Ch8BeforeRandallPlyler

Randall Plyler

2/20/2022

```
#### Table 8.4

library(e1071)
delays.df <- read.csv("C:/Users/randa/Dropbox/Masters/Winter/TBANLT 560 Data Mining/Files/DMBA-R-datase
# change numerical variables to categorical first
delays.df$DAY_WEEK <- factor(delays.df$DAY_WEEK)
delays.df$DEP_TIME <- factor(delays.df$DEP_TIME)
# create hourly bins departure time
delays.df$CRS_DEP_TIME <- factor(round(delays.df$CRS_DEP_TIME/100))
#####
delays.df$CARRIER <- factor(delays.df$CARRIER)
delays.df$DEST <- factor(delays.df$DEST)
delays.df$ORIGIN<- factor(delays.df$ORIGIN)
delays.df$Flight.Status<- factor(delays.df$Flight.Status)
#####
```

```
# Create training and validation sets.
selected.var <- c(10, 1, 8, 4, 2, 13)
train.index <- sample(c(1:dim(delays.df)[1]), dim(delays.df)[1]*0.6)
train.df <- delays.df[train.index, selected.var]
valid.df <- delays.df[-train.index, selected.var]

# run naive bayes
delays.nb <- naiveBayes(Flight.Status ~ ., data = train.df)
delays.nb
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##   delayed    ontime
## 0.1878788 0.8121212
##
## Conditional probabilities:
##          DAY_WEEK
## Y                 1         2         3         4         5         6
##   delayed 0.21774194 0.14919355 0.14516129 0.10080645 0.17741935 0.05241935
```

```
##   ontime  0.11287313 0.14272388 0.14925373 0.18843284 0.15671642 0.13152985
##          DAY_WEEK
## Y                 7
##   delayed 0.15725806
##   ontime  0.11847015
##
##          CRS_DEP_TIME
## Y                 6          7          8          9         10         11
##   delayed 0.02016129 0.05241935 0.05645161 0.02822581 0.02822581 0.01209677
##   ontime  0.06156716 0.06343284 0.08115672 0.06063433 0.04850746 0.03078358
##          CRS_DEP_TIME
## Y                12         13         14         15         16         17
##   delayed 0.04838710 0.04838710 0.04435484 0.21370968 0.08467742 0.15725806
##   ontime  0.06996269 0.06996269 0.05223881 0.12033582 0.08022388 0.09421642
##          CRS_DEP_TIME
## Y                18         19         20         21
##   delayed 0.03225806 0.08467742 0.01209677 0.07661290
##   ontime  0.03731343 0.04850746 0.02425373 0.05690299
##
##          ORIGIN
## Y               BWI        DCA        IAD
##   delayed 0.09677419 0.51209677 0.39112903
##   ontime  0.05410448 0.65951493 0.28638060
##
##          DEST
## Y              EWR       JFK       LGA
##   delayed 0.3830645 0.2177419 0.3991935
##   ontime  0.2854478 0.1567164 0.5578358
##
##          CARRIER
## Y                 CO          DH          DL          MQ          OH
##   delayed 0.056451613 0.314516129 0.116935484 0.189516129 0.012096774
##   ontime  0.040111940 0.227611940 0.178171642 0.123134328 0.013992537
##          CARRIER
## Y                 RU          UA          US
##   delayed 0.241935484 0.004032258 0.064516129
##   ontime  0.176305970 0.013059701 0.227611940
```

```
#### Table 8.5

# use prop.table() with margin = 1 to convert a count table to a proportion table,
# where each row sums up to 1 (use margin = 2 for column sums).
prop.table(table(train.df$Flight.Status, train.df$DEST), margin = 1)
```

```
##
##                EWR       JFK       LGA
##   delayed 0.3830645 0.2177419 0.3991935
##   ontime  0.2854478 0.1567164 0.5578358
```

```
#### Table 8.6

## predict probabilities
pred.prob <- predict(delays.nb, newdata = valid.df, type = "raw")
```

```
## predict class membership
pred.class <- predict(delays.nb, newdata = valid.df, type= "class")

df <- data.frame(actual = valid.df$Flight.Status, predicted = pred.class, pred.prob)

df[valid.df$CARRIER == "DL" & valid.df$DAY_WEEK == 7 & valid.df$CRS_DEP_TIME == 10 &
     valid.df$DEST == "LGA" & valid.df$ORIGIN == "DCA",]
```

```
##     actual predicted    delayed    ontime
## 303 ontime    ontime 0.06117786 0.9388221
## 702 ontime    ontime 0.06117786 0.9388221
```

```
#### Table 8.7

library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
# training
pred.class <- predict(delays.nb, newdata = train.df)
confusionMatrix(pred.class, train.df$Flight.Status)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction delayed ontime
##    delayed      50     62
##    ontime      198   1010
##
##                Accuracy : 0.803
##                  95% CI : (0.7805, 0.8242)
##     No Information Rate : 0.8121
##     P-Value [Acc > NIR] : 0.8112
##
##                   Kappa : 0.1822
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.20161
##             Specificity : 0.94216
##          Pos Pred Value : 0.44643
##          Neg Pred Value : 0.83609
##              Prevalence : 0.18788
##          Detection Rate : 0.03788
##    Detection Prevalence : 0.08485
##       Balanced Accuracy : 0.57189
##
##        'Positive' Class : delayed
##
```

