R-ML-Titanic

EMMA

Saturday, August 26, 2017

本周内容：

机器学习实战：泰坦尼克乘客生还预测

教材位置：

http://trevorstephens.com/kaggle-titanic-tutorial/getting-started-with-r/

西瓜书 http://pan.baidu.com/s/1b7mbQu （感谢@逍遥子晴分享）

MATLAB 机器学习宣传手册
https://github.com/simon19891101/Books/tree/master/MATLAB

http://pan.baidu.com/s/1c0FRdW 《R for Data Science - Import, Tidy, Transform, Visualize and Model Data》(感谢@想分享)

http://pan.baidu.com/s/1gf6Bw35 《R 实战第二版中文》(感谢@想分享)

学习要求：

注册 Kaggle 账号， 对泰坦尼克乘客数据进行分析并预测，方式不限，算法不限， 将分析流程生成 pdf 文档上传至 github，并将预测结果上传至 Kaggle， 最终得分需附加在 pdf 报告中.

泰坦尼克乘客生还预测过程

设置工作区间

```
getwd()
```

```
## [1] "D:/Data-Analysis/R/R_ML/R_ML-Titanic"
```

```
setwd("D:/Data-Analysis/R/R_ML/R_ML-Titanic")
getwd()
```

```
## [1] "D:/Data-Analysis/R/R_ML/R_ML-Titanic"
```

导入数据

点击 Import Dataset 按钮选择数据源，数据自动导入。数据路径有中文时会报错可写代码导入 欠报错截图，自动生成导数代码截图

```
train<- read.csv("D:/Data-Analysis/R/R_ML/R_ML-Titanic/Data/train.csv")
test<- read.csv("D:/Data-Analysis/R/R_ML/R_ML-Titanic/Data/test.csv")
```

了解数据摘要

```
str(train)
```

```
## '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",..: 10
9 191 358 277 16 559 520 629 417 581 ...
##  $ 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 59
7 670 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 ...
```

```
str(test)
```

```
## 'data.frame':    418 obs. of  11 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : Factor w/ 418 levels "Abbott, Master. Eugene Jose
ph",..: 210 409 273 414 182 370 85 58 5 104 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 1
 2 1 2 ...
##  $ Age        : num  34.5 47 62 27 22 14 30 26 18 21 ...
##  $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
##  $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Ticket     : Factor w/ 363 levels "110469","110489",..: 153 22
```

```
2 74 148 139 262 159 85 101 270 ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : Factor w/ 77 levels "","A11","A18",..: 1 1 1 1 1
1 1 1 1 1 ...
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1
 3 ...
```

```
table(train$Survived)#table()统计汇总函数
```

```
##
##   0   1
## 549 342
```

```
prop.table(table(train$Survived))#占比
```

```
##
##         0         1
## 0.6161616 0.3838384
```

处理数据

预测一：生还率较小

假设所有的数据都为 0，从训练集数据来看有 62%的正确率

```
head(test$Survived)
```

```
## NULL
```

```
test$Survived<-rep(0,418)#为Survied 变量设值为0（死），rep()重复设置值函数。
str(test)
```

```
## 'data.frame':    418 obs. of  12 variables:
##  $ PassengerId: int  892 893 894 895 896 897 898 899 900 901 ...
##  $ Pclass     : int  3 3 2 3 3 3 3 2 3 3 ...
##  $ Name       : Factor w/ 418 levels "Abbott, Master. Eugene Jose
ph",..: 210 409 273 414 182 370 85 58 5 104 ...
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 2 2 1 2 1
 2 1 2 ...
##  $ Age        : num  34.5 47 62 27 22 14 30 26 18 21 ...
##  $ SibSp      : int  0 1 0 0 1 0 0 1 0 2 ...
##  $ Parch      : int  0 0 0 0 1 0 0 1 0 0 ...
##  $ Ticket     : Factor w/ 363 levels "110469","110489",..: 153 22
2 74 148 139 262 159 85 101 270 ...
##  $ Fare       : num  7.83 7 9.69 8.66 12.29 ...
##  $ Cabin      : Factor w/ 77 levels "","A11","A18",..: 1 1 1 1 1
1 1 1 1 1 ...
```

```
##  $ Embarked   : Factor w/ 3 levels "C","Q","S": 2 3 2 3 3 3 2 3 1
 3 ...
##  $ Survived   : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
submit <- data.frame(PassengerId = test$PassengerId, Survived = test
$Survived)
write.csv(submit, file = "theyallperish.csv", row.names = FALSE)
```

