Class Probability Prediction

Blake Pappas

2023-12-17

Class Probability Prediction in R

Load the following packages:

```
library(dplyr)
library(caret)
library(e1071)
library(pROC)
```

In this exercise, we use the "bank.csv" file. The goal is to predict outcome "y" - whether a customer makes a deposit as a result of the bank's marketing activity. Note: the delimiter of this .csv file is not comma, but ";"

P1: Import the dataset:

```
bank = read.csv("bank.csv", sep = ";")
```

P2: Split the dataset into 75% training and 25% testing

```
library(caret)

train_rows = createDataPartition(y = bank$y,
                                 p = 0.75, list = FALSE)

bank_train = bank[train_rows, ]

bank_test = bank[-train_rows, ]
```

P3: Build a naive Bayes model

```
library(e1071)

NB_model = naiveBayes(y ~ ., data = bank_train)
```

P4: Evaluate the performance of your model. Report AUC.

```
# Make Categorical Predictions
pred_nb = predict(NB_model, bank_test, type = "class")

# Make Class Probability Predictions
prob_pred_nb = predict(NB_model, bank_test, type = "raw")

# Evaluate Model Performance
confusionMatrix(pred_nb, as.factor(bank_test$y),
                mode = "prec_recall", positive = "yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  9204  601
##        yes  776  721
##
##                Accuracy : 0.8782
##                  95% CI : (0.872, 0.8841)
##     No Information Rate : 0.883
##     P-Value [Acc > NIR] : 0.9472
##
##                   Kappa : 0.4422
##
##  Mcnemar’s Test P-Value : 2.745e-06
##
##               Precision : 0.48163
##                  Recall : 0.54539
##                      F1 : 0.51153
##              Prevalence : 0.11697
##          Detection Rate : 0.06379
##    Detection Prevalence : 0.13245
##       Balanced Accuracy : 0.73382
##
##        ’Positive’ Class : yes
##
```

```
# Make ROC Curve for Class "yes"
library(pROC)

roc_nb = roc(response = ifelse(bank_test$y == "yes", 1, 0),
             predictor = prob_pred_nb[, 2])
```

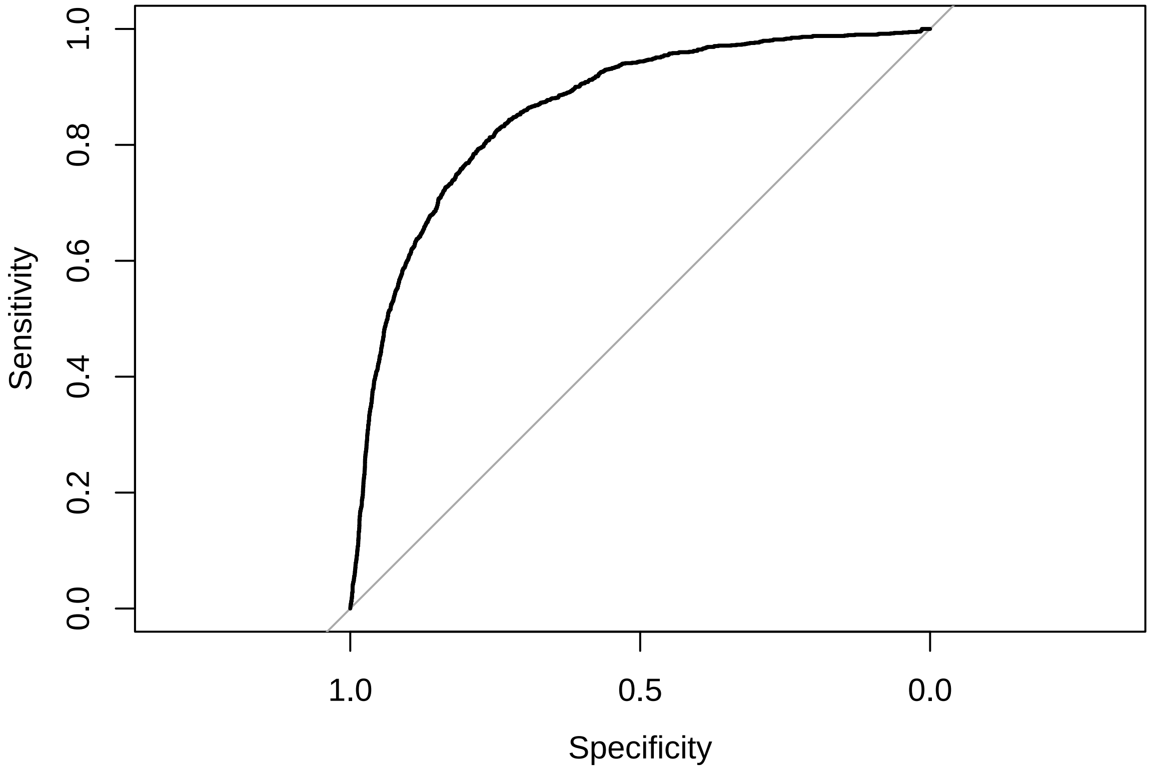
```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

