ROC and AUC for Logistic Regression

Yuan Bian

University of Western Ontario

2022/11/24

Dataset *Default*

```
library(ISLR2)
options(digits=3)
str(Default)
```

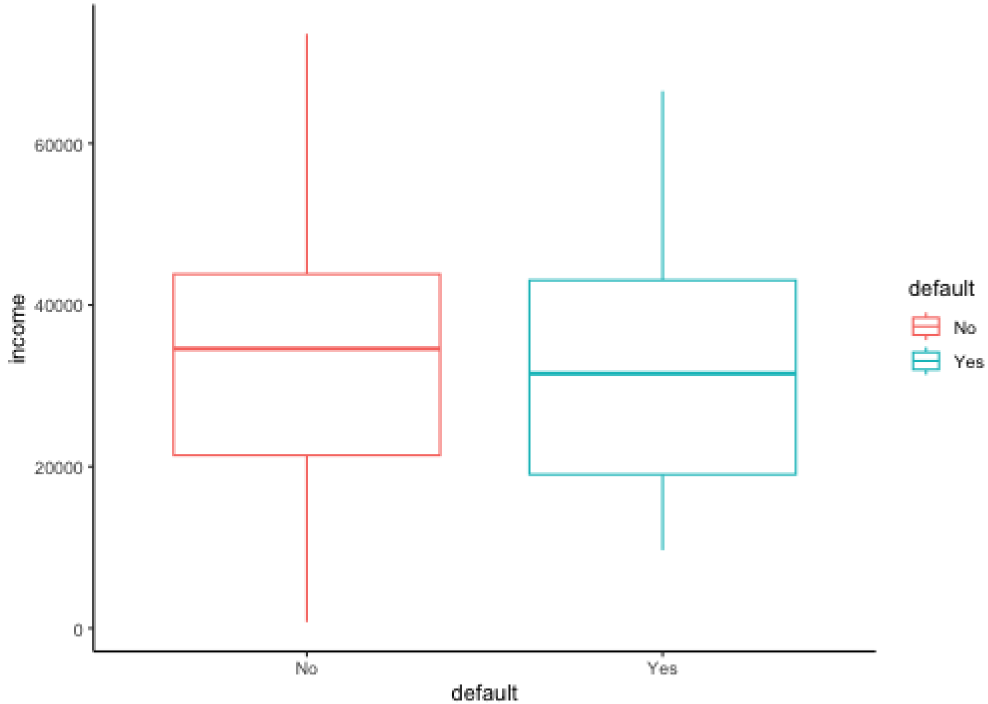
```
'data.frame':	10000 obs. of  4 variables:
 $ default: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
 $ student: Factor w/ 2 levels "No","Yes": 1 2 1 1 1 2 1 2 1 1 ...
 $ balance: num  730 817 1074 529 786 ...
 $ income : num  44362 12106 31767 35704 38463 ...
```

```
mean(Default$default == "Yes") * 100
```

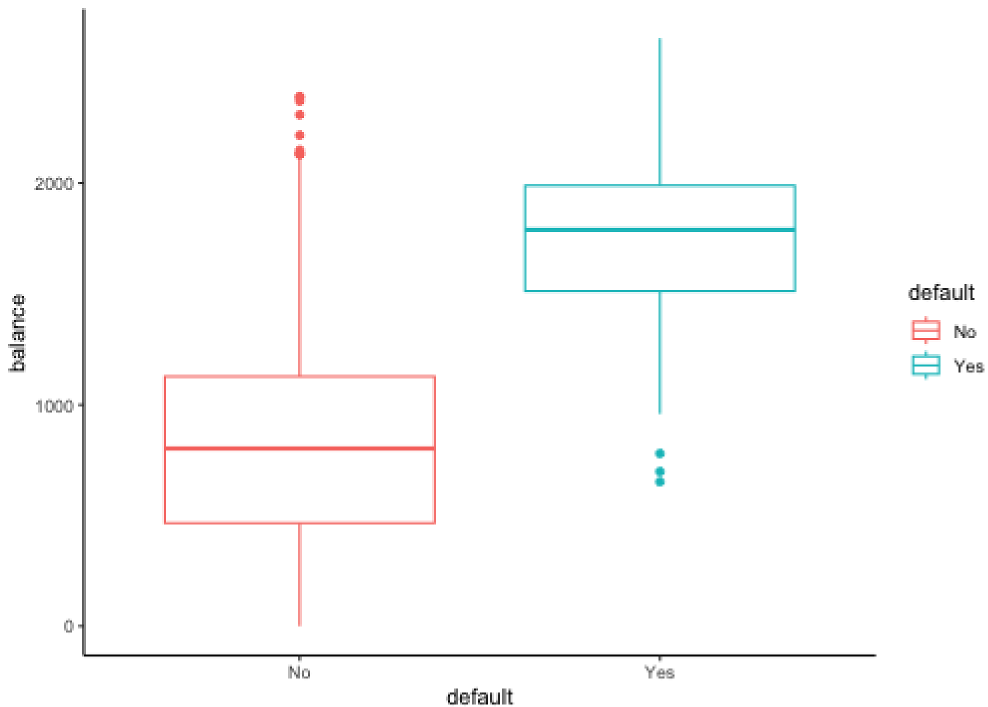
```
[1] 3.33
```

Goal: to predict whether an individual will default (fail to pay) his/her credit card payment based on different predictors

```r
library(ggplot2)
ggplot(Default, aes(x=default, y=income, color=default)) +
  geom_boxplot() + theme_classic()
```



```r
ggplot(Default, aes(x=default, y=balance, color=default)) +
  geom_boxplot() + theme_classic()
```



Quick Overview of Logistic Regression

- $X_i = (X_{i1}, \cdots, X_{iq})^\top$ is a vector of q predictors.
- Y_i is a response variable for $i = 1, \cdots, n$ taking value zero or one
- $p_i \triangleq P(Y_i = 1 | X_i = x_i)$ and $1 - p_i = P(Y_i = 0 | X_i = x_i)$
- Logit link with a linear predictor

$$\eta_i \triangleq \log \frac{p_i}{1 - p_i} = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_q x_{iq},$$

or equivalently

$$p_i = \frac{\exp\left(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_q x_{iq}\right)}{1 + \exp\left(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_q x_{iq}\right)},$$

$$p_i = \frac{\exp\left(\eta_i\right)}{1 + \exp\left(\eta_i\right)} = \frac{1}{1 + \exp\left(-\eta_i\right)}.$$

Training a logistic regression model

Creating training and test sets

```
data = Default
len_x = dim(data)[1]
set.seed(123)
index_ran = sample(1:len_x, len_x)
train_size = len_x * 0.8
train = data[index_ran[1:train_size],]
test = data[index_ran[(train_size+1):len_x],]
mean(train$default == "Yes")
```

```
[1] 0.0334
```

```
x_train = train
x_train$default <- NULL
x_test = test
x_test$default <- NULL
```

```r
logistic_fit <- glm(default~student+balance+income, data=train,
                    family=binomial)
library(faraway)
sumary(logistic_fit)
```

```
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.10e+01   5.49e-01  -20.00   <2e-16
studentYes  -6.34e-01   2.62e-01   -2.42    0.015
balance      5.77e-03   2.60e-04   22.16   <2e-16
income       4.45e-06   9.10e-06    0.49    0.625

n = 8000 p = 4
Deviance = 1252.3 Null Deviance = 2340.6 (Difference = 1088.3)
```

Variable selection

- select for variables via the Aikaike Information Criterion (AIC):

```
best_logistic <- step(logistic_fit, trace=0)
sumary(best_logistic)
```

```
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -10.79808    0.41409  -26.08  < 2e-16
studentYes   -0.73333    0.16576   -4.42  9.7e-06
balance       0.00577    0.00026   22.18  < 2e-16

n = 8000 p = 3
Deviance = 1252.5 Null Deviance = 2340.6 (Difference = 1088.0)
```

Interpretation of coefficients

```
exp(coefficients(best_logistic))
```

```
(Intercept)  studentYes     balance 
   2.04e-05    4.80e-01    1.01e+00 
```

- Holding student fixed, the odds of default will increase by $(e^{\hat{\beta}_{balance}} - 1) = 0.58\%$ when increasing balance by one unit (one dollar).