预测二:性别生还率不同

```
summary(train$Sex)
```

```
## female   male
##    314    577
```

```
prop.table(table(train$Sex))#占比
```

```
##
##   female     male
## 0.352413 0.647587
```

```
prop.table(table(train$Sex,train$Survived))#
```

```
##
##                   0          1
##   female 0.09090909 0.26150393
##   male   0.52525253 0.12233446
```

```
##女性的得救率是 26.25/35.24=75%
##假设女性都被得救
test$Survived<-0
test$Survived[test$Sex=='female']<-1
submit <- data.frame(PassengerId = test$PassengerId, Survived = test
$Survived)
write.csv(submit, file = "theyallperish.csv", row.names = FALSE)
```

年龄

```
summary(train$Age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.42   20.12   28.00   29.70   38.00   80.00     177
```

```
train$Age<-as.numeric(train$Age)
train$Age[train$Age==0]<-0.01
breaks<-c(0,0.01,12,18,24,30,35,40,50,60,70,80)
lables<-c('空值','0-12','13-18','19-24','25-30','31-35','36-40','41-5
0','51-60','61-70','71-80')
train$年龄分组<-cut(train$Age,breaks=breaks,lables=lables)
aggregate(Survived~年龄分组+Sex,data=train,FUN=sum)
```

```
##      年龄分组    Sex Survived
## 1  (0.01,12] female       19
## 2    (12,18] female       27
## 3    (18,24] female       39
## 4    (24,30] female       29
## 5    (30,35] female       26
## 6    (35,40] female       20
## 7    (40,50] female       21
## 8    (50,60] female       13
## 9    (60,70] female        3
## 10 (0.01,12]   male       21
## 11   (12,18]   male        3
## 12   (18,24]   male        9
## 13   (24,30]   male       19
## 14   (30,35]   male       15
## 15   (35,40]   male        8
## 16   (40,50]   male       12
## 17   (50,60]   male        4
## 18   (60,70]   male        1
## 19   (70,80]   male        1
```

```
aggregate(PassengerId~年龄分组+Sex,data=train,FUN=length)
```

```
##      年龄分组    Sex PassengerId
## 1  (0.01,12] female          32
## 2    (12,18] female          36
## 3    (18,24] female          49
## 4    (24,30] female          41
## 5    (30,35] female          30
## 6    (35,40] female          25
## 7    (40,50] female          31
## 8    (50,60] female          14
## 9    (60,70] female           3
## 10 (0.01,12]   male          37
## 11   (12,18]   male          34
## 12   (18,24]   male          89
## 13   (24,30]   male          91
## 14   (30,35]   male          58
## 15   (35,40]   male          42
## 16   (40,50]   male          55
## 17   (50,60]   male          28
## 18   (60,70]   male          14
## 19   (70,80]   male           5
```

```
str(train)
```

```
## '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",..: 10
9 191 358 277 16 559 520 629 417 581 ...
##  $ 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 59
7 670 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 ...
##  $ 年龄分组   : Factor w/ 11 levels "(0,0.01]","(0.01,12]",..: 4 7
 5 6 6 NA 9 2 5 3 ...
```

```
aggregate(Survived~年龄分组+Sex,data=train,FUN=function(x)
  {sum(x)/length(x)})
```

```
##     年龄分组    Sex   Survived
## 1  (0.01,12] female 0.59375000
## 2    (12,18] female 0.75000000
## 3    (18,24] female 0.79591837
## 4    (24,30] female 0.70731707
## 5    (30,35] female 0.86666667
## 6    (35,40] female 0.80000000
## 7    (40,50] female 0.67741935
## 8    (50,60] female 0.92857143
## 9    (60,70] female 1.00000000
## 10 (0.01,12]   male 0.56756757
## 11   (12,18]   male 0.08823529
## 12   (18,24]   male 0.10112360
## 13   (24,30]   male 0.20879121
## 14   (30,35]   male 0.25862069
## 15   (35,40]   male 0.19047619
## 16   (40,50]   male 0.21818182
## 17   (50,60]   male 0.14285714
## 18   (60,70]   male 0.07142857
## 19   (70,80]   male 0.20000000
```

