Optimizing Offensive Efficiency in the NBA

ENMGT 5930: Units1-5 Project

Fall2021

Cornell University

Team member:

Ruixin Zhang

Steven He

Chengyu Zhang

Introduction

In this project, we investigate how NBA teams can improve their offensive efficiency. We set out to determine what key factors play in generating an open shot and what shot are the best shots to take. To do this, we firstly used PCA tool to re-integrate several variables of dataset into a new vector. Then, we had a further investigation of regression for finding a better fitted model, as well as investigating correlations between the independent variables and the shot result.

Dataset Description & Explanation

This dataset gives us shot data from NBA games from the 2014-2015 NBA season. The data variable includes, the matchup, date, location (home or away), result of the game (win or loss), shot number, period, time on the game clock, time on the shot clock, number of dribbles, touch time, the shot distance, the shot type, the shot result, closest defender, the distance of the closest defender, whether the field goal was made or missed, the points scored from the shot, and the player who took the shot. Several variables should be paid more attention. Shot clock represents the time that a team may possess the ball before attempting to score a field goal. Game clock, which is different from shot clock, represents the time remaining in the period of play. Final margin is how many points a team won or lose at the end of game.

Data Processing

Attempt on PCA: We processed out data with PCA to try to determine the various factors that can affect the open shot in the NBA game and decide what shot are the best shots to take. So, we use PCA as a tool for integration and feature extraction.

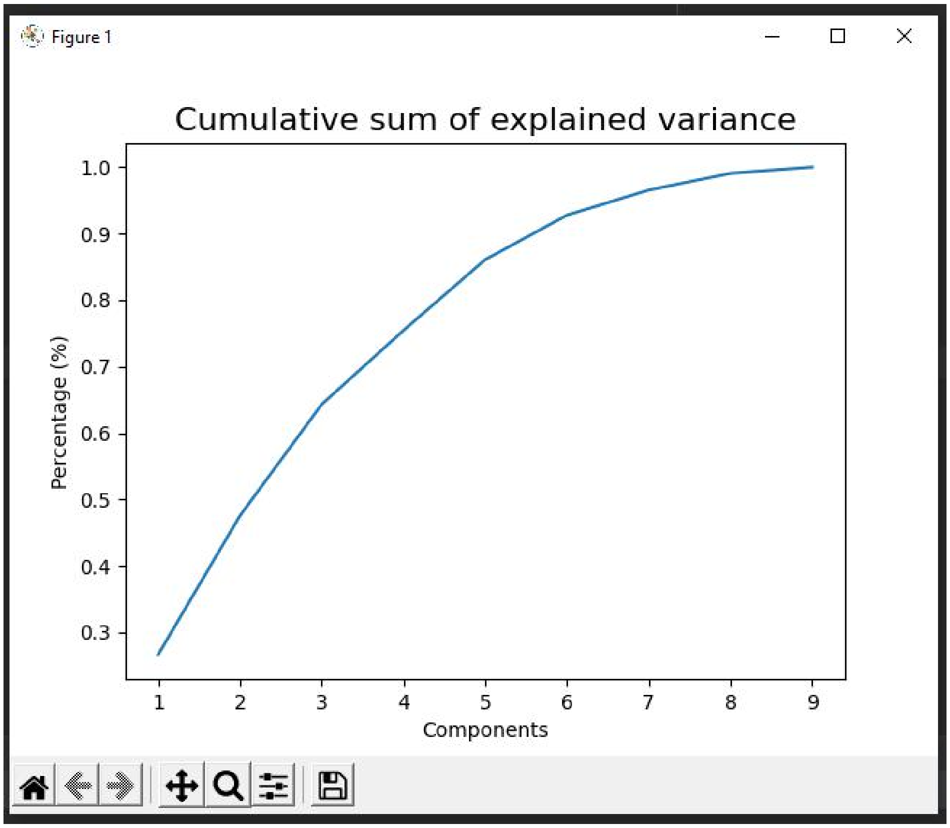
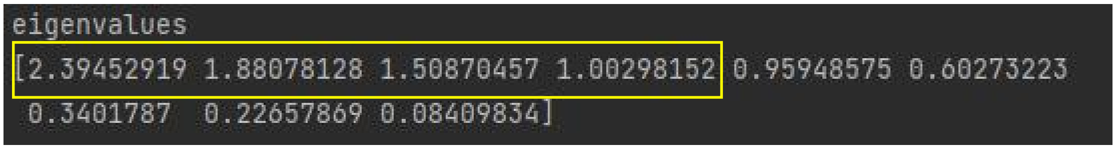


Figure 1 – Variance explained vs PCs

PCA (Principal Component Analysis) is an integration method that re-aggregates many variables into a new vector. When we are studying a sporting event, we cannot fully reflect all the judgment criteria in one picture (such as various technical movements in rhythmic gymnastics; jumps in diving, flips in the air, and the size of splashes), so we It is necessary to re-establish a new coordinate system to reflect our evaluation indicators, so that observers can intuitively feel the pros and cons of athletes.

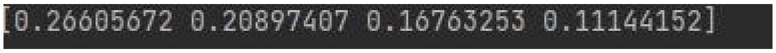
In our project this time, we are studying the impact of various NBA actions before shooting on the hit rate, but these influencing factors cannot be presented in one picture at the same time. We need to use PCA for integration. After testing (as shown in the figure above), we found that when the number of components reaches 6, about 90% of variance can be explained by the principal components. But at the same time, we were afraid if all 6 components are needed to represent our data, so we calculated the eigenvalue and used it as a parameter of factor significance.



```
eigenvalues
[2.39452919 1.88078128 1.50870457 1.00298152 0.95948575 0.60273223
 0.3401787  0.22657869 0.08409834]
```

Figure 2 – List of Eigenvalues

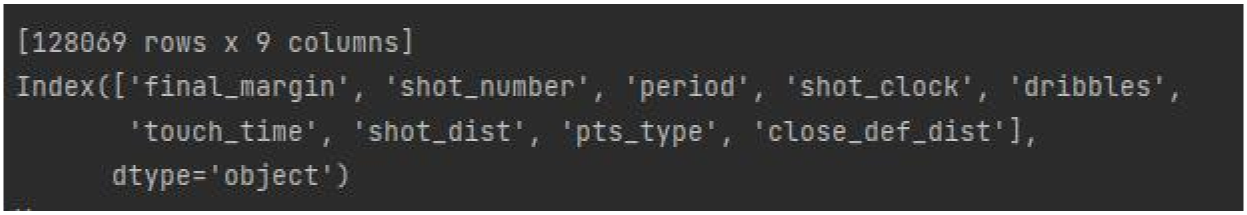
Then, from the eigenvalues, if we set 1.0 as our criterion, we find that 4 principal components would be significant to our dataset. But this also means that we cannot simply explain our model by drawing a two-dimensional Bi-plot as lectured, as it can only visualize two components and e when the number of principal components is only 2, we can only explain less than 50% of the variables, which is not statistically significant.



```
[0.26605672 0.20897407 0.16763253 0.11144152]
```

