****

**NBA player 5-year career longevity prediction with logistic regression.**



**Course: ALY 6020 Predictive Analytics**

**Instructor: Dr. Marco Montes de Oca**

**Submitted by: Nithish Saravanan**

Data is read using read.csv command and stored as myData1 object. The head(myData1) gives the details about the columns in the dataset along with few values.

> head(myData1)

Name GP MIN PTS FGM FGA FG. X3P.Made X3PA X3P. FTM FTA FT. OREB DREB REB AST

1 Brandon Ingram 36 27.4 7.4 2.6 7.6 34.7 0.5 2.1 25.0 1.6 2.3 69.9 0.7 3.4 4.1 1.9

2 Andrew Harrison 35 26.9 7.2 2.0 6.7 29.6 0.7 2.8 23.5 2.6 3.4 76.5 0.5 2.0 2.4 3.7

3 JaKarr Sampson 74 15.3 5.2 2.0 4.7 42.2 0.4 1.7 24.4 0.9 1.3 67.0 0.5 1.7 2.2 1.0

4 Malik Sealy 58 11.6 5.7 2.3 5.5 42.6 0.1 0.5 22.6 0.9 1.3 68.9 1.0 0.9 1.9 0.8

5 Matt Geiger 48 11.5 4.5 1.6 3.0 52.4 0.0 0.1 0.0 1.3 1.9 67.4 1.0 1.5 2.5 0.3

6 Tony Bennett 75 11.4 3.7 1.5 3.5 42.3 0.3 1.1 32.5 0.4 0.5 73.2 0.2 0.7 0.8 1.8

STL BLK TOV TARGET\_5Yrs

1 0.4 0.4 1.3 0

2 1.1 0.5 1.6 0

Performing Exploratory Data Analysis on the data set gives descriptions and observations about the data set.

> ExpData(data=data\_nba, type=1)

Descriptions Obs

1 Sample size (Nrow) 1340

2 No. of Variables (Ncol) 21

3 No. of Numeric Variables 20

4 No. of Factor Variables 1

5 No. of Text Variables 0

6 No. of Logical Variables 0

7 No. of Date Variables 0

8 No. of Zero variance Variables (Uniform) 0

9 %. of Variables having complete cases 95.24% (20)

10 %. of Variables having <50% missing cases 4.76% (1)

11 %. of Variables having >50% missing cases 0% (0)

12 %. of Variables having >90% missing cases 0% (0)

From this we were able to see that Next step is to handle the missing data in the data set.

> #Handling the missing values

> nba\_missing = myData1 %>%

+ summarise\_all((funs(sum(is.na(.)))))

> nba\_missing

Name GP MIN PTS FGM FGA FG. X3P.Made X3PA X3P. FTM FTA FT. OREB DREB REB AST STL BLK TOV

1 0 0 0 0 0 0 0 0 0 11 0 0 0 0 0 0 0 0 0 0

TARGET\_5Yrs

1 0

TARGET\_5Yrs

1 0

There are 11 tuples in X3P factor missing. We can substitute with the mean value of the variable in the tuples that are missing.

> #Hadling missing values for the feature that has null values

> myData1$X3P. <- ifelse(is.na(myData1$X3P.),

+ ave(myData1$X3P.,FUN = function(x) mean(x,na.rm = TRUE)),

+ myData1$X3P.)

The mean value of the variable X3P gets filled in the place of the 11 tuples where the data was missing. This can be verified by doing EDA again

> #EDA of the data given after cleaning

> ExpData(data=myData1, type=1)

Descriptions Obs

1 Sample size (Nrow) 1340

2 No. of Variables (Ncol) 21

3 No. of Numeric Variables 20

4 No. of Factor Variables 1

5 No. of Text Variables 0

6 No. of Logical Variables 0

7 No. of Date Variables 0

8 No. of Zero variance Variables (Uniform) 0

9 %. of Variables having complete cases 100% (21)

10 %. of Variables having <50% missing cases 0% (0)

11 %. of Variables having >50% missing cases 0% (0)

12 %. of Variables having >90% missing cases 0% (0)

Now we segregate the data based upon the number of one and zeros present in the target variable.

