final-titanic

2154177 何应豪　　2151298 杨滕超　　2151299 苏家铭　　2151294 马威

目录

1 研究问题和假设

1.1 背景

泰坦尼克号的沉没是历史上最著名的沉船事故之一。1912 年 4 月 15 日，在处女航中，被认为 “永不沉没” 的皇家邮轮泰坦尼克号与冰山相撞后沉没。不幸的是，船上没有足够的救生艇容纳所有人，导致 2224 名乘客和船员中的 1502 人死亡。尽管全世界依然沉浸在惨烈的事故所带来的伤痛当中，但是有一些人却认为

虽然生存有一些运气因素，但似乎有些群体比其他群体更有可能生存下来，因此他们尽可能收集到了一些乘客数据（无论是否已故），想看看什么样的人更有可能存活下来。我们很幸运地获取到了这些数据，希望通过乘客信息建立分类预测模型，尝试解答 “什么样的人更有可能活下来” 的迷思。

1.2 现有研究

已搜集到近 900 名乘客的信息，包括舱位等级、姓名、性别、年龄、同乘兄弟姐妹/配偶数量、票价、登船港口等。是否存活是一个二分类变量，0 为死亡，1 为存活。

1.3 研究问题

什么样的人更有可能活下来？

1.4 研究假设

1. 你敢假定？

2 数据分析方法选择

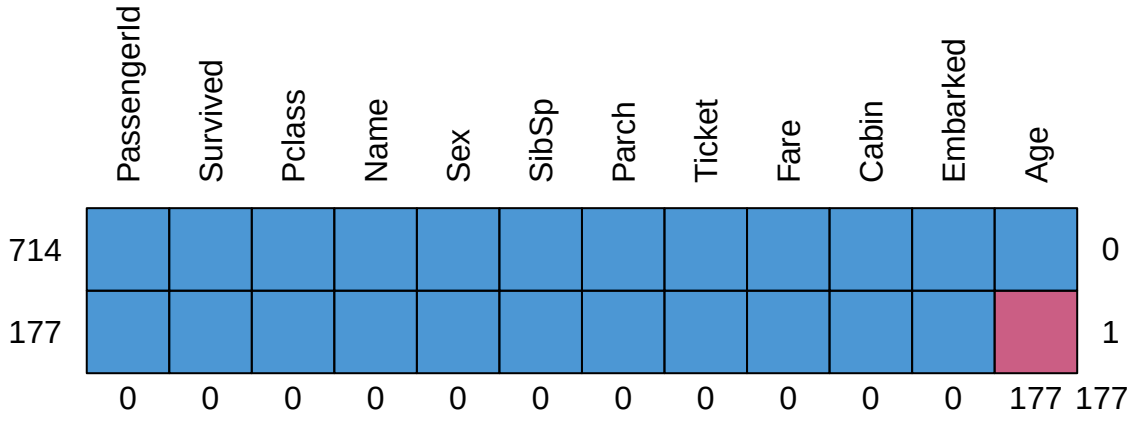
1. 本实验需要进行预测，因变量是否存活是一个 “0/1” 二分类问题，且有多个解释变量。适合使用广义线性回归建立预测模型

3 基础分析

3.1 准备工作

```
# 导入必要的包
library(caret)     # 生成训练集和测试集
library(ROCR)      # glm
library(pROC)      # ROC 面积计算
library(ggplot2)   # 画图
library(mice)      # 缺失值可视化

# 清除当前镜像中的数据
rm(list = ls())

# 导入数据
```

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

# 预处理判断
md.pattern(data, rotate.names = TRUE)    # 检测缺失值
```




```
##     PassengerId Survived Pclass Name Sex SibSp Parch Ticket Fare Cabin Embarked
## 714           1        1      1    1   1     1     1      1    1     1        1
## 177           1        1      1    1   1     1     1      1    1     1        1
##               0        0      0    0   0     0     0      0    0     0        0
##     Age
## 714   1   0
## 177   0   1
##     177 177
```

```r
print(length(which(duplicated(data))))  # 检测重复行
```

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

3.2 数据预处理

3.2.1 处理缺失值

```
# 年龄（缺失形式：NA）
data$Age[is.na(data$Age)] <- mean(data$Age, na.rm = TRUE)  # 使用平均值填充

# 登船港口（缺失形式："")
freq.table <- table(data$Embarked)  # 使用 table 函数计算频数
mode <- names(freq.table)[which.max(freq.table)]  # 找到频数最大的元素（众数）
data$Embarked[nchar(data$Embarked) == 0] <- mode  # 使用众数填补
rm(freq.table)
rm(mode)
```

3.2.2 数据规范

```
data$Fare <- scale(data$Fare)  # 票价过于分散，因此进行规范处理
```

3.2.3 指定因子水平

```
data$Survived <- as.factor(data$Survived)  # 是否存活
data$Sex <- as.factor(data$Sex)            # 性别
data$Embarked <- as.factor(data$Embarked)  # 登船港口
```

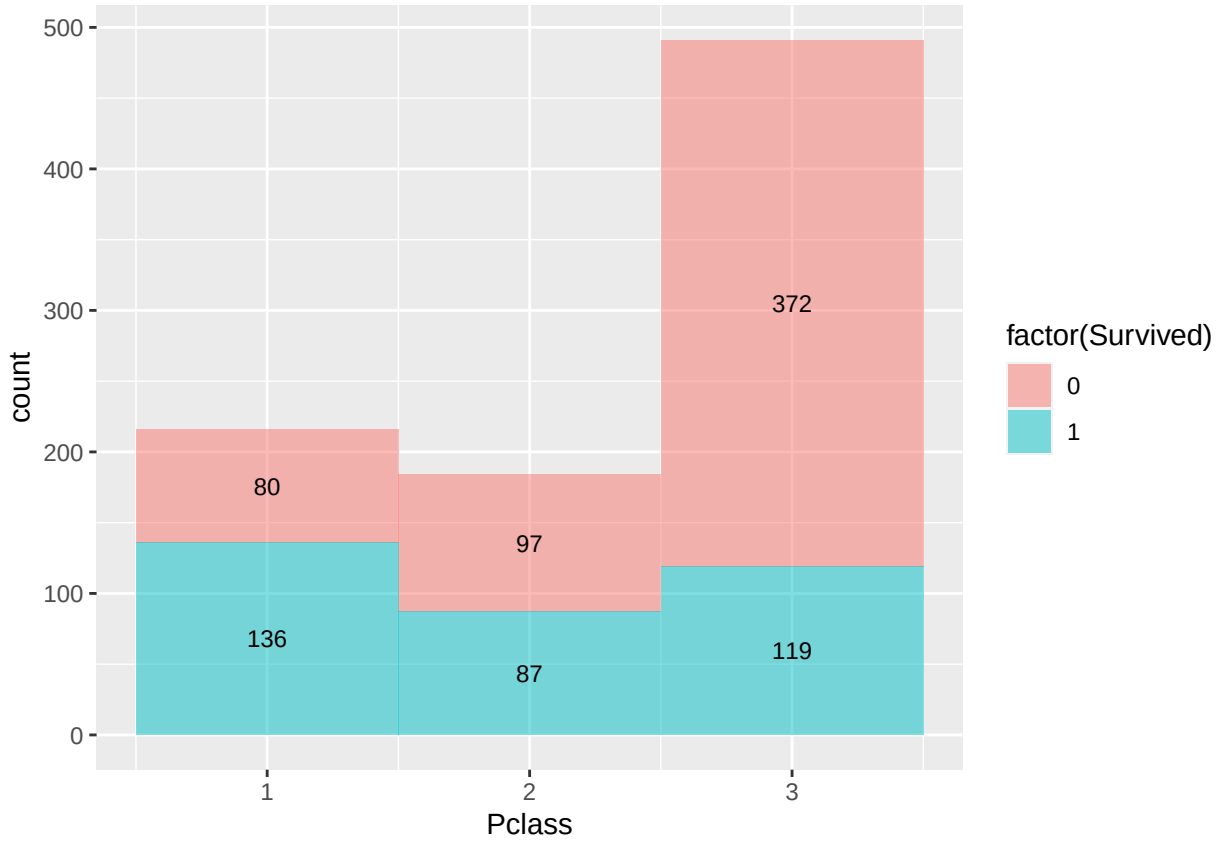
3.3 初步分析

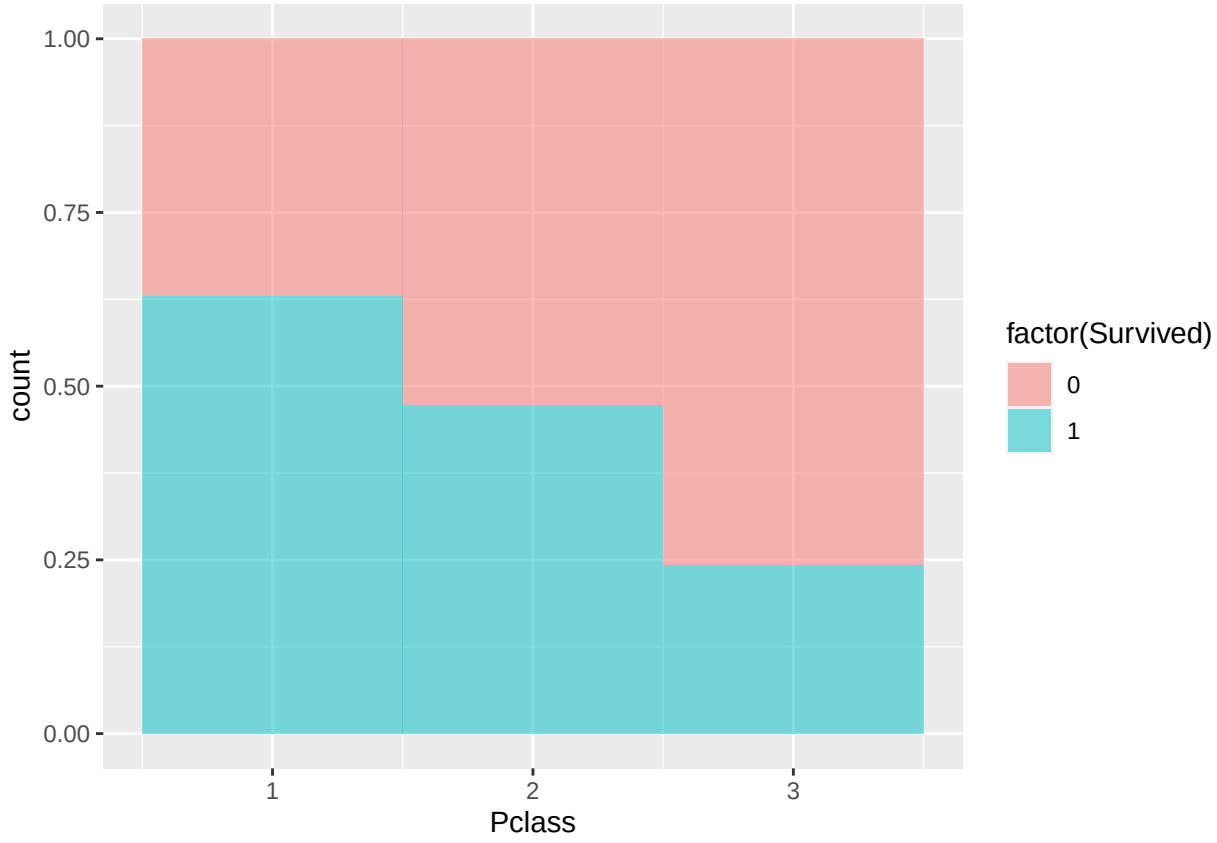
```
# 选取所需数据进行初步分析
data <- data[, c("Survived", "Pclass", "Sex", "Age", "SibSp", "Fare", "Embarked")]
summary(data)

# 各因素的可视化
# 舱位等级
ggplot(data, aes(x = Pclass, fill = factor(Survived))) +
  geom_bar(binwidth = 1, position = "stack", stat = "bin", alpha = 0.5) +
  geom_text(
    stat = "count",
    aes(label = stat(count)),
    position = position_stack(vjust = 0.5),
```

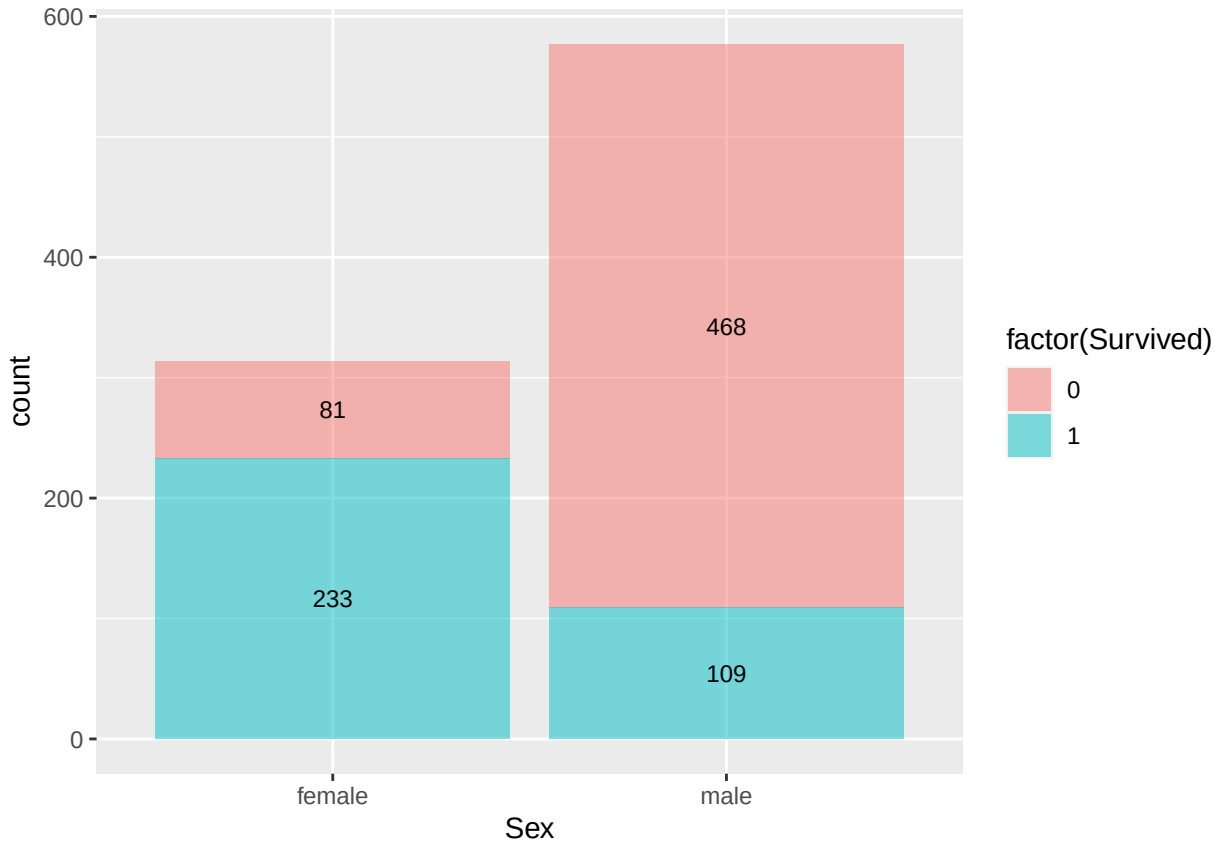
```
    size = 3
  )
ggplot(data, aes(x = Pclass, fill = factor(Survived))) +
  geom_bar(binwidth = 1, position = "fill", stat = "bin", alpha = 0.5)