预测三:船舱位置，费用

```
##赋值
train$Fare2<-'30+'
train$Fare2[train$Fare<30&train$Fare>=20]<-'20-30'
train$Fare2[train$Fare<20&train$Fare>=10]<-'10-20'
train$Fare2[train$Fare<10]<-'10'
aggregate(Survived ~ Fare2 + Pclass +年龄分组+ Sex, data=train, FUN=f
unction(x) {sum(x)/length(x)})
```

```
##     Fare2 Pclass  年龄分组    Sex   Survived
## 1     30+      1 (0.01,12] female 0.00000000
## 2   20-30      2 (0.01,12] female 1.00000000
## 3     30+      2 (0.01,12] female 1.00000000
## 4   10-20      3 (0.01,12] female 0.81818182
## 5   20-30      3 (0.01,12] female 0.16666667
## 6     30+      3 (0.01,12] female 0.16666667
## 7     30+      1   (12,18] female 1.00000000
## 8   10-20      2   (12,18] female 1.00000000
## 9   20-30      2   (12,18] female 1.00000000
## 10    30+      2   (12,18] female 1.00000000
## 11     10      3   (12,18] female 0.75000000
## 12  10-20      3   (12,18] female 0.28571429
## 13    30+      3   (12,18] female 0.00000000
## 14  20-30      1   (18,24] female 1.00000000
## 15    30+      1   (18,24] female 1.00000000
## 16  10-20      2   (18,24] female 0.83333333
## 17  20-30      2   (18,24] female 1.00000000
## 18    30+      2   (18,24] female 1.00000000
## 19     10      3   (18,24] female 0.53333333
## 20  10-20      3   (18,24] female 0.75000000
## 21    30+      3   (18,24] female 0.00000000
## 22    30+      1   (24,30] female 0.87500000
## 23  10-20      2   (24,30] female 1.00000000
## 24  20-30      2   (24,30] female 0.75000000
## 25    30+      2   (24,30] female 1.00000000
## 26     10      3   (24,30] female 0.42857143
## 27  10-20      3   (24,30] female 0.57142857
## 28  20-30      3   (24,30] female 0.00000000
## 29    30+      1   (30,35] female 1.00000000
## 30  10-20      2   (30,35] female 1.00000000
## 31  20-30      2   (30,35] female 1.00000000
## 32    30+      2   (30,35] female 1.00000000
## 33     10      3   (30,35] female 0.33333333
## 34  10-20      3   (30,35] female 0.33333333
```

```
## 35  20-30      3   (30,35] female 1.00000000
## 36    30+      1   (35,40] female 1.00000000
## 37  10-20      2   (35,40] female 0.80000000
## 38  20-30      2   (35,40] female 1.00000000
## 39    30+      2   (35,40] female 1.00000000
## 40     10      3   (35,40] female 0.00000000
## 41  10-20      3   (35,40] female 1.00000000
## 42  20-30      3   (35,40] female 0.00000000
## 43    30+      3   (35,40] female 0.50000000
## 44  20-30      1   (40,50] female 0.75000000
## 45    30+      1   (40,50] female 1.00000000
## 46  10-20      2   (40,50] female 1.00000000
## 47  20-30      2   (40,50] female 0.75000000
## 48    30+      2   (40,50] female 1.00000000
## 49     10      3   (40,50] female 0.00000000
## 50  10-20      3   (40,50] female 0.00000000
## 51  20-30      3   (40,50] female 0.00000000
## 52    30+      3   (40,50] female 0.00000000
## 53  20-30      1   (50,60] female 1.00000000
## 54    30+      1   (50,60] female 1.00000000
## 55  10-20      2   (50,60] female 0.50000000
## 56  20-30      2   (50,60] female 1.00000000
## 57    30+      1   (60,70] female 1.00000000
## 58     10      3   (60,70] female 1.00000000
## 59    30+      1 (0.01,12]   male 1.00000000
## 60  10-20      2 (0.01,12]   male 1.00000000
## 61  20-30      2 (0.01,12]   male 1.00000000
## 62    30+      2 (0.01,12]   male 1.00000000
## 63     10      3 (0.01,12]   male 1.00000000
## 64  10-20      3 (0.01,12]   male 0.83333333
## 65  20-30      3 (0.01,12]   male 0.22222222
## 66    30+      3 (0.01,12]   male 0.11111111
## 67    30+      1   (12,18]   male 0.50000000
## 68  10-20      2   (12,18]   male 0.00000000
## 69  20-30      2   (12,18]   male 0.00000000
## 70    30+      2   (12,18]   male 0.00000000
## 71     10      3   (12,18]   male 0.10526316
## 72  10-20      3   (12,18]   male 0.00000000
## 73  20-30      3   (12,18]   male 0.00000000
## 74    30+      3   (12,18]   male 0.00000000
## 75    30+      1   (18,24]   male 0.14285714
## 76  10-20      2   (18,24]   male 0.08333333
## 77    30+      2   (18,24]   male 0.00000000
## 78     10      3   (18,24]   male 0.10000000
```

```
## 79  10-20      3   (18,24]   male 0.20000000
## 80  20-30      3   (18,24]   male 0.00000000
## 81  20-30      1   (24,30]   male 0.50000000
## 82    30+      1   (24,30]   male 0.58333333
## 83  10-20      2   (24,30]   male 0.00000000
## 84  20-30      2   (24,30]   male 0.00000000
## 85    30+      2   (24,30]   male 0.00000000
## 86     10      3   (24,30]   male 0.20000000
## 87  10-20      3   (24,30]   male 0.14285714
## 88  20-30      3   (24,30]   male 0.00000000
## 89    30+      3   (24,30]   male 0.50000000
## 90     10      1   (30,35]   male 0.00000000
## 91  20-30      1   (30,35]   male 1.00000000
## 92    30+      1   (30,35]   male 0.60000000
## 93  10-20      2   (30,35]   male 0.25000000
## 94  20-30      2   (30,35]   male 0.16666667
## 95    30+      2   (30,35]   male 0.00000000
## 96     10      3   (30,35]   male 0.14814815
## 97  10-20      3   (30,35]   male 0.00000000
## 98  20-30      3   (30,35]   male 0.00000000
## 99    30+      3   (30,35]   male 1.00000000
## 100    10      1   (35,40]   male 0.00000000
## 101 20-30      1   (35,40]   male 0.50000000
## 102   30+      1   (35,40]   male 0.55555556
## 103 10-20      2   (35,40]   male 0.00000000
## 104 20-30      2   (35,40]   male 0.00000000
## 105    10      3   (35,40]   male 0.09090909
## 106 10-20      3   (35,40]   male 0.00000000
## 107 20-30      3   (35,40]   male 0.00000000
## 108   30+      3   (35,40]   male 0.00000000
## 109 20-30      1   (40,50]   male 0.50000000
## 110   30+      1   (40,50]   male 0.33333333
## 111 10-20      2   (40,50]   male 0.20000000
## 112 20-30      2   (40,50]   male 0.00000000
## 113    10      3   (40,50]   male 0.10526316
## 114 10-20      3   (40,50]   male 0.00000000
## 115 20-30      1   (50,60]   male 0.25000000
## 116   30+      1   (50,60]   male 0.30000000
## 117 10-20      2   (50,60]   male 0.00000000
## 118 20-30      2   (50,60]   male 0.00000000
## 119   30+      2   (50,60]   male 0.00000000
## 120    10      3   (50,60]   male 0.00000000
## 121 20-30      1   (60,70]   male 0.00000000
## 122   30+      1   (60,70]   male 0.00000000
```

```
## 123 10-20      2   (60,70]   male 0.33333333
## 124    10      3   (60,70]   male 0.00000000
## 125   30+      1   (70,80]   male 0.33333333
## 126    10      3   (70,80]   male 0.00000000
```

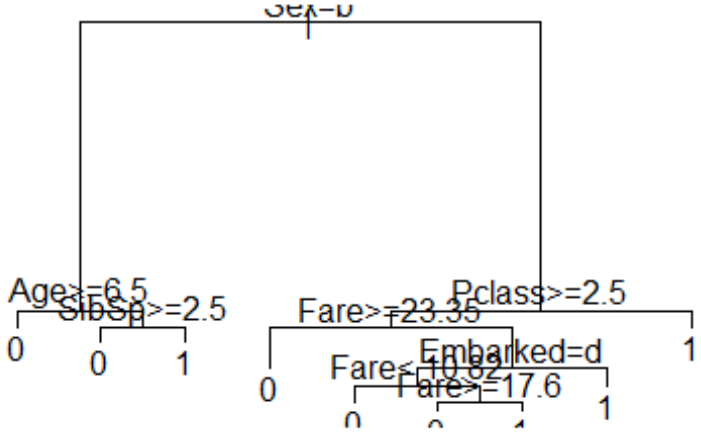
预测三导出：按特征更新生还率女性 0，男性 1

```
test$Survived <- 0
test$Survived[test$Sex == 'female'] <- 1
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare >=
 30&test$Age==0.01] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 1 & test$Fare >=
 30 & test$Age<=12& test$Age>0.01] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare >=
 20 &test$Age<=12& test$Age>0.01] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare >=
 30 &test$Age<=24& test$Age>12] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare<=3
0 & test$Fare >20 &test$Age<=30& test$Age>24] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare<=2
0 & test$Fare >10 &test$Age<=18& test$Age>12] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare<=3
0 & test$Fare >20 &test$Age<=40& test$Age>35] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare<=1
0 &test$Age<=40& test$Age>35] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare<=2
0 &test$Age<=35& test$Age>30] <- 0
test$Survived[test$Sex == 'female' & test$Pclass == 3 &  test$Age<=5
0& test$Age>40] <- 0
test$Survived[test$Sex == 'male' & test$Pclass == 2 & test$Fare<=30
& test$Fare >10 &test$Age<=12& test$Age>0.01] <- 1
test$Survived[test$Sex == 'male' & test$Pclass == 3 & test$Fare<10 &
test$Age<=12& test$Age>0.01] <- 1
test$Survived[test$Sex == 'male' & test$Pclass == 1 & test$Fare>30 &
test$Age<=12& test$Age>0.01] <- 1
test$Survived[test$Sex == 'male' & test$Pclass == 3 & test$Fare>30 &
test$Age<=35& test$Age>30] <- 1
test$Survived[test$Sex == 'male' & test$Pclass == 1 & test$Fare<=30
&test$Fare>20 &test$Age<=35& test$Age>30] <- 1
test$Survived[test$Sex == 'female' & test$Pclass == 3 & test$Fare >=
 20] <- 0
submit <- data.frame(PassengerId = test$PassengerId, Survived = test
$Survived)
write.csv(submit, file = "theyallperish4.csv", row.names = FALSE)
```