```
# Calculate AUC
auc(roc_nb)
```

```
## Area under the curve: 0.8595
```

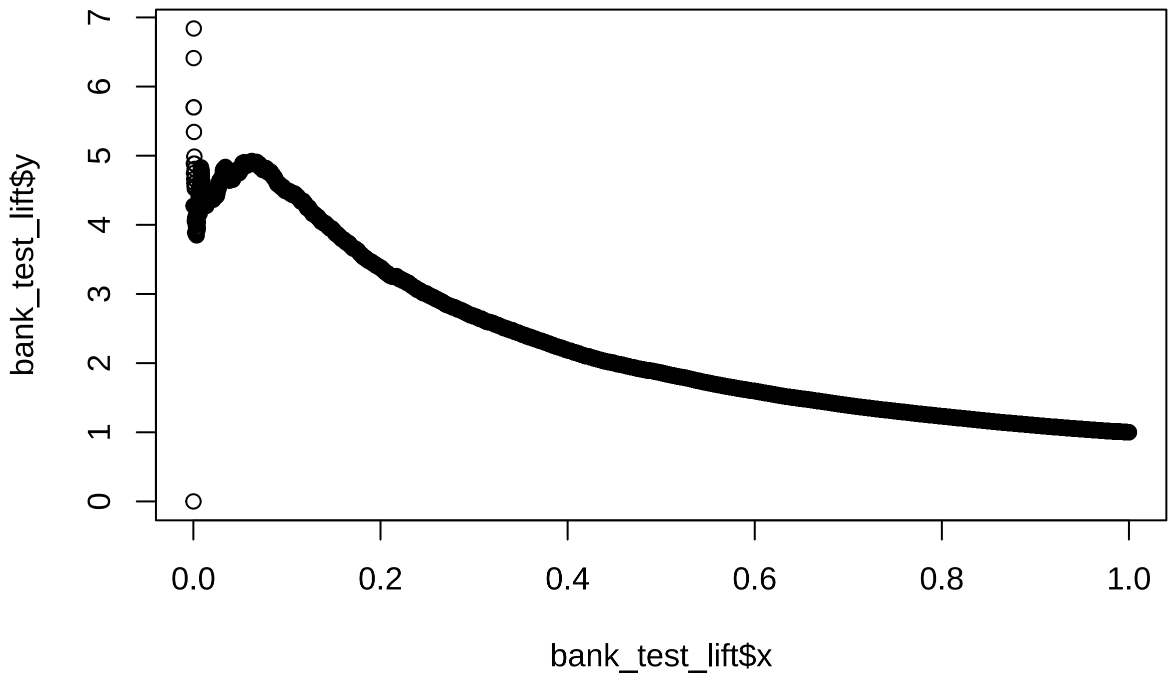
P5: Plot the ROC curve

```
plot(roc_nb)
```



P6: Plot the lift curve. What is the lift ratio for the top 20% of customers? What does that mean?

```
bank_test_lift = bank_test %>%
  mutate(prob = prob_pred_nb[, 2]) %>%
  arrange(desc(prob)) %>%
  mutate(y_yes = ifelse(y == "yes", 1, 0)) %>%
  # The Following Two Lines Make the Lift Curve
  mutate(x = row_number() / nrow(bank_test),
         y = (cumsum(y_yes) / sum(y_yes)) / x)

plot(bank_test_lift$x, bank_test_lift$y)
```



Answer: The lift ratio for the top 20% of customers is about three. This means that the model's top 20% of customers are about three times as good at predicting the outcome than random guesses.