Northeastern University
College of Professional Studies

Northeastern University

Instructor: Dr. Vladimir Shapiro

Date:20th February, 2022

Introduction

Basketball is a beloved sport across the globe and can be played by anyone. These games are played not only for entertainment but also for the fans who are passionate about this game. In such a case, it has become imperative for fans and the team's management to watch out for their favorite player's or team's statistics in terms of their previous game or against their opponent. Women's basketball is no different and is only gaining more popularity each day. For our project, I intend to explore and identify the different patterns for the winning team statistics and understand the mindsets of the professional Women NBA players.

The WNBA 2014 data set used in this analysis was chosen from Sports Statistics. From this analysis, I intend to answer the following business queries:

- What is the effect of a player's overall match statistics on a player's future efficiency?
- What is the effect of a team's overall match statistics on a team winning the game?

To conduct the analyses for the above research queries, predictive models such as the linear regression model and the logistic regression model will be used.

Regression tests are conducted to find the cause-and-effect relationships. These tests are performed to get an estimation on the effect of one or more continuous variables with another variable.

A Linear Regression Model predicts the value of a variable based on another variable's value. It uses a straight line to display the relationship between the variables. The variable that is predicted is known as the dependent variable and the variable that is being used to predict the dependent variable is known as the independent variable. There are two types of linear regression. They are the Simple Linear Regression and Multiple Linear Regression. In our analysis, we will be working with the Multiple Linear Regression Model.

A Logistic Regression Model is used to understand the relationship between the dependent variable and one or more than one independent variables and helps us in predicting the likelihood of an event. In this model, the dependent variable is a categorical or a finite variable.

The WNBA dataset variables that were considered for this analysis were:

bucket	Variable_Name	Description	Data_Type
1	Player	Name of the Player	Factor
2	player_id	Player ID	Integer
3	team	Team Name	Factor

bucket	Variable_Name	Description	Data_Type
4	date	Date of the Game	Integer
5	home	Home or Away	Integer
6	opponent	Opponent Team Name	Factor
7	tin	Win or Loss	Integer
8	team_pts	Total Team Points	Integer
9	opp_pts	Total Opponent Points	Integer
10	minutes	Number of minutes played by a player	Integer
11	fgmade	Field Goals	Integer
12	fgatt	Field Goal Attempts	Integer
13	made3	Number of 3 pointers made	Integer
14	att3	Number of 3 pointer attempts	Integer
15	made1	Number of 1 pointers made	Integer
16	att1	Number of 1 pointer attempts	Integer
17	offrb	Offensive Rebound	Integer
18	defrb	Defensive Rebound	Integer
19	totrb	Total Rebounds	Integer
20	assist	Number of assists	Integer
21	steal	Number of steals	Integer
22	block	Number of blocks	Integer
23	turnover	Number of turnovers	Integer
24	fouls	Number of fouls	Integer
25	points	Total Player Points	Integer
26	efficiency	Player Efficiency	Integer

Methods

BQ1 -> What is the effect of a player's overall match statistics on a player's future efficiency?

To answer the above question, the multiple linear regression was used. This method was chosen to estimate or predict the player's efficiency with respect to her previous game statistics.

In this model, the efficiency of the player is the dependent variable and the variables assist, fouls, turnover, made1 and fgmade are the independent variables. To make this model, the player's statistics was subsetted and the necessary assumptions were checked i.e., the check for multicollinearity, linearity, normal distribution of the dependent variable and the homoscedasticity.

BQ2 -> What is the effect of a team's overall match statistics on a team winning the game?

To answer the above question, the Logistic Regression Model was chosen. Since we wanted to determine the probability that a team would win the game and there are only two possible results i.e., win or lose, this model was appropriate to answer our question. Since the logistic regression model returns a probability between 0 and 1, we will need to decide on the limit that will be considered as a win or loss. If the probability from our model is greater than 0.5, then it'll be considered as a win and vice versa.

In this model, the efficiency of the player is the dependent variable and the variables team_pts, home, efficiency, fgmade and assist are the independent variables.

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
library(psych)
library(DataExplorer)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.9
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggplot2::%+%()   masks psych::%+%()
## x ggplot2::alpha() masks psych::alpha()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(ggpubr)
library(dplyr)
library(car)
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

## The following object is masked from 'package:purrr':
##
##     some

## The following object is masked from 'package:psych':
##
##     logit
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(AICcmodavg)
library(leaps)
library(magrittr)
```

```
##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
#Importing the dataset
WNBA <- read.csv("/Users/jillienchu/Desktop/Courses/Winter 2022/ALY6015/Project/WNBA.csv")
```

```
#Exploring the data set
plot_intro(WNBA, title = "Basic Information of the Data Set", ggtheme = theme_gray())
```

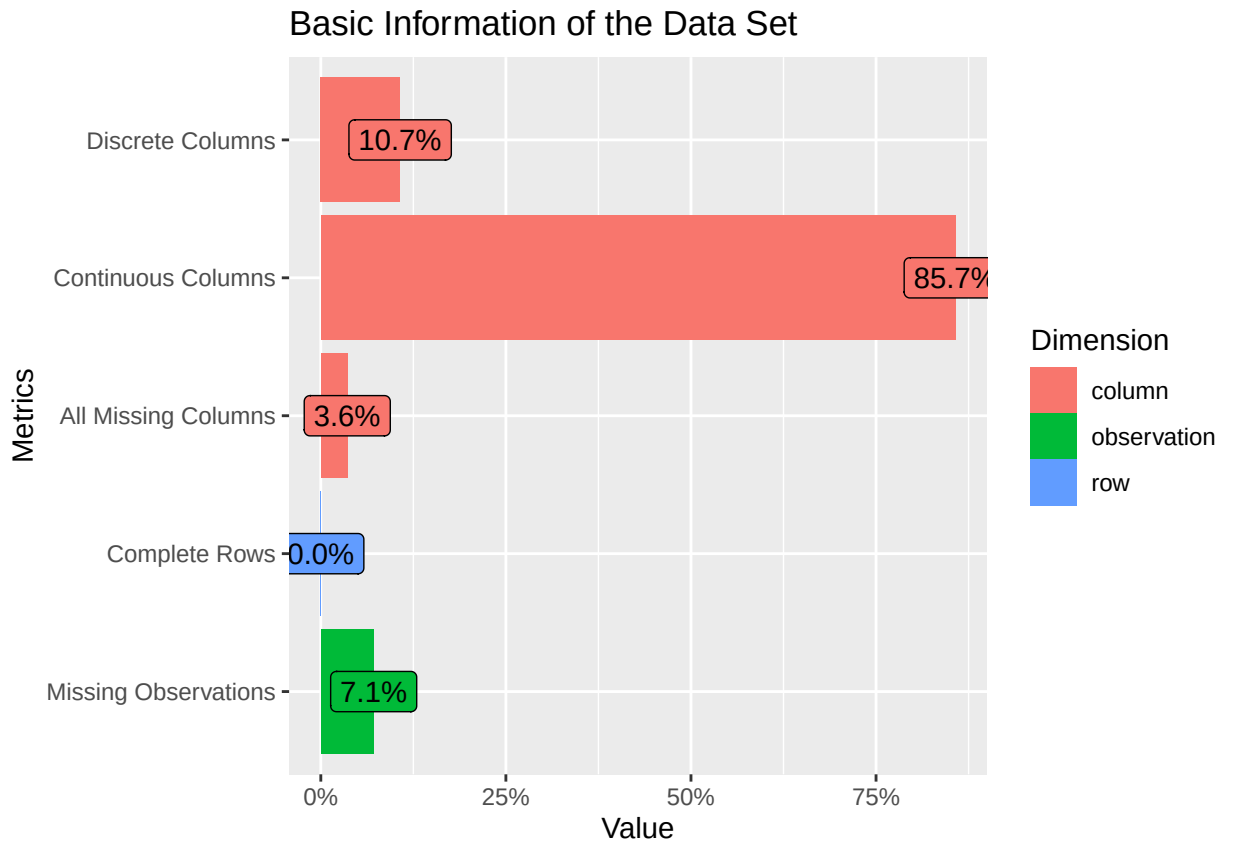


Figure 1: Information of the WNBA data set

From the above, we can infer that there are 23% discrete and 77% continuous columns in the data set. The total data set consisted of 7.1% observations that were missing, and all the missing columns totaled up to 3.6%.

```
plot_missing(WNBA, title = "Visualization of Missing Values", ggtheme = theme_gray())
```

From the above plot, I find that the column X and X.1 have missing values. After further analysis, it was found that this could have occurred due to a data set entry error. Thus, these two columns have been removed from our analysis as it doesn't provide any additional information.

```
#Removing Columns X and X.1
WNBA$X <- NULL
WNBA$X.1 <- NULL

#Check for duplicate values
WNBA[duplicated(WNBA),]
```

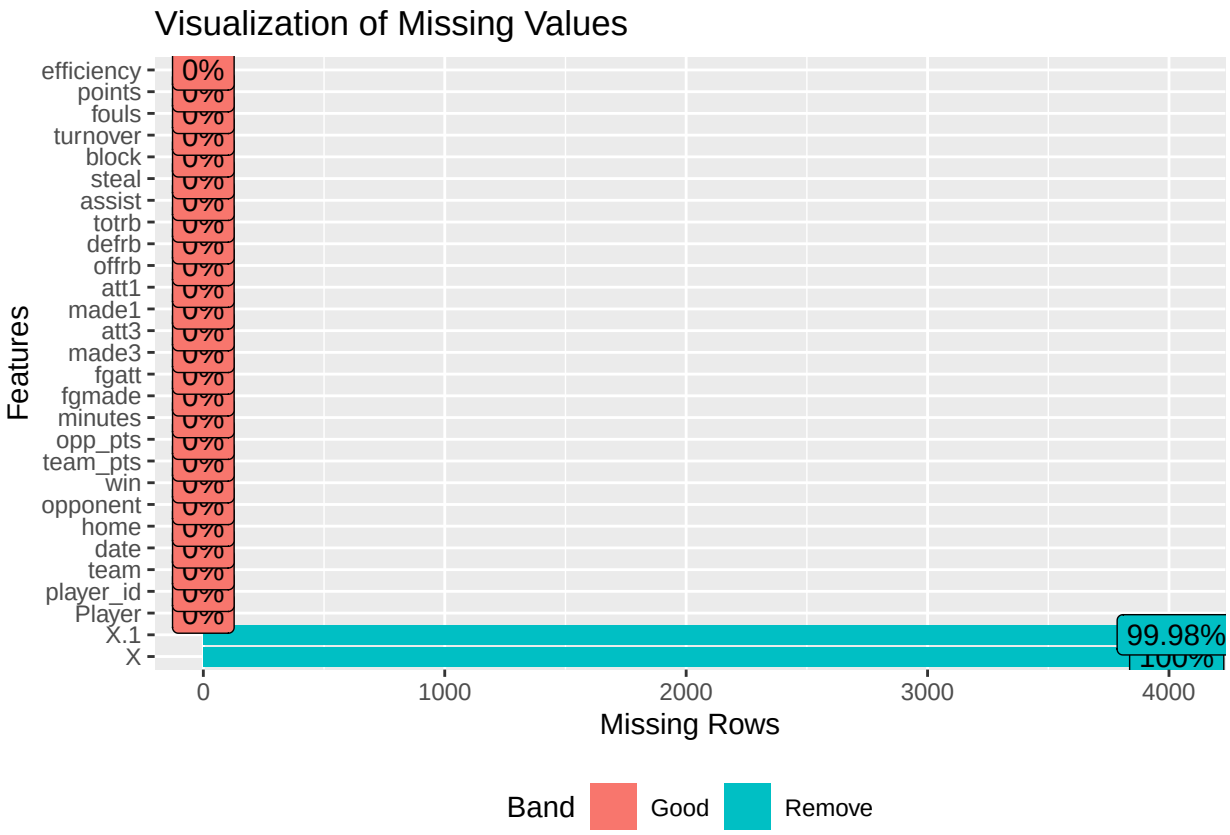


Figure 2: Missing values of the WNBA data set

```
##  [1] Player     player_id  team       date       home       opponent
##  [7] win        team_pts   opp_pts    minutes    fgmade     fgatt
## [13] made3      att3       made1      att1       offrb      defrb
## [19] totrb      assist     steal      block      turnover   fouls
## [25] points     efficiency
## <0 rows> (or 0-length row.names)
```

From the above outputs, it is clear that there are no duplicates in the data set.

```
#Replacing data entry errors with the correct values
WNBA$opp_pts = gsub('-100', '100', WNBA$opp_pts)
WNBA$opp_pts = gsub('-102', '102', WNBA$opp_pts)
WNBA$opp_pts = gsub('-105', '105', WNBA$opp_pts)
WNBA$opp_pts = gsub('-107', '107', WNBA$opp_pts)
WNBA$opp_pts = gsub('-108', '108', WNBA$opp_pts)
WNBA$opp_pts = gsub('-112', '112', WNBA$opp_pts)
WNBA$opp_pts = gsub('-74', '74', WNBA$opp_pts)
WNBA$opp_pts = gsub('-78', '78', WNBA$opp_pts)
WNBA$opp_pts = gsub('-82', '82', WNBA$opp_pts)
WNBA$opp_pts = gsub('-85', '85', WNBA$opp_pts)
WNBA$opp_pts = gsub('-88', '88', WNBA$opp_pts)
WNBA$opp_pts = gsub('-99', '99', WNBA$opp_pts)
WNBA$team_pts[3814] <- 112
WNBA$team_pts[3838] <- 112
WNBA$team_pts[3885] <- 112
WNBA$team_pts[3437] <- 101
WNBA$team_pts[3558] <- 105
WNBA$team_pts[3574] <- 105
WNBA$team_pts[4000] <- 107
WNBA$team_pts[3475] <- 100
WNBA$team_pts[3716] <- 100
WNBA$team_pts[3653] <- 102
WNBA$team_pts[3537] <- 108
WNBA$team_pts[3934] <- 101
WNBA$team_pts[3960] <- 101
WNBA$team_pts[3888] <- 100
WNBA$team_pts[3907] <- 100
WNBA$team_pts[3553] <- 101
WNBA$team_pts[3895] <- 101
WNBA$team_pts[3912] <- 101
WNBA$team_pts[3769] <- 102
WNBA$team_pts[3805] <- 102
WNBA$team_pts[3982] <- 105
WNBA$team_pts[3999] <- 105
WNBA$team_pts[3569] <- 101
WNBA$team_pts <- as.integer(WNBA$team_pts)
WNBA$opp_pts <- as.integer(WNBA$opp_pts)
```

Replaced the incorrect data entry values with the correct values above.