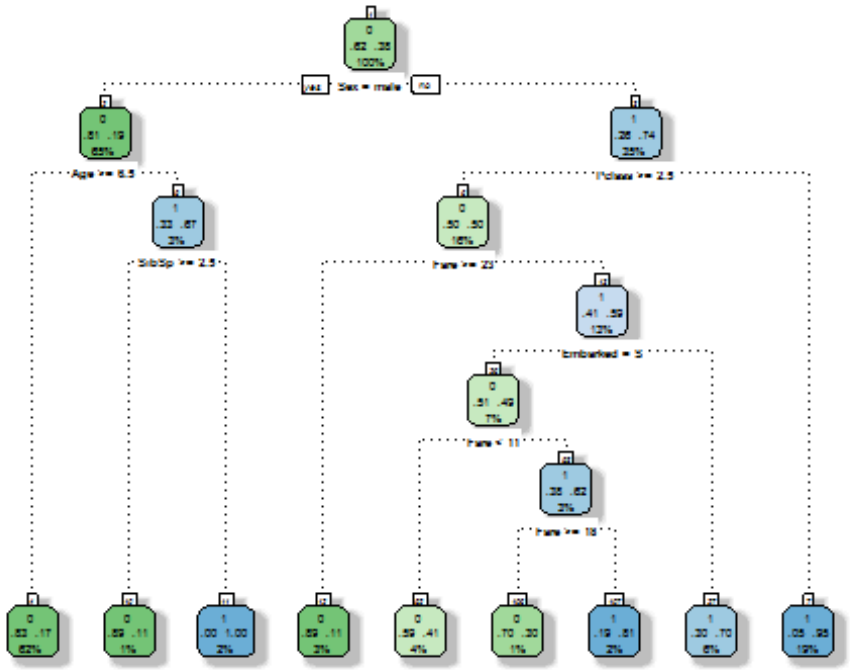
建模预测

决策树

```
train<- read.csv("D:/Data-Analysis/R/R_ML/R_ML-Titanic/Data/train.csv")
test<- read.csv("D:/Data-Analysis/R/R_ML/R_ML-Titanic/Data/test.csv")
library(rpart)

fit <- rpart(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked,
             data=train,
             method="class")

plot(fit)
text(fit)
```

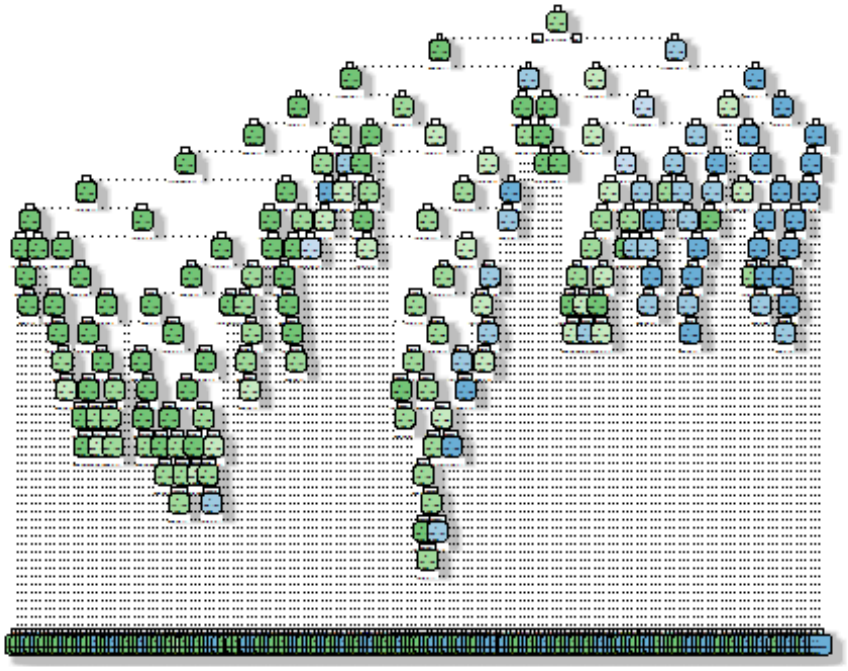



```
library(rattle)

## Rattle: A free graphical interface for data mining with R.
## XXXX 4.1.0 Copyright (c) 2006-2015 Togaware Pty Ltd.
## 键入'rattle()'去轻摇、晃动、翻滚你的数据。

library(rpart.plot)
library(RColorBrewer)
fancyRpartPlot(fit)#美化图片，提高可视化
```



