Lab 2

Sebastián Soriano Pérez | Juan David Martínez Gordillo - NetID: ss1072 | jdm127

September 20, 2019

Exercise 1

- Make exploratory plots to explore the relationships between Win and the following variables: Home, TeamPoints, FieldGoals., Assists, Steals, Blocks and Turnovers. Don't include any of the plots, just briefly describe the relationships.

EDA

```
   Home       TeamPoints     FieldGoals.        Assists
 Away:164   Min.   : 75.0   Min.   :0.3180   Min.   : 7.00
 Home:164   1st Qu.: 98.0   1st Qu.:0.4285   1st Qu.:17.00
            Median :106.0   Median :0.4605   Median :20.00
            Mean   :106.3   Mean   :0.4616   Mean   :20.58
            3rd Qu.:114.0   3rd Qu.:0.4963   3rd Qu.:24.00
            Max.   :133.0   Max.   :0.5920   Max.   :38.00
     Steals          Blocks          Turnovers          Win
 Min.   : 1.000   Min.   : 0.000   Min.   : 4.00   Min.   :0.0000
 1st Qu.: 6.000   1st Qu.: 3.000   1st Qu.: 9.00   1st Qu.:0.0000
 Median : 8.000   Median : 5.000   Median :12.00   Median :1.0000
 Mean   : 7.787   Mean   : 5.201   Mean   :12.22   Mean   :0.6555
 3rd Qu.: 9.000   3rd Qu.: 7.000   3rd Qu.:15.00   3rd Qu.:1.0000
 Max.   :17.000   Max.   :16.000   Max.   :23.00   Max.   :1.0000
```

The relationship between *Win* and *TeamPoints* depicts the obvious positive relation of winning with a higher number of points in a match. When the team wins they score more points compared to when they lose. The same positive relation exists when comparing *Win* to *FieldGoals* and *Assists*. However, when we compare *Win* to *Steals* and *Blocks* the differential is not that clear, this means that there is no clear relationship between the number of steals and blocks, and the team winning a match. Lastly, for the case of home and away games, it is clear that the team performs better at home, as it would've been expected.

Exercise 2

- There are several combinations of variables we should not include as predictors in the logistic model. Identify at least two pairs and explain in at most two sentences, why we should not include them in the model at the same time.

```r
cor <- cor(nba_reduced[, -c(1, 2, 3, 4, 5, 6, 7)])

cor[upper.tri(cor)] <- NA

cor_df <- as.data.frame(as.table(cor))

colnames(cor_df) <- c("Variable 1", "Variable 2", "Corr")

high_risk_pairs <- subset(cor_df, (abs(Corr) >= 0.8) & (abs(Corr) <
    1))

high_risk_pairs[order(-high_risk_pairs$Corr), ]
```

Variable 1	Variable 2	Corr
FreeThrowsAttempted	FreeThrows	0.945
Opp.FreeThrowsAttempted	Opp.FreeThrows	0.923
Opp.FieldGoals	OpponentPoints	0.829
FieldGoals	TeamPoints	0.819
Opp.TotalFouls	FreeThrowsAttempted	0.814

We should not include the variables presented in the table since they have a big correlation and would bring problems of multicollinearity.

Exercise 3

- Fit a logistic regression model for Win (or WinorLoss) using Home, TeamPoints, FieldGoals., Assists, Steals, Blocks and Turnovers. as your predictors. Using the vif function, are there are any concerns regarding multicollinearity in this model?

```r
logit_nba <- glm(Win ~ Home + TeamPoints + FieldGoals. + Assists +
    Steals + Blocks + Turnovers, family = binomial(link = logit),
    data = nba_reduced_train)

summary(logit_nba)
```

```
Call:
```

```
glm(formula = Win ~ Home + TeamPoints + FieldGoals. + Assists +
    Steals + Blocks + Turnovers, family = binomial(link = logit),
    data = nba_reduced_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2374  -1.0420   0.5708   0.8943   1.8076

Coefficients:
            Estimate Std. Error z value    Pr(>|z|)
(Intercept) -9.83235    1.93471  -5.082 0.000000373 ***
HomeHome     0.46939    0.29767   1.577     0.11482
TeamPoints   0.02534    0.02027   1.250     0.21128
FieldGoals. 11.96916    4.13707   2.893     0.00381 **
Assists      0.02782    0.03675   0.757     0.44901
Steals       0.08611    0.05628   1.530     0.12603
Blocks       0.09207    0.06026   1.528     0.12655
Turnovers    0.03936    0.04185   0.941     0.34690
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 323.10  on 245  degrees of freedom
Residual deviance: 276.94  on 238  degrees of freedom
AIC: 292.94

Number of Fisher Scoring iterations: 4
```

```r
vif(logit_nba)
```

```
       Home  TeamPoints FieldGoals.     Assists      Steals      Blocks
   1.035553    1.853202    1.786862    1.249512    1.101457    1.014406
  Turnovers
   1.174134
```

None of the variables has a VIF value greater than 10, which means we should not worry about multicollinearity in this model. In fact, all values are pretty close to 1, which implies that the predictor variables are not correlated or just moderately correlated. **However**, when we see the summary of the model, the coefficient for *FieldGoals.* is extremely high as well as its standard error. A likely cause for the incredibly large odd ratio and very large standard error is the multicollinearity among the independent variables of our model. As we saw in the previous item, *FieldGoals.* and *TeamPoints* have a very high correlation coefficient, leading us to think that this pair of variables is bringing problems due to multicollinearity. This is why we decide to drop the variable *FieldGoals.* and re-run the logit model.

Exercise 4

- Present the output of the fitted model and interpret the significant coefficients in terms of the odds of your team winning an NBA game.

```r
logit_nba2 <- glm(Win ~ Home + TeamPoints + Assists + Steals +
    Blocks + Turnovers, family = binomial(link = logit), data = nba_reduced_train)

summary(logit_nba2)
```

```
Call:
glm(formula = Win ~ Home + TeamPoints + Assists + Steals + Blocks +
    Turnovers, family = binomial(link = logit), data = nba_reduced_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0816  -1.0633   0.6151   0.9300   1.6504

Coefficients:
            Estimate Std. Error z value   Pr(>|z|)
(Intercept) -8.58112    1.81754  -4.721 0.00000234 ***
HomeHome     0.45739    0.29175   1.568   0.116938
TeamPoints   0.06084    0.01627   3.739   0.000185 ***
Assists      0.05238    0.03482   1.504   0.132479
Steals       0.06716    0.05479   1.226   0.220316
Blocks       0.07279    0.05888   1.236   0.216387
Turnovers    0.06215    0.04052   1.534   0.125025
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 323.10  on 245  degrees of freedom
Residual deviance: 285.91  on 239  degrees of freedom
AIC: 299.91

Number of Fisher Scoring iterations: 4
```

```r
exp(coefficients(logit_nba2))
```

```
 (Intercept)     HomeHome   TeamPoints      Assists       Steals
0.0001876143 1.5799431924 1.0627237820 1.0537760884 1.0694617671
      Blocks    Turnovers
1.0755070774 1.0641272682
```

The output of the logit model shows that the only significant coefficient is the one for *TeamPoints*.

The coefficient for the variable *TeamPoints* is 0.06084. This means that for a one-unit increase in *TeamPoints*, we expect a 0.06084 increase in the log-odds of the dependent variable *Win*, holding all other independent variables constant.

To interpret better this coefficient, we can exponentiate it and analyze its value in terms of odds ratios. In this case, we can say for a one-unit increase in *TeamPoints*, we expect to see about a 6.27% increase in the odds of winning a game.

Exercise 5

- Using 0.5 as your cutoff for predicting wins or losses (1 vs 0) from the predicted probabilities, what is the accuracy of this model? Plot the roc curve for the fitted model. What is the AUC value?

```
cutoff = 0.5

Conf_mat <- confusionMatrix(as.factor(ifelse(fitted(logit_nba2) >=
    cutoff, "W", "L")), nba_reduced_train$WINorLOSS, positive = "W")
Conf_mat$table
```

```
          Reference
Prediction   L   W
         L  39  21
         W  51 135
```

```
Conf_mat$overall["Accuracy"]
```

```
 Accuracy
0.7073171
```

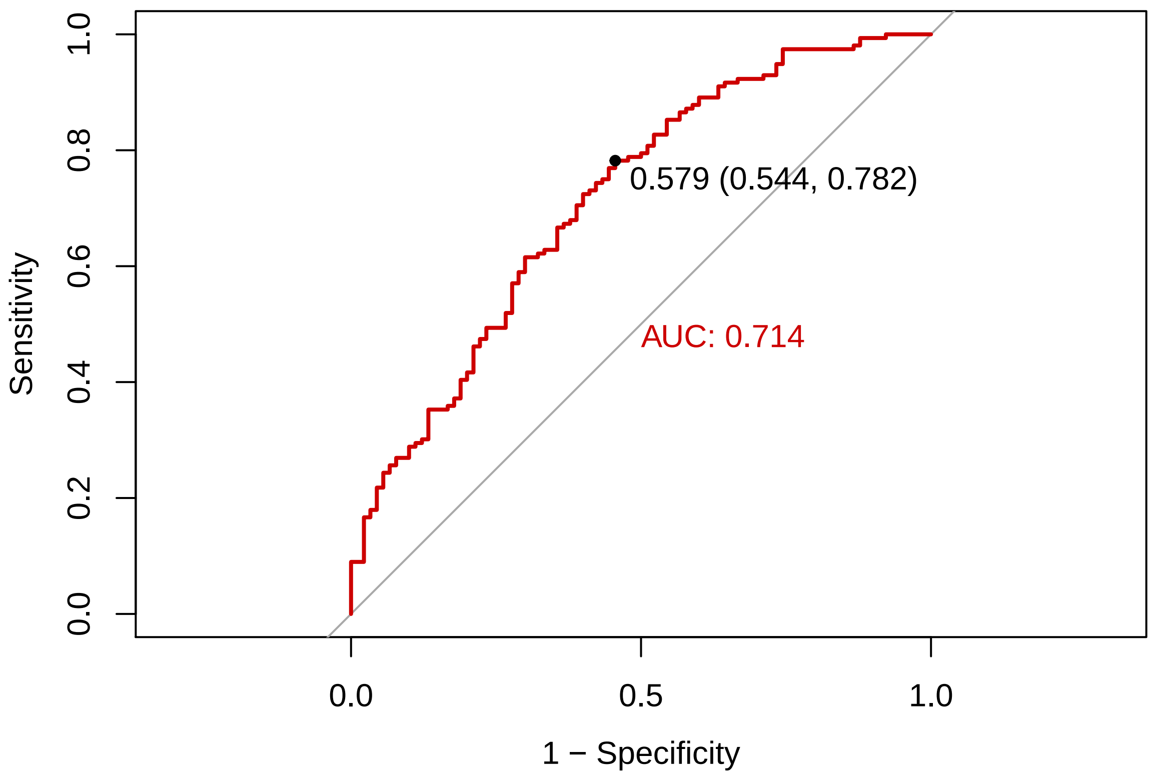
```
Conf_mat$byClass[c("Sensitivity", "Specificity")]
```

```
Sensitivity Specificity
  0.8653846   0.4333333
```

```
invisible(roc(nba_reduced_train$WINorLOSS, fitted(logit_nba2),
    plot = T, print.thres = "best", legacy.axes = T, print.auc = T,
    col = "red3"))
```

```
Setting levels: control = L, case = W
```

```
Setting direction: controls < cases
```



The accuracy of the model is 0.707 and the AUC is 0.714.

Exercise 6

- Now add Opp.FieldGoals. as a predictor to the previous model. Is the coefficient significant? If yes, interpret the coefficient in the context of the question.

```
logit_nba3 <- glm(Win ~ Home + TeamPoints + Opp.FieldGoals. +
    Assists + Steals + Blocks + Turnovers, family = binomial(link = logit),
    data = nba_reduced_train)

summary(logit_nba3)
```

```
Call:
glm(formula = Win ~ Home + TeamPoints + Opp.FieldGoals. + Assists +
    Steals + Blocks + Turnovers, family = binomial(link = logit),
    data = nba_reduced_train)
```

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3622  -0.5746   0.2370   0.6026   2.2600

Coefficients:
                  Estimate Std. Error z value         Pr(>|z|)
(Intercept)       2.063886   2.657404   0.777          0.43736
HomeHome          0.357192   0.366844   0.974          0.33021
TeamPoints        0.126979   0.023378   5.432 0.00000005588228 ***
Opp.FieldGoals. -39.646819   5.619550  -7.055 0.00000000000172 ***
Assists           0.112029   0.046404   2.414          0.01577 *
Steals            0.217024   0.074434   2.916          0.00355 **
Blocks           -0.078905   0.075014  -1.052          0.29286
Turnovers        -0.009045   0.052209  -0.173          0.86245
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 323.10  on 245  degrees of freedom
Residual deviance: 194.48  on 238  degrees of freedom
AIC: 210.48

Number of Fisher Scoring iterations: 6
```

```
exp(coefficients(logit_nba3))
```

```
               (Intercept)                   HomeHome
7.876520538222242961978736 1.429310318449516836736279
                TeamPoints            Opp.FieldGoals.
1.135393737155658167026218 0.000000000000000006047909
                   Assists                     Steals
1.118544952209291176714601 1.242373410778876641202828
                    Blocks                  Turnovers
0.924128092741786644204183 0.990995328150996890315128
```

The coefficient for *Opp.FieldGoals.* is significant. We can interpret from the model's output that for a one-unit increase in *Opp.FieldGoals.*, we expect to see about 99.9999999999999993952091% decrease in the odds of winning a game.

Exercise 7

- What is the accuracy of this new model? Plot the ROC curve for the fitted model. What is the new AUC value? Which model predicts the odds of winning better?

```
cutoff = 0.5

Conf_mat <- confusionMatrix(as.factor(ifelse(fitted(logit_nba3) >=
    cutoff, "W", "L")), nba_reduced_train$WINorLOSS, positive = "W")
Conf_mat$table
```

```
          Reference
Prediction   L   W
         L  66  17
         W  24 139
```

```
Conf_mat$overall["Accuracy"]
```

```
 Accuracy
0.8333333
```

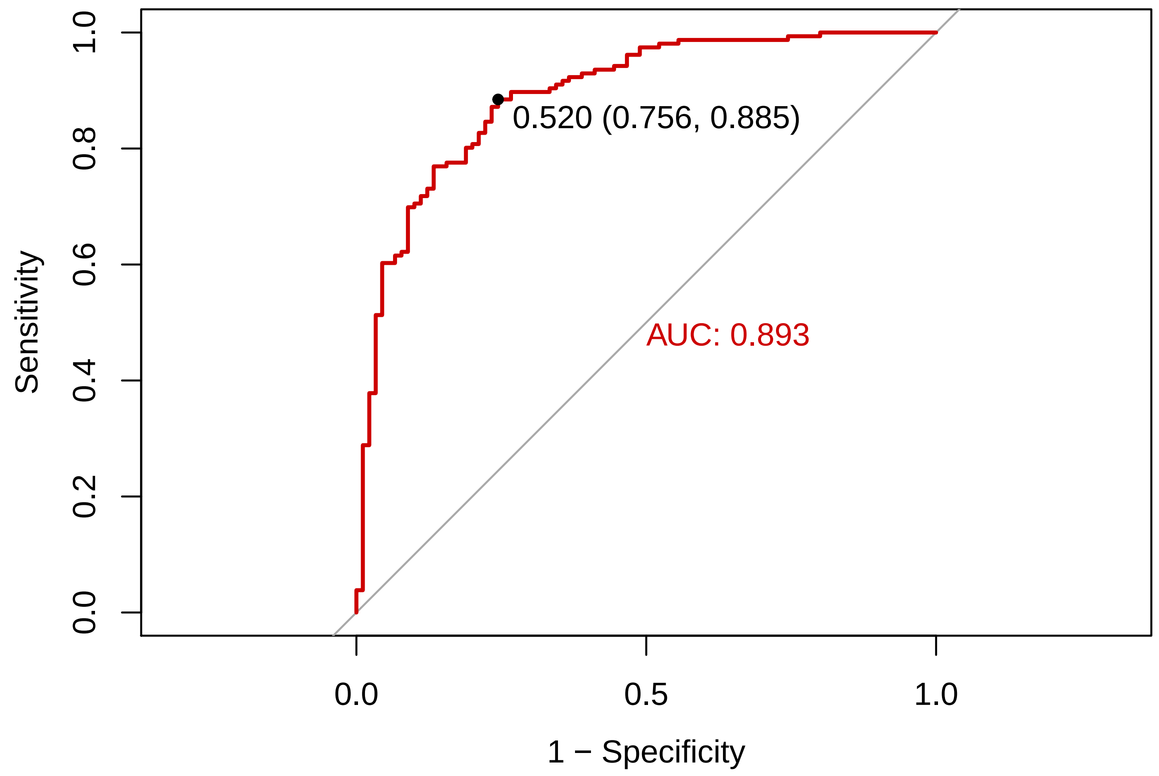
```
Conf_mat$byClass[c("Sensitivity", "Specificity")]
```

```
Sensitivity Specificity
  0.8910256   0.7333333
```

```
invisible(roc(nba_reduced_train$WINorLOSS, fitted(logit_nba3),
    plot = T, print.thres = "best", legacy.axes = T, print.auc = T,
    col = "red3"))
```

```
Setting levels: control = L, case = W
```

```
Setting direction: controls < cases
```



The accuracy of the model increases dramatically to 0.83. Also, the AUC increased to 0.893. It is very clear that this new model is better when it comes to predicting the odds of winning a game.

Exercise 8

- Using the results of the model with the better predictive ability, what suggestions do you have for the coach of your team trying to improve the odds of his team winning a regular season game?

Besides from the coefficient for *Opp.FieldGoals.*, our new model depicts the following insights:

-
  - With a one-unit increase in *TeamPoints*, we expect to see about a 13.5% increase in the odds of winning a game.
  - With a one-unit increase in *Assists*, we expect to see about an 11.8% increase in the odds of winning a game.
  - With a one-unit increase in *Steals*, we expect to see about a 24.2% increase in the odds of winning a game.

This being said, our suggestion to the coach would be to emphasize a strategy in which the team focuses more on stealing the ball from the opponent and doing more assists. Additionally, there should be a focus as well on increasing the team's points and reducing the opponent's points (Although the effect of this is minimal, as seen on the previous items)

Exercise 9

- Use this model to predict out-of-sample probabilities for the nba_reduced_test data. Using 0.5 as your cutoff for predicting wins or losses (1 vs 0) from the out-of-sample predicted probabilities, what is the out-of-sample accuracy? How well does your model do in predicting data for the 2017/2018 season?

```
cutoff = 0.5

Conf_mat <- confusionMatrix(as.factor(ifelse(predict(logit_nba3,
    nba_reduced_test, type = "response") >= cutoff, "W", "L")),
    nba_reduced_test$WINorLOSS, positive = "W")

Conf_mat$table
```

```
          Reference
Prediction  L  W
         L 10  1
         W 13 58
```

```
Conf_mat$overall["Accuracy"]
```

```
 Accuracy
0.8292683
```

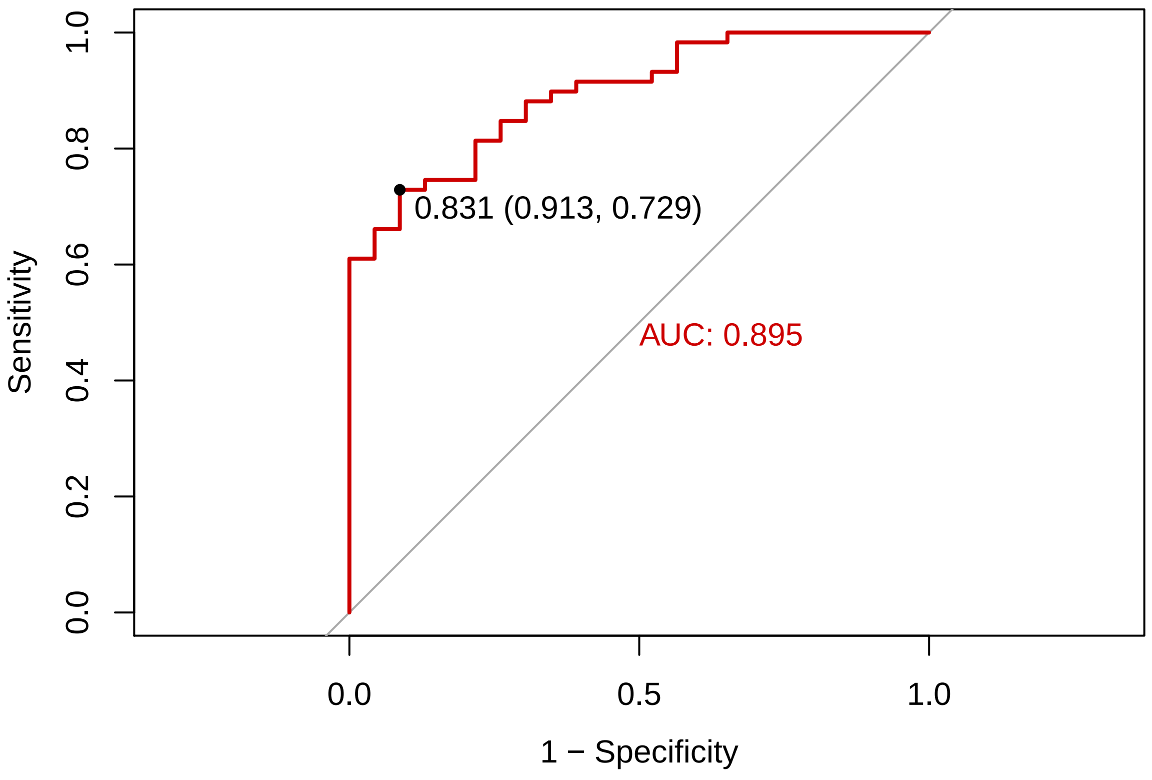
```
Conf_mat$byClass[c("Sensitivity", "Specificity")]
```

```
Sensitivity Specificity
  0.9830508   0.4347826
```

```
invisible(roc(nba_reduced_test$WINorLOSS, predict(logit_nba3,
    nba_reduced_test, type = "response"), plot = T, print.thres = "best",
    legacy.axes = T, print.auc = T, col = "red3"))
```

```
Setting levels: control = L, case = W
```

```
Setting direction: controls < cases
```



The out-of-sample accuracy is 82.92 and the out-of-sample AUC is 0.895. Our model does very well in predicting data for the 2017/2018 season since the in-sample values for accuracy and AUC are maintained even for the new data coming in the test set.

Exercise 10

- Using the change in deviance test, test whether including Opp.Assists and Opp.Blocks in the model at the same time would improve the model. Is there any other variable in this dataset which we did not consider that you think might improve our model? Which one and why?