Figure 3 – Percentage of variance explained

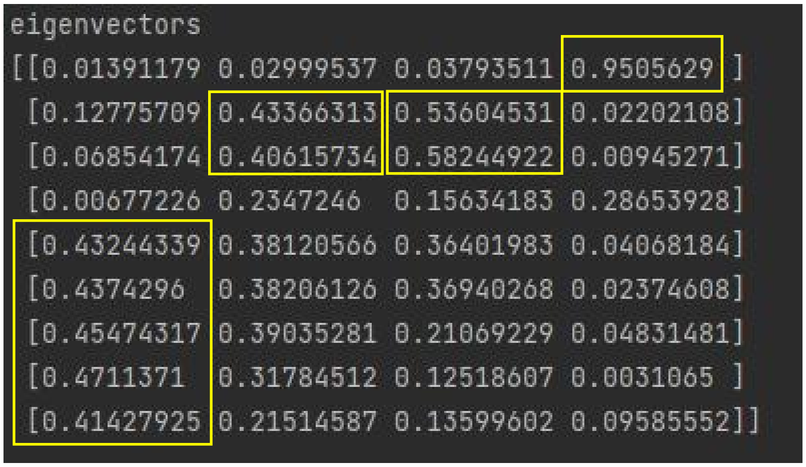
The above graph shows the result of our Cos^2, or the percentage of variance explained by each principal component. As shown, four PCs can explain around 75.4 percent of the variance of the data set.



```
[128069 rows x 9 columns]
Index(['final_margin', 'shot_number', 'period', 'shot_clock', 'dribbles',
       'touch_time', 'shot_dist', 'pts_type', 'close_def_dist'],
      dtype='object')
```

Figure 4 – List of variables

The screenshot above is the nine variances of our model. They are ‘final_margin’, ‘shot_number’, ‘period’, ‘shot_clock’, ‘dribbles’, ‘touch-time’, ‘shot-dist’, ‘pts-type’, and ‘close-def-dist’. Before the analysis, we have eliminated the variables that are not very relevant, and finally found that these nine variables have the most direct relationship with the shooting percentage of NBA players.



```
eigenvectors
[[0.01391179 0.02999537 0.03793511 0.9505629 ]
 [0.12775709 0.43366313 0.53604531 0.02202108]
 [0.06854174 0.40615734 0.58244922 0.00945271]
 [0.00677226 0.2347246  0.15634183 0.28653928]
 [0.43244339 0.38120566 0.36401983 0.04068184]
 [0.4374296  0.38206126 0.36940268 0.02374608]
 [0.45474317 0.39035281 0.21069229 0.04831481]
 [0.4711371  0.31784512 0.12518607 0.0031065 ]
 [0.41427925 0.21514587 0.13599602 0.09585552]]
```

Figure 4 – Table of eigenvectors corresponds to PCs and variables

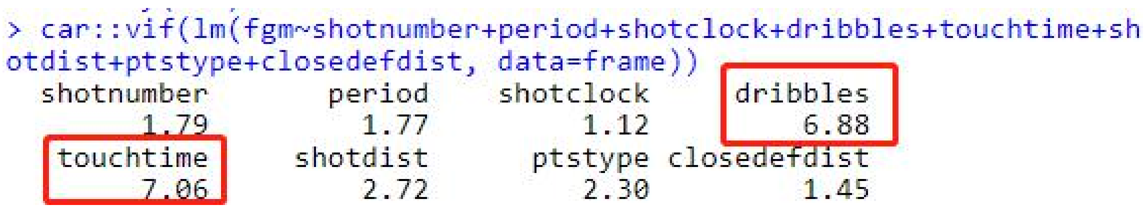
The eigenvalue result figure above is the explanation level of the 9 variables for the four principal component we got.

we set 0.4 as our criterion, we find that the four principal components represent most of the variability in our data. PC1 highly represents ‘dribbles’, ‘touch-time’, ‘shot-dist’, ‘pts-type’, and ‘close-def-dist’. PC2 and PC3 mainly explained by ‘shot number’ and ‘period’. PC4 has the strong positive correlation with the ‘final margin’, which means if the ‘final margin’ is large in our data analysis, the reaction of PC4 in the vector diagram will be very obvious (very large). We can be bold to draw our conclusion that PC1 is mostly a measure of points type and shot distance, PC2 can be represented by a measure of shot number, PC3 by period, and that PC4 is just the final margin of the game.

Multiple Regression Analysis

Multicollinearity Check and Variable Selection

There are eight selected variables, which are the number of shots, time period, shot clock, dribbles, touch time, shot distance, type of points, and the closest defender’s distance. Two variables (dribbles, and touch time) are found that the value is larger than 5 after processing the variance inflation factor analysis as shown in figure 5, which means multicollinearity is a particular concern in the model. Based on the result, dribbles and touch time has not significant connection with offensive efficiency in the NBA game. Afterwards, the rest of 6 variables are selected.

