EDA in R (NBA Statistics)

Malik

2023-12-23

About Project

NBA Exploratory Data Analysis (EDA) Project

This repository hosts an in-depth exploration of NBA (National Basketball Association) data from the 2018-2019 season, employing R and RMarkdown for comprehensive Exploratory Data Analysis (EDA) techniques. The project dives into various facets of data analysis, including data cleaning, outlier detection, handling missing values, correlation analysis, and descriptive statistics.

Datasets Used:

- **Salaries Data:** 2018-19 NBA player salaries.
- **Payroll:** 2019-20 NBA team payroll data.
- **Player Statistics:** Detailed statistics of NBA players from the 2018-19 season.
- **Team Statistics 1 & 2:** Team statistics datasets for the 2018-19 NBA season, providing comprehensive insights into team performances.

Key Highlights:

- **Data Cleaning:** The project initiates with thorough data cleaning processes, ensuring data quality and consistency.
- **Outlier Detection:** Identification and handling of outliers within the datasets.
- **Handling Missing Values:** Strategies and methodologies used to deal with missing data points.
- **Correlation Analysis:** Understanding relationships and correlations between various NBA metrics.
- **Descriptive Statistics:** Summarizing and visualizing key statistical measures for better comprehension of the data.

This project is an illustrative demonstration of Exