

UNITED STATES MILITARY ACADEMY

MA388: SCORE PROJECT PRESENTATION

MA388: SABERMETRICS

SECTION D1

LTC MICHAEL POWELL

BY

CDT WILLIAM BOEHLKE '24 CO E1  
CDT KARLY PARCELL '24 CO F2

WEST POINT, NEW YORK

06 MAY 2024

WP I CERTIFY THAT I HAVE COMPLETELY DOCUMENTED ALL SOURCES  
THAT I USED TO COMPLETE THIS ASSIGNMENT AND THAT I ACKNOWLEDGED  
ALL ASSISTANCE RECEIVED IN THE COMPLETION OF THIS ASSIGNMENT

\_\_\_\_ I CERTIFY THAT I DID NOT USE ANY SOURCES OR RECEIVE ANY  
ASSISTANCE REQUIRING DOCUMENTATION WHILE COMPLETING THIS  
ASSIGNMENT

SIGNATURE: \_\_\_\_\_

Karly Parcell

## WORKS CITED

Assistance received from CDT John Beggs and CDT Skyler Chauff, the peer revisions done in class were used to edit and change our final product.

# Womens College Basketball Predicting Shoting Plays with logistic Regression

```
knitr::opts_chunk$set(echo = TRUE,  
  message = FALSE,  
  warning = FALSE)
```

Learning Goals: 1) Visualize the success and failures of shots made in college women basketball. 2) Understand when to use logistic regression. 3) Create a (glm) generalized linear model in R 4) Interpret results from a glm given in R 5) Compare 2 models using anova in R

```
# load in these library's before running the code below  
library(tidyverse)  
library(Lahman)  
library(knitr)  
library(dplyr)  
library(broom)  
# library(wehoop)
```

Introduction: Women college basketball is becoming more popular given the recent success of March Madness. With the 2024 game of Iowa vs UConn being the most watched basketball game on ESPN. In this module we will teach you how to perform logistic regression in r-studio, which is used as a way to estimate the probability of an event occurring. For this module, we will create a generalized linear model using womens college basketball data to estimate the probability of a basket being made during a womens college basketball game. We will determine which coefficients are most significant in determining the probability estimate. Additionally we will compare a model with the variable quarter, to a model without accounting for quarter in order to determine if quarter has a significant effect on determining the probability of a shot made.

Data: The data we are using is from the **wehoop** package. This can be installed by un-commenting a line in the block of code below. From this package you can then load in the college basketball data (wbb\_pbp) which we later name college. This data contains many variables scrapped from ESPN, so the information loaded into ESPN about the game, is extracted and used to make up the different variables used to create the data set for womens basketball. The data contains information on all games that ESPN has information for in the 2024 season. The College data frame below shows all variables we are using and gives a brief explanation above the column name for what the values mean. In total, we have 867657 observations in the college data set.

```
# uncomment the line below to install the wehoop dataset  
# install.packages("wehoop")
```

```
# how to load in college basketball data  
tictoc::tic()  
progressr::with_progress({  
  wbb_pbp <- wehoop::load_wbb_pbp()  
})  
tictoc::toc()
```

```
## 12.62 sec elapsed
```

```
wbb_pbp <- wehoop::load_wbb_pbp()
```

Below we create the college data set that will be used in our example. We select a multitude of variables from the wbb\_pbp data set. Each variable is described above their selection. Many of the variables left unselected for the college data contain information that is accounted for within the variables that were chosen to be included in the college data.

```
college <- wbb_pbp %>%  
  filter(shooting_play == TRUE) %>%  
  select(  
    # x coordinate of where shot was taken  
    coordinate_x,  
    # y coordinate of where shot was taken  
    coordinate_y,  
    # away teams score  
    away_score,  
    # home teams score  
    home_score,  
    # quarter of the game  
    qtr,  
    # how many points the shot was worth (3 pointer = 3, inside 3 point line = 2,  
    # free throw = 1) these are not just made shots,  
    # this is any shot taken -> missed or made  
    score_value,  
    # TRUE = shot was made, FALSE = shot was missed - turned 1 = made, 0 = missed  
    scoring_play,  
    # number of minutes left on the clock  
    clock_minutes,  
    # number of seconds left on the clock  
    clock_seconds)  
  
# view the first 6 columns and rows of our college data set  
kable(head(college[,1:6]))
```

coordinate_x	coordinate_y	away_score	home_score	qtr	score_value
40.75	3	0	0	1	2
-18.75	-10	3	0	1	3
35.75	6	3	0	1	2
-40.75	3	3	0	1	2
26.75	0	3	0	1	2
-42.75	-5	5	0	1	2