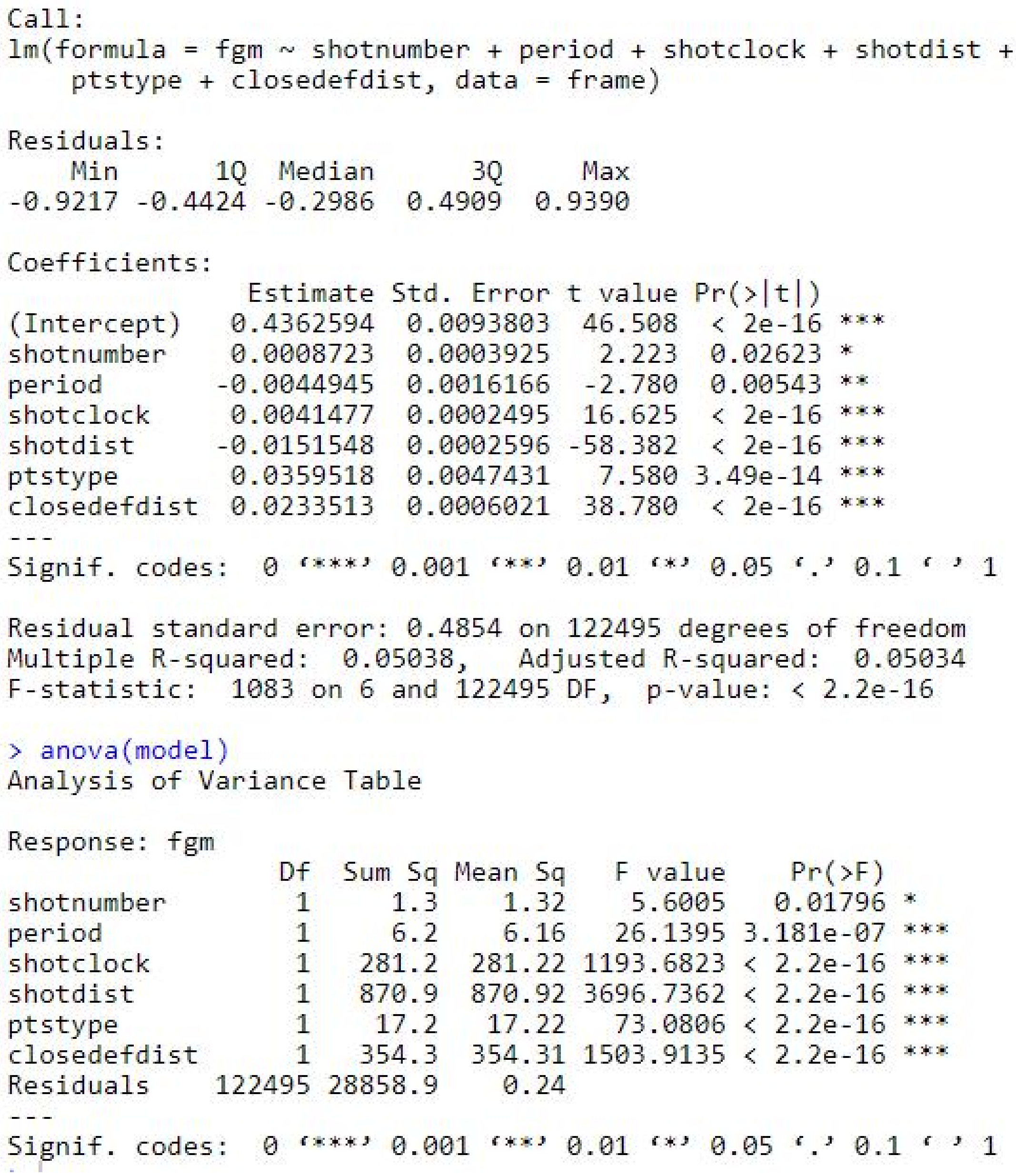


```
> car::vif(lm(fgm~shotnumber+period+shotclock+dribbles+touchtime+sh
otdist+ptstype+closedefdist, data=frame))
  shotnumber       period    shotclock     dribbles
        1.79         1.77         1.12         6.88
   touchtime     shotdist      ptstype closedefdist
        7.06         2.72         2.30         1.45
```

Figure 5 – Variance Inflation Factor Analysis for 8 Variables

Multiple Regression with 6 variables

The multiple regression was conducted with the selected 6 variables (the number of shots, time period, shot clock, shot distance, type of points, and close defense distance) after the multicollinearity check. As shown in figure 6, all the variables are statistically significant, and the nova part’s p values for the 6 variables are all statistically significant as well. Based on the regression result, we can interpret that the offensive efficiency is negatively correlated with the shooting distance and game period, and positively correlated with shotting number, shot clock, points type, and the closest defender’s distance. Shooting distance is key factor for affecting the offensive efficiency. A longer shooting distance will decrease the scoring percentage. Since a longer shooting distance requires the player to control the angle, direction, strength of shots more precisely to score compared to a shorter distance. Game period is another factor that have a negative relationship with offensive efficiency. Game period 1 and 2 normally has the highest hit rate, and game period 4 has the low hit rate as player’s ability to control the ball decrease. It is known that a player's physical strength will decrease with time and energy expenditure. And thus, the hit rate will decrease when a player has less physical strength to control the ball and take the shot. With regards to the closet defender’s distance, any shot in which a player is six or more feet away from the defender is an optimal shot to take because the player is open and no defender can disrupt the shooter to shot. In the case of shotting number, the more shots a player shots , the better his chances of scoring. As for point type, shots between 22 and 26 feet generate a higherthan-average expected value per shot attempt. Three-point shots have become more and more prevalent in today's NBA and for good reason. A three-pointer is more effective than a two-pointer. It correlates with our regression analysis result in figure 6. Points type(from two-pointer to three-pointer) has a positive relationship with offensive efficiency in the regression analysis.



```
Call:
lm(formula = fgm ~ shotnumber + period + shotclock + shotdist +
    ptstype + closedefdist, data = frame)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.9217 -0.4424 -0.2986  0.4909  0.9390 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)   0.4362594  0.0093803  46.508  < 2e-16 ***
shotnumber    0.0008723  0.0003925   2.223  0.02623 *  
period       -0.0044945  0.0016166  -2.780  0.00543 ** 
shotclock     0.0041477  0.0002495  16.625  < 2e-16 ***
shotdist     -0.0151548  0.0002596 -58.382  < 2e-16 ***
ptstype       0.0359518  0.0047431   7.580 3.49e-14 ***
closedefdist  0.0233513  0.0006021  38.780  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4854 on 122495 degrees of freedom
Multiple R-squared:  0.05038,   Adjusted R-squared:  0.05034 
F-statistic:  1083 on 6 and 122495 DF,  p-value: < 2.2e-16

> anova(model)
Analysis of Variance Table

Response: fgm
                 Df  Sum Sq Mean Sq   F value    Pr(>F)    
shotnumber        1     1.3    1.32    5.6005   0.01796 *  
period            1     6.2    6.16   26.1395 3.181e-07 ***
shotclock         1   281.2  281.22 1193.6823 < 2.2e-16 ***
shotdist          1   870.9  870.92 3696.7362 < 2.2e-16 ***
ptstype           1    17.2   17.22   73.0806 < 2.2e-16 ***
closedefdist      1   354.3  354.31 1503.9135 < 2.2e-16 ***
Residuals    122495 28858.9    0.24                        
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 6 – Initial Multiple Regression

K-fold Cross Validation

An important part of any statistical analysis is some type of validation. In order to verify the model’s predictability is on the right track, k-fold cross validation tool is used in this project. A 6-fold cross validation is processed for the final six variable model. As shown in the figure 7, 6 folds' regression line is clear and straight. The sum of squares is 4802 and the mean square is 0.24, suggesting that the model is in a good, predicted condition.

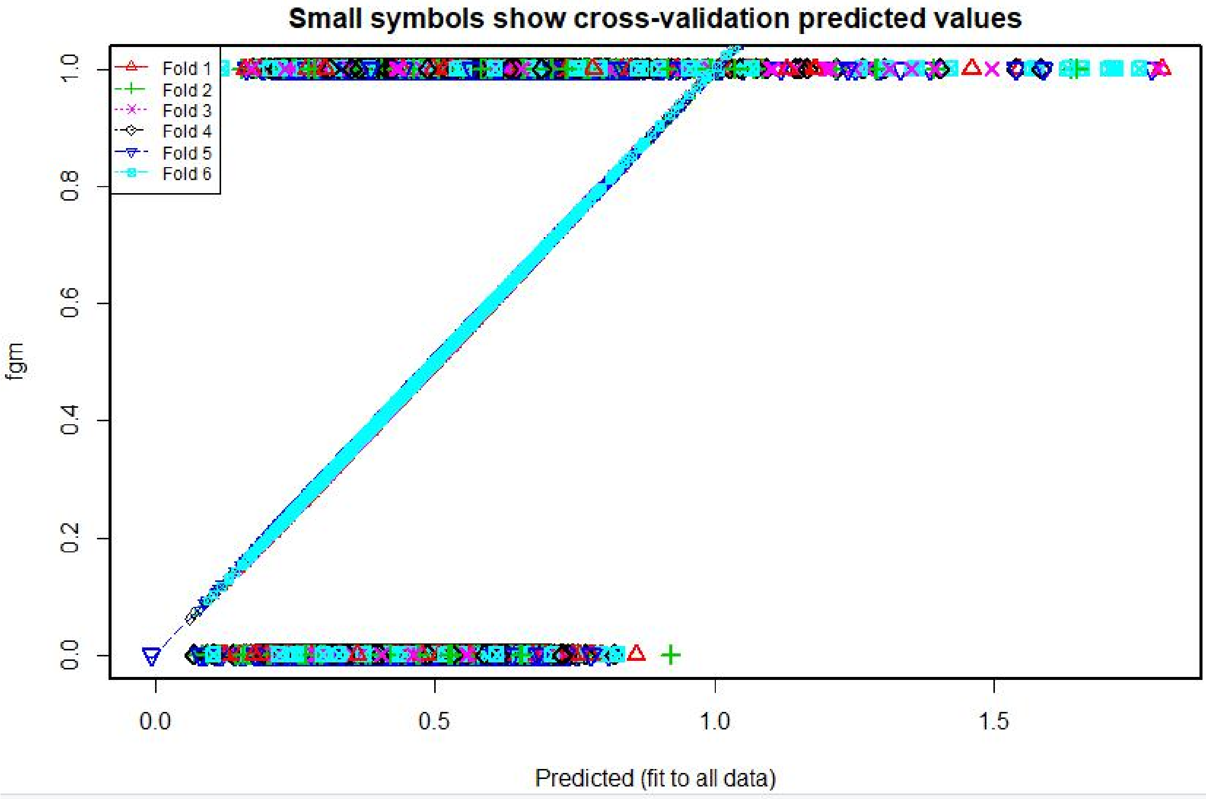


Figure 7 – K-fold Cross Validation

Check for Heteroskedasticity

After performing the K-fold cross validation, the heteroskedasticity check process is conducted by using the residuals plot (figure 8), the Q-Q plot (figure 9), and the Breusch Pagan test (figure 6). For the residuals, as the fitted values increases, the variance of the residuals should also increase. However, based on our residuals plot, the variance of the residuals decreases with the fitted values increase, probably suggesting that it's not the telltale pattern for heteroscedasticity. For the Q-Q plot, a linear trend is shown from the lower range to the upper range in figure 5, which is a typical “bimodal” appearance of QQ- plots. For the Breusch Pagan test, the output p-value is less then 2e-16. Since the p- value is below an appropriate threshold (e.g. p < 0.05) , the null hypothesis of homoskedasticity is rejected and heteroskedasticity assumed.

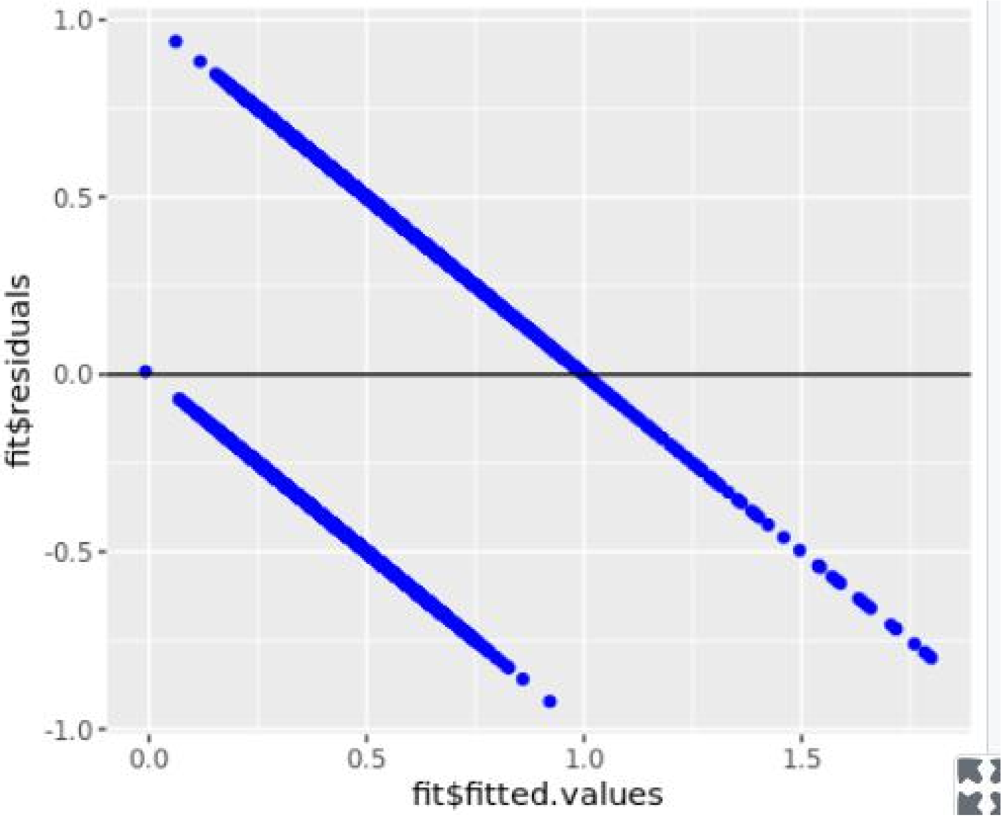


Figure 8 – Residuals Plot

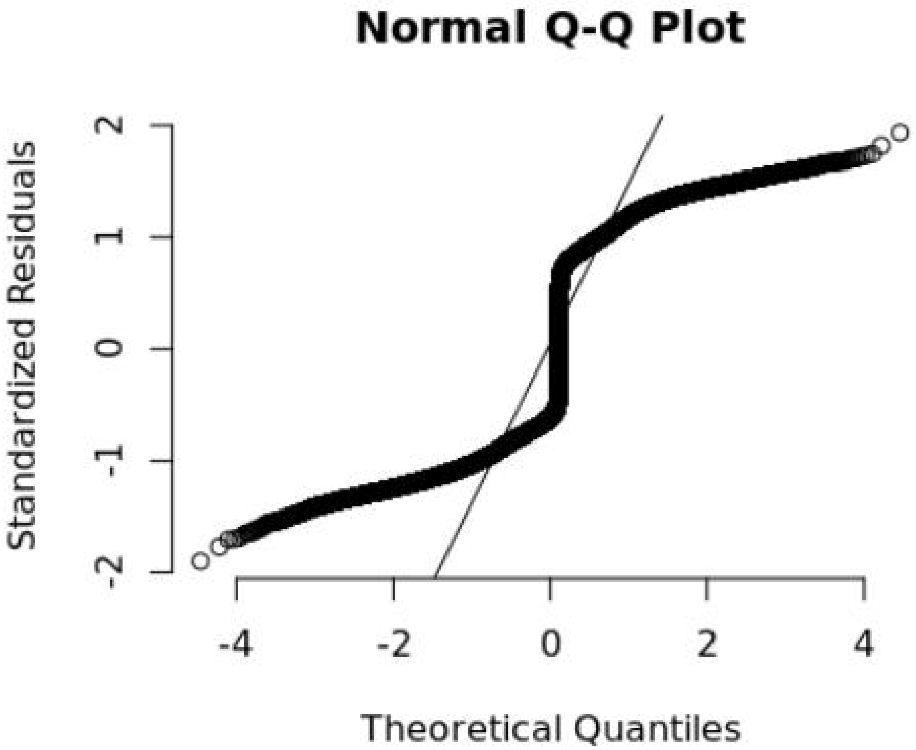
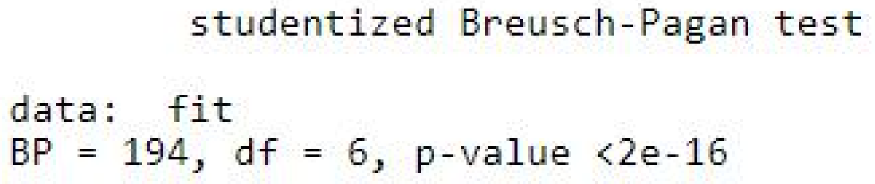


Figure 5 – Q-Q Plot

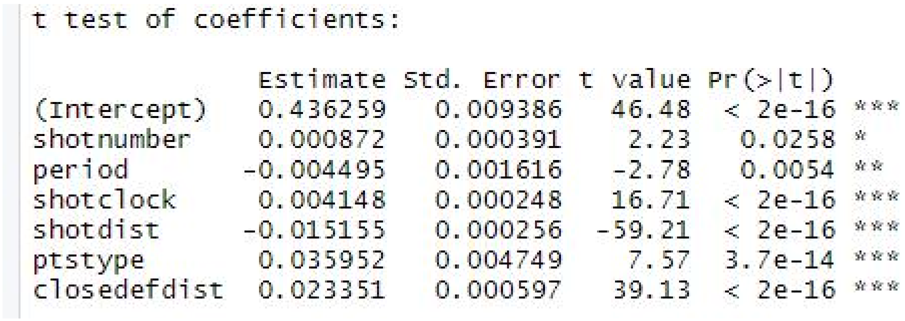