> dataOnes<-myData1[which(myData1$TARGET\_5Yrs==1),]

> dataZeros<-myData1[which(myData1$TARGET\_5Yrs==0),]

After setting the seed value we will start creating the training and test data set with 80% to 20% ratio.

> set.seed(100)

> #\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Creating Training Data\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

> trainingOnesRows<-sample(1:nrow(dataOnes),0.8\*nrow(dataOnes))

> trainingZerosRows<-sample(1:nrow(dataZeros),0.8\*nrow(dataZeros))

> trainingOnes<-dataOnes[trainingOnesRows,]

> trainingZeros<-dataZeros[trainingZerosRows,]

> trainingData<-rbind(trainingOnes,trainingZeros)

> #\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Creating Test Data\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

> testOnes<-dataOnes[-trainingOnesRows,]

> testZeros<-dataZeros[-trainingZerosRows,]

> testData<-rbind(testOnes,testZeros)

Now we create a general linear regression model with training data and linear model family as logit as we are doing a logistic regression.

> Model1<- glm(TARGET\_5Yrs ~ GP+MIN+PTS+FGM+FGA+X3PA+FTM+FTA+OREB+DREB+REB+AST+STL+BLK+TOV,data = trainingData,family= binomial(link="logit"))

> summary(Model1)

Call:

glm(formula = TARGET\_5Yrs ~ GP + MIN + PTS + FGM + FGA + X3PA +

FTM + FTA + OREB + DREB + REB + AST + STL + BLK + TOV, family = binomial(link = "logit"),

data = trainingData)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.9617 -0.9792 0.5172 0.8915 2.0423

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -2.509479 0.282216 -8.892 < 2e-16 \*\*\*

GP 0.041471 0.005288 7.842 4.44e-15 \*\*\*

MIN -0.035569 0.037410 -0.951 0.342

PTS 1.238234 0.779677 1.588 0.112

FGM -2.298012 1.639563 -1.402 0.161

FGA -0.065367 0.156347 -0.418 0.676

X3PA -0.424317 0.306971 -1.382 0.167

FTM -0.324197 0.915285 -0.354 0.723

FTA -0.477938 0.383904 -1.245 0.213

OREB -0.422566 1.444226 -0.293 0.770

DREB -1.349293 1.448131 -0.932 0.351

REB 1.216962 1.437924 0.846 0.397

AST 0.194068 0.122623 1.583 0.114

STL 0.132578 0.345093 0.384 0.701

BLK 0.326337 0.296845 1.099 0.272

TOV -0.109496 0.300640 -0.364 0.716

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1422.4 on 1070 degrees of freedom

Residual deviance: 1181.8 on 1055 degrees of freedom

AIC: 1213.8

Number of Fisher Scoring iterations: 5

As the model created is in canonical correlation we will convert this into inverse relation using exponential function such that we can have a clear idea about the odds and the variables that are to be considered with more importance for a long career.

> exp(coef(Model1))

(Intercept) GP MIN PTS FGM FGA X3PA

0.08228623 1.04236184 0.95486195 0.71224240 1.72077072 1.10597260 0.26388027

X3P.Made FTM FTA OREB DREB REB AST

55.78973314 3.20686762 0.67065643 0.59365225 0.23794013 3.76298346 1.22454855

STL BLK TOV

1.23758408 1.39553234 0.88045464

From this we can find out that the variables such as 3 points made, rebounds, assists, FTM will have more impact on the longevity of the player. Now we predict the model with test data

> pred<-plogis(predict(Model1,testData,type = "response"))

We use plogis () because we wanted to predict the outcome as 0’s and 1’s.