导出预测结果

```
Prediction <- predict(fit, test, type = "class")
submit <- data.frame(PassengerId = test$PassengerId, Survived = Pred
iction)
write.csv(submit, file = "myfirstdtree.csv", row.names = FALSE)

fit <- rpart(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare +
Embarked,
               data=train,
               method="class",
               control=rpart.control(minsplit=2, cp=0))
fancyRpartPlot(fit)#美化图片，提高可视化
```



调整决策树

fit <- rpart(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare + Embarked, data=train, method="class", control=rpart.control(your controls))##调整条件 new.fit <- prp(fit,snip=TRUE)$obj fancyRpartPlot(new.fit)

整合新的变量

```
train$Name[1]

## [1] Braund, Mr. Owen Harris
## 891 Levels: Abbing, Mr. Anthony ... Zimmerman, Mr. Leo

test$Survived <- NA


combi <- rbind(train, test)
combi$Name <- as.character(combi$Name)
combi$Name[1]

## [1] "Braund, Mr. Owen Harris"

strsplit(combi$Name[1], split='[,.]')#字符串拆分的功能

## [[1]]
## [1] "Braund"       " Mr"          " Owen Harris"

strsplit(combi$Name[1], split='[,.]')[[1]][2]#在文本部分之前加上索引

## [1] " Mr"
```

```
combi$Title <- sapply(combi$Name, FUN=function(x) {strsplit(x, split
='[,.]')[[1]][2]})
##遍历名称向量的行，并将每个名称发送到函数。
##所有这些字符串分割的结果都被组合成一个向量作为 sapply 函数的输出，
##然后我们存储到我们原始数据帧中的一个新列:标题
combi$Title <- sub(' ', '', combi$Title)##
table(combi$Title)

##
##         Capt          Col          Don         Dona           Dr
##            1            4            1            1            8
##     Jonkheer         Lady        Major       Master         Miss
##            1            1            2           61          260
##         Mlle          Mme           Mr          Mrs           Ms
##            2            1          757          197            2
##          Rev          Sir the Countess
##            8            1            1

combi$Title[combi$Title %in% c('Mme', 'Mlle')] <- 'Mlle'
combi$Title[combi$Title %in% c('Capt', 'Don', 'Major', 'Sir')] <- 'S
ir'
combi$Title[combi$Title %in% c('Dona', 'Lady', 'the Countess', 'Jonk
heer')] <- 'Lady'
combi$Title <- factor(combi$Title)
combi$FamilySize <- combi$SibSp + combi$Parch + 1
combi$Surname <- sapply(combi$Name, FUN=function(x) {strsplit(x, spl
it='[,.]')[[1]][1]})
combi$FamilyID <- paste(as.character(combi$FamilySize), combi$Surnam
e, sep="")
combi$FamilyID[combi$FamilySize <= 2] <- 'Small'
table(combi$FamilyID)

##
##            11Sage           3Abbott         3Appleton          3Be
ckwith
##                11                 3                 1
     2
##           3Boulos           3Bourke            3Brown          3Ca
ldwell
##                 3                 3                 4
     3
##          3Christy          3Collyer          3Compton           3C
ornell
##                 2                 3                 3
     1
##           3Coutts           3Crosby           3Danbom            3
```

```
Davies
##                 3                 3                 3
     5
##            3Dodge          3Douglas             3Drew
3Elias
##                 3                 1                 3
     3
##       3Frauenthal        3Frolicher 3Frolicher-Stehli        3Gol
dsmith
##                 1                 1                 2
     3
##       3Gustafsson       3Hamalainen           3Hansen
 3Hart
##                 2                 2                 1
     3
##             3Hays          3Hickman         3Hiltunen          3Hi
rvonen
##                 2                 3                 1
     1
##         3Jefferys          3Johnson             3Kink    3Kink-He
ilmann
##                 2                 3                 2
     2
##           3Klasen         3Lahtinen           3Mallet
3McCoy
##                 3                 2                 3
     3
##          3Minahan         3Moubarek            3Nakid          3Na
vratil
##                 1                 3                 3
     3
##           3Newell           3Newsom         3Nicholls           3P
eacock
##                 1                 1                 1
     3
##            3Peter            3Quick         3Richards           3R
osblom
##                 3                 3                 2
     3
##           3Samaan        3Sandstrom           3Silven           3S
pedden
##                 3                 3                 1
     3
##            3Strom          3Taussig           3Thayer            3
```

```
Thomas
##                 1                 3                 3
     1
##            3Touma     3van Billiard         3Van Impe    3Vander
Planke
##                 3                 3                 3
     2
##            3Wells             3Wick          3Widener          4A
llison
##                 3                 3                 3
     4
##        4Backstrom          4Baclini           4Becker           4
Carter
##                 1                 4                 4
     4
##         4Davidson            4Dean           4Herman          4H
ocking
##                 1                 4                 4
     2
##        4Jacobsohn         4Johnston          4Laroche           4
Renouf
##                 1                 4                 4
     1
##    4Vander Planke             4West             5Ford          5H
ocking
##                 1                 4                 5
     1
##    5Kink-Heilmann          5Lefebre          5Palsson          5R
yerson
##                 1                 5                 5
     5
##          6Fortune           6Panula             6Rice         6Ri
chards
##                 6                 6                 6
     1
##            6Skoog        7Andersson          7Asplund          8G
oodwin
##                 6                 9                 7
     8
##             Small
##              1025

famIDs <- data.frame(table(combi$FamilyID))##保存到数据框
famIDs <- famIDs[famIDs$Freq <= 2,]
combi$FamilyID[combi$FamilyID %in% famIDs$Var1] <- 'Small'
```

```
combi$FamilyID <- factor(combi$FamilyID)
train <- combi[1:891,]
test <- combi[892:1309,]
fit <- rpart(Survived ~ Pclass + Sex + Age + SibSp + Parch + Fare +
Embarked + Title + FamilySize + FamilyID,data=train, method="class")
##导出结果
submit <- data.frame(PassengerId = test$PassengerId, Survived = Pred
iction)
write.csv(submit, file = "mydtree2.csv", row.names = FALSE)
```