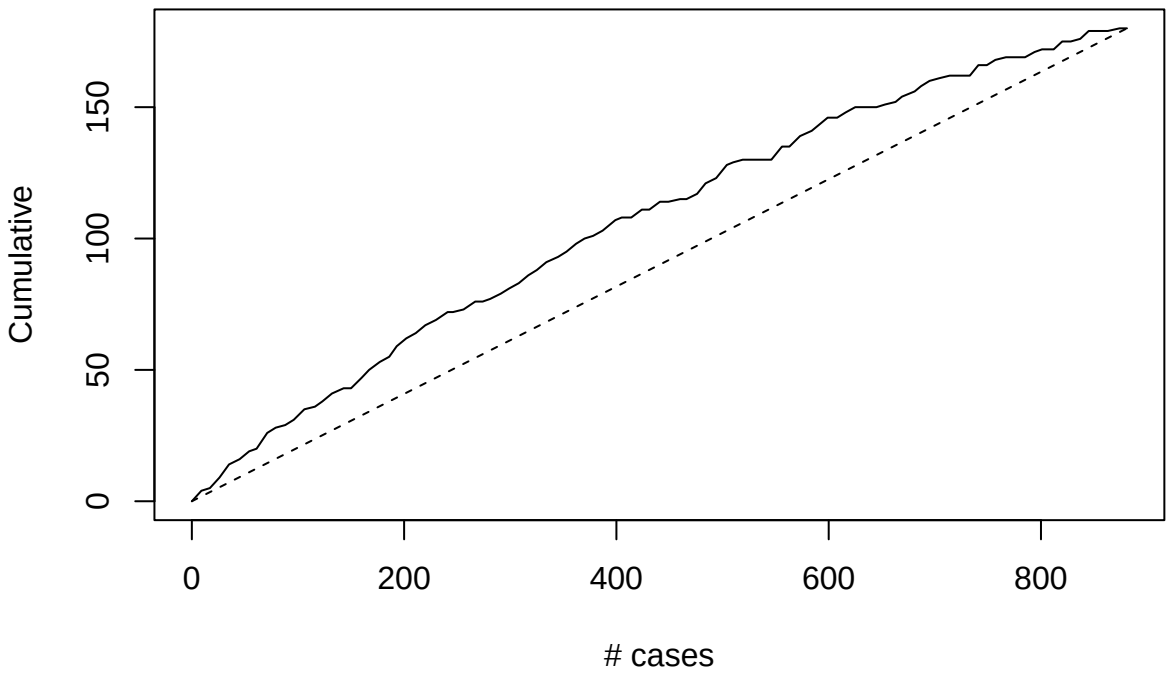
```
# validation
pred.class <- predict(delays.nb, newdata = valid.df)
confusionMatrix(pred.class, valid.df$Flight.Status)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction delayed ontime
##    delayed      29     54
##    ontime      151    647
##
##                Accuracy : 0.7673
##                  95% CI : (0.738, 0.7948)
##     No Information Rate : 0.7957
##     P-Value [Acc > NIR] : 0.9823
##
##                   Kappa : 0.1051
##
##  Mcnemar's Test P-Value : 2.015e-11
##
##             Sensitivity : 0.16111
##             Specificity : 0.92297
##          Pos Pred Value : 0.34940
##          Neg Pred Value : 0.81078
##              Prevalence : 0.20431
##          Detection Rate : 0.03292
##    Detection Prevalence : 0.09421
##       Balanced Accuracy : 0.54204
##
##        'Positive' Class : delayed
##
```

```
#### Figure 8.1

library(gains)
gain <- gains(ifelse(valid.df$Flight.Status=="delayed",1,0), pred.prob[,1], groups=100)

plot(c(0,gain$cume.pct.of.total*sum(valid.df$Flight.Status=="delayed"))~c(0,gain$cume.obs),
     xlab="# cases", ylab="Cumulative", main="", type="l")
lines(c(0,sum(valid.df$Flight.Status=="delayed"))~c(0, dim(valid.df)[1]), lty=2)
```


Cumulative
0
50
100
150
0
200
400
600
800
cases