# 性别
ggplot(data, aes(x = Sex, fill = factor(Survived))) +
  geom_bar(position = "stack", alpha = 0.5) +
  geom_text(
    stat = "count",
    aes(label = stat(count)),
    position = position_stack(vjust = 0.5),
    size = 3
  )
ggplot(data, aes(x = Sex, fill = factor(Survived))) +
  geom_bar(position = "fill", alpha = 0.5)

# 年龄
ggplot(data, aes(x = Age, fill = factor(Survived))) +
  geom_bar(binwidth = 2, position = "stack", stat = "bin", alpha = 0.5) +
  geom_text(
    stat = "count",
    aes(label = stat(count)),
    position = position_stack(),
    size = 1
  )
ggplot(data, aes(x = Age, fill = factor(Survived))) +
  geom_bar(binwidth = 2, position = "fill", stat = "bin", alpha = 0.5)

# 同乘兄弟姐妹/配偶
ggplot(data, aes(x = SibSp, fill = factor(Survived))) +
  geom_bar(binwidth = 1, position = "stack", stat = "bin", alpha = 0.5) +
  geom_text(
    stat = "count",
    aes(label = stat(count)),
    position = position_stack(vjust = 0.5),
    size = 3
  )
ggplot(data, aes(x = SibSp, fill = factor(Survived))) +
  geom_bar(binwidth = 1, position = "fill", stat = "bin", alpha = 0.5)
```

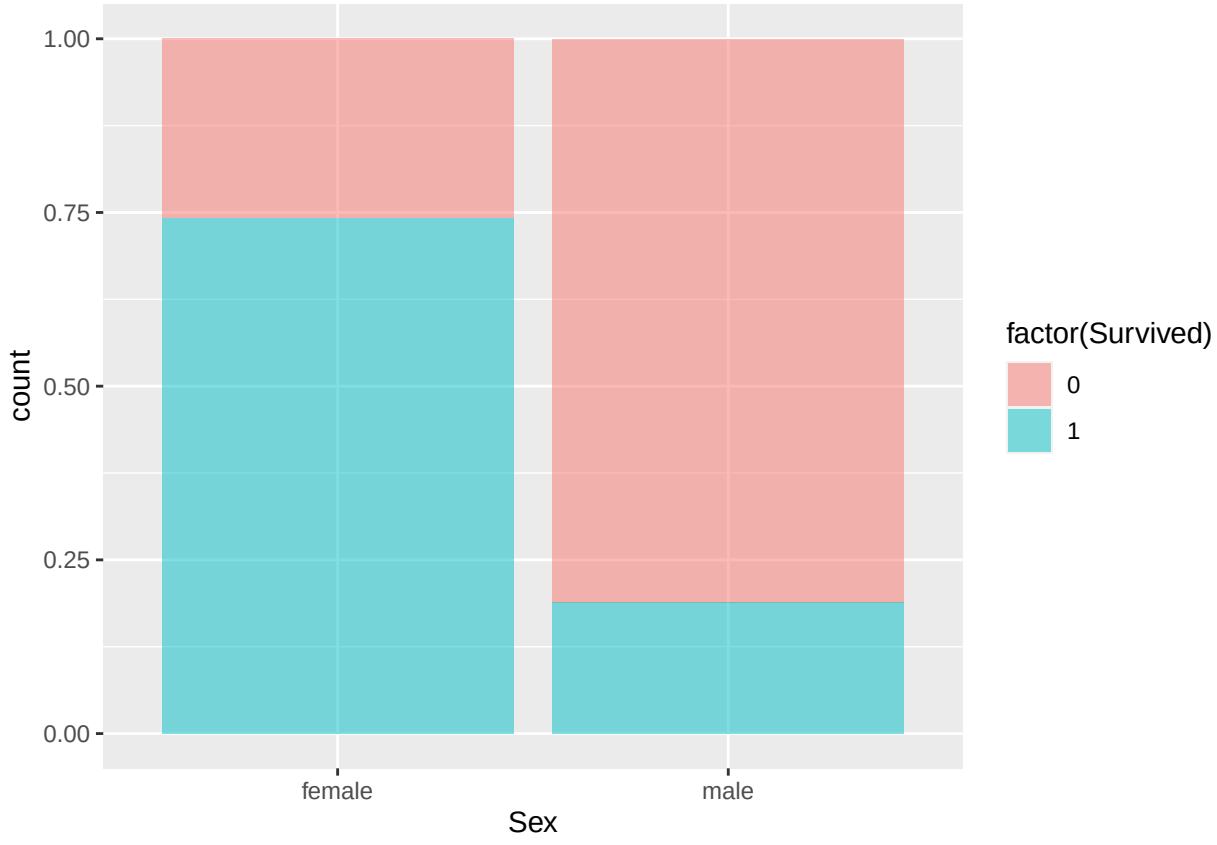
```
# 票价
ggplot(data, aes(x = Fare, fill = factor(Survived))) +
  geom_bar(binwidth = 0.2, position = "stack", stat = "bin", alpha = 0.5)
ggplot(data, aes(x = Fare, fill = factor(Survived))) +
  geom_bar(binwidth = 0.2, position = "fill", stat = "bin", alpha = 0.5)

# 登船港口
ggplot(data, aes(x = Embarked, fill = factor(Survived))) +
  geom_bar(position = "stack", alpha = 0.5) +
  geom_text(
    stat = "count",
    aes(label = stat(count)),
    position = position_stack(vjust = 0.5),
    size = 3
  )
ggplot(data, aes(x = Embarked, fill = factor(Survived))) +
  geom_bar(position = "fill", alpha = 0.5)
```

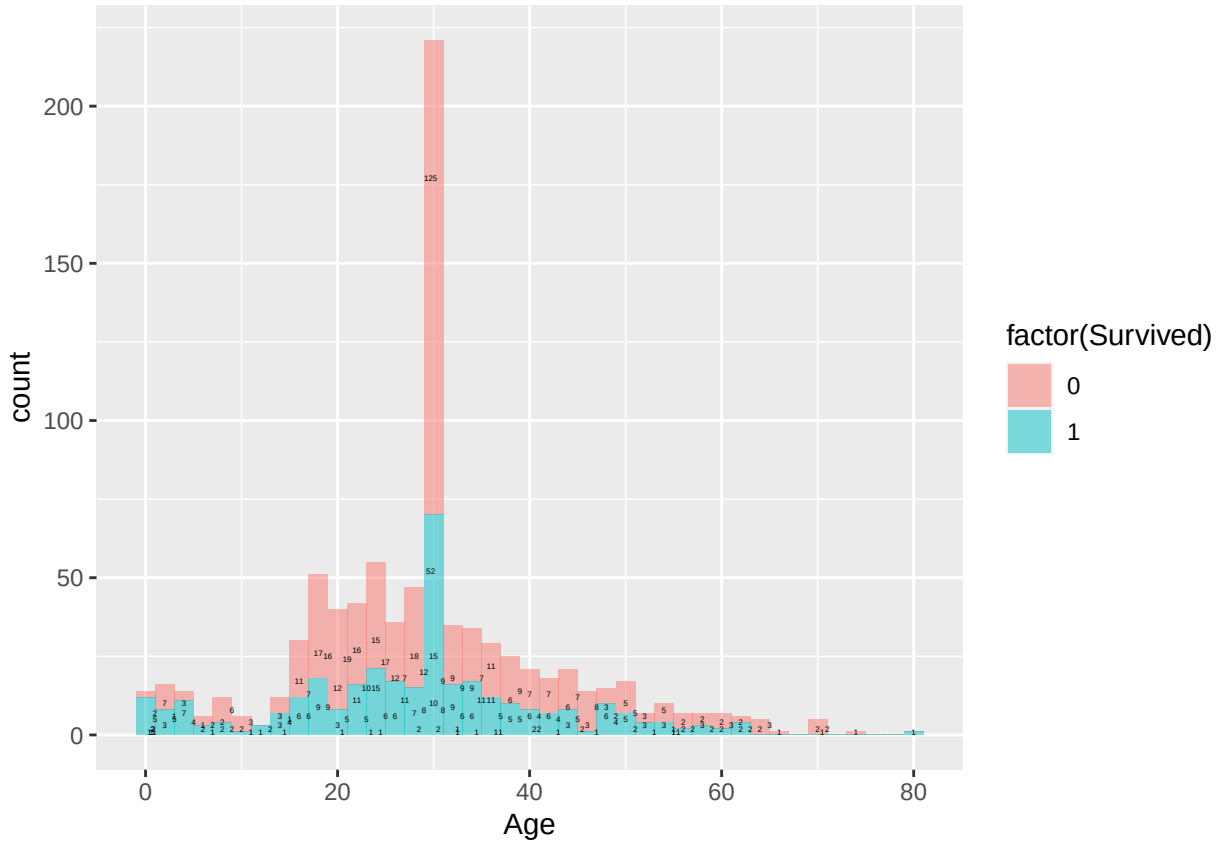
```
##  Survived     Pclass          Sex           Age            SibSp
##  0:549    Min.   :1.000   female:314   Min.   : 0.42   Min.   :0.000
##  1:342    1st Qu.:2.000   male  :577   1st Qu.:22.00   1st Qu.:0.000
##           Median :3.000                Median :29.70   Median :0.000
##           Mean   :2.309                Mean   :29.70   Mean   :0.523
##           3rd Qu.:3.000                3rd Qu.:35.00   3rd Qu.:1.000
##           Max.   :3.000                Max.   :80.00   Max.   :8.000
##        Fare.V1       Embarked
##  Min.   :-0.648058   C:168
##  1st Qu.:-0.488874   Q: 77
##  Median :-0.357190   S:646
##  Mean   : 0.000000
##  3rd Qu.:-0.024233
##  Max.   : 9.661740
```

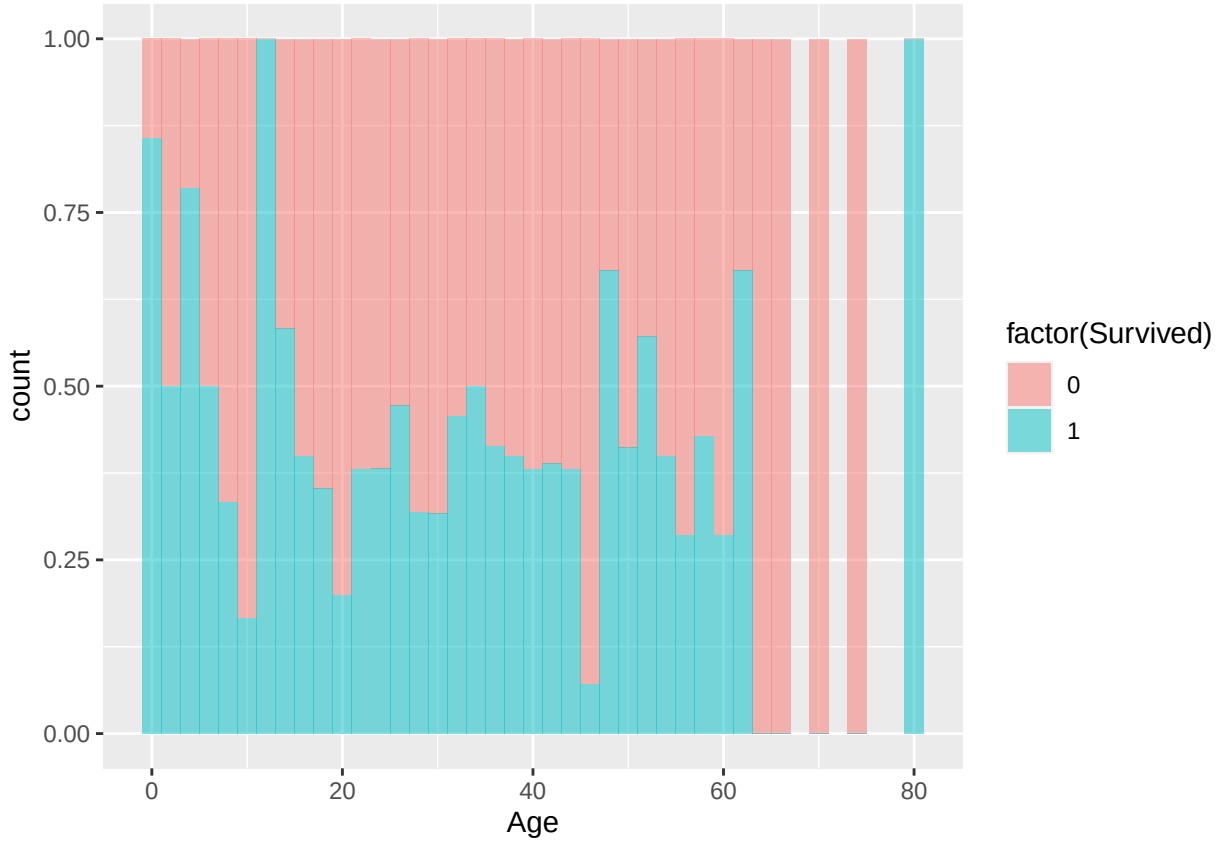

500
400
300
200
100
0