Let's visualize what our made and missed shots look like on the court.

```
# Showing location of all shots taken  
# Create a graph that depicts the missed and made shots from all observations  
# in the college data set by their x and y coordinates.  
  
#Filter to fit dimensions of court, 100 feet in length (x) and 50 feet wide (y)
```

```
working_wbb_coordinates <- college %>%
  drop_na(coordinate_x, coordinate_y) %>%
  filter(coordinate_y >= -25, coordinate_y <= 25,
         coordinate_x >= -50, coordinate_x <= 50)

# Plot of X and Y coordinates (locations) on the court where baskets are made. Alpha
# indicates the density of these shots, where darker happened at a higher frequency
# than lighter. This is to give a visual aid initially when examining the LOCATION
# aspect of where shots are.

ggplot(working_wbb_coordinates, aes(x = coordinate_x, y = coordinate_y, color =
                                   scoring_play)) +
  geom_point(alpha = 0.05) + # Set the transparency to 0.5 (adjust as needed)
  labs(x = "Processed X Coordinate", y = "Processed Y Coordinate") +
  ggtitle("Coordinates of All Shots Taken")
```



From the visual above, do you notice any variables from the college data set that you think should be standardized before creating a model? Why?

Your Answer Here:

If you said x and y coordinates, you're right!

We want to standardize these variables because currently, as you can see in the visual, as a player gets closer to the basket, their x position is increasing in one direction and decreasing for the other. Similar for the y position, as a player gets further from the basket, they could either be increasing or decreasing in their y

position. This will cause problems for our model, as we have increasing and decreasing values likely meaning the same thing and not being of a linear relationship. To fix this problem for our model, we will standardize the x and y coordinates for use in our logistic regression models by taking their absolute values.

```
college <- college %>%
  summarize(
    # absolute value of x coordinate of where shot was taken
    coordinate_x = abs(coordinate_x),
    # absolute value of y coordinate of where shot was taken
    coordinate_y = abs(coordinate_y),
    away_score,
    home_score,
    qtr,
    score_value,
    scoring_play,
    clock_minutes,
    clock_seconds)

kable(head(college[,1:6]))
```

coordinate_x	coordinate_y	away_score	home_score	qtr	score_value
40.75	3	0	0	1	2
18.75	10	3	0	1	3
35.75	6	3	0	1	2
40.75	3	3	0	1	2
26.75	0	3	0	1	2
42.75	5	5	0	1	2