```
#Descriptive Statistics of the Data set
stargazer(WNBA, type = "text", title = "Descriptive Statistics of the Data Set", header = FALSE, single
```

```
##
```

```
## Descriptive Statistics of the Data Set
## ============================================================
## Statistic    N    Mean   St. Dev. Min Pctl(25) Pctl(75) Max
## ------------------------------------------------------------
## player_id  4,032 64.287   39.062   1     31       93    152
## date       4,032 675.514  95.489  516   610      729    817
## home       4,032  0.498   0.500    0     0        1      1
## win        4,032  0.499   0.500    0     0        1      1
## team_pts   4,032 76.961   10.476  46     69       83    112
## opp_pts    4,032 76.975   10.416  46     69       83    112
## minutes    4,032 20.480   10.476   0     12       29    52
## fgmade     4,032  2.933   2.656    0     1        4     16
## fgatt      4,032  6.704   4.923    0     3        10    30
## made3      4,032  0.467   0.912    0     0        1      7
## att3       4,032  1.421   1.972    0     0        2     13
## made1      4,032  1.466   2.075    0     0        2     16
## att1       4,032  1.881   2.469    0     0        3     18
## offrb      4,032  0.940   1.295    0     0        1      9
## defrb      4,032  2.464   2.399    0     1        4     17
## totrb      4,032  3.404   3.168    0     1        5     22
## assist     4,032  1.729   1.981    0     0        3     13
## steal      4,032  0.770   1.038    0     0        1      7
## block      4,032  0.379   0.820    0     0        1     11
## turnover   4,032  1.344   1.371    0     0        2     11
## fouls      4,032  1.849   1.458    0     1        3      6
## points     4,032  7.798   6.852    0     2        12    48
## efficiency 4,032  8.549   8.176   -7     2        13    44
## ------------------------------------------------------------
```

```
describe(WNBA)
```

```
##            vars    n   mean    sd median trimmed    mad min max range  skew
## Player*       1 4032  75.42 44.19     77   75.45  59.30   1 152   151 -0.01
## player_id     2 4032  64.29 39.06     62   62.59  45.96   1 152   151  0.28
## team*         3 4032   6.53  3.46      7    6.54   4.45   1  12    11 -0.02
## date          4 4032 675.51 95.49    703  677.63 131.95 516 817   301 -0.09
## home          5 4032   0.50  0.50      0    0.50   0.00   0   1     1  0.01
## opponent*     6 4032   6.47  3.46      6    6.46   4.45   1  12    11  0.02
## win           7 4032   0.50  0.50      0    0.50   0.00   0   1     1  0.01
## team_pts      8 4032  76.96 10.48     76   76.57  10.38  46 112    66  0.35
## opp_pts       9 4032  76.98 10.42     76   76.59  10.38  46 112    66  0.34
## minutes      10 4032  20.48 10.48     21   20.60  12.60   0  52    52 -0.08
## fgmade       11 4032   2.93  2.66      2    2.60   2.97   0  16    16  1.04
## fgatt        12 4032   6.70  4.92      6    6.26   4.45   0  30    30  0.78
## made3        13 4032   0.47  0.91      0    0.25   0.00   0   7     7  2.38
## att3         14 4032   1.42  1.97      0    1.03   0.00   0  13    13  1.62
## made1        15 4032   1.47  2.07      0    1.05   0.00   0  16    16  1.90
## att1         16 4032   1.88  2.47      1    1.41   1.48   0  18    18  1.68
## offrb        17 4032   0.94  1.30      1    0.68   1.48   0   9     9  1.88
## defrb        18 4032   2.46  2.40      2    2.11   1.48   0  17    17  1.43
## totrb        19 4032   3.40  3.17      3    2.94   2.97   0  22    22  1.40
## assist       20 4032   1.73  1.98      1    1.38   1.48   0  13    13  1.55
## steal        21 4032   0.77  1.04      0    0.58   0.00   0   7     7  1.69
## block        22 4032   0.38  0.82      0    0.19   0.00   0  11    11  3.61
```

```
## turnover     23 4032   1.34  1.37      1    1.15   1.48   0  11    11  1.19
## fouls        24 4032   1.85  1.46      2    1.74   1.48   0   6     6  0.55
## points       25 4032   7.80  6.85      6    6.99   5.93   0  48    48  1.04
## efficiency   26 4032   8.55  8.18      7    7.70   7.41  -7  44    51  0.93
##            kurtosis   se
## Player*       -1.23 0.70
## player_id     -0.85 0.62
## team*         -1.22 0.05
## date          -1.08 1.50
## home          -2.00 0.01
## opponent*     -1.22 0.05
## win           -2.00 0.01
## team_pts       0.15 0.16
## opp_pts        0.16 0.16
## minutes       -1.00 0.16
## fgmade         0.93 0.04
## fgatt          0.29 0.08
## made3          6.55 0.01
## att3           2.55 0.03
## made1          4.46 0.03
## att1           3.26 0.04
## offrb          4.24 0.02
## defrb          2.49 0.04
## totrb          2.16 0.05
## assist         2.75 0.03
## steal          3.54 0.02
## block         21.73 0.01
## turnover       1.73 0.02
## fouls         -0.44 0.02
## points         1.04 0.11
## efficiency     0.65 0.13
```

```
str(WNBA)
```

```
## 'data.frame':    4032 obs. of  26 variables:
##  $ Player    : Factor w/ 152 levels "Alana Beard",..: 97 97 97 97 97 97 97 97 97 97 ...
##  $ player_id : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ team      : Factor w/ 12 levels "ATL","CHI","CON",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ date      : int  516 518 523 524 526 530 601 606 608 613 ...
##  $ home      : int  0 1 0 1 0 1 0 0 0 0 ...
##  $ opponent  : Factor w/ 12 levels "ATL","CHI","CON",..: 12 3 11 7 2 9 9 10 5 1 ...
##  $ win       : int  1 1 1 1 1 1 1 0 1 0 ...
##  $ team_pts  : int  89 90 94 87 75 88 87 62 85 82 ...
##  $ opp_pts   : int  77 87 93 82 72 72 79 65 72 85 ...
##  $ minutes   : int  36 43 35 31 35 34 40 33 40 34 ...
##  $ fgmade    : int  11 12 13 11 5 10 8 6 5 2 ...
##  $ fgatt     : int  21 24 21 19 14 18 19 15 17 11 ...
##  $ made3     : int  3 3 5 5 1 2 1 0 0 2 ...
##  $ att3      : int  7 7 9 8 3 4 5 4 4 9 ...
##  $ made1     : int  9 6 7 3 3 4 1 0 2 4 ...
##  $ att1      : int  11 6 7 3 3 6 2 0 3 4 ...
##  $ offrb     : int  4 3 3 0 1 5 4 3 3 3 ...
##  $ defrb     : int  5 9 10 3 6 4 5 3 10 6 ...
##  $ totrb     : int  9 12 13 3 7 9 9 6 13 9 ...
```

```
##  $ assist    : int  3 2 2 2 3 5 4 1 3 5 ...
##  $ steal     : int  1 5 2 0 3 2 4 0 0 2 ...
##  $ block     : int  1 1 0 2 0 0 1 2 1 0 ...
##  $ turnover  : int  1 6 4 3 2 2 3 1 1 1 ...
##  $ fouls     : int  3 5 2 2 2 2 4 1 3 4 ...
##  $ points    : int  34 33 38 30 14 26 18 12 12 10 ...
##  $ efficiency: int  35 35 43 26 16 30 21 11 15 16 ...
```

From the descriptive statistics, we infer that our analysis will consist of 4032 observations and 26 variables in total. The stargazer function displays the mean, median, quartiles, standard deviation, minimum and maximum values for each variable in this dataset. From this, we observe that this dataset holds data for 152 players across 12 teams. The team points ranged between 46 and 112 whereas, the opponent points ranged between 46 and 112. The longest time a player played a match was for 52 minutes. The minimum points a player scored in a game was 0 and the maximum points a player scored in a game was 48. The efficiency of these players ranged between -7 and 44.

```
plot_bar(WNBA, by= "win", title = "Discrete Variable based on the variable 'win'")
```

```
## 1 columns ignored with more than 50 categories.
## Player: 152 categories
```

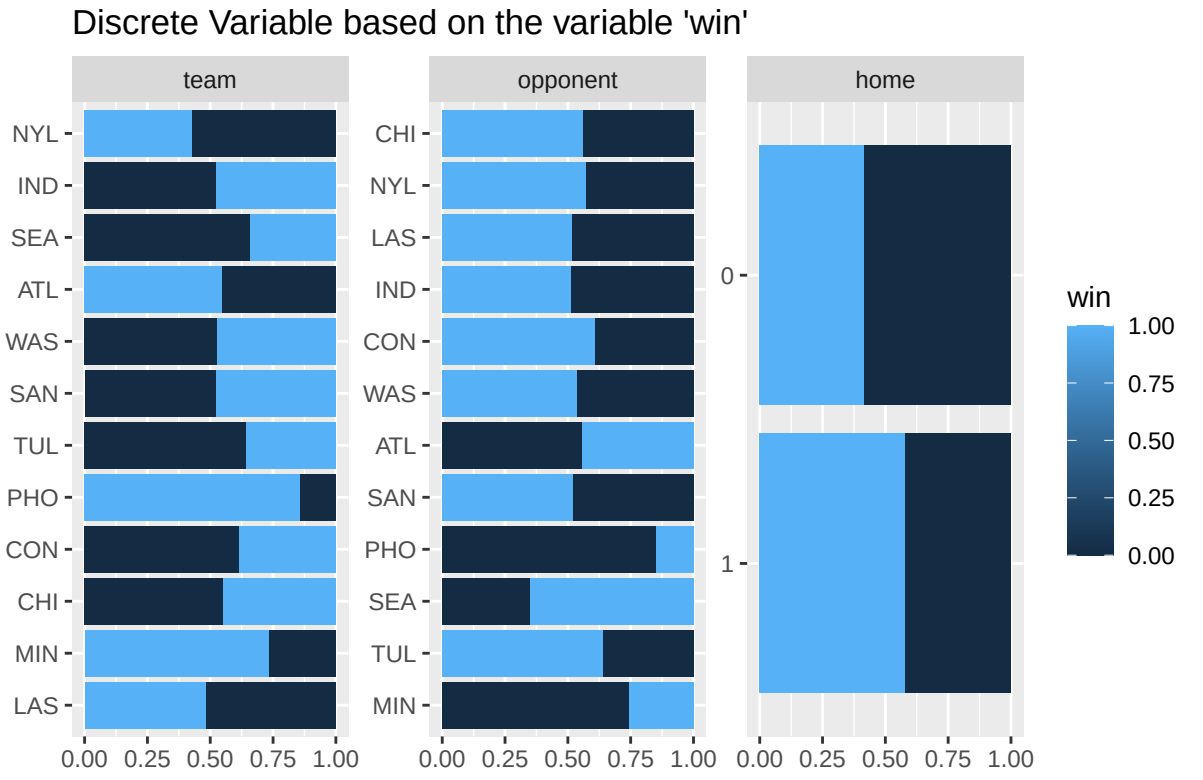


Figure 3: Discrete Variable based on 'win'

From the above plot, we can infer that there was approximately 60% of home games were won.

```
plot_histogram(WNBA, title = "Histograms of each feature in the dataset")
```

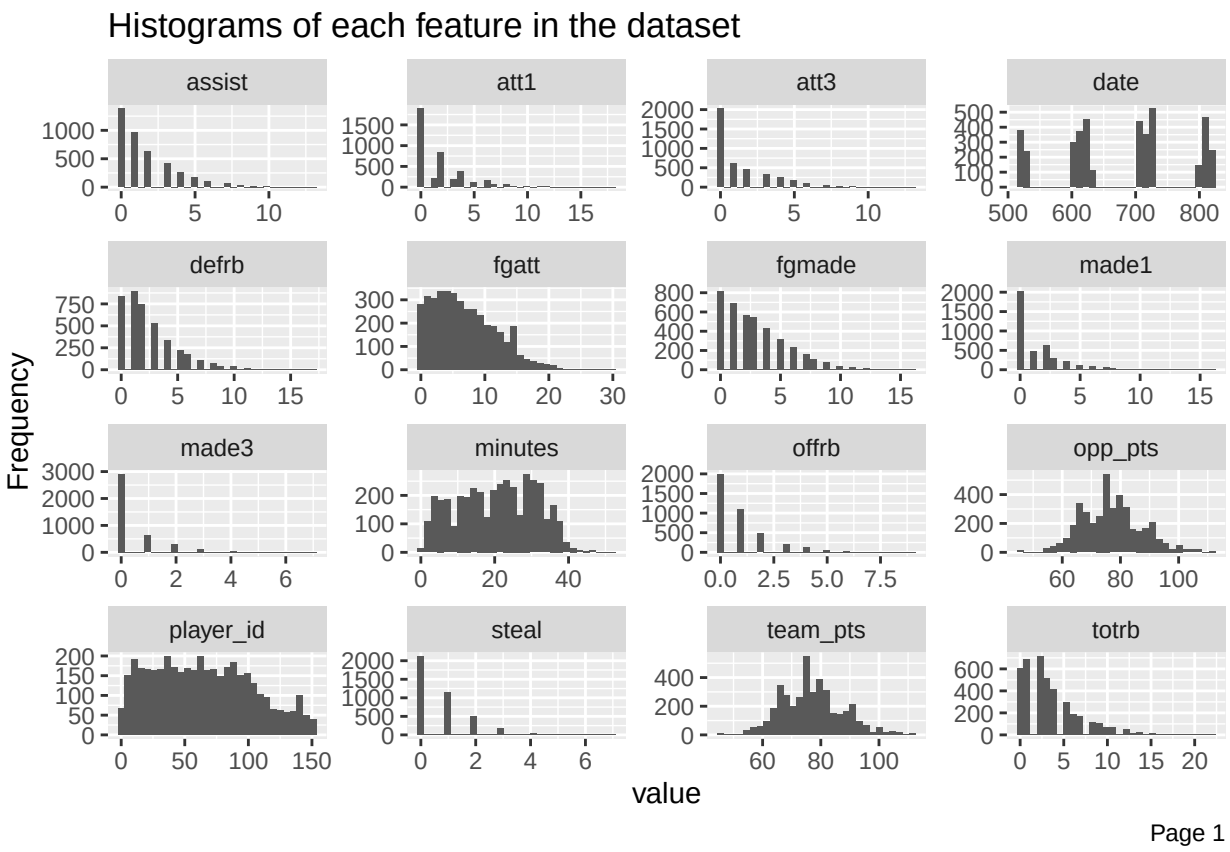


Figure 4: Histograms for the WNBA data set

From the above plots, we can observe that team_pts, opp_pts and efficiency are normally distributed.