```
logit_nba4 <- glm(Win ~ Home + TeamPoints + Opp.FieldGoals. +
    Assists + Steals + Blocks + Turnovers + Opp.Assists + Opp.Blocks,
    family = binomial(link = logit), data = nba_reduced_train)

summary(logit_nba4)
```

```
Call:
glm(formula = Win ~ Home + TeamPoints + Opp.FieldGoals. + Assists +
    Steals + Blocks + Turnovers + Opp.Assists + Opp.Blocks, family = binomial(link = logit),
    data = nba_reduced_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3572  -0.4618   0.2333   0.5682   2.2854

Coefficients:
                 Estimate Std. Error z value      Pr(>|z|)
(Intercept)      2.207981   2.807760   0.786       0.43164
HomeHome         0.175610   0.385770   0.455       0.64895
TeamPoints       0.138617   0.025580   5.419 0.00000005997 ***
Opp.FieldGoals. -35.563571   5.886091  -6.042 0.00000000152 ***
Assists          0.129440   0.050278   2.575       0.01004 *
Steals           0.237594   0.077500   3.066       0.00217 **
Blocks          -0.061778   0.077900  -0.793       0.42776
Turnovers        0.002786   0.054686   0.051       0.95936
Opp.Assists     -0.148535   0.052996  -2.803       0.00507 **
Opp.Blocks      -0.099947   0.081322  -1.229       0.21906
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 323.10  on 245  degrees of freedom
Residual deviance: 182.93  on 236  degrees of freedom
AIC: 202.93

Number of Fisher Scoring iterations: 6
```

```
anova(logit_nba3, logit_nba4, test = "Chisq")
```

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
238	194			
236	183	2	11.5	0.00311

At the 0.05 level, we can conclude that including the variables *Opp.Assists* and *Opp.Blocks* contributes to enhancing our logit model.

Also, to improve our model we would add more information regarding the performance of the rival team in each match. The variables *Opp.TotalFouls*, *Opp.Turnovers*, *Opp.Steals* and *Opp.FreeThrows.* offer a good overall view of the rival team and would definitely add more predictive power to the model:

```r
logit_nba5 <- glm(Win ~ Home + TeamPoints + Opp.FieldGoals. +
    Assists + Steals + Blocks + Turnovers + Opp.Assists + Opp.Blocks +
    Opp.TotalFouls + Opp.Turnovers + Opp.Steals + Opp.FreeThrows.,
    family = binomial(link = logit), data = nba_reduced_train)

summary(logit_nba5)
```

```

Call:
glm(formula = Win ~ Home + TeamPoints + Opp.FieldGoals. + Assists +
    Steals + Blocks + Turnovers + Opp.Assists + Opp.Blocks +
    Opp.TotalFouls + Opp.Turnovers + Opp.Steals + Opp.FreeThrows.,
    family = binomial(link = logit), data = nba_reduced_train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.3552  -0.3141   0.1201   0.4360   2.4335

Coefficients:
                 Estimate Std. Error z value       Pr(>|z|)
(Intercept)       5.17594    3.62538   1.428       0.153379
HomeHome          0.23739    0.43140   0.550       0.582126
TeamPoints        0.18452    0.03313   5.569 0.000000025554 ***
Opp.FieldGoals. -47.05832    7.62774  -6.169 0.000000000686 ***
Assists           0.13685    0.05883   2.326       0.020007 *
Steals           -0.06283    0.12181  -0.516       0.605998
Blocks           -0.05115    0.09156  -0.559       0.576379
Turnovers        -0.08867    0.09411  -0.942       0.346061
Opp.Assists      -0.17675    0.06180  -2.860       0.004238 **
Opp.Blocks       -0.12758    0.09689  -1.317       0.187898
Opp.TotalFouls    0.03880    0.05644   0.687       0.491859
Opp.Turnovers     0.36032    0.09712   3.710       0.000207 ***
Opp.Steals        0.06361    0.12688   0.501       0.616138
Opp.FreeThrows.  -5.97685    2.05787  -2.904       0.003680 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 323.10  on 245  degrees of freedom
Residual deviance: 151.58  on 232  degrees of freedom
AIC: 179.58

Number of Fisher Scoring iterations: 6
```

```r
anova(logit_nba4, logit_nba5, test = "Chisq")
```

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
236	183			
232	152	4	31.3	2.6e-06

After running the Analysis of Deviance test we can conclude that the new features included in our model are significant.