count
80
136
97
87
372
119
1
2
3
Pclass
factor(Survived)
0
1

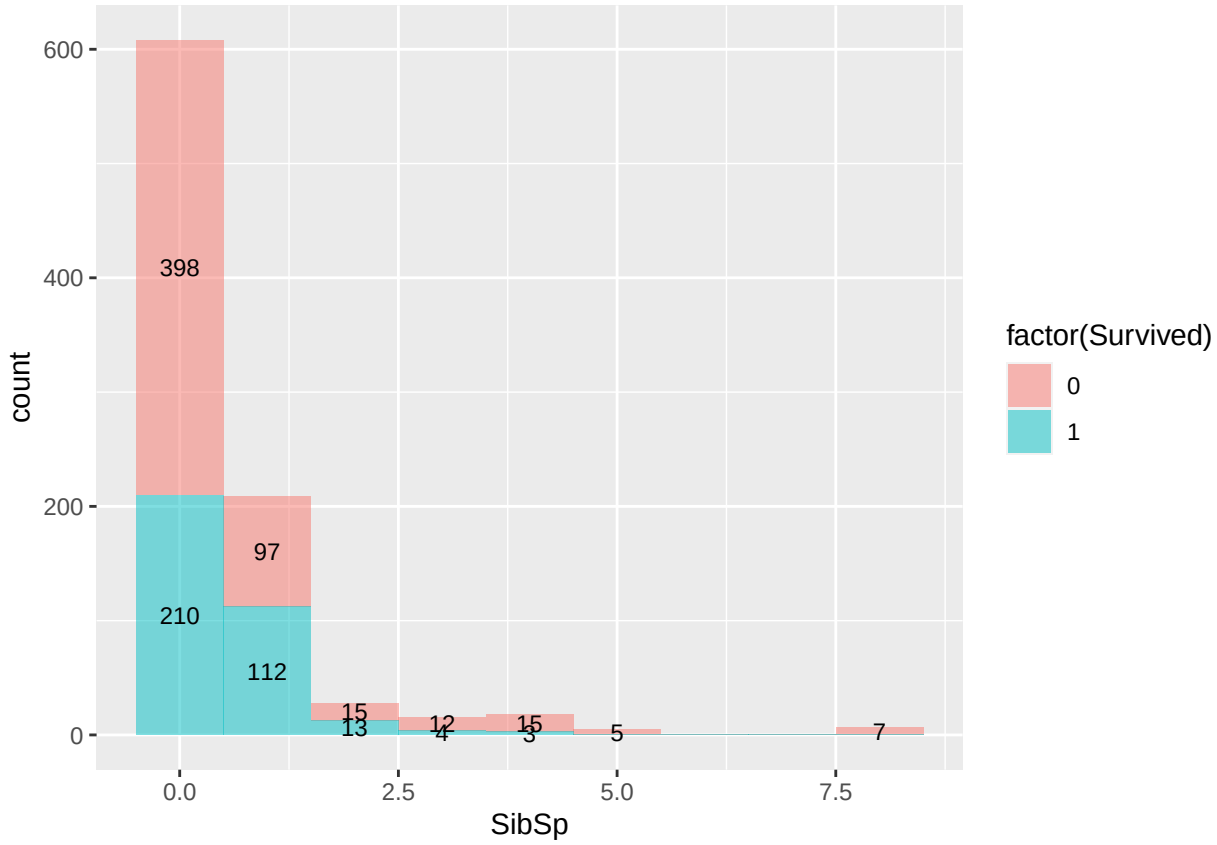

1.00
0.75
0.50
0.25
0.00
count
1
2
3
Pclass
factor(Survived)
0
1

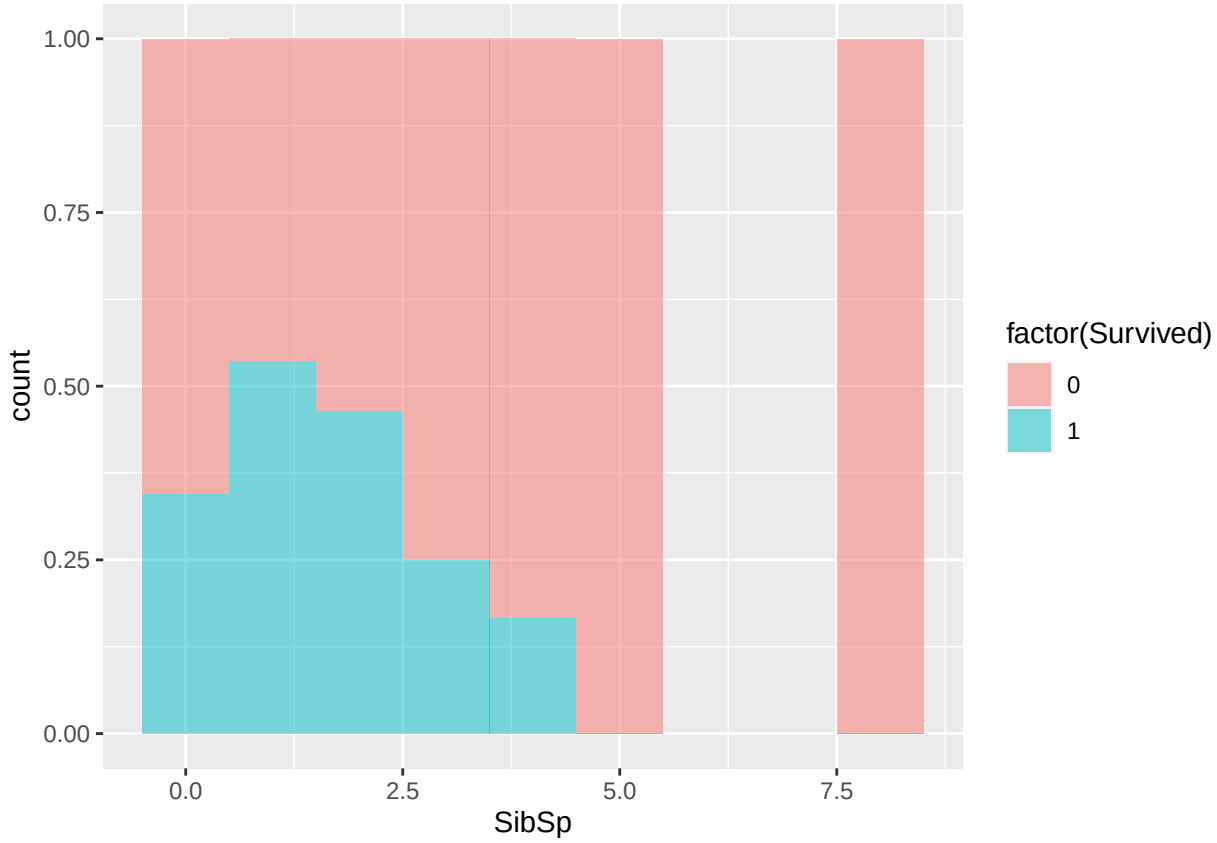

600
400
200
0
count
81
233
468
109
female
male
Sex
factor(Survived)
0
1

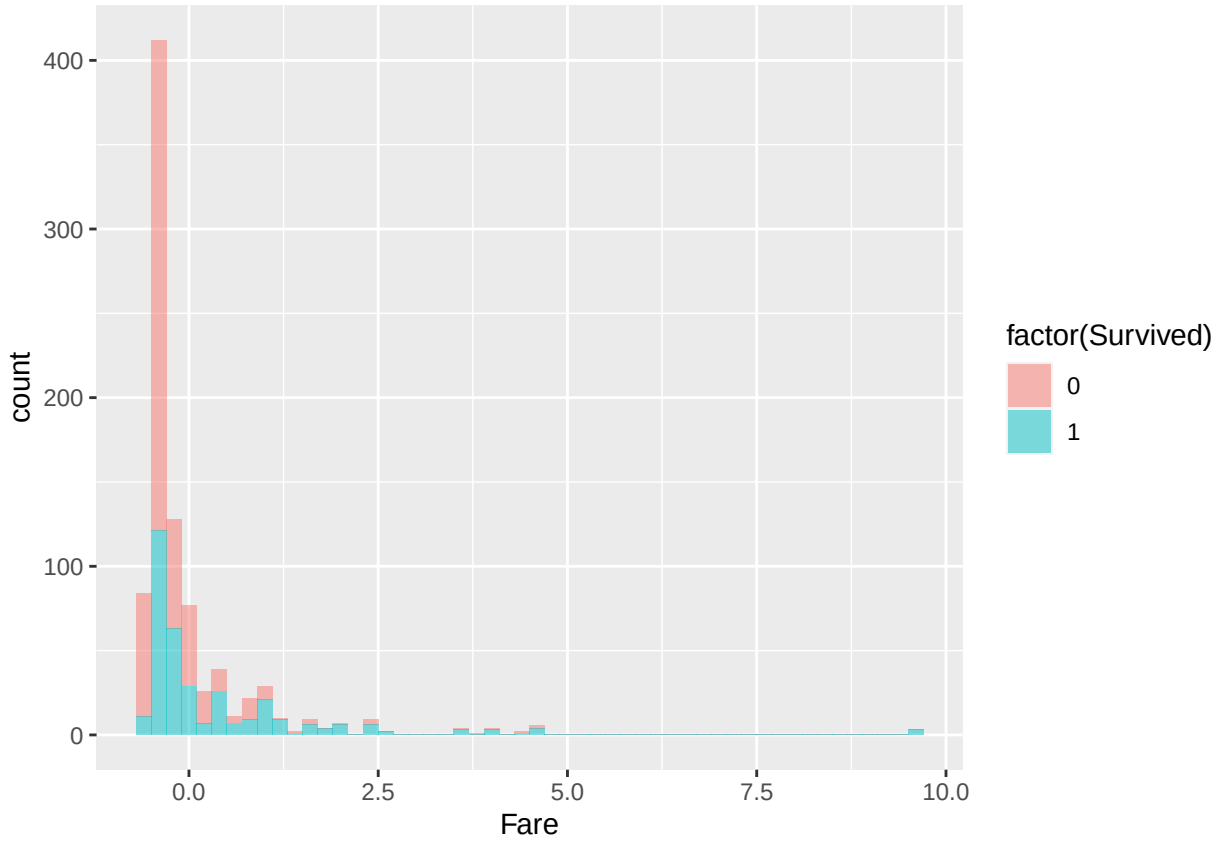

1.00
0.75
0.50
0.25
0.00
count
female
male
Sex
factor(Survived)
0
1

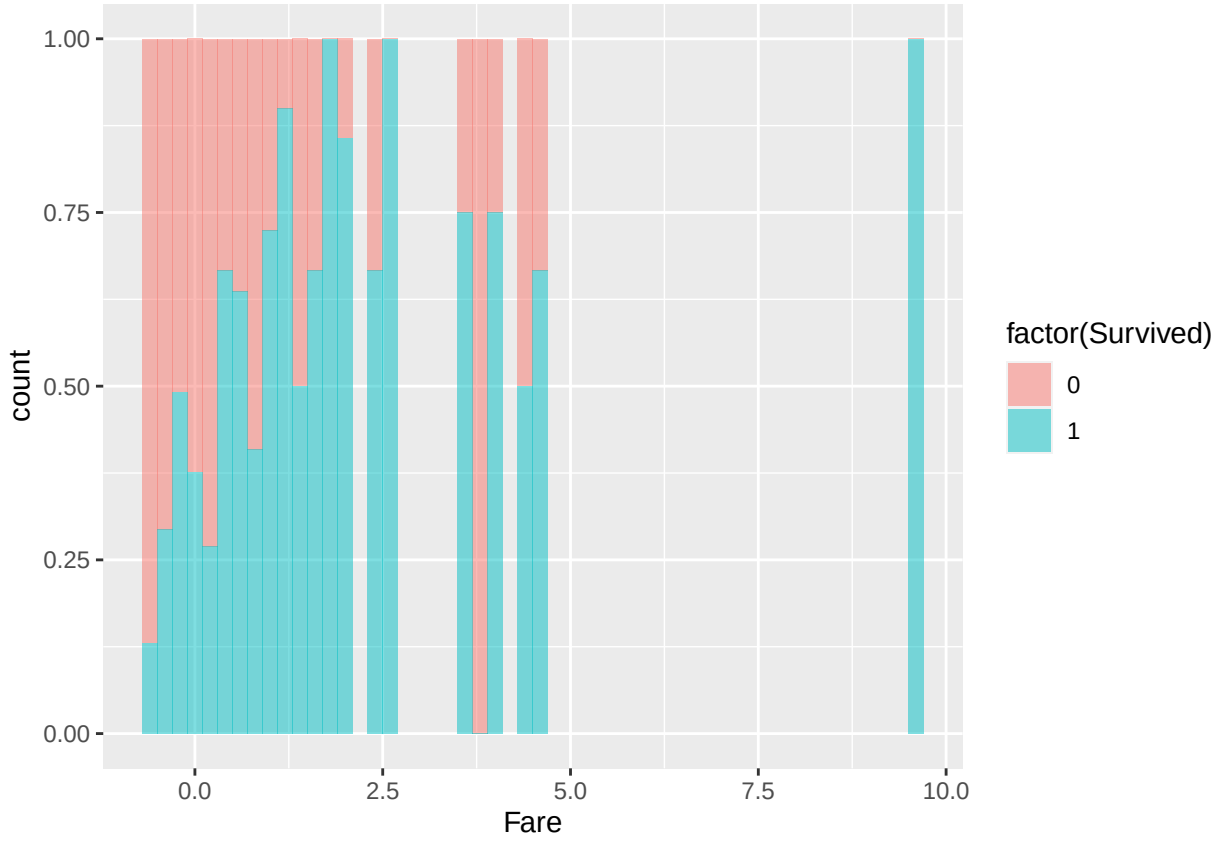

count
200
150
100
50
0
0
20
40
60
80
Age
factor(Survived)
0
1

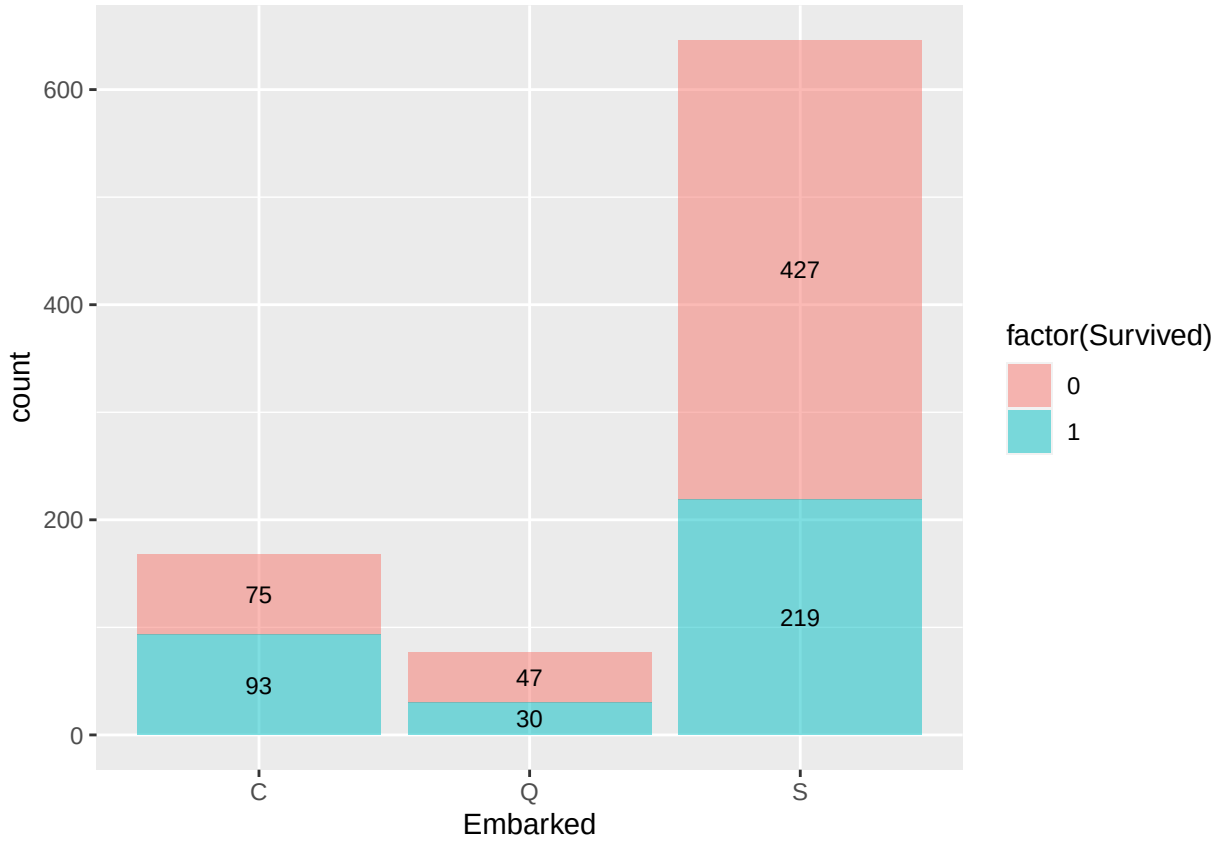

count
1.00
0.75
0.50
0.25
0.00
0