```
plot_correlation(na.omit(WNBA), type = c("continuous"), ggtheme = theme_gray(), title = "Correlation of
```

From the above matrix, we notice that there are a couple of variables that are correlated to each other. The variables 'minutes' and 'fgmade', 'minutes' and 'fgatt', 'fgmade' and 'fgatt', 'minutes' and 'points', 'minutes' and 'efficiency', 'fgmade' and 'points', 'fgmade' and 'efficiency', 'fgatt' and 'points', 'fgatt' and 'efficiency', 'made3' and 'att3', 'made1' and 'att1', 'offrb' and 'totrb', and 'defrb' and 'totrr' have a strong positive correlation. As such, another correlation plot is made to gain a better understanding of the data set.

```
che <- WNBA[, c("win", "team_pts", "points", "efficiency", "fgmade", "fgatt", "made3", "att3", "made1",
pairs.panels(che, method = "pearson", hist.col = "blue", density = TRUE, ellipses = TRUE)
```

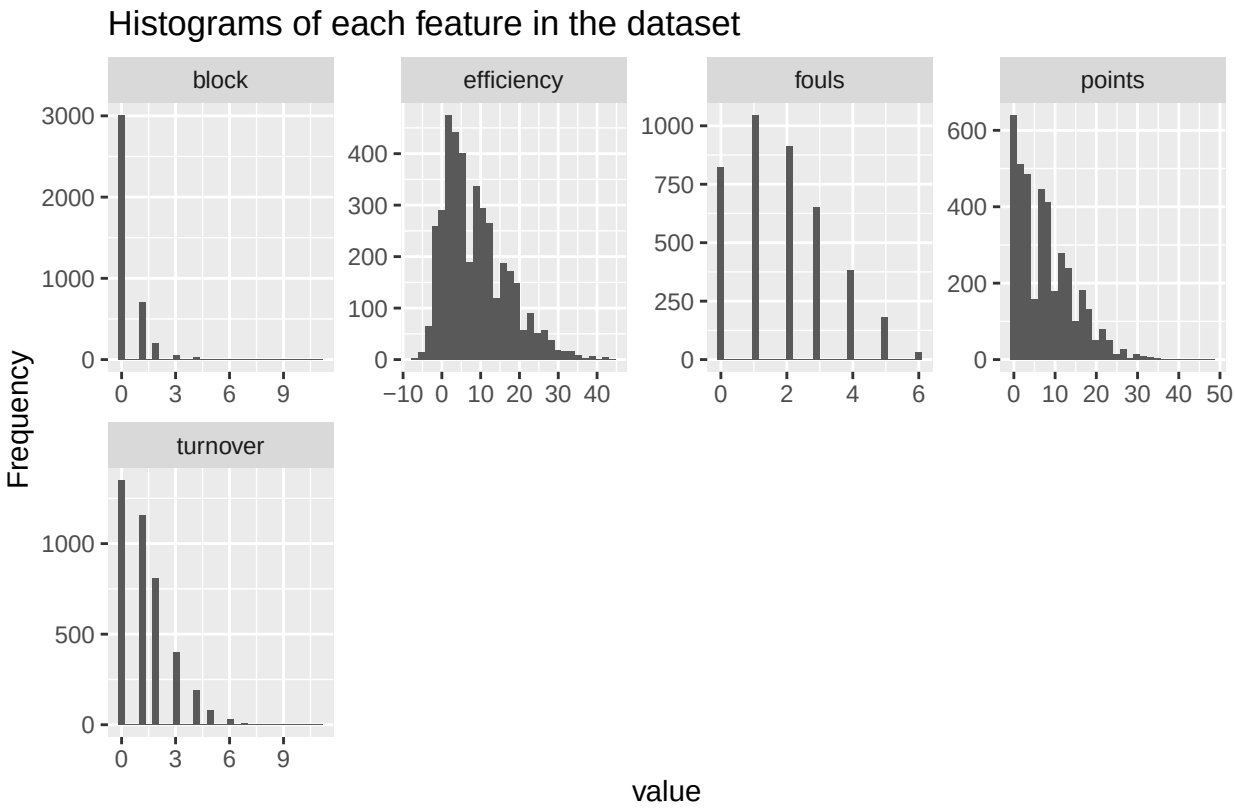


Figure 5: Histograms for the WNBA data set

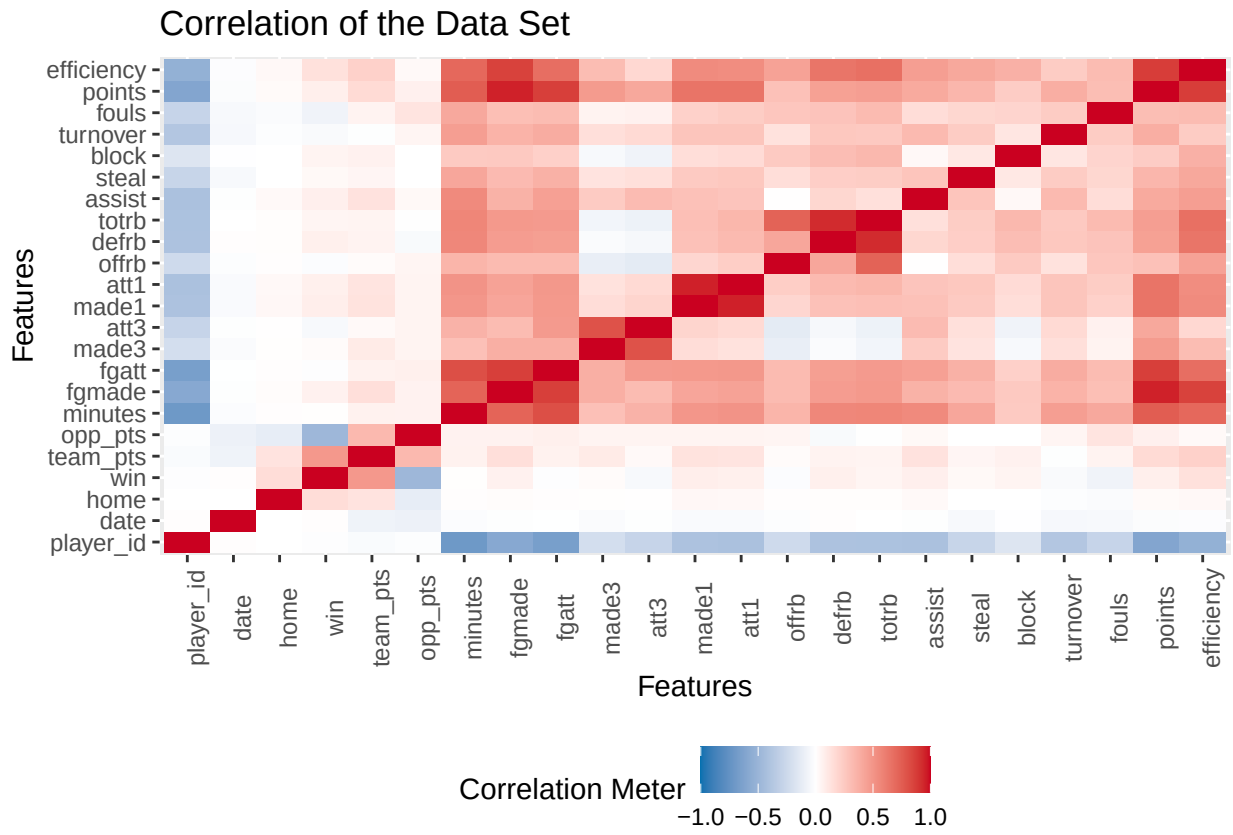
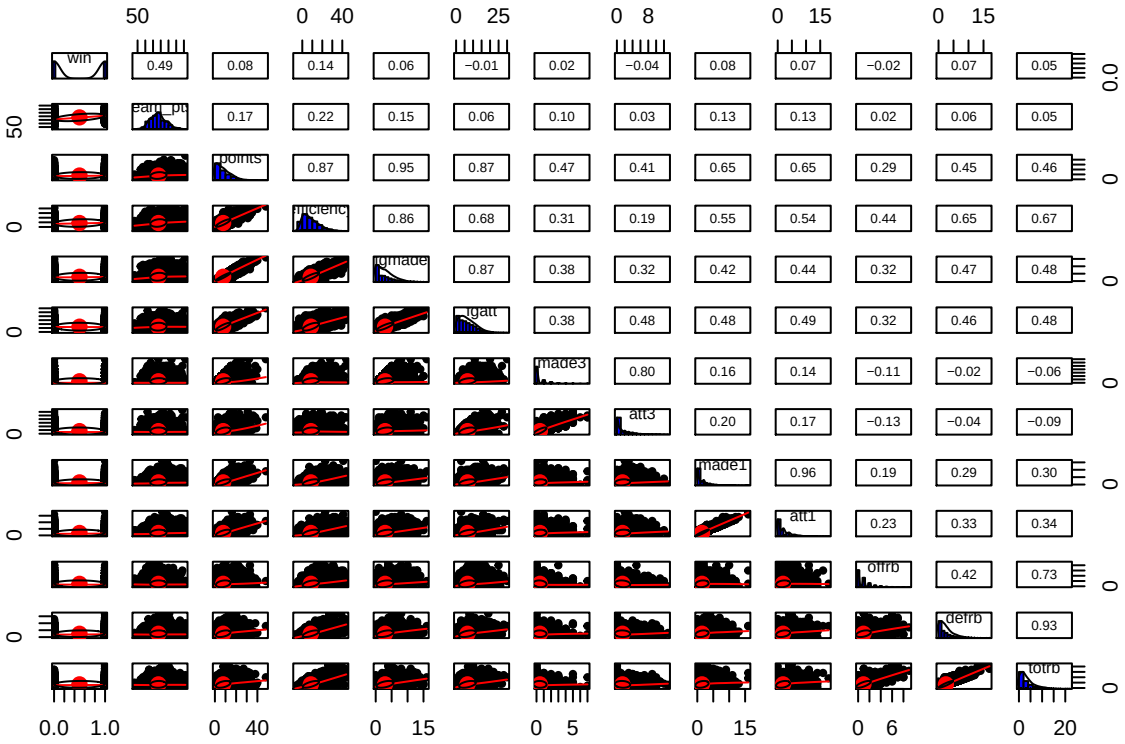


Figure 6: Correlation Matrix of the WNBA Data Set



From the above, it was found that 'points' and 'efficiency', 'points' and 'fgmade', 'points' and 'fgatt', 'efficiency' and 'fgmade', 'efficiency' and 'fgatt', 'fgmade' and 'fgatt', 'made3' and 'att3', 'points' and 'made1', 'efficiency' and 'made1', 'fgatt' and 'made1', 'points' and 'att1', 'efficiency' and 'att1', 'fgatt' and 'att1', 'made1' and 'att1', 'efficiency' and 'defrb', 'fgmade' and 'totrb', 'fgatt' and 'totrb', 'defrb' and 'totrb', 'offrb' and 'totrb' as well as 'efficiency' and 'totrb' have a strong positive correlation.