```r
cutoff = 0.5

Conf_mat <- confusionMatrix(as.factor(ifelse(predict(logit_nba5,
    nba_reduced_test, type = "response") >= cutoff, "W", "L")),
    nba_reduced_test$WINorLOSS, positive = "W")

Conf_mat$table
```

```
          Reference
Prediction  L  W
         L 10  1
         W 13 58
```

```r
Conf_mat$overall["Accuracy"]
```

```
 Accuracy
0.8292683
```

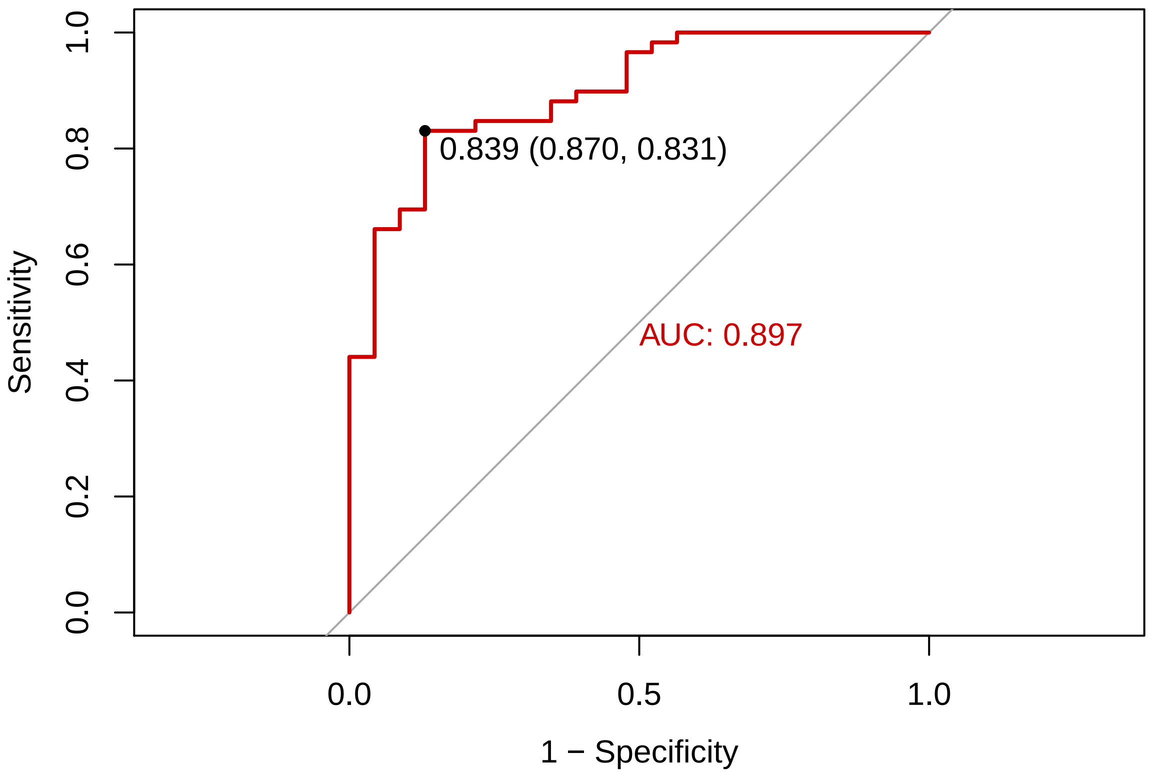
```r
Conf_mat$byClass[c("Sensitivity", "Specificity")]
```

```
Sensitivity Specificity
  0.9830508   0.4347826
```

```r
invisible(roc(nba_reduced_test$WINorLOSS, predict(logit_nba5,
    nba_reduced_test, type = "response"), plot = T, print.thres = "best",
    legacy.axes = T, print.auc = T, col = "red3"))
```

```
Setting levels: control = L, case = W

Setting direction: controls < cases
```



In terms of predictive power, our new model behaves just as well as our past model, only outperforming it by a marginal value in the AUC. **Nonetheless**, our new model is better when it comes to balance between sensitivity and specificity.