Now we need to find the optimal cutoff point for the predicted values to increase the accuracy and reduce the misclassification error.

> optCutOff <- optimalCutoff(testData$TARGET\_5Yrs, pred)[1]

> optCutOff

[1] 0.6273455

> #vif(pred)

> err=misClassError(testData$TARGET\_5Yrs, pred, threshold = optCutOff)

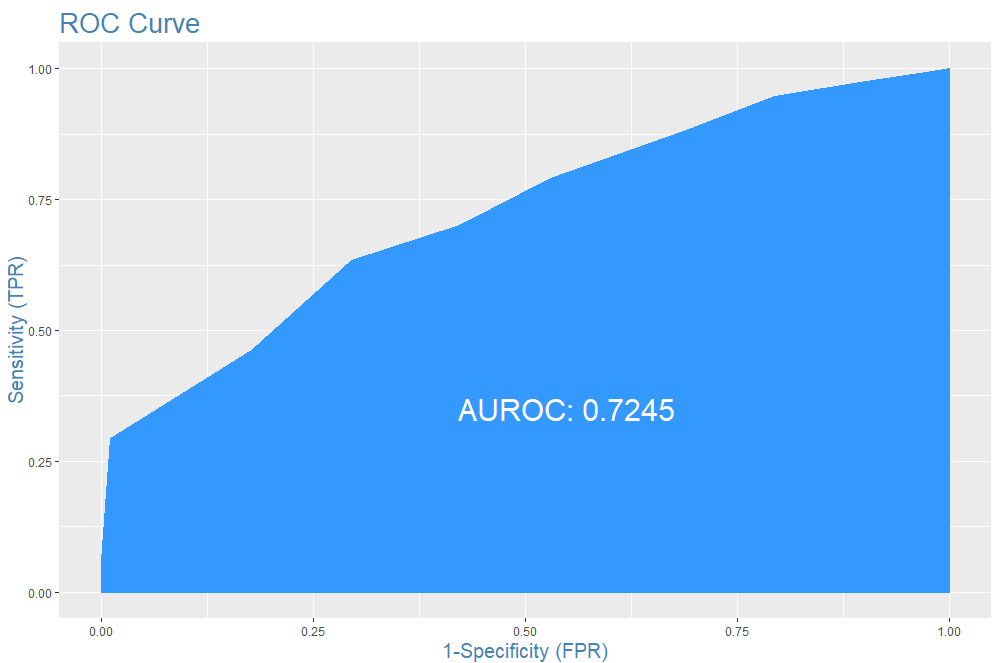
> err

[1] 0.3123

So we were able to get the cut off point as 0.63 and misclassification error as 0.3123 respectively.

The next step is to find the area under receiving operating characteristic curve by using the plotroc() method .

> plotROC(testData$TARGET\_5Yrs, pred)



The model created has an area of 72.45% under the curve which is good.

Now the concordance of the model is calculated in addition to have scores.

> Concordance(testData$TARGET\_5Yrs, pred)

$Concordance

[1] 0.7324175

$Discordance

[1] 0.2675825

$Tied

[1] 5.551115e-17

$Pairs

[1] 17034

The value of concordance is greater than the value of discordance and this show that the quality of model is good. For the available NBA data set the concordance is 73.24% which is a good quality model.

> sensitivity(testData$TARGET\_5Yrs, pred, threshold = optCutOff)

[1] 0.7784431

> specificity(testData$TARGET\_5Yrs, pred, threshold = optCutOff)

[1] 0.5392157

With sensitivity and specificity we can say that the model has predicted 77.84% and 53.92% respectively with the test data.

Now we get the confusion matrix by

> confusionMatrix(testData$TARGET\_5Yrs, pred, threshold = optCutOff)

0 1

0 55 37

1 47 130

This say that 55 players in the NBA list are predicted to be True Positive and 130 player are predicted to be True negative.