```
#Best players in a particular team
options(repr.plot.width = 12, repr.plot.height = 8)

WNBA %>%
  filter(team == "LAS") %>%
  select(Player, points, efficiency) %>%
  arrange(-points) %>%
  head(5) %>%
  gather(variable, Exp, -Player) %>%
  ggplot(aes(Player, Exp, fill = variable))+
  geom_col(position = "dodge")+
  geom_text(aes(label = Exp),position = position_dodge(width = 0.9), vjust = -0.5)+
  scale_fill_manual(values = c("#DA291C", "#004170"))+
  theme_minimal()+
  theme(legend.position = "right")+
  labs(y = "Points", x = "Players", title = "Best Players in LAS")
```

From the above plot, it can be observed that Candace Parker was the best player for the team LAS with the highest scored points of 34 followed by Nneka Ogwumike and Kristi Toliver. Also, the best player based on efficiency was Nneka Ogwumike followed by Candace Parker and Kristi Toliver.

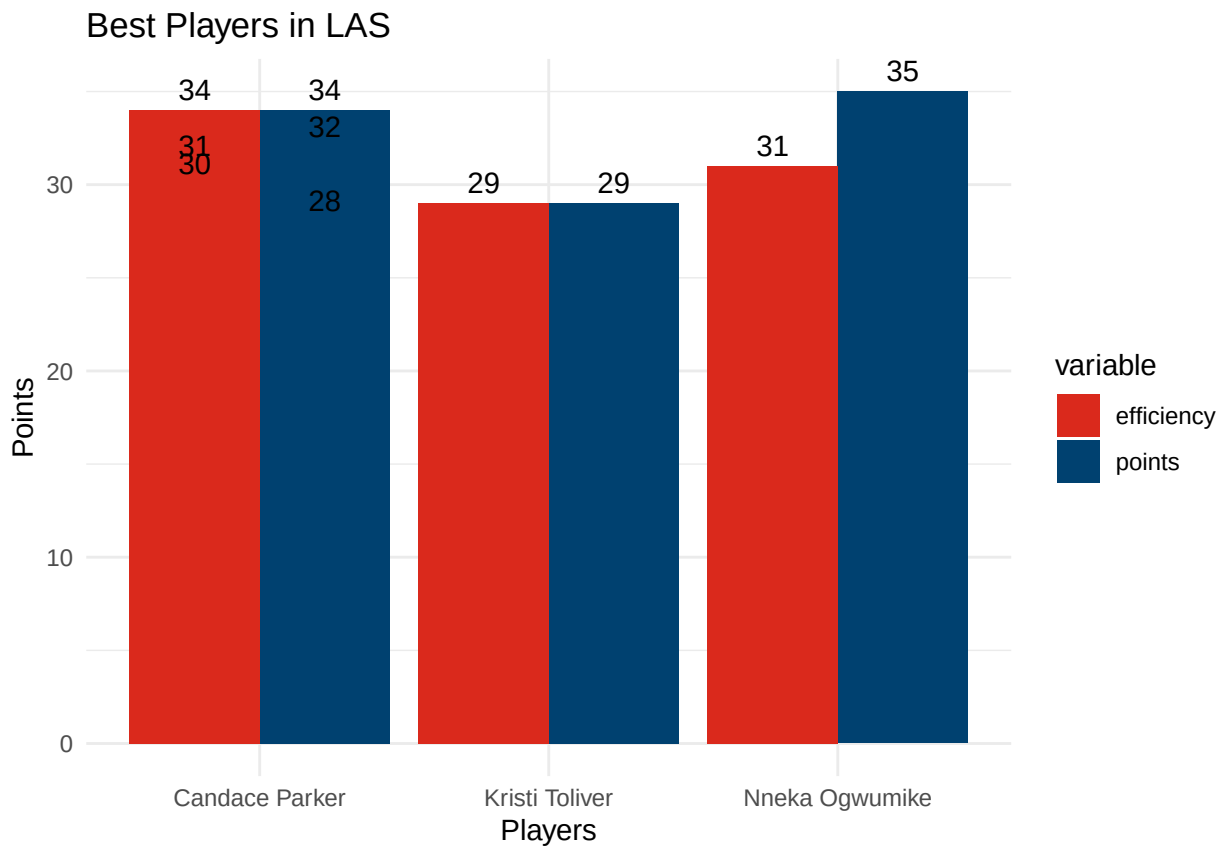


Figure 7: Best players in the ATL team based on their points and efficiency

Analysis

Exploratory Data Analysis

Business Question 1 - What is the effect of a player's overall match statistics on a player's future efficiency?

Motivation

A player's efficiency changes according to a number of factors. I wanted to explore the data and find the best predictors for our target variable (efficiency) that will estimate the future efficiency of the players. To find the prediction, I chose the multiple linear regression model.

Research

The focus of this research was to build an accurate predictive model that would estimate the efficiency of a player with a high accuracy. As an essential part of building such a model, I aimed to provide answers to the following questions:

- Which features or statistics are the most useful for estimating the future efficiency of a player?

Development

The data used for this research was a subset of Maya Moore's player statistics from the WNBA dataset. It consisted of all her match statistics of every single game she played. The variables that were considered for this model were:

bucket	Variable_Name	Description	Data_Type
1	fgmade	Field Goals	Integer
2	made3	Number of 3 pointers made	Integer
3	made1	Number of 1 pointers made	Integer
4	att1	Number of 1 pointer attempts	Integer
5	assist	Number of Assists	Integer
6	steal	Number of Steals	Integer
7	block	Number of Blocks	Integer
8	fouls	Number of Fouls	Integer
9	turnover	Number of Turnovers	Integer
10	efficiency	Player Efficiency	Integer

Code

```
Maya <- WNBA %>% filter(Player == "Maya Moore" ) %>% select(efficiency, assist, steal, block, fouls, tu
```

The data set was subsetted with the player Maya Moore game statistics.

```
efficiency <- Maya$efficiency
assist <- Maya$assist
steal <- Maya$steal
block <- Maya$block
fouls <- Maya$fouls
turnover <- Maya$turnover
made1 <- Maya$made1
made3 <- Maya$made3
fgmade <- Maya$fgmade
att1 <- Maya$att1
```

```
par(mfrow=c(2,2))
scatterplot(efficiency~assist,data=Maya)
```

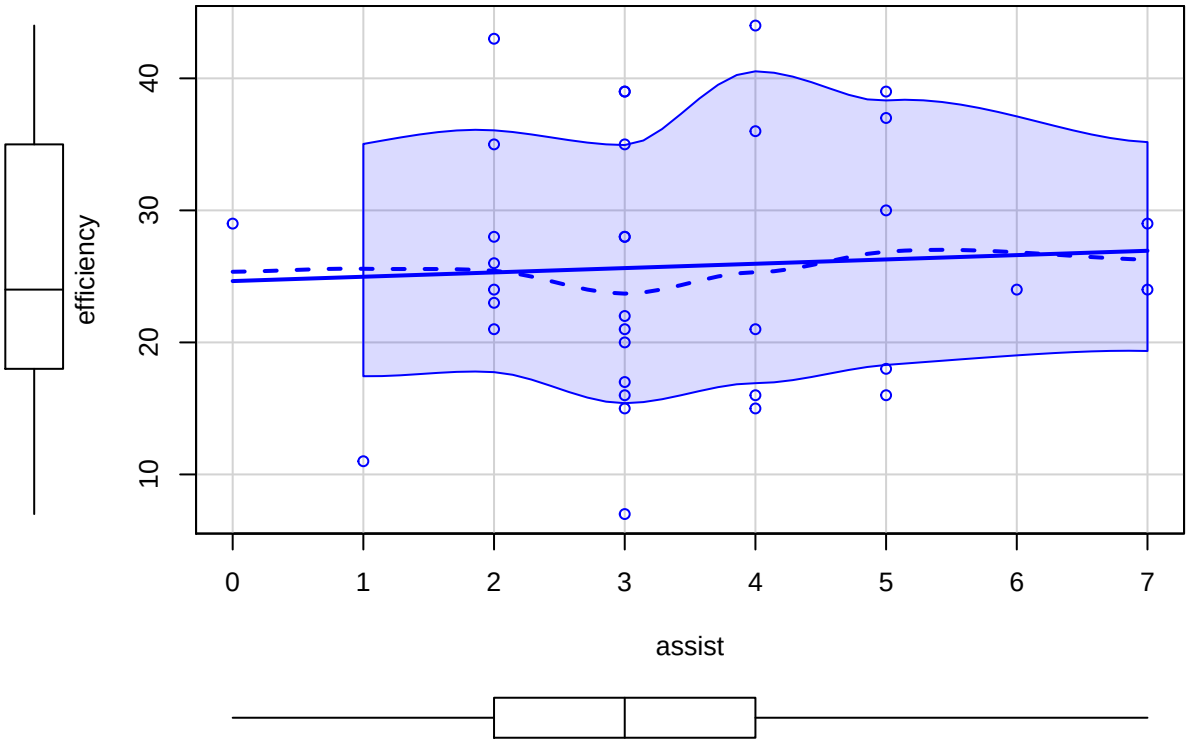


Figure 8: Scatterplots with efficiency

```
scatterplot(efficiency~fgmade,data=Maya)
```

```
scatterplot(efficiency~turnover,data=Maya)
```

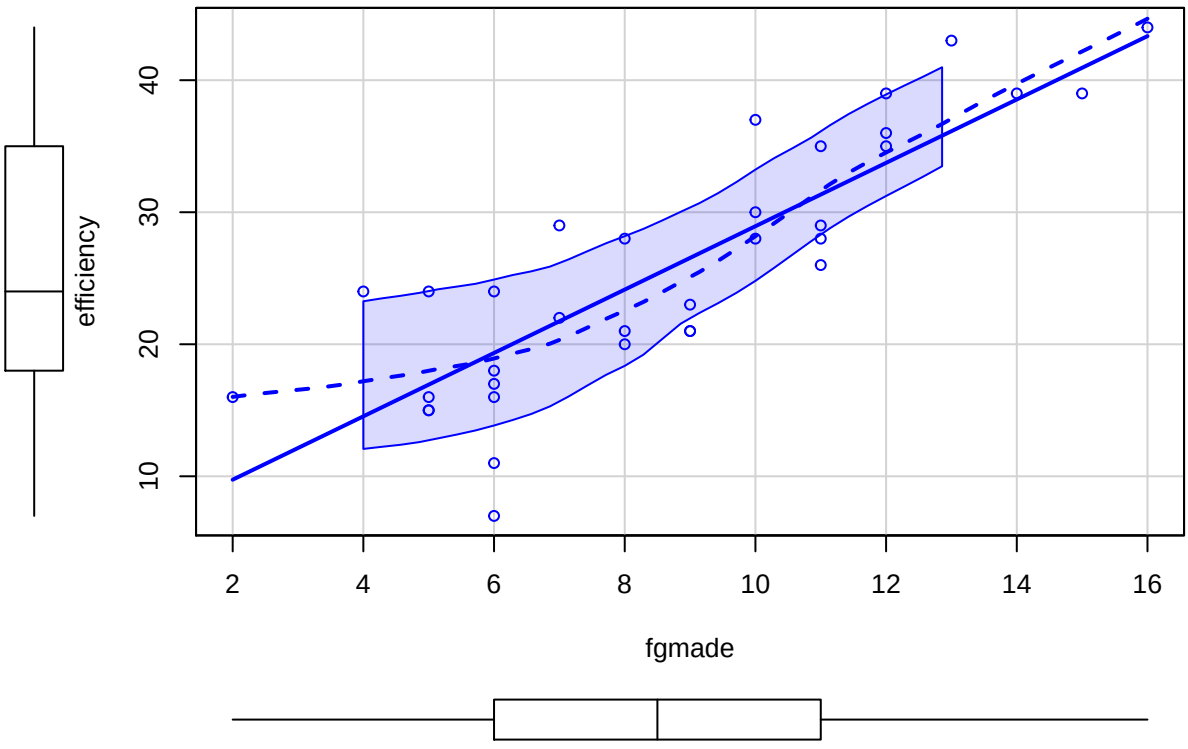


Figure 9: Scatterplots with efficiency

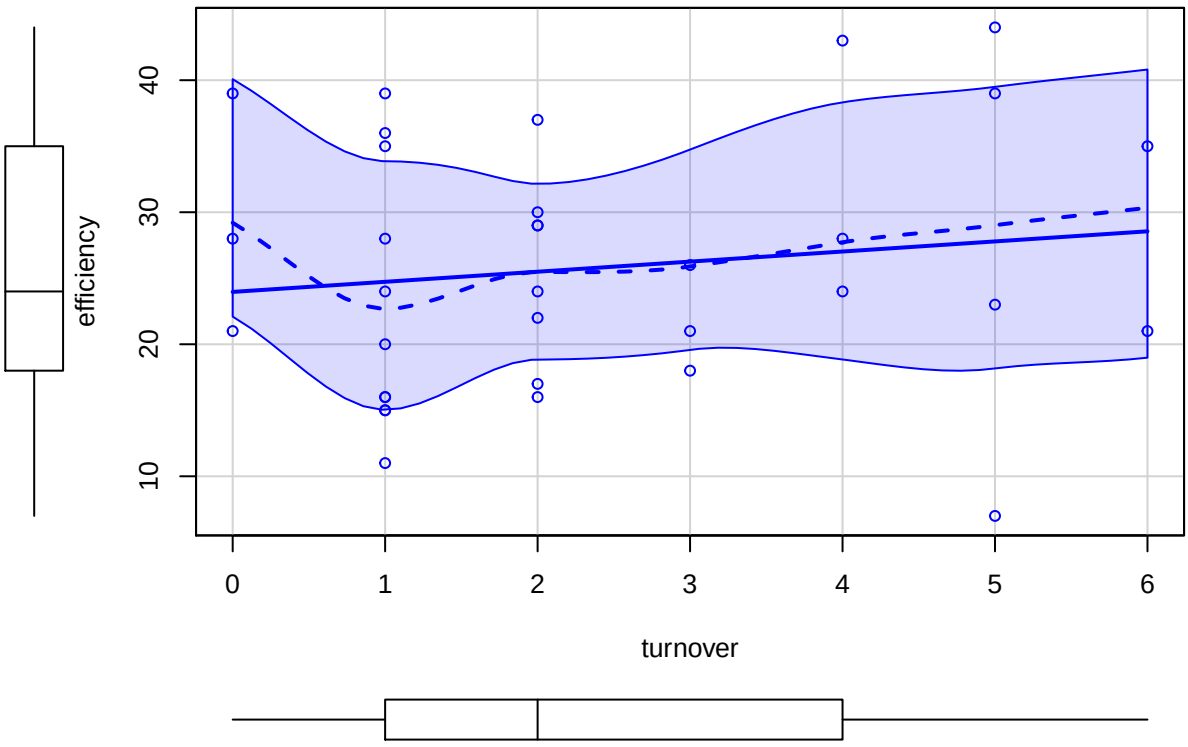
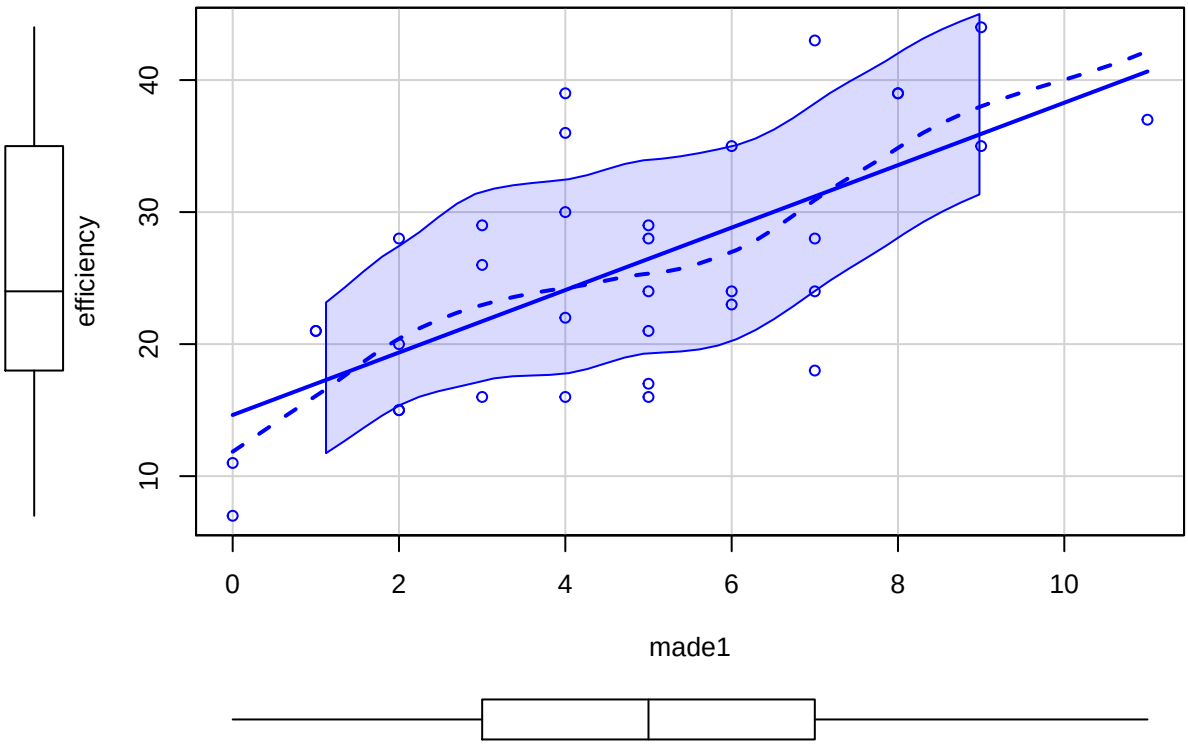


Figure 10: Scatterplots with efficiency

```
scatterplot(efficiency~made1,data=Maya)
```



The above scatter plots represents the variables efficiency with assist, fgmade, turnover and made1. From these plots, it was inferred that the variables had a positive linear relationship.

```
cors2 <- cor(Maya[, c('efficiency','assist','steal','block','fouls','turnover','made1', 'made3', 'fgmad
#corrplot(cors,col="blue",addCoef.col = "black")
plot_correlation(na.omit(cors2), type = c("continuous"), ggtheme = theme_gray(), title = "Correlation o
```

The correlation matrix represents the correlation between efficiency, assist, steal, block, fouls, turnover, made1, made3 and fgmade. From this matrix, it can be observed that the correlation between 'efficiency' and 'fgmade', 'efficiency' and 'made3', 'efficiency' and 'made1', 'fgmade' and 'made1', 'efficiency' and 'made3', and 'fgmade' and 'made3' have strong positive relationships. The variables 'fgmade' and 'assist', 'made3' and 'assist', 'turnover' and 'assist', as well as 'block' and 'steal' have strong negative relationships.

```
fit1 <- lm(efficiency ~., data = Maya)
summary(fit1)
```

```
##
## Call:
## lm(formula = efficiency ~ ., data = Maya)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7557 -1.7416 -0.5771  1.8915  7.0489
```

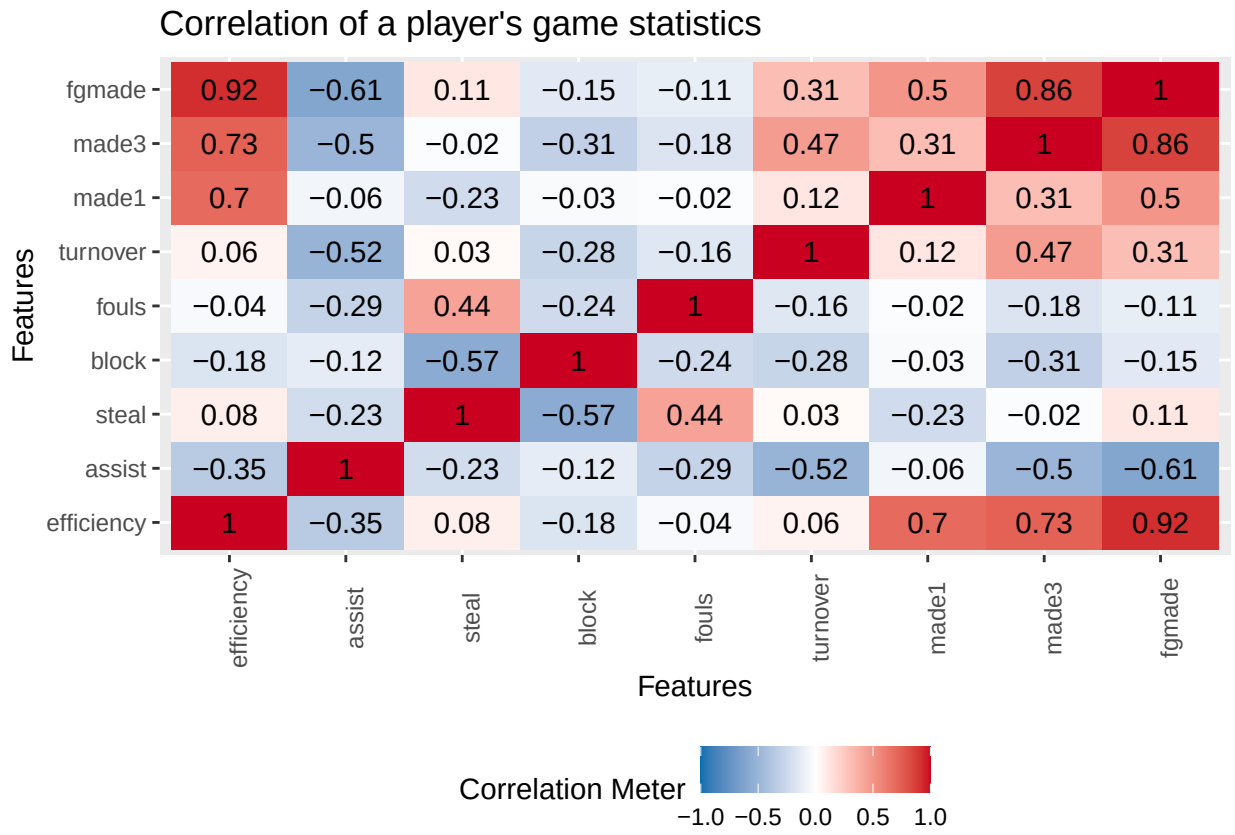


Figure 11: Correlation of Maya Moore's game statistics

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.1580     3.3292  -0.648   0.5230
## assist        1.0675     0.4644   2.298   0.0306 *
## steal         0.5859     0.5113   1.146   0.2631
## block         0.2712     0.6687   0.405   0.6887
## fouls         0.7638     0.6364   1.200   0.2418
## turnover     -0.9501     0.4041  -2.351   0.0273 *
## made1         1.9043     0.9812   1.941   0.0641 .
## made3         0.7387     0.5582   1.323   0.1982
## fgmade        1.9189     0.3163   6.067 2.89e-06 ***
## att1         -0.6993     0.8543  -0.819   0.4211
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.46 on 24 degrees of freedom
## Multiple R-squared:  0.9032, Adjusted R-squared:  0.8669
## F-statistic: 24.88 on 9 and 24 DF,  p-value: 4.556e-10
```

The above predictive model represents all the game statistics of Maya Moore. From this, it can be understood that the variables 'fgmade', 'assist', 'turnover' and 'made1' are significant to estimate the efficiency of the player.

```
AIC(fit1)
```

```
## [1] 191.0581
```

```
BIC(fit1)
```

```
## [1] 207.848
```

The AIC of the above model was 191.0581 and the BIC was 207.848.

```
fit2 <- lm(efficiency ~ fgmade + made1 + turnover + assist, data = Maya)
summary(fit2)
```

```
##
## Call:
## lm(formula = efficiency ~ fgmade + made1 + turnover + assist,
##     data = Maya)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.0442 -1.7106 -0.4506  1.9426  7.7054
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.0648     2.6014  -0.409 0.685307
## fgmade        2.2233     0.2239   9.929 7.76e-11 ***
## made1         1.1459     0.2804   4.086 0.000317 ***
## turnover     -0.6957     0.3778  -1.841 0.075823 .
```

```
## assist        1.1090     0.4416   2.511 0.017853 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.566 on 29 degrees of freedom
## Multiple R-squared:  0.8758, Adjusted R-squared:  0.8586
## F-statistic:  51.1 on 4 and 29 DF,  p-value: 1.008e-12
```

The multiple linear regression model represents the estimation of efficiency. The equation to estimate efficiency is efficiency = (2.2233)fgmade + (1.1459)made1 + (-0.6957)turnover + (1.1090)assist - 1.0648.The regression model, shows that the adjusted R-squared is 0.8586 which indicates that 85.86% of the variance in efficiency can be estimated by 'fgmade', 'made1', 'turnover' and 'assist'. From the above model, it can be interpreted that the 'efficiency' increases by 1 unit when 'fgmade' increases by a factor of 2.2233 provided there is no change in the other predictor variables. Likewise, 'efficiency' will increase by 1 unit when 'made1' increases by a factor of 1.1459 or when 'turnover' increases by a factor of -0.6957 or when 'assist' increases by a factor of 1.1090.

```
AIC(fit2)
```

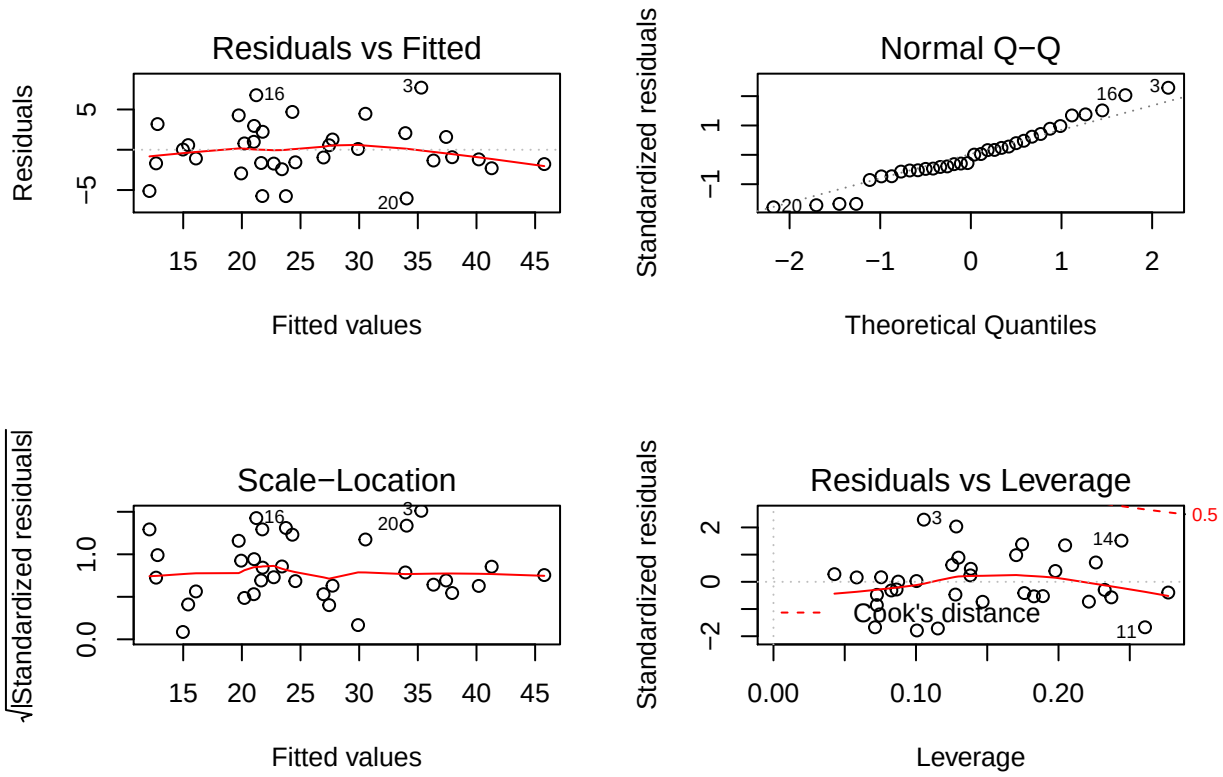
```
## [1] 189.5366
```

```
BIC(fit2)
```

```
## [1] 198.6948
```

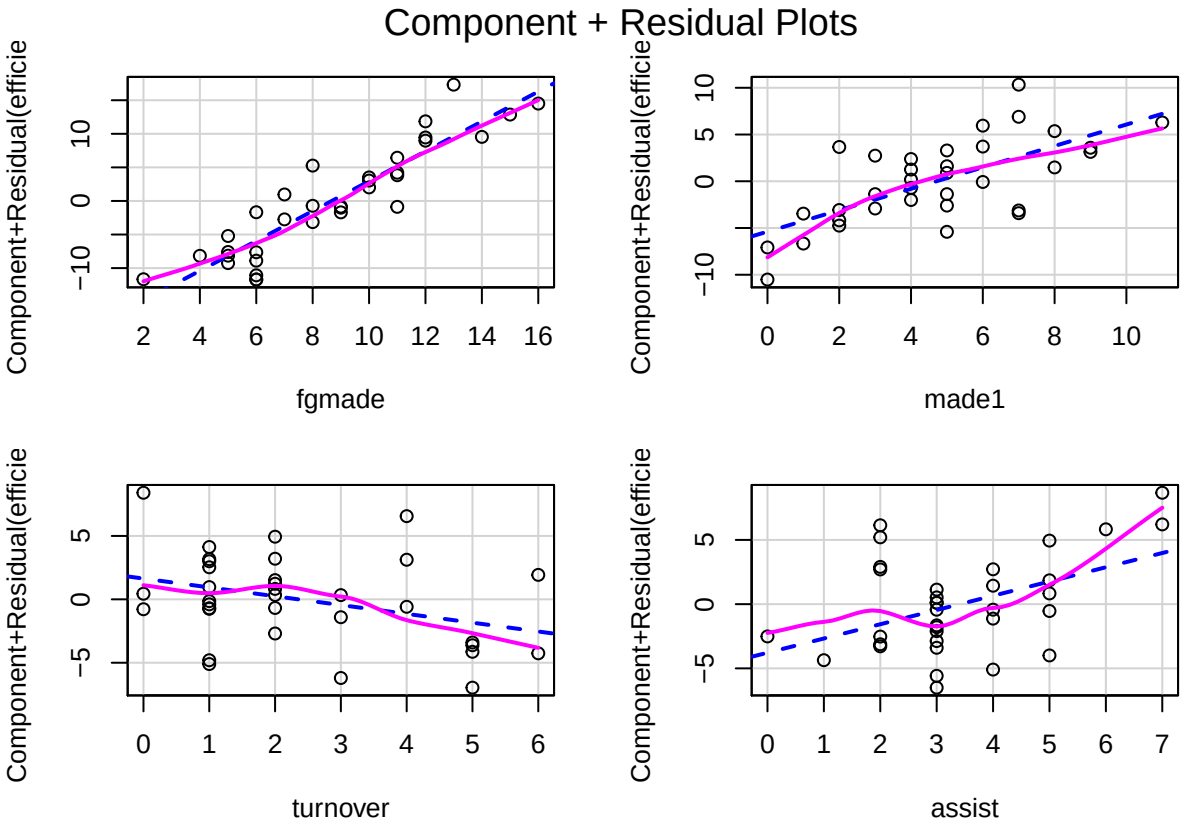
The AIC of the above model was 189.5366 and the BIC was 198.6948.

```
par(mfrow=c(2,2))
plot(fit2)
```

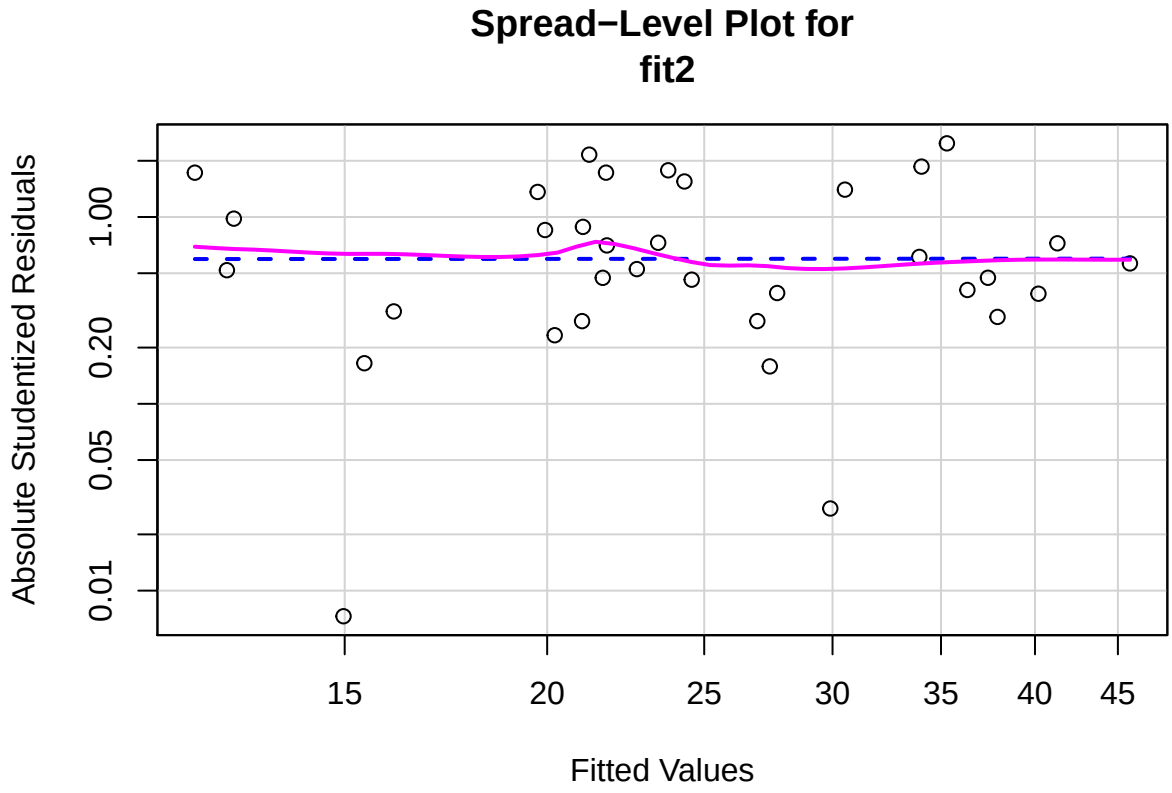


The first type of plot represents the residual vs fitted plot. This plot depicts if the residuals have any non-linear pattern. From this, I noticed that the red line deviated very slightly from the horizontal line. Thus, the residuals followed a linear pattern and the linear regression model was appropriate for this data set. The second plot represents the Normal Q-Q plot which determines if the residuals are normally distributed. From the Q-Q Plot, it was inferred that the data was roughly on the diagonal line. Thus, the residuals was not enough to declare that it was non-normally distributed. The third type of plot represents the Scale-Location Plot. It is used to check the assumption of homoscedasticity among the residuals. In this plot, the red line wasn't horizontal across the plot and showed the red line was moving upwards which indicated that the residuals spread wider on the x-axis. The fourth type of plot is the residuals vs leverage plot which is used to find the unusual observations. This plot displayed that there wasn't any data in the 1 or 0.5 area which indicated that there was no outliers or observations that needed extra attention.

```
crPlots(model=fit2)
```




```
spreadLevelPlot(fit2)
```




```
##
## Suggested power transformation:  0.9962918
```

```
vif(fit2)
```

```
##   fgmade    made1 turnover   assist
## 1.464761 1.466259 1.142571 1.230404
```

The variance inflation factor (VIF) was conducted to check for multicollinearity. From this, it was found that the VIF values for the predictors are less than 1.5 which indicated that the predictors had some correlation. Thus, it didn't need any modification.

```
b.mod <- regsubsets(efficiency ~., data = Maya, nvmax = 5)
res_sum <- summary(b.mod)
res_sum
```

```
## Subset selection object
## Call: regsubsets.formula(efficiency ~ ., data = Maya, nvmax = 5)
## 9 Variables  (and intercept)
##          Forced in Forced out
## assist       FALSE      FALSE
## steal        FALSE      FALSE
## block        FALSE      FALSE
## fouls        FALSE      FALSE
```

```
## turnover     FALSE      FALSE
## made1        FALSE      FALSE
## made3        FALSE      FALSE
## fgmade       FALSE      FALSE
## att1         FALSE      FALSE
## 1 subsets of each size up to 5
## Selection Algorithm: exhaustive
##          assist steal block fouls turnover made1 made3 fgmade att1
## 1  ( 1 ) " "    " "   " "   " "   " "      " "   " "   "*"    " "
## 2  ( 1 ) " "    " "   " "   " "   " "      "*"   " "   "*"    " "
## 3  ( 1 ) "*"    " "   " "   " "   " "      "*"   " "   "*"    " "
## 4  ( 1 ) "*"    " "   " "   "*"   " "      "*"   " "   "*"    " "
## 5  ( 1 ) "*"    " "   " "   "*"   "*"      "*"   " "   "*"    " "
```

The above summary displays the best set of independent variables for the dependent variable "efficiency". The best model for estimating the efficiency was represented with an asterisk. From the above, it was observed that if the best model was to be estimated by 5 independent variables, then the variables would be 'assist', 'fouls', 'turnover', 'made1' and 'fgmade'.

```
data.frame(Adj.R2 = which.max(res_sum$adjr2),
           CP = which.min(res_sum$cp),
           BIC = which.min(res_sum$bic))
```

```
##   Adj.R2 CP BIC
## 1      5  5   5
```

From these results, it was interpreted that a model with 5 independent variables was the best model.

```
best_reg_model <- lm(formula = efficiency ~assist + fouls + turnover + made1 + fgmade, data = Maya)
summary(best_reg_model)
```

```
##
## Call:
## lm(formula = efficiency ~ assist + fouls + turnover + made1 +
##     fgmade, data = Maya)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1215 -1.8947 -0.5618  1.8968  8.6116
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -3.9971     2.9151  -1.371  0.18121
## assist        1.2399     0.4276   2.899  0.00719 **
## fouls         1.0783     0.5590   1.929  0.06393 .
## turnover     -0.6736     0.3615  -1.864  0.07291 .
## made1         1.0047     0.2780   3.615  0.00117 **
## fgmade        2.2623     0.2151  10.519 3.11e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.41 on 28 degrees of freedom
## Multiple R-squared:  0.8903, Adjusted R-squared:  0.8707
## F-statistic: 45.46 on 5 and 28 DF,  p-value: 1.386e-12
```

The multiple linear regression model equation to estimate efficiency was efficiency = (1.2399)assist + (1.0783)fouls + (-0.6736)turnover + (1.0047)made1 + (2.2623)fgmade - 3.9971. The adjusted R-squared was 0.8707 which indicated that 87.07% of the variance in 'efficiency' can be estimated by 'assist', 'fouls', 'turnover', 'made1' and 'fgmade'.

```
set.seed(123)
trainIndex <-createDataPartition(Maya$efficiency, p = 0.70, list = FALSE)
train <- Maya[trainIndex,]
test <- Maya[-trainIndex,]
```

```
fit3 <- lm(formula = efficiency ~assist + fouls + turnover + made1 + fgmade, data = train)
summary(fit3)
```

```
##
## Call:
## lm(formula = efficiency ~ assist + fouls + turnover + made1 +
##     fgmade, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.6331 -2.2479 -0.6624  2.0295  8.8638
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.6615     3.5975  -1.574   0.1312
## assist        1.4457     0.5676   2.547   0.0192 *
## fouls         1.0969     0.6729   1.630   0.1187
## turnover     -0.7417     0.4170  -1.779   0.0905 .
## made1         0.8887     0.3682   2.414   0.0255 *
## fgmade        2.4199     0.2990   8.093 9.75e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.754 on 20 degrees of freedom
## Multiple R-squared:  0.8864, Adjusted R-squared:  0.858
## F-statistic: 31.22 on 5 and 20 DF,  p-value: 8.602e-09
```

The adjusted R-squared is 0.858 which indicates that 85.8% of the variance in efficiency can be estimated by 'assist', 'fouls', 'turnover', 'made1' and 'fgmade'.

```
fit4 <- lm(formula = efficiency ~assist + fouls + turnover + made1 + fgmade, data = test)
summary(fit4)
```

```
##
## Call:
## lm(formula = efficiency ~ assist + fouls + turnover + made1 +
##     fgmade, data = test)
##
## Residuals:
##       5       6      10      12      23      31      33      34
## -1.3255 -0.6036  1.5909  2.5670 -1.2879  0.3350 -2.2366  0.9607
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.33070   15.48613   0.021   0.9849
## assist       1.04224    1.34845   0.773   0.5204
## fouls        0.08788    2.71143   0.032   0.9771
## turnover     0.17670    2.69361   0.066   0.9537
## made1        1.07266    1.12660   0.952   0.4415
## fgmade       2.02419    0.58448   3.463   0.0742 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.077 on 2 degrees of freedom
## Multiple R-squared:  0.961,  Adjusted R-squared:  0.8636
## F-statistic: 9.865 on 5 and 2 DF,  p-value: 0.09459
```

The adjusted R-squared is 0.8636 which indicates that 86.36% of the variance in efficiency can be estimated by assist, fouls, turnover, made1 and fgmade. The multiple linear regression model equation to estimate efficiency is efficiency = (1.04224)assist + (0.08788)fouls + (0.1767)turnover + (1.07266)made1 + (2.02419)fgmade + 0.3307.

```
AIC(fit4)
```

```
## [1] 43.59673
```

```
BIC(fit4)
```

```
## [1] 44.15282
```

```
AIC(fit3,fit4)
```

```
## Warning in AIC.default(fit3, fit4): models are not all fitted to the same number
## of observations
```

```
##      df       AIC
## fit3  7 149.74862
## fit4  7  43.59673
```

```
BIC(fit3,fit4)
```

```
## Warning in BIC.default(fit3, fit4): models are not all fitted to the same number
## of observations
```

```
##      df       BIC
## fit3  7 158.55529
## fit4  7  44.15282
```

When comparing the AIC and BIC of the train and test set, it was found that 'fit4' i.e., the test data set had a lower AIC and BIC value.

Results

The multiple linear regression model was conducted on the Maya data set which was taken from the WNBA data set. From this, it was found that significant variables were 'turnover', 'fgmade', 'made1' and 'assist' with which a second model was created which gave an adjusted R-squared value of 0.8586. Numerous checks were performed to assess the assumptions. The residual vs fitted plot depicted that the residuals followed a linear pattern and the linear regression model was appropriate for the data set. From the Q-Q plot, I inferred that the data was roughly on the diagonal line, however, the residuals was not enough to declare that it was non-normally distributed. The Scale-Location Plot was used to check the homoscedasticity among the residuals and it was observed that the residuals spread wider on the x-axis. The residuals vs leverage plot displayed that there were no outliers in the data used. Also, the variance inflation factor (VIF) indicated that the predictors had some correlation as the VIF values for the predictors were less than 1.5.

After conducting the following checks, the regsubset function was performed where it was found that the model with five independent variables gave the best result. The significant variables were 'assist', 'fouls', 'turnover', 'made1' and 'fgmade' which gave an adjusted R-squared value of 0.8707. The data was split into two i.e., the train and test set. From this, it was observed that the test data set had a higher adjusted R-squared value of 0.8638 compared to the train data set which was 0.858. Also, the AIC and BIC value of the test set data was lesser than the train set data which indicated that it was a good model to estimate the future efficiency of a player.