20
40
60
80
Age
factor(Survived)
0
1

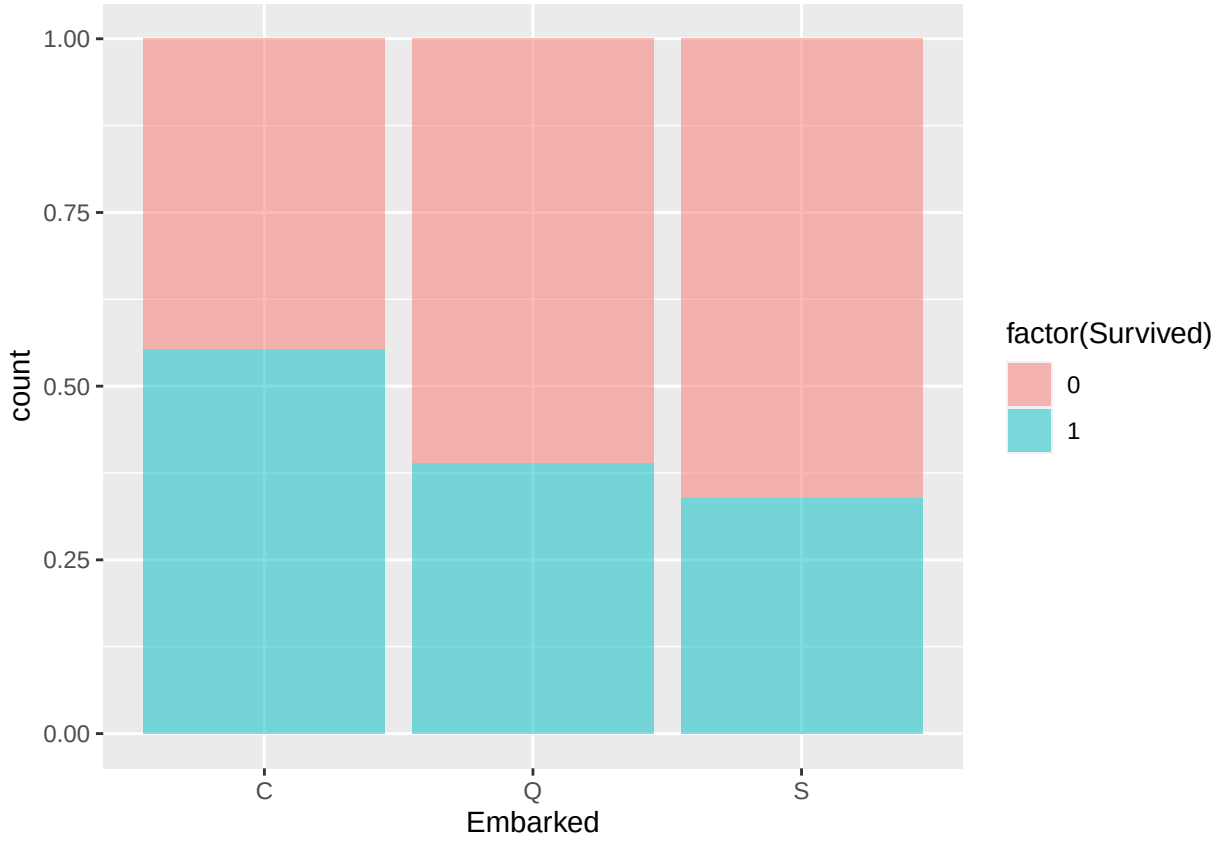

600
400
200
0
count
398
97
210
112
15
13
12
4
15
3
5
7
0.0
2.5
5.0
7.5
SibSp
factor(Survived)
0
1


1.00
0.75
0.50
0.25
0.00
count
0.0
2.5
5.0
7.5
SibSp
factor(Survived)
0
1


400
300
200
100
0
count
0.0
2.5
5.0
7.5
10.0
Fare
factor(Survived)
0
1


1.00
0.75
0.50
0.25
0.00
count
0.0
2.5
5.0
7.5
10.0
Fare
factor(Survived)
0
1


600
400
200
0
count
427
75
219
93
47
30
C
Q
S
Embarked
factor(Survived)
0
1


1.00
0.75
0.50
0.25
0.00
count
C
Q
S
Embarked
factor(Survived)
0
1

4 主要分析

4.1 进行数据划分