- $e^{\hat{\beta}_0}$ is the odds of defaulting for a non-student with zero balance.

or

- Holding balance fixed, $e^{\hat{\beta}_{student}}$ is the default odds ratio for students versus non-students

- Holding student fixed, $e^{\hat{\beta}_{balance}}$ is the default odds ratio corresponding to an increase of 1 dollar in balance

Goodness of fit

- Hosmer-Lemeshow (HL) test to assess of the null hypothesis that the model fits the data well.

```
library(ResourceSelection)
hoslem.test(best_logistic$y, fitted(best_logistic))
```

```
    Hosmer and Lemeshow goodness of fit (GOF) test

data:  best_logistic$y, fitted(best_logistic)
X-squared = 4, df = 8, p-value = 0.8
```

- We do not reject the null hypothesis and conclude that the model fits the data well.

Discrimination Analysis - building our classifier

- when $\hat{p}_i < 0.5$, no default
- when $\hat{p}_i \geq 0.5$, yes default

Would the cutoff of 0.5 useful for classification? Let us check.

- First let us use the trained logistic regression model (with student and income) to obtain predicted probabilities for the test set:

```
library(dplyr)
testm <- mutate(test, predprob=predict(best_logistic,
                    newdata=x_test, type="response"))
head(testm,3)
```

```
     default student balance income predprob
8367      No     Yes    1908  17643 0.374866
667       No      No     362  33675 0.000165
1253      No      No     559  55007 0.000516
```

Confusion matrix:

```
testm <- mutate(testm, predout = ifelse(predprob < 0.5, "no", "yes"))
xtabs( ~ default + predout, testm)
```

```
       predout
default   no  yes
    No  1928    6
    Yes   46   20
```

Overall correct classification rate (accuracy):

```
(1928+20)/(2000)
```

```
[1] 0.974
```

Sensitivity and Specificity

- Sensitivity = true positive rate, that is, the proportion of actual positives that are correctly identified as such

# of predicted subjects who have defaulted / # of observed subjects who have defaulted

$$20/(46 + 20) = 0.303$$

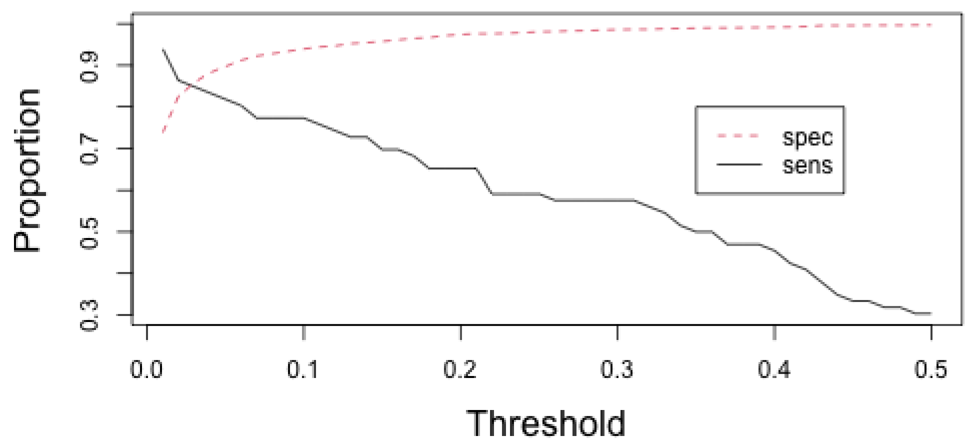
- Specificity = true negative rate, that is, the proportion of actual negatives that are correctly identified as such