Interpretation

The multiple linear regression model was conducted to find the effect of a player's overall match statistics on a player's future efficiency. From the above analyses, it was found that model with five independent variables i.e., assist, fouls, turnover, made1 and fgmade gave the best model to estimate the future efficiency of the player. This model gave an adjusted R-squared value of 0.8707 which indicated that 87.07% of the variance in efficiency can be estimated by assist, fouls, turnover, made1 and fgmade. When the data was split to train and test set, it was found that the test data set performed better than the train set as the adjusted R-squared value was higher than the train set and the AIC as well as the BIC value of the test set was lower than the train set which indicated that this was a good predictive model to estimate the future efficiency of a player.

Business Question 2 - What is the effect of a team's overall match statistics on a team winning the game?

Motivation

A game can have two outcomes i.e., win or lose, which can change depending on various factors. As such, we wanted to explore the data and find the best predictors for our target variable "win" that will predict the wins of a particular team. To find our prediction, the logistic regression model was used.

Research

The focus of this research was to build an accurate predictive model that would predict the outcome of a game with high accuracy. As an essential part of building such a model, I aimed to provide answers to the following questions:

- Which features or statistics are the most useful for predicting the outcomes?
- What is the accuracy that I can predict the winner of each game?

Development

The data used for this research was a subset of a team's statistics i.e., for Los Angeles Sparks(LAS). It consisted of the team's match statistics for each game.

bucket	Variable_Name	Description	Data_Type
1	win	Number of Wins	Integer
2	team_pts	Team Points	Integer
3	efficiency	Player Efficiency	Integer
4	assist	Number of Assists	Integer
5	fouls	Number of Fouls	Integer
6	block	Number of Blocks	Integer
7	steal	Number of Steals	Integer
8	home	Home Stadium	Integer
9	fgmade	Field Goals	Integer
10	made3	Number of 3 pointers made	Integer
11	made1	Number of 1 pointers made	Integer
12	defrb	Defense Rebound	Integer
13	offrb	Offense Rebound	Integer
14	turnover	Number of turnovers	Integer

Code

```
Team_LAS <- WNBA %>% filter(team == "LAS") %>% select(win, team_pts, efficiency, assist, fouls, block, s
```

The data set was subsetted with the team Los Angeles Sparks (LAS) game statistics.

```
ones <- Team_LAS[which(Team_LAS$win == 1),]
zero <- Team_LAS[which(Team_LAS$win == 0),]
set.seed(123)
no_bs <- nrow(ones)
zero_no_bs <- head(zero, no_bs)
ones_no_bs <- ones

#Split balanced data set to train and test sets
r_ind <- sample(1:nrow(ones_no_bs), 0.70*no_bs)
ones_train <- ones_no_bs[r_ind,]
z_train <- zero_no_bs[r_ind,]
train_data <- rbind(ones_train, z_train)

#To create test data
test_1 <- ones_no_bs[-r_ind,]
test_0 <- zero_no_bs[-r_ind,]
test_data <- rbind(test_1, test_0)
cat("Training dataset size is = [", dim(train_data), "] testing is = [", dim(test_data), "]\n")
```

```
## Training dataset size is = [ 208 14 ] testing is = [ 90 14 ]
```

```
table(train_data$win)
```

```
##
##   0   1
## 104 104
```

```
table(test_data$win)
```

```
##
##  0  1
## 45 45
```

To avoid overfitting the model, the data set was split into 70/30 train and test sets, where 70% of the observations were used in the training data set and 30% of the observations were used in the testing data set in order to carry out further analyses.

```
str(Team_LAS)
```

```
## 'data.frame':    307 obs. of  14 variables:
##  $ win       : int  1 0 1 0 0 1 0 0 0 0 ...
##  $ team_pts  : int  80 69 83 84 85 102 72 67 74 77 ...
##  $ efficiency: int  24 23 29 34 18 23 29 18 23 18 ...
##  $ assist    : int  4 3 6 6 7 6 3 2 4 3 ...
##  $ fouls     : int  1 3 0 5 2 1 2 2 2 1 ...
##  $ block     : int  2 2 0 0 1 0 1 3 2 0 ...
##  $ steal     : int  2 1 4 6 3 1 3 2 2 1 ...
##  $ home      : int  0 1 1 0 0 0 1 0 0 1 ...
##  $ fgmade    : int  8 8 8 10 7 8 9 5 7 7 ...
##  $ made3     : int  0 0 1 1 0 0 2 0 0 0 ...
##  $ made1     : int  2 7 6 11 1 1 4 8 2 2 ...
##  $ defrb     : int  5 9 7 9 4 5 9 4 5 4 ...
##  $ offrb     : int  1 2 0 3 1 1 2 1 0 1 ...
##  $ turnover  : int  3 3 4 3 3 2 1 1 1 1 ...
```

```
#Fitting the model
model1 <- glm(win ~ ., data= train_data, family = binomial(link = "logit"))
summary(model1)
```

```
##
## Call:
## glm(formula = win ~ ., family = binomial(link = "logit"), data = train_data)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.79585  -1.03119  -0.01798   0.97507   1.99903
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.43889    1.49727  -2.297  0.02163 *
## team_pts     0.06085    0.01923   3.165  0.00155 **
## efficiency   0.16602    0.06930   2.396  0.01660 *
```

```
## assist      -0.26753    0.10096  -2.650  0.00805 **
## fouls       -0.25888    0.11250  -2.301  0.02139 *
## block       -0.28374    0.25506  -1.112  0.26595
## steal       -0.01703    0.14663  -0.116  0.90754
## home        -0.70480    0.31645  -2.227  0.02593 *
## fgmade      -0.25755    0.15317  -1.681  0.09267 .
## made3       -0.10141    0.24266  -0.418  0.67600
## made1       -0.06620    0.10138  -0.653  0.51377
## defrb       -0.04706    0.09277  -0.507  0.61198
## offrb       -0.33479    0.14989  -2.234  0.02551 *
## turnover     0.04552    0.13703   0.332  0.73974
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 288.35  on 207  degrees of freedom
## Residual deviance: 250.34  on 194  degrees of freedom
## AIC: 278.34
##
## Number of Fisher Scoring iterations: 4
```

In the above model, the train data set with all the variables were fitted into the model where it was found that the variables i.e., 'team_pts', 'efficiency', 'assist', 'fouls', 'home', 'fgmade' and 'offrb' were significant to predict the team wins and this model had an AIC of 278.34 with the difference between the two deviances at approximately 38.

```
model2 <- glm(win ~ team_pts + efficiency + assist + fouls + home + fgmade + offrb, data= train_data, fa
model2
```

```
##
## Call:  glm(formula = win ~ team_pts + efficiency + assist + fouls +
##     home + fgmade + offrb, family = binomial(link = "logit"),
##     data = train_data)
##
## Coefficients:
## (Intercept)     team_pts   efficiency       assist        fouls         home
##    -3.59327      0.06223      0.12043     -0.24992     -0.26519     -0.69173
##      fgmade        offrb
##    -0.21990     -0.31123
##
## Degrees of Freedom: 207 Total (i.e. Null);  200 Residual
## Null Deviance:       288.3
## Residual Deviance: 252.1     AIC: 268.1
```

In the above model, only the significant variables i.e., team_pts, efficiency, assist, fouls, home, and fgmade were fitted into the model. The AIC of this model was 268.1 and the difference between the two deviances was approximately 36. Comparing the above two models, it can be observed that the second model performed better since it had a lower AIC than the first model. However, the second model's difference between the two deviance was slightly lower than the first model. As such, further analyses was conducted to ascertain which model would accurately predict the winning team.

```
model3 <- glm(win ~ team_pts + efficiency + assist + fouls + home + fgmade + offrb, data= test_data, fa
model3
```

```
##
## Call:  glm(formula = win ~ team_pts + efficiency + assist + fouls +
##     home + fgmade + offrb, family = binomial(link = "logit"),
##     data = test_data)
##
## Coefficients:
## (Intercept)     team_pts   efficiency       assist        fouls         home
##    -5.07443      0.07500      0.11938     -0.05316     -0.25692      0.16133
##      fgmade        offrb
##    -0.31578     -0.23321
##
## Degrees of Freedom: 89 Total (i.e. Null);  82 Residual
## Null Deviance:       124.8
## Residual Deviance: 107   AIC: 123
```

In the above model, the test data set was fitted into the model with the significant variables. The AIC of this model was 123 and the difference between the two deviances was approximately 18.

```
AICc(model1)
```

```
## [1] 280.5145
```

```
AICc(model2)
```

```
## [1] 268.8343
```

From the above AICc function performed, it was found that the model2 performed better than the model1 as the AICc value of model2 was lower than model1.

```
AIC(model1,model2, model3)
```

```
## Warning in AIC.default(model1, model2, model3): models are not all fitted to the
## same number of observations
```

```
##        df      AIC
## model1 14 278.3384
## model2  8 268.1107
## model3  8 122.9881
```

```
BIC(model1,model2, model3)
```

```
## Warning in BIC.default(model1, model2, model3): models are not all fitted to the
## same number of observations
```

```
##        df      BIC
## model1 14 325.0639
## model2  8 294.8110
## model3  8 142.9866
```

To find out which model is giving better results, the AIC and BIC of the two models were compared. From this, we found that model2 was the preferred model as it had a low AIC and BIC value compared to model1. Also, when comparing model2 and model3, it was found that the testing model performed better than the training model as the AIC and BIC values were lower in the testing model.

```
coef(model2)
```

```
## (Intercept)    team_pts  efficiency      assist       fouls        home
## -3.59327107  0.06223026  0.12043116 -0.24992092 -0.26518967 -0.69172926
##      fgmade       offrb
## -0.21990324 -0.31123123
```

```
exp(coef(model2))
```

```
## (Intercept)    team_pts  efficiency      assist       fouls        home
##   0.0275082   1.0642074   1.1279831   0.7788624   0.7670604   0.5007095
##      fgmade       offrb
##   0.8025965   0.7325445
```

From the above, it can be observed that the probability of the team LAS winning the game increases by a factor of 0.06223026 for each 1 unit increase in 'team_pts' provided all the other variables remain the same. Likewise, the probability of winning the game increases by a factor of 0.12043116 for each 1 unit increase in 'efficiency', or by a factor of -0.24992092 for each 1 unit increase in 'assist', or by a factor of -0.26518967 for each 1 unit increase in 'fouls', or by a factor of -0.21990324 for each 1 unit increase in 'fgmade', or by a factor of -0.69172926 for each 1 unit increase in 'home', or by a factor of -0.31123123 for each 1 unit increase in 'offrb'.

```
#Creating a dataset to observe the probability changes for the different values
pr <- data.frame((team_pts = c(min(Team_LAS$team_pts), mean(Team_LAS$team_pts), max(Team_LAS$team_pts))
pr$probability <- predict(model2, pr, type = "response")
pr
```

```
##   X.team_pts...c.min.Team_LAS.team_pts...mean.Team_LAS.team_pts...
## 1                                                          64.00000
## 2                                                          77.42671
## 3                                                         102.00000
##   X.efficiency...c.min.Team_LAS.efficiency...mean.Team_LAS.efficiency...
## 1                                                             -4.000000
## 2                                                              9.674267
## 3                                                             36.000000
##   X.assist...c.min.Team_LAS.assist...mean.Team_LAS.assist...max.Team_LAS.assist...
## 1                                                                        0.000000
## 2                                                                        2.055375
## 3                                                                        9.000000
##   X.fouls...c.min.Team_LAS.fouls...mean.Team_LAS.fouls...max.Team_LAS.fouls...
## 1                                                                    0.000000
## 2                                                                    1.905537
## 3                                                                    6.000000
##   X.fgmade...c.min.Team_LAS.fgmade...mean.Team_LAS.fgmade...max.Team_LAS.fgmade...
## 1                                                                        0.000000
## 2                                                                        3.397394
## 3                                                                       12.000000
```

```
##   X.home...c.min.Team_LAS.home...mean.Team_LAS.home...max.Team_LAS.home...
## 1                                                                 0.000000
## 2                                                                 0.485342
## 3                                                                 1.000000
##   X.offrb...c.min.Team_LAS.offrb...mean.Team_LAS.offrb...max.Team_LAS.offrb...
## 1                                                                    0.0000000
## 2                                                                    0.9739414
## 3                                                                    6.0000000
##   probability
## 1   0.4769515
## 2   0.4962855
## 3   0.1247012
```

From the results, it was observed that the minimum values probability was 0.48, mean values probability was 0.50 and the max values probability was 0.12.

```
train_data$win <- as.factor(train_data$win)
#To make predictions on the data
prob_train <- predict(model2, newdata = train_data, type = "response")
pred <- as.factor(ifelse(prob_train >= 0.5, 1, 0))

#Confusion Matrix - For Model Accuracy
confusionMatrix(pred, train_data$win, positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 70 29
##          1 34 75
##
##                Accuracy : 0.6971
##                  95% CI : (0.6298, 0.7587)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : 6.384e-09
##
##                   Kappa : 0.3942
##
##  Mcnemar's Test P-Value : 0.6143
##
##             Sensitivity : 0.7212
##             Specificity : 0.6731
##          Pos Pred Value : 0.6881
##          Neg Pred Value : 0.7071
##              Prevalence : 0.5000
##          Detection Rate : 0.3606
##    Detection Prevalence : 0.5240
##       Balanced Accuracy : 0.6971
##
##        'Positive' Class : 1
##
```

In the above matrix, it can be inferred that the True Positive value is 75, the True Negative value is 50, the False Positive value is 34 and the False Negative value is 29. The Accuracy was 0.6971 which indicated

that from the positive and negative classes, 69.71% of the data were predicted accurately. The Sensitivity aka True Positive rate or Recall is 0.7212 which indicated that from all the positive classes, 72.12% were predicted accurately and the Specificity aka the False Positive rate is 0.6731 which indicated that from all the negative classes, 67.31% were predicted accurately. The Precision aka the Positive Pred value is 0.6881 which indicated that from all the classes that were predicted as positive, 68.81% of the data was positive. The Negative Pred value is 0.7071 which indicated that from all the classes that were predicted as negative, 70.71% of the data was negative.

```
test_data$win <- as.factor(test_data$win)
prob_test <- predict(model2, newdata = test_data, type = "response")
pred2 <- as.factor(ifelse(prob_test >= 0.5, 1, 0))

#Confusion Matrix - For Model Accuracy
confusionMatrix(pred2, test_data$win, positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 30 19
##          1 15 26
##
##                Accuracy : 0.6222
##                  95% CI : (0.5138, 0.7223)
##     No Information Rate : 0.5
##     P-Value [Acc > NIR] : 0.01315
##
##                   Kappa : 0.2444
##
##  Mcnemar's Test P-Value : 0.60691
##
##             Sensitivity : 0.5778
##             Specificity : 0.6667
##          Pos Pred Value : 0.6341
##          Neg Pred Value : 0.6122
##              Prevalence : 0.5000
##          Detection Rate : 0.2889
##    Detection Prevalence : 0.4556
##       Balanced Accuracy : 0.6222
##
##        'Positive' Class : 1
##
```