随机森林

```
##处理缺失值
Agefit <- rpart(Age ~ Pclass + Sex + SibSp + Parch + Fare + Embarked
 + Title + FamilySize,
                data=combi[!is.na(combi$Age),],
                method="anova")
combi$Age[is.na(combi$Age)] <- predict(Agefit, combi[is.na(combi$Ag
e),])
summary(combi)
```

```
##   PassengerId      Survived          Pclass          Name

##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309

##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character

##  Median : 655   Median :0.0000   Median :3.000   Mode  :character

##  Mean   : 655   Mean   :0.3838   Mean   :2.295

##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000

##  Max.   :1309   Max.   :1.0000   Max.   :3.000

##                 NA's   :418

##      Sex           Age            SibSp            Parch
##  female:466   Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  male  :843   1st Qu.:22.00   1st Qu.:0.0000   1st Qu.:0.000
##               Median :28.86   Median :0.0000   Median :0.000
##               Mean   :29.70   Mean   :0.4989   Mean   :0.385
##               3rd Qu.:36.50   3rd Qu.:1.0000   3rd Qu.:0.000
##               Max.   :80.00   Max.   :8.0000   Max.   :9.000
##
```

```
##       Ticket          Fare                        Cabin      Embarke
d
##  CA. 2343:  11   Min.   :  0.000                 :1014    :  2
##  1601    :   8   1st Qu.:  7.896   C23 C25 C27    :   6   C:270
##  CA 2144 :   8   Median : 14.454   B57 B59 B63 B66:   5   Q:123
##  3101295 :   7   Mean   : 33.295   G6             :   5   S:914
##  347077  :   7   3rd Qu.: 31.275   B96 B98        :   4
##  347082  :   7   Max.   :512.329   C22 C26        :   4
##  (Other) :1261   NA's   :1         (Other)        : 271
##      Title       FamilySize       Surname                FamilyID
##  Mr     :757   Min.   : 1.000   Length:1309        Small     :107
4
##  Miss   :260   1st Qu.: 1.000   Class :character   11Sage    :  1
1
##  Mrs    :197   Median : 1.000   Mode  :character   7Andersson:
9
##  Master : 61   Mean   : 1.884                      8Goodwin  :
8
##  Dr     :  8   3rd Qu.: 2.000                      7Asplund  :
7
##  Rev    :  8   Max.   :11.000                      6Fortune  :
6
##  (Other): 18                                       (Other)   : 19
4
```

```
summary(combi$Embarked)##combi$Embarked 存在空白值
```

```
##       C   Q   S
##   2 270 123 914
```

```
which(combi$Embarked == '')##找出空白值的位置
```

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

```
combi$Embarked[c(62,830)] = "S"##赋值，914/(270+123+914)=70%为 S
combi$Embarked <- factor(combi$Embarked)
summary(combi$Fare)##combi$Fare 存在异常值
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   7.896  14.450  33.300  31.280 512.300       1
```

```
which(is.na(combi$Fare))
```