# of predicted subjects who have not defaulted / # of obs. subjects who have not defaulted

$$1928/(1928 + 6) = 0.997$$

$\rightarrow$ Very low sensitivity, we should search for a better cutoff

```r
thresh <- seq(0.01,0.5,0.01)
Sens<- numeric(length(thresh))
Spec <- numeric(length(thresh))
for(j in seq(along=thresh)){
  pp <- ifelse(testm$predprob < thresh[j],"no","yes")
  xx <- xtabs( ~ default + pp, testm)
  Spec[j] <- xx[1,1]/(xx[1,1]+xx[1,2])
  Sens[j] <- xx[2,2]/(xx[2,1]+xx[2,2])
}
matplot(thresh,cbind(Sens,Spec),type="l", xlab="Threshold",
        ylab="Proportion", lty=1:2, cex.lab=1.5)
legend(.35, 0.8, c("spec", "sens"), lty=c(2,1), col=c(2,1))
```



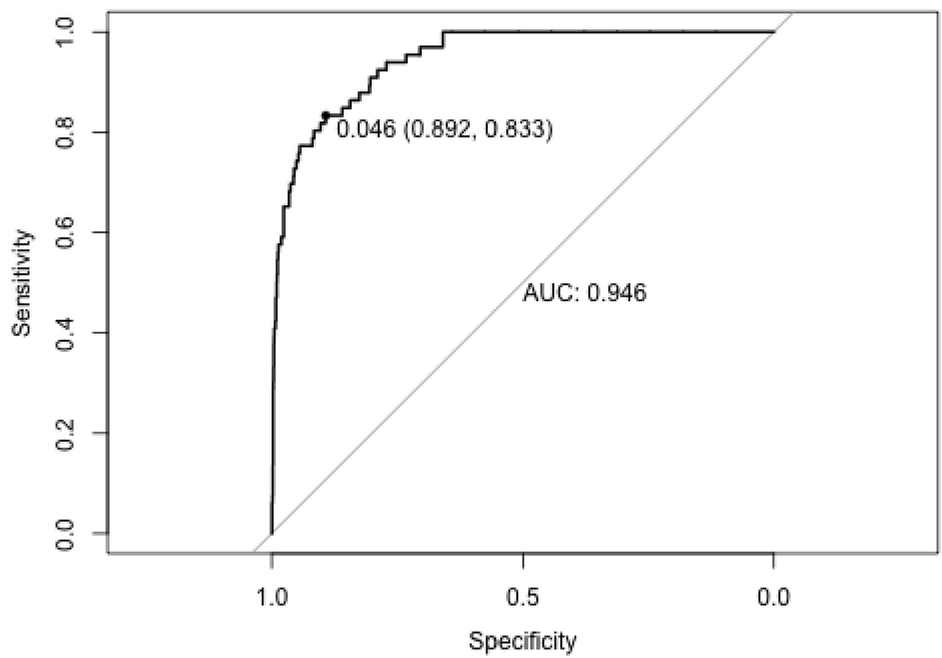
Receiver operating characteristic (ROC) curve

- The ROC curve is a commonly used way to assess the trade-off between sensitivity and specificity over all possible thresholds
- Overall performance of a classifier can be summarized using the area under the ROC curve (AUC)
- An ideal ROC curve will reach the top left corner, so the larger the AUC the better the classifier

```r
library(pROC)
roc_obj <- roc(response=testm$default, predictor=testm$predprob)
AUC <- auc(roc_obj)
roc_logistic <- c(coords(roc_obj, "b",
ret=c("threshold","se","sp","accuracy"), best.method="youden"), AUC)
names(roc_logistic) <- c("Threshold","Sensitivity","Specificity",
                         "Accuracy","AUC")
t(roc_logistic)
```

```
     Threshold Sensitivity Specificity Accuracy AUC
[1,] 0.0462    0.833       0.892       0.89     0.946
```

```r
plot(roc_obj, legacy.axes=F, print.auc=T, print.thres=T)
```



Reference

James, G, Witten, D, Hastie, T, Tibshirani, R (2013) *An Introduction to Statistical Learning*. Springer, New York, Second edition.