```
          studentized Breusch-Pagan test

data:  fit
BP = 194, df = 6, p-value <2e-16
```

Figure 9 – Breusch Pagan Test Output

Additionally, we need to verify if heteroskedasticity affect our model. Given the data we conducted with a large fraction of extreme outliers, a robust estimator is needed to guarantee the returned value is still within the non-outlier part of the data. The output for the robust estimator test is shown in Figure 10. As shown on the result, all the variables are statistically significant.

And thus, the heteroskedasticity does not influence the model a lot. Therefore, the 6-variable regression model is still good to use.

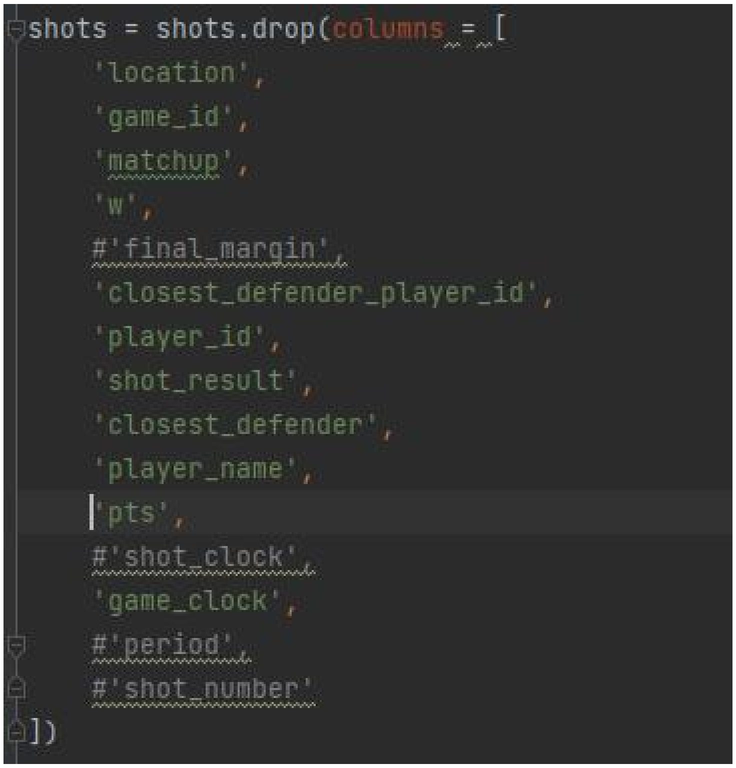


```
t test of coefficients:

             Estimate Std. Error t value Pr(>|t|)    
(Intercept)   0.436259   0.009386   46.48  < 2e-16 ***
shotnumber    0.000872   0.000391    2.23   0.0258 *  
period       -0.004495   0.001616   -2.78   0.0054 ** 
shotclock     0.004148   0.000248   16.71  < 2e-16 ***
shotdist     -0.015155   0.000256  -59.21  < 2e-16 ***
ptstype       0.035952   0.004749    7.57  3.7e-14 ***
closedefdist  0.023351   0.000597   39.13  < 2e-16 ***
```

Figure 10 – Robust Estimator Test

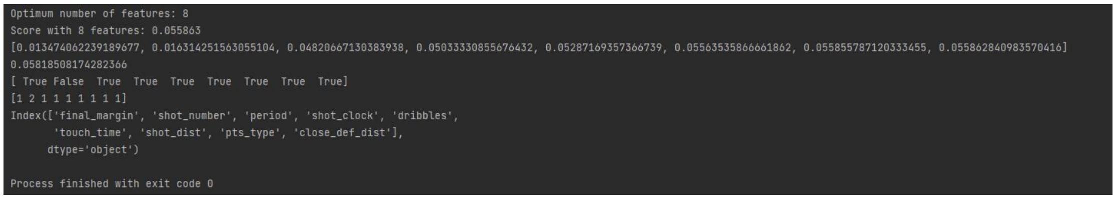
Machine learning model training and result.

As our goal incorporates a part of predicting the result of a shot, I started to examine the selected 8 features for the model, they are: 'shot_number', 'shot_clock', 'dribbles','touch_time', 'shot_dist', 'pts_type', 'close_def_dist', and 'pts', and thinking about if there is a feature that can be trained to a machine learning model that predicts shot-result, which is denoted by 0, or 1, 1 for hit and 0 for miss. After examination, I determined pts, which is points got after a shot is basically exactly the same as hit rate, if the hit misses the point is 0, and if the hit counts points can either be 3 or 2, the correlation can be 1 for the two variables, so I decided to drop the ‘pts’ variable from the set and retrain the model.



```
shots = shots.drop(columns = [
    'location',
    'game_id',
    'matchup',
    'w',
    #'final_margin',
    'closest_defender_player_id',
    'player_id',
    'shot_result',
    'closest_defender',
    'player_name',
    'pts',
    #'shot_clock',
    'game_clock',
    #'period',
    #'shot_number'
])
```

Figure 11 – Variables used in the model



```
Optimum number of features: 8
Score with 8 features: 0.055863
[0.013474062239189677, 0.016314251563055104, 0.04820667130383938, 0.05033330855676432, 0.05287169357366739, 0.05563535866661862, 0.055855787120333455, 0.055862840983570416]
0.05818508174282366
[ True False  True  True  True  True  True  True  True]
[1 2 1 1 1 1 1 1 1]
Index(['final_margin', 'shot_number', 'period', 'shot_clock', 'dribbles',
       'touch_time', 'shot_dist', 'pts_type', 'close_def_dist'],
      dtype='object')

Process finished with exit code 0
```

Figure 12 – Resulting predicting rate of the second model

As shown above, the new model still consists of 8 features, with only shot number being eliminated from the list of 9 kept, but the resulting prediction score is just too low. 5.59% is nowhere near the needed result of a good prediction model. This shows that there are limited, close to no correlation between the variables and shot result. The data set might be too complex with close to 200000 data points and the amount of hard to quantify features might be making it hard to get a good model from feature selection and regression model.

Summary

Based on the multiple regression analysis, the relationship between offensive efficiency and 6 variables are conducted clearly. Offensive efficiency is positively correlated with shotting number, shot clock, points type, and the closest defender's distance and is negatively correlated with shooting distance and game period. Offensive efficiency does not have significant correlation with dribbles and touch time in the NBA game.

Conclusion:

The purpose of this study was to find a way of demonstrating the correlations between variables in a NBA match before taking a shot, and the corresponding result of the shot. The PCA analysis functioned as a feature extraction tool to tell us the reduced number of needed variables to explain the variance of the data set, helped us to understand if there are only four principal components, what variables are being the most contribution and correspondingly what are the most important factors to consider before taking a shot. The regression model we created based on the important variables we considered gave us a better view of what are the variables that are being statistically significant when we are studying the correlations between dataset, and the internal correlations of these statistically significant variables and the shot result we want to predict.

The result we derived from our study mostly demonstrated that when explaining the dataset, only four principal components would be considered significant, and from the regression model, shooting efficiency is positively correlated with shotting number, shot clock, points type, and the closest defender's distance and is negatively correlated with shooting distance and game period.

Offensive efficiency does not have significant correlation with dribbles and touch time in the NBA game. When a player plays on the field, now he can understand what to pay more attention to if he wants to score for his team.

Limitations and Future Directions:

As data analysis at the current stage only focuses on relationships with quantified variables, other hidden affecting factors such as location of the match and more are not investigated in the research. And, just as presented in our previous report, the overall result of the research, especially the machine learning model trained to predict a shot result, is rather disappointing. That might not be the true picture of the dataset though, it could be a representation of our lacked understanding of machine learning and model training, which provided insights for future study and dedication.

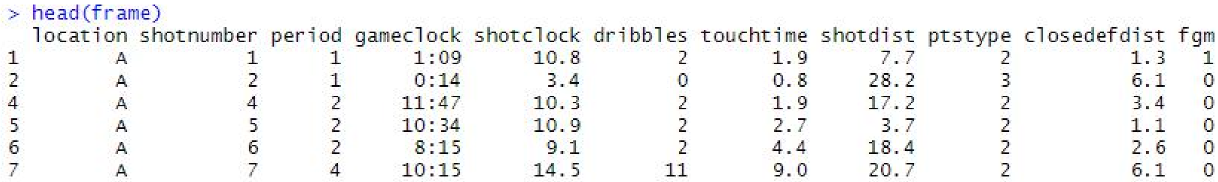
Lessons Learned:

1. A good dataset is not easy to locate, our team spent a considerable amount of time finding a dataset of interest and can at the same time worth studying.
2. Dataset selected might not be suitable for conducting data analysis, in our case, the dataset is very messy and contains potentially too much data to train for the sake of a half-term project.
3. During the analysis, data processing and cleaning are very important, null data could result in a total failure of the analysis, just like in our data, such points should be located and removed at the beginning.
4. During analysis, the reasonability of results should be examined frequently, for instance, does the result drawn align with the common sense.

Logistic regression

Now as the dependent variable we are studying is binary, we will use discrete choice model to investigate the relationships between our independent variables and the dependent one, we are using logistic regression to power our study.

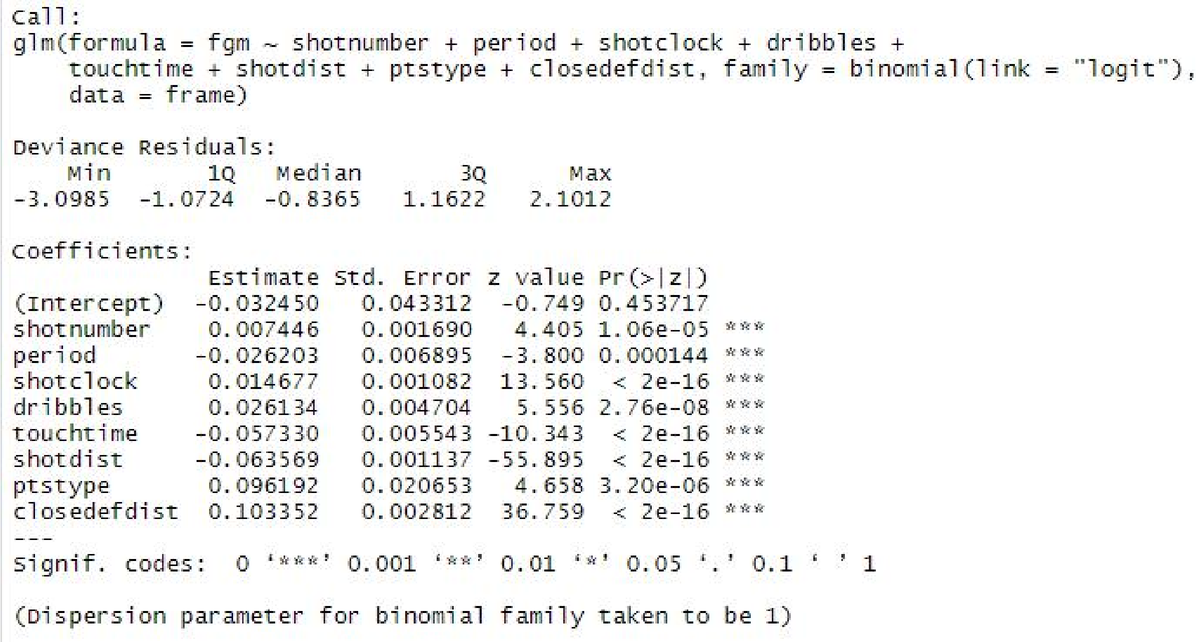
First, we introduce the dataset to R as a data frame, popping all the null values in the dataset at first to eliminate the risk of having errors. The data frame's content and general look is represented in the figure below.