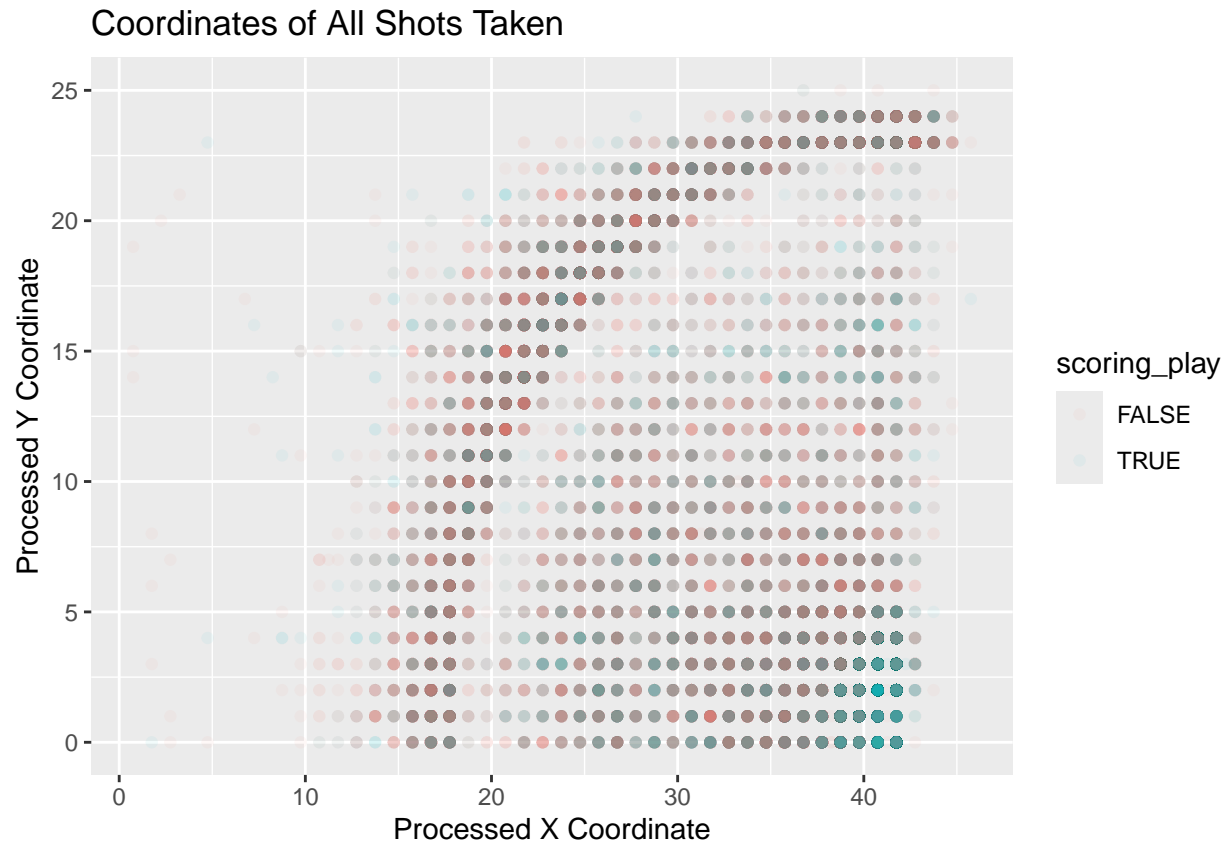
Now we recreate our visual of the court from above, but with the standardized coordinates. What do you expect this figure to look like in comparison to the one above?

Your Answer Here:

This is the new visualization for made and missed shots from our new college data set

```
working_wbb_coordinates <- college %>%
  drop_na(coordinate_x, coordinate_y) %>%
  filter(coordinate_y >= 0, coordinate_y <= 25,
         coordinate_x >= 0, coordinate_x <= 50)

ggplot(working_wbb_coordinates, aes(x = coordinate_x, y = coordinate_y, color =
  scoring_play)) +
  geom_point(alpha = 0.05) + # Set the transparency to 0.05 (adjust as needed)
  labs(x = "Processed X Coordinate", y = "Processed Y Coordinate") +
  ggtitle("Coordinates of All Shots Taken")
```



Now that we have a visual to represent our data that will be going through our model, we want to make one last change to the `scoring_play` variable, changing it from the values true or false to numeric, for interpretability, with 1 as shot made and 0 as shot missed.

```
college <- college %>%
  summarize(
    coordinate_x,
    coordinate_y,
    away_score,
    home_score,
    qtr,
    score_value,
    # changing scoring play to as.numeric for our model
    scoring_play = as.numeric(scoring_play),
    clock_minutes,
    clock_seconds)

kable(head(college[,1:6]))
```

coordinate_x	coordinate_y	away_score	home_score	qtr	score_value
40.75	3	0	0	1	2
18.75	10	3	0	1	3
35.75	6	3	0	1	2
40.75	3	3	0	1	2
26.75	0	3	0	1	2

coordinate_x	coordinate_y	away_score	home_score	qtr	score_value
42.75	5	5	0	1	2

### What is logistic regression?

Logistic regression allows us to see the effect of the variables in our model. In our case, in predicting whether the play is a scoring play. Logistic regression is a statistical analysis method that allows for modeling the probability of a binary outcome. An example model can be seen below:

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 * x_1 + \dots + \beta_n * x_n$$

To understand logistic regression, it is first important to understand the variables and coefficients involved.

$$\pi_i$$

is the probability that an event occurred, often coded as a 1 or 0.

$$\beta_0$$

is the intercept term of the model and represents the log odds of the event occurring when the predictor variable is 0. The predictor variable,

$$\beta_1$$

is the coefficient of the predictor variable that represents the change in the log odds of the event occurring for a one-unit change in the predictor variable. Similarly,

$$\beta_2$$

would represent the change in the log odds of the event occurring for a one-unit change in the second predictor variable. This is the same for any additional

$$\beta_n$$

terms.

The following logistic regression model takes the template model from above and fits it to what this project is attempting to analyze below: predicting whether the play is a scoring play.

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 \text{coordinate\_x}_i + \beta_2 \text{coordinate\_y}_i + \beta_3 \text{away\_score}_i + \beta_4 \text{home\_score}_i + \beta_5 \text{qtr}_i + \beta_6 \text{score\_value}_i + \beta_7 \text{scoring\_play}_i + \beta_8 \text{clock\_minutes}_i + \beta_9 \text{clock\_seconds}_i$$

$$\beta_0$$

in this model represents the log odds of a scoring play when all other variables are zero. This project uses two separate logistic regression models. One with the

$$\text{qtr}_i$$

variable and another without the

$$\text{qtr}_i$$

variable. Instead of creating four separate logistic regressions that attempted to see the effect of each quarter, creating two logistic regression with and without the variable

$$\text{qtr}_i$$



will enable the user to see if quarter has an effect, if any, on a scoring play. The model without

$$qtr_i$$

can be seen below:

$$\log\left(\frac{\pi_i}{1 - \pi_i}\right) = \beta_0 + \beta_1 \text{coordinate\_x}_i + \beta_2 \text{coordinate\_y}_i + \beta_3 \text{away\_score}_i + \beta_4 \text{home\_score}_i + \beta_5 \text{score\_value}_i + \beta_6 \text{scoring\_play}_i + \beta_7 \text{clock\_minutes}_i + \beta_8 \text{clock\_seconds}_i$$

This model examines the same effects of variables, but controlling for the effect that the quarter has on the scoring play.

### Logistic Regression in Use

Create a logistic regression model including all variables from the college data set above. Be sure to making scoring play your response variable for the model, as we are looking to see what variables have a significant effect on a scoring play.

### YOUR CODE HERE ###

```
making_shot_glm <- glm(scoring_play ~ ., data = college, family = "binomial")
```

Print a table for the summary of results for the model produced above

### YOUR CODE HERE ###

```
tidy1_glm <- tidy(making_shot_glm)
kable(tidy1_glm, digits = 4)
```

term	estimate	std.error	statistic	p.value
(Intercept)	1.4223	0.0541	26.3131	0.0000
coordinate_x	0.0000	0.0000	1.3691	0.1710
coordinate_y	-0.0120	0.0022	-5.5371	0.0000
away_score	0.0159	0.0012	13.1435	0.0000
home_score	0.0132	0.0011	11.6821	0.0000
qtr	-0.4545	0.0275	-16.5569	0.0000
score_value	-0.7603	0.0238	-31.9607	0.0000
clock_minutes	0.0590	0.0045	13.1217	0.0000
clock_seconds	0.0008	0.0006	1.2688	0.2045

Which variables from the table above are considered to be significant for estimating a scoring play based on their p-value? (Hint: a p-value of 0.05 or less is considered significant)

Your Answer Here:

coordinate\_y, away\_score, home\_score, qtr, score\_value, clock\_minutes

Make another GLM, include all of the predictors from above, except for qtr.

