 39 27 19 21 26 32
```

```
table(trailData$fitness, trailData$Survived)
```

```
## 
##            0   1
##   Strong 427 250
##   Weak   122  92
```

```
plot1 <- ggplot(trailData, aes(fitness, ..count..)) + geom_bar(aes(fill = Survived), position = "dodge"
plot1
```



Data Variables

Processing and Cleaning Data