Modeling

Shukry Zablah

05 December, 2018

Contents

Imports

```
library(dplyr)
library(ROCR)
```

Load Data

```
train <- readRDS(file = "../data/PIMA_train.Rds")
test <- readRDS(file = "../data/PIMA_test.Rds")
```

Baseline

```
tally(~ hasDiabetes, data = train)
```

```
## hasDiabetes
##   0   1
## 196  98
```

Any model we choose has to have an accuracy higher than 98/(98+196) = 33%. This is the baseline accuracy score.

Logistic Regression Classifier

Training

```
clf <- glm(hasDiabetes ~ ., family = binomial(link='logit'), data = train)
summary(clf)
```

```
##
## Call:
## glm(formula = hasDiabetes ~ ., family = binomial(link = "logit"),
##     data = train)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8732  -0.6344  -0.3526   0.5918   2.2488
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -1.090e+01  1.516e+00  -7.190 6.48e-13 ***
## pregnancies               1.310e-01  6.738e-02   1.944  0.05192 .
## glucoseConcentration      4.080e-02  6.740e-03   6.054 1.41e-09 ***
## bloodPressure             2.284e-03  1.367e-02   0.167  0.86732
## skinThickness            -3.725e-06  2.118e-02   0.000  0.99986
## insulin                  -1.764e-03  1.544e-03  -1.142  0.25325
## bmi                       9.483e-02  3.472e-02   2.731  0.00631 **
## diabetesPedigreeFunction  1.472e+00  4.967e-01   2.963  0.00305 **
## age                       2.227e-02  2.099e-02   1.061  0.28881
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 374.27  on 293  degrees of freedom
## Residual deviance: 251.84  on 285  degrees of freedom
## AIC: 269.84
##
## Number of Fisher Scoring iterations: 5
```

```
saveRDS(clf, file = "../models/LogisticRegressionClassifier_Full.Rds")
```

Model Evaluation on Train Set

```
predict <- predict(clf, type = 'response')

with(train,
     table(hasDiabetes, predict > 0.3))
```

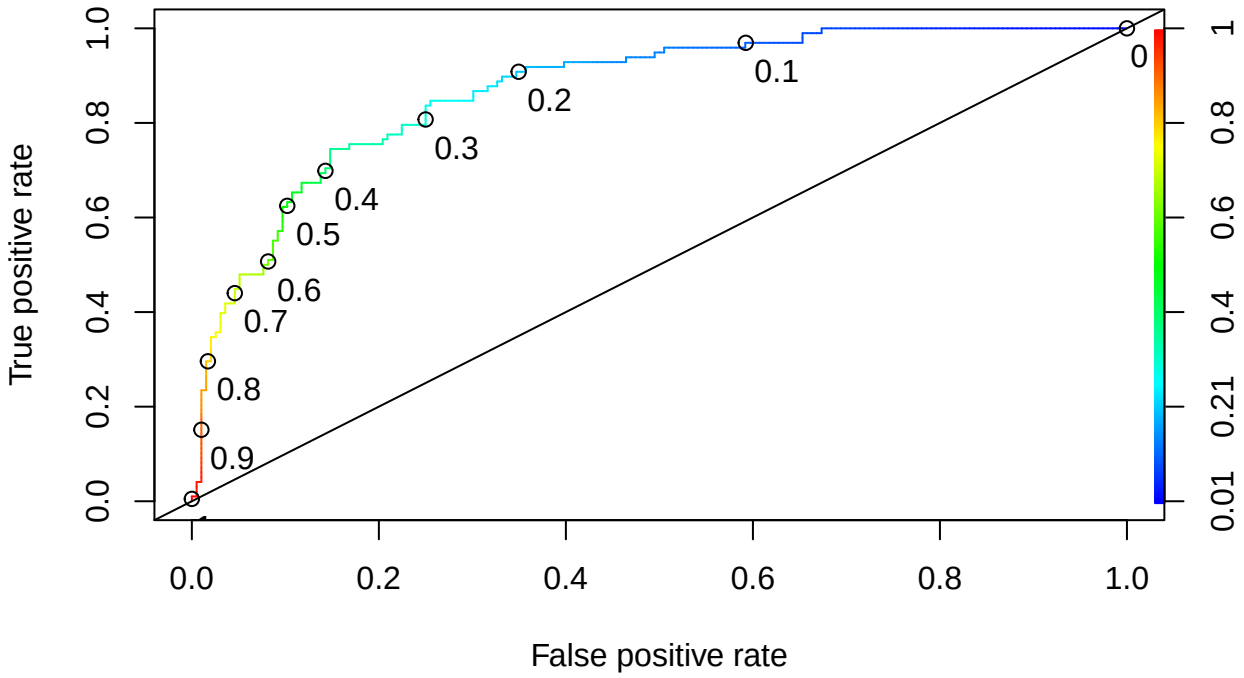
```
##
## hasDiabetes FALSE TRUE
##           0   147   49
##           1    19   79
```

We can see in the confusion matrix that our accuracy is $(79 + 147)/(147 + 49 + 19 + 79) = 0.7687$. This is with a cutoff of 0.3.

```
ROCRpred <- with(train,
                 prediction(predict, hasDiabetes))

ROCRperf <- performance(ROCRpred, 'tpr','fpr')

plot(ROCRperf, colorize = TRUE, print.cutoffs.at = seq(0,1,0.1), text.adj = c(-0.2,1.7)); abline(0,1)
```



In the ROC curve we can see that our model is good (the curve is away from the diagonal). Since we care about not predicting a negative result for someone that is actually positive for diabetes (false negative rate), we want to have a larger true positive rate (1 - TPR = FNR). This means that we choose a cutoff near the blue part of the curve, the lower the cutoff the more cautious our model and the less accurate.

Model Evaluation on Test Set

```
#only after making decision of cutoff and dealing with missing values....
```