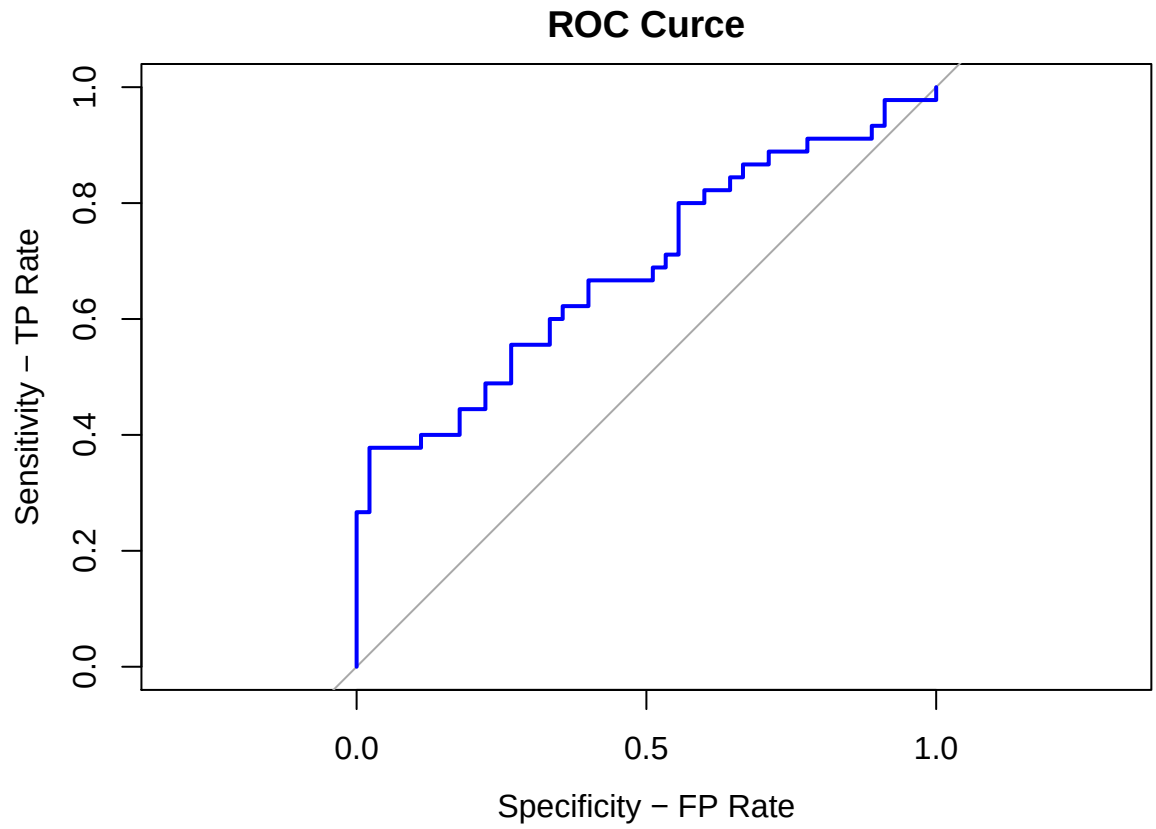
In the above matrix, it can be inferred that the True Positive value was 26, the True Negative value was 30, the False Positive value is 15 and the False Negative value is 19. The Accuracy was 0.6222 which indicated that from the positive and negative classes, 62.22% of the data were predicted accurately. The Sensitivity aka True Positive rate or Recall was 0.5778 which indicated that from all the positive classes, 57.78% of the data were predicted accurately and the Specificity aka the False Positive rate is 0.6667 which indicated that from all the negative classes, 66.67% of the data were predicted accurately. The Precision aka the Positive Pred value is 0.6341 which indicated that from all the classes that were predicted as positive, 63.41% of the data was positive. The Negative Pred value is 0.6122 which indicated that from all the classes that were predicted as negative, 61.22% of the data was negative.

```
ROC1 <- roc(test_data$win, prob_test)
```

```
## Setting levels: control = 0, case = 1
```

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

```
#ROC_ch <- coords(ROC1, "best")
plot(ROC1, col = "blue", ylab = "Sensitivity - TP Rate", xlab = "Specificity - FP Rate", main = "ROC Cu
```



The above plot indicates that the model moves upwards and has a curve which indicates that the model is good.

```
auc <- auc(ROC1)
auc
```

```
## Area under the curve: 0.6884
```

The area under the curve is the area that is between the blue line and the grey diagonal line. The AUC of our model is approximately 0.69 which indicates that it has a good measure of separability.

Results

The logistic regression model was conducted on the Team_LAS data set which was taken from the WNBA data set. The data was validated and split into two i.e., the train and test set. From this, we found that

significant variables were 'team_pts', 'efficiency', 'assist', 'fouls', 'home', 'fgmade', and 'offrb' with which a second model was created which gave an AIC of 268.1. Numerous checks were performed to assess which model performed better. By comparing the AIC and BIC of the models, it was found that model2 performed better than model1 as model 2 had a low AIC and BIC values as compared to model1.

The confusion matrix was conducted for the train and test set from which it was found that in the training data set, the Accuracy was 0.6971 which indicated that from the positive and negative classes, 69.71% of the data were predicted accurately. The Sensitivity aka True Positive rate or Recall is 0.7212 which indicated that from all the positive classes, 72.12% were predicted accurately and the Specificity aka the False Positive rate is 0.6731 which indicated that from all the negative classes, 67.31% were predicted accurately. The Precision aka the Positive Pred value is 0.6881 which indicated that from all the classes that were predicted as positive, 68.81% of the data was positive. The Negative Pred value is 0.7071 which indicated that from all the classes that were predicted as negative, 70.71% of the data was negative. Also, from the test data set, the Accuracy was 0.6222 which indicated that from the positive and negative classes, 62.22% of the data were predicted accurately. The Sensitivity aka True Positive rate or Recall was 0.5778 which indicated that from all the positive classes, 57.78% of the data were predicted accurately and the Specificity aka the False Positive rate is 0.6667 which indicated that from all the negative classes, 66.67% of the data were predicted accurately. The Precision aka the Positive Pred value is 0.6341 which indicated that from all the classes that were predicted as positive, 63.41% of the data was positive. The Negative Pred value is 0.6122 which indicated that from all the classes that were predicted as negative, 61.22% of the data was negative. The ROC curve was plotted and the Area under the curve (AUC) was 0.69 which indicated that the model had a good measure of separability.

Interpretation

The logistic regression model was conducted to find the effect of a team's overall match statistics on the team winning the game. From the above analyses, it was found that significant variables were 'team_pts', 'efficiency', 'assist', 'fouls', 'home', 'fgmade', and 'offrb' which gave the best model to predict the team wins. The best performing model was model2 (268.1, 294.8) as it had a low AIC and BIC value as compared to model1 (278.3, 325.1). The confusion matrix was conducted for the training and testing sets, where it was found that the test set had an accuracy of 62.22%, the recall had 57.78% of the data that were accurately predicted from both, the positive and negative classes. The precision in the model consisted of 63.41% of the data that was accurately predicted as positive and the negative pred value indicated that 61.22% of the data was accurately predicted as negative. The ROC curve was plotted where the area under the curve was 0.69 which indicated that the model was good.

Conclusion

From this project, the following conclusions were made: * There are 4032 observations and 26 variables in total. There were a total of 152 players across 12 teams. The team points ranged between 46 and 112 whereas, the opponent points range between 46 and 112. The longest time a player played a match was for 52 minutes. The minimum points a player scored in a game was 0 and the maximum points a player scored in a game was 48. The efficiency of these players ranged between -7 and 44. There were no duplicate values.

- The following findings were found after conducting the EDA - there were approximately 60% of home games that were won in the series, the variables 'team_pts', 'opp_pts' and 'efficiency' were normally distributed. The variables 'minutes' and 'fgmade', 'minutes' and 'fgatt', 'fgmade' and 'fgatt', 'minutes' and 'points', 'minutes' and 'efficiency', 'fgmade' and 'points', 'fgmade' and 'efficiency', 'fgatt' and 'points', 'fgatt' and 'efficiency', 'made3' and 'att3', 'made1' and 'att1', 'offrb' and 'totrb', and 'defrb' and 'totrr', 'points' and 'efficiency', 'points' and 'made1', 'efficiency' and 'made1', 'fgatt' and 'made1', 'points' and 'att1', 'efficiency' and 'att1', 'fgatt' and 'att1', 'efficiency' and 'defrb', 'fgmade' and 'totrb', 'fgatt' and 'totrb', as well as 'efficiency' and 'totrb', 'efficiency' and 'made3', 'efficiency'

and 'made1', 'fgmade' and 'made1', 'efficiency' and 'made3', and 'fgmade' and 'made3' have a strong positive relationship. The variables 'fgmade' and 'assist', 'made3' and 'assist', 'turnover' and 'assist', as well as 'block' and 'steal' have a strong negative relationship. It was also found that from the Team LAS, Candace Parker was the best player on the basis of highest scored points at 34 followed by Nneka Ogwumike and Kristi Toliver and the best player based on efficiency was Nneka Ogwumike followed by Candace Parker and Kristi Toliver.

- The multiple linear regression model was conducted on the Maya data set which was taken from the WNBA data set. From this, it was found that significant variables were 'turnover', 'fgmade', 'made1' and 'assist' with which a second model was created which gave an adjusted R-squared value of 0.8586 which indicated that 85.86% of the variance in efficiency can be estimated by 'fgmade', 'made1', 'turnover' and 'assist'.

- The residual vs fitted plot depicted that the residuals followed a linear pattern and the linear regression model was appropriate for the data set. From the Q-Q plot, it could be inferred that the data was roughly on the diagonal line. However, the residuals was not enough to declare that it was non-normally distributed. The Scale-Location Plot was used to check the homoscedasticity among the residuals and it was observed that the residuals spread wider on the x-axis. The residuals vs leverage plot displayed that there weren't any outliers in the data. Also, the variance inflation factor (VIF) indicated that the predictors had some correlation as the VIF values for the predictors were less than 1.5.

- The regsubset function was performed where it was found that the model with five independent variables gave the best result. The significant variables were 'assist', 'fouls', 'turnover', 'made1' and 'fgmade' which gave an adjusted R-squared value of 0.8707 which indicated that 87.07% of the variance in efficiency can be estimated by 'assist', 'fouls', 'turnover', 'made1' and 'fgmade'. The data was split into two i.e., the train and test set. From this, we observed that the test data set had a higher adjusted R-squared value of 0.8638 compared to the train data set which was 0.858. Also, the AIC and BIC value of the test set data was lesser than the train set data which indicated that it was a good model to estimate the future efficiency of a player.

- The logistic regression model was conducted on the Team_LAS data set which was taken from the WNBA data set. The data was validated and split into two i.e., the train and test set. From the first model, it was found that the significant variables which were 'team_pts', 'efficiency', 'assist', 'fouls', 'home', 'fgmade', and 'offrb' with which a second model was created which gave an AIC of 268.1.

- To assess which model performed better, the AIC and BIC values of the models were compared, where it was found that the model2 performed better than model1 as it had a lower AIC and BIC values as compared to model1.

- The confusion matrix was conducted for the train and test set from which it was found that in the training data set, the Accuracy was 0.6971 which indicated that from the positive and negative classes, 69.71% of the data were predicted accurately. The Sensitivity aka True Positive rate or Recall is 0.7212 which indicated that from all the positive classes, 72.12% were predicted accurately and the Specificity aka the False Positive rate is 0.6731 which indicated that from all the negative classes, 67.31% were predicted accurately. The Precision aka the Positive Pred value is 0.6881 which indicated that from all the classes that were predicted as positive, 68.81% of the data was positive. The Negative Pred value is 0.7071 which indicated that from all the classes that were predicted as negative, 70.71% of the data was negative.

- From the test data set, the Accuracy was 0.6222 which indicated that from the positive and negative classes, 62.22% of the data were predicted accurately. The Sensitivity aka True Positive rate or Recall was 0.5778 which indicated that from all the positive classes, 57.78% of the data were predicted accurately and the Specificity aka the False Positive rate is 0.6667 which indicated that from all the negative classes, 66.67% of the data were predicted accurately. The Precision aka the Positive Pred value is 0.6341 which indicated that from all the classes that were predicted as positive, 63.41% of the data was positive. The Negative Pred value is 0.6122 which indicated that from all the classes that

were predicted as negative, 61.22% of the data was negative. The ROC curve was plotted and the Area under the curve (AUC) was 0.69 which indicated that the model had a good measure of separability.

References

- Sports Statistics. (n.d.). WNBA 2014 player stats by game. Retrieved on February 5, 2022 from https://sports-statistics.com/sports-data/sports-data-sets-for-data-modeling-visualization-predictions-machine-learning/
- kassambara. (2018, March 11). Regression Model Diagnostics. Retrieved on February 9, 2022 from http://www.sthda.com/english/articles/39-regression-model-diagnostics/161-linear-regression-assumptions-and-diagnostics-in-r-essentials/#example-of-data.
- Bevans, Rebecca. (2020, January 28). Choosing the Right Statistical Test | Types and Examples. Retrieved on February 9, 2022 from https://www.scribbr.com/statistics/statistical-tests/.
- JournalDev. (n.d.). Confusion Matrix in R | A Complete Guide. Retrieved on February 18, 2022 from https://www.journaldev.com/46732/confusion-matrix-in-r.