```
set.seed(123)
split <- createDataPartition(data$Survived, p = 0.7, list = FALSE)
train_data <- data[split, ]  # 训练集
test_data <- data[-split, ]  # 测试集
```

4.2 建立预测模型（全因素）

```
model <- glm(Survived ~ ., data = train_data, family = binomial)
summary(model)
```

```
##
## Call:
## glm(formula = Survived ~ ., family = binomial, data = train_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.986042   0.688987   8.688  < 2e-16 ***
## Pclass      -1.244643   0.178360  -6.978 2.99e-12 ***
## Sexmale     -2.880914   0.245019 -11.758  < 2e-16 ***
## Age         -0.039244   0.009513  -4.125 3.70e-05 ***
## SibSp       -0.338020   0.140778  -2.401  0.01635 *
## Fare         0.008242   0.139717   0.059  0.95296
## EmbarkedQ   -0.313126   0.454168  -0.689  0.49054
## EmbarkedS   -0.791667   0.284583  -2.782  0.00541 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 832.49  on 624  degrees of freedom
## Residual deviance: 522.46  on 617  degrees of freedom
## AIC: 538.46
##
## Number of Fisher Scoring iterations: 5
```

4.3 建立预测模型（显著影响因素）

```
model <- glm(Survived ~ Pclass+Sex+Age+SibSp+Embarked, data = train_data, family = binomial)
summary(model)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Age + SibSp + Embarked,
##     family = binomial, data = train_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  6.001759   0.635738   9.441  < 2e-16 ***
## Pclass      -1.250068   0.152922  -8.175 2.97e-16 ***
## Sexmale     -2.882160   0.244132 -11.806  < 2e-16 ***
## Age         -0.039289   0.009482  -4.144 3.42e-05 ***
## SibSp       -0.336351   0.137875  -2.440  0.01471 *
## EmbarkedQ   -0.315139   0.452946  -0.696  0.48658
## EmbarkedS   -0.794794   0.279609  -2.843  0.00448 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 832.49  on 624  degrees of freedom
## Residual deviance: 522.46  on 618  degrees of freedom
## AIC: 536.46
##
## Number of Fisher Scoring iterations: 5
```

4.4 模型评估

```
pred <- predict(model, test_data, type = "response")
auc <- roc(test_data$Survived, pred)
```

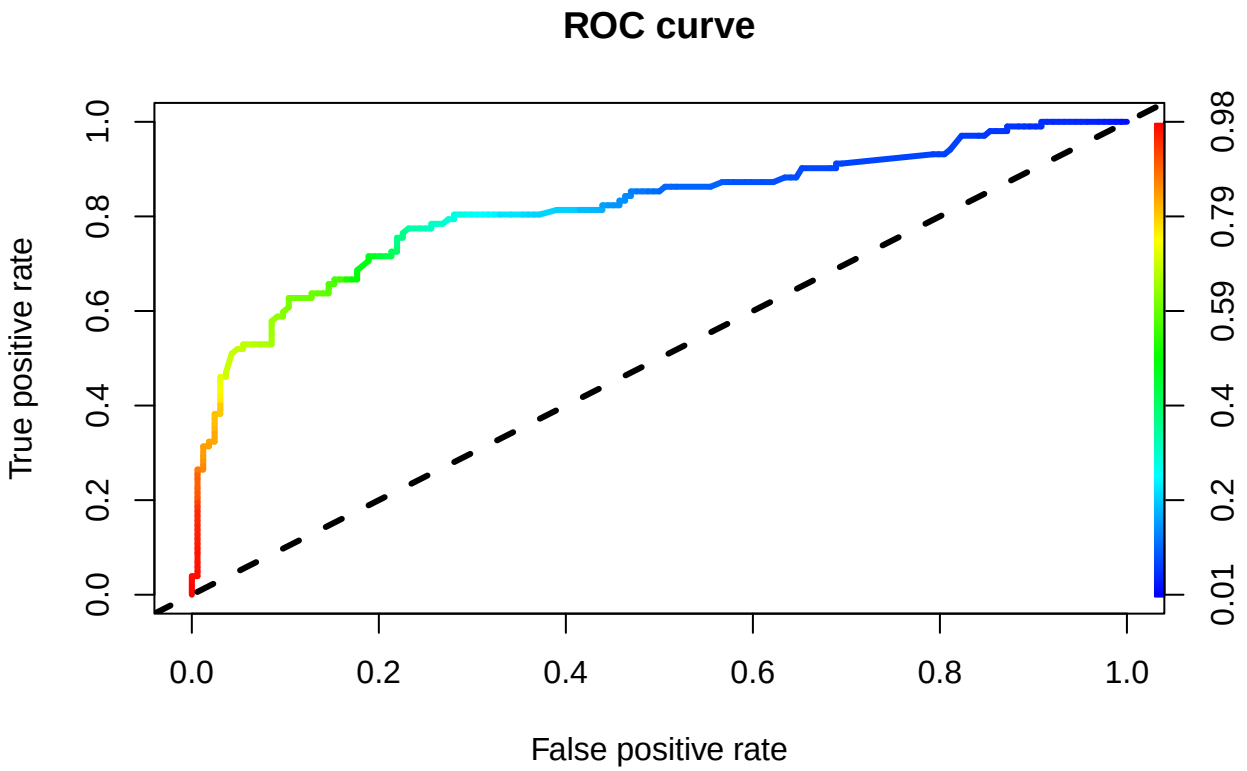
```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
auc$auc
pred <- prediction(pred, test_data$Survived)
plot(performance(pred, "tpr", "fpr"), colorize = T, lwd = 3, main = "ROC curve")
```

```r
abline(a = 0, b = 1, lty = 2, lwd = 3, col = "black")
```

```
## Area under the curve: 0.8166
```



5 分析结果解读

5.1 对生存率的各影响因素（由于 1 为生存，所以将 odds 称为生存几率）：

```r
print(exp(model$coefficients))
```

```
##  (Intercept)       Pclass      Sexmale          Age        SibSp    EmbarkedQ
## 404.13917928   0.28648518   0.05601362   0.96147289   0.71437219   0.72968760
##    EmbarkedS
##   0.45167409
```

5.1.1 舱位等级

- **预测：**每下降一级（等级数 +1），生存几率**降为**原来的 **0.286 倍**
- **解读：**能坐到更高等级舱位的人可能会有更高的地位，在紧急情况下就有可能被“优待”得更多。也可能是富人并没有参与救生艇的共同分配，能自己搞到救生艇提前逃跑。

5.1.2 性别

- **预测：**为男性时，生存几率**降为**原来的 **0.056 倍**
- **解读：**女士优先

5.1.3 年龄

- **预测：**每大一岁，生存几率**降为**原来的 **0.961 倍**
- **解读：**小孩优先

5.1.4 同乘兄弟姐妹/配偶数量

- **预测：**每多 1 个，生存几率**降为**原来的 **0.714 倍**
- **解读：**落单的人优先

5.1.5 登船港口

- **预测：**在 S 港（Southampton）登船时，生存几率**降为**原来的 **0.452 倍**
- **解读：**在南安普顿港登船的人可能社会阶级更高，也可能是在登船港口进行缺失值处理时进行了众数填补，从而导致了更多生还的人被划入 S 港登船人群

5.2 总结

1. 就搜集到的数据以及生成的预测模型上看，的确一些因素对于生存率有显著影响
2. 高等级舱位/女性/年轻人/落单/不在 Southampton 登船，满足以上部分因素的人更有可能活下来
3. 在主要分析中生成的模型里，仅保留显著影响因素的模型 AIC 值较小，因此选择它进行后续评估分析。其 AUC 值为 $0.817 > 0.8$，测试集结果说明，该模型具有相对较好的分类能力
4. 总而言之，现在再回头分析 “哪些人更有可能活下来” 这类问题，我们可以窥视到历史上一场惨烈事故中的众生百态：舍己为人、尔虞我诈、争先恐后、绝望等待……同样，也可以看到当时社会的一些价值观体现。但今时不同以往，了解到这段历史的沉痛后，在新的社会环境下，随着航海技术、人们的观念等发生改变，或许这类问题有了重新研究的价值。但我们更希望的是事故永远不会发生，大家平平安安，一帆风顺。