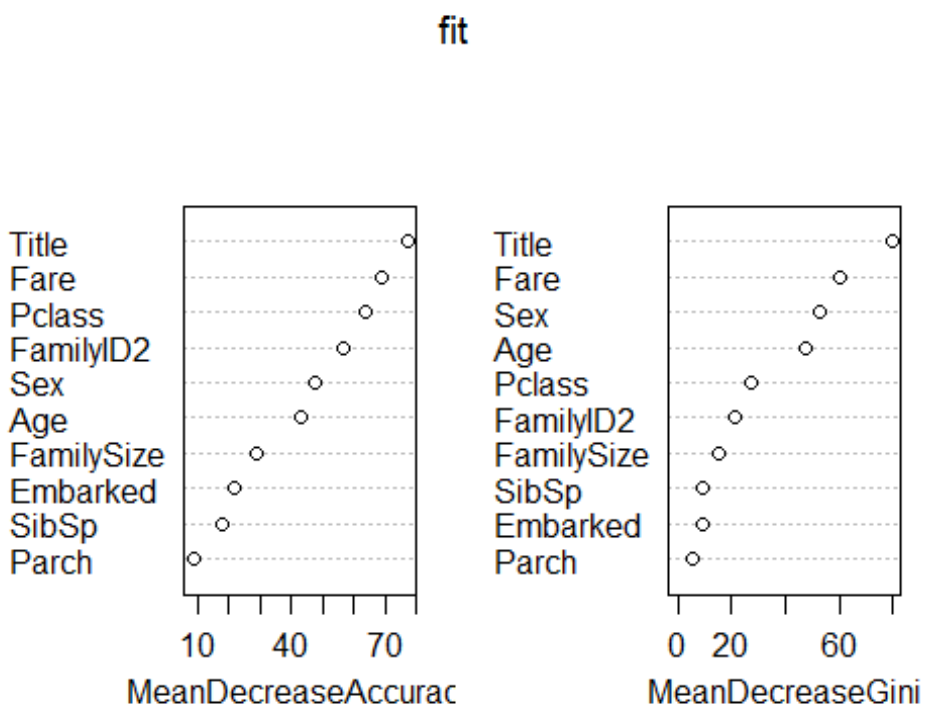
```
## [1] 1044
```

```
combi$Fare[1044] <- median(combi$Fare, na.rm=TRUE)#

## R can only digest factors with up to 32 levels，降级
combi$FamilyID2 <- combi$FamilyID
combi$FamilyID2 <- as.character(combi$FamilyID2)
combi$FamilyID2[combi$FamilySize <= 3] <- 'Small'
combi$FamilyID2 <- factor(combi$FamilyID2)
train <- combi[1:891,]
test <- combi[892:1309,]
library(randomForest)
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
set.seed(415)
##内部数字并不重要，但要确保每次使用相同的种子编号，以便在随机森林功能中生成相同的随机数。
fit <- randomForest(as.factor(Survived) ~ Pclass + Sex + Age + SibSp
 + Parch + Fare +
                                                Embarked + Title + Famil
ySize + FamilyID2,
data=train,importance=TRUE,ntree=2000)
varImpPlot(fit)
```

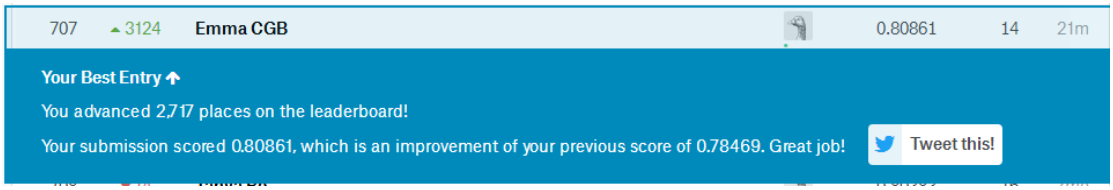



```
Prediction <- predict(fit, test)
submit <- data.frame(PassengerId = test$PassengerId, Survived = Pred
iction)
write.csv(submit, file = "firstforest.csv", row.names = FALSE)
```

条件推理树

```
library(party)

set.seed(415)
fit <- cforest(as.factor(Survived) ~ Pclass + Sex + Age + SibSp + Pa
rch + Fare +Embarked + Title + FamilySize + FamilyID,
                 data = train,
                 controls=cforest_unbiased(ntree=2000, mtry=3))
Prediction <- predict(fit, test, OOB=TRUE, type = "response")
submit <- data.frame(PassengerId = test$PassengerId, Survived = Pred
iction)
write.csv(submit, file = "LTREE.csv", row.names = FALSE)
```

预测最佳结果：条件推理树



注意事项

1. 为了避免上一个处理过程影响下一个预测结果，每次处理前记得重新加载数据。
2. 导出文档时注意把导出内容中安装包语句删除，避免出错。

有待解决疑问

1. 中文导出 PDF 格式又不知道哪出问题了，要把这个问题彻底解决。
2. 是否因为预测的结果是随机的，所以预测分数与教材不一致？
3. Rmarkdown 导出图片的大小是否可控？