```
> head(frame)
  location shotnumber period gameclock shotclock dribbles touchtime shotdist ptstype closedefdist fgm
1        A          1      1      1:09      10.8        2       1.9      7.7       2          1.3   1
2        A          2      1      0:14       3.4        0       0.8     28.2       3          6.1   0
4        A          4      2     11:47      10.3        2       1.9     17.2       2          3.4   0
5        A          5      2     10:34      10.9        2       2.7      3.7       2          1.1   0
6        A          6      2      8:15       9.1        2       4.4     18.4       2          2.6   0
7        A          7      4     10:15      14.5       11       9.0     20.7       2          6.1   0
```

Figure 13 – View of the data frame

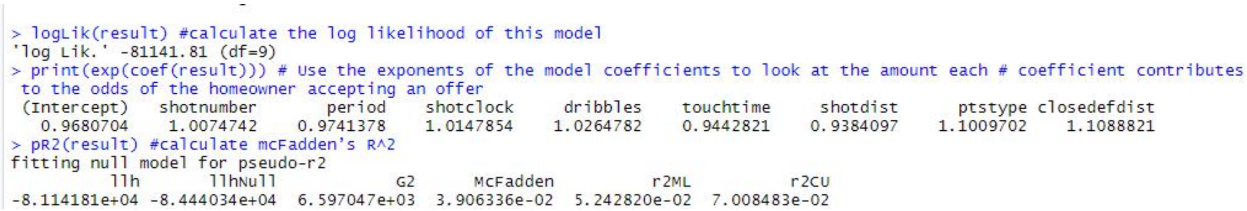
Now we perform logistic regression in R to eliminate potential insignificant terms, and to get the coefficients of variables that could be used to investigate degree affected of the result by the variable. Based on the result shown in figure 14, all the variables we included in the model are statistically significant, and their corresponding coefficients can be used to study the odds ratio associating the variable with the shot result. For instance, shot distance have an odd ratio of e^ (-0.064) = 0.938 which means that the group launching the shot afar would have a 6.2% lesser chance of landing a shot than that of the group shooting close.



```
Call:
glm(formula = fgm ~ shotnumber + period + shotclock + dribbles + 
    touchtime + shotdist + ptstype + closedefdist, family = binomial(link = "logit"), 
    data = frame)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-3.0985  -1.0724  -0.8365   1.1622   2.1012  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  -0.032450   0.043312  -0.749 0.453717    
shotnumber    0.007446   0.001690   4.405 1.06e-05 ***
period       -0.026203   0.006895  -3.800 0.000144 ***
shotclock     0.014677   0.001082  13.560  < 2e-16 ***
dribbles      0.026134   0.004704   5.556 2.76e-08 ***
touchtime    -0.057330   0.005543 -10.343  < 2e-16 ***
shotdist     -0.063569   0.001137 -55.895  < 2e-16 ***
ptstype       0.096192   0.020653   4.658 3.20e-06 ***
closedefdist  0.103352   0.002812  36.759  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

Figure 14 – Result of the logistic regression

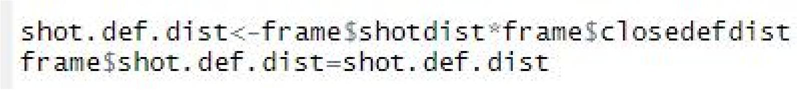
The log likelihood of the model is calculated and presented in figure 15 below, as we can see the result is -8114.81, as a measure of fit, that seems to be not ideal. And the changes in log likelihood with one unit change in variables is also presented below, from this we can see that the more important variables for our model are close defender distance and points type, which makes common sense as well. The McFadden pseudo-r squared is about 0.03906, which indicates that the covariance of the independent variable is important as there is a big difference between them and the result using only the intercept.



```
> logLik(result) #calculate the log likelihood of this model
'log Lik.' -81141.81 (df=9)
> print(exp(coef(result))) # Use the exponents of the model coefficients to look at the amount each # coefficient contributes
 to the odds of the homeowner accepting an offer
 (Intercept)   shotnumber       period    shotclock     dribbles    touchtime     shotdist      ptstype closedefdist 
   0.9680704    1.0074742    0.9741378    1.0147854    1.0264782    0.9442821    0.9384097    1.1009702    1.1088821 
> pR2(result) #calculate mcFadden's R^2
fitting null model for pseudo-r2
          llh       llhNull            G2      McFadden          r2ML          r2CU 
-8.114181e+04 -8.444034e+04  6.597047e+03  3.906336e-02  5.242820e-02  7.008483e-02 
```

Figure 15 – Result of the logistic regression 2

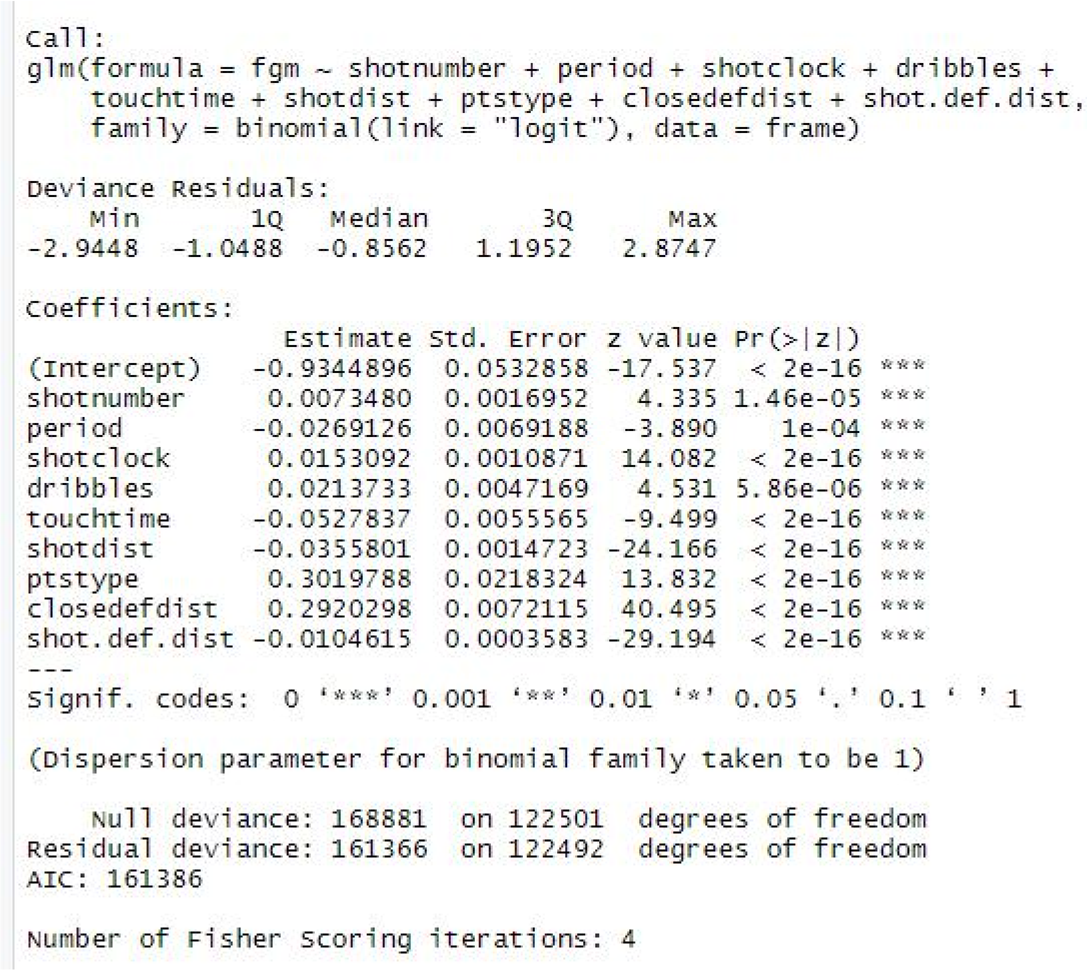
Now we decided to add an interactive variable shot & close defender distance to run our model again and see the effect of this variable, whether it is significant.



```
shot.def.dist<-frame$shotdist*frame$closedefdist
frame$shot.def.dist=shot.def.dist
```

Figure 16 – New variable introduced

We ran the logistic regression again and found that the new variable is significant, with an overall small negative impact on the shot result.

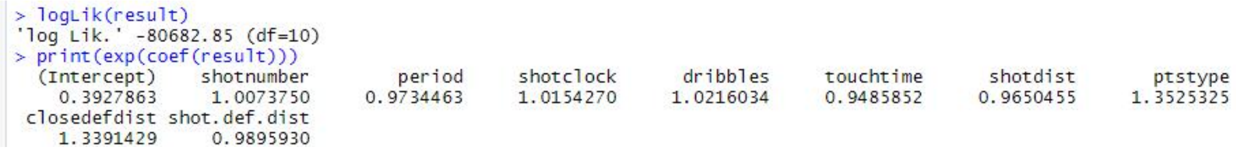


```
Call:
glm(formula = fgm ~ shotnumber + period + shotclock + dribbles + 
    touchtime + shotdist + ptstype + closedefdist + shot.def.dist, 
    family = binomial(link = "logit"), data = frame)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-2.9448  -1.0488  -0.8562   1.1952   2.8747  

Coefficients:
                Estimate Std. Error z value Pr(>|z|)    
(Intercept)   -0.9344896  0.0532858 -17.537  < 2e-16 ***
shotnumber     0.0073480  0.0016952   4.335 1.46e-05 ***
period        -0.0269126  0.0069188  -3.890    1e-04 ***
shotclock      0.0153092  0.0010871  14.082  < 2e-16 ***
dribbles       0.0213733  0.0047169   4.531 5.86e-06 ***
touchtime     -0.0527837  0.0055565  -9.499  < 2e-16 ***
shotdist      -0.0355801  0.0014723 -24.166  < 2e-16 ***
ptstype        0.3019788  0.0218324  13.832  < 2e-16 ***
closedefdist   0.2920298  0.0072115  40.495  < 2e-16 ***
shot.def.dist -0.0104615  0.0003583 -29.194  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 168881  on 122501  degrees of freedom
Residual deviance: 161366  on 122492  degrees of freedom
AIC: 161386

Number of Fisher Scoring iterations: 4
```

Figure 17 – Result of the logistic regression with new variable

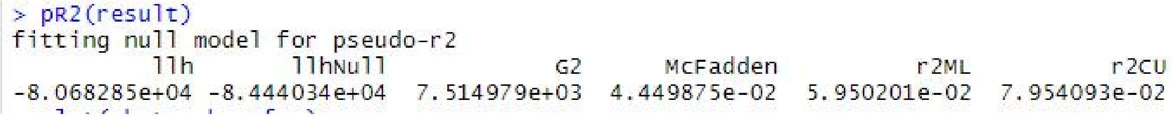
Now to see the resulting loglikelihood value, the change of loglikelihood value and McFadden pseudo-r squared, we present figure 18 and 19. The overall log likelihood is decreased to

80682.85, and the more important variable for the model is not changed, with the new variable not making a great impact. Based on the McFadden pseudo-r squared, the new model's covariance of the independent variable is slightly less important, changing from 0.0306 to 0.0445.



```
> logLik(result)
'log Lik.' -80682.85 (df=10)
> print(exp(coef(result)))
  (Intercept)    shotnumber        period     shotclock      dribbles     touchtime      shotdist       ptstype
    0.3927863     1.0073750     0.9734463     1.0154270     1.0216034     0.9485852     0.9650455     1.3525325
 closedefdist shot.def.dist
    1.3391429     0.9895930
```

Figure 18 – Result of the logistic regression with new variable 2



```
> pR2(result)
fitting null model for pseudo-r2
          llh       llhNull            G2      McFadden          r2ML          r2CU
-8.068285e+04 -8.444034e+04  7.514979e+03  4.449875e-02  5.950201e-02  7.954093e-02
```

Figure 19 – McFadden pseudo-r squared with new variable