This will give us two models to later compare and interpret.

```
### YOUR CODE HERE ###
```

```
making_shot_noqtr_glm <- glm(scoring_play ~ coordinate_x + coordinate_y + away_score +  
                             home_score + score_value + clock_minutes +  
                             clock_seconds, data = college, family = "binomial")
```

Now make a table for the summary results from the glm made above with no qtr variable

```
### YOUR CODE HERE ###
```

```
tidy2_glm <- tidy(making_shot_noqtr_glm)  
kable(tidy2_glm, digits = 4)
```

term	estimate	std.error	statistic	p.value
(Intercept)	1.2342	0.0525	23.4884	0.0000
coordinate_x	0.0000	0.0000	1.1338	0.2569
coordinate_y	-0.0127	0.0022	-5.9014	0.0000
away_score	0.0038	0.0010	3.9815	0.0001
home_score	0.0014	0.0009	1.6635	0.0962
score_value	-0.7341	0.0236	-31.0970	0.0000
clock_minutes	0.0199	0.0038	5.2248	0.0000
clock_seconds	0.0001	0.0006	0.2386	0.8114

Answer the questions below for interpreting different betas from the model above:

Which variables have changed in significance from our model with qtr, to the one without?

home\_score is no longer significant

Interpret the intercept in the model above:

With all other variables at 0, the log odds of a womens college basketball player making a basket is 1.3826.

Interpret  $\beta_{\text{clock\_minutes}}$ :

For each additional minute on the clock, the log odds of making a basket increase by 0.0199

Interpret  $\beta_{\text{score\_value}}$ :

For every unit the score\_value increases, going from a 1, to 2, to 3 point shot, the log odds of making a basket decreases by 0.7341.

### ANOVA

You can use anova to compare two nested models, nested means that one model is contained within the other. For this problem we will look at comparing our model that contains quarter to the model without the quarter variable, keeping all other variables constant. The Chi value will tell us the significance of adding the variable qtr to our model. If the Pr(Chi) is less than or equal to 0.05, then the addition of the variable is considered as a significant predictor for our scoring play prediction abilities.

Now use anova to compare the two models you made above.

```
anova(making_shot_noqtr_glm, making_shot_glm, test = "Chisq")
```

```
## Analysis of Deviance Table  
##
```

```
## Model 1: scoring_play ~ coordinate_x + coordinate_y + away_score + home_score +
##      score_value + clock_minutes + clock_seconds
## Model 2: scoring_play ~ coordinate_x + coordinate_y + away_score + home_score +
##      qtr + score_value + clock_minutes + clock_seconds
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1      39144      51069
## 2      39143      50790  1    278.75 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the results above, is `qtr` a significant predictor for a shooting play?

It is a significant predictor, as the chi squared is much less than 0.05, meaning the addition of `qtr` to our model, significantly improves our models ability to predict a scoring play.

### Learning Goals Revisited

#### Conclusion

Now you should understand how to use logistic regression to determine the effect of the variable `qtr` on making a shot for a womens basketball game. You should understand how to interpret coefficients given by running logistic regression. Additionally, you now should know how to make a plot to visualize missed vs made shots for a basketball game from our college data set made from the `wehoop` library.

#### Additional Application

Another use of logistic regression is to adjust for categorical variables, for example, if you wanted to know how the position of a player effects making a shot. Those values are entered as guard, forward, center, you can factorize this variable to determine the effect of position on a basketball player making a shot during a game, with all other variables constant.

Outside of basketball, logistic regression can be used many other ways. For example, estimating the probability of a politician winning an election given different variables of those voting. Using those variables to estimate the probability of winning and election.

#### Additional Sources:

For more information please visit the sources below

The first source used to understand Binary Logistic Regression was from Penn State (<https://online.stat.psu.edu/stat504/>), this text book goes over how to set up Binary Logistic Regression and when to use Binary vs Multinomial vs Ordinal Logistic Regression. In this exercise we will only use Binary. (Multinomial is for more 3 or more response options, the same with Ordinal but for Ordinal, the responses are ordered (such as a survey where you respond with “very bad”, “bad”, “okay”, “good”, “very good”)).

Our second source is <https://quantifyinghealth.com/how-to-run-and-interpret-a-logistic-regression-model-in-r/> which goes over how to interpret logistic regression results produced in r-code, such as interpreting coefficients and determining if a coefficient is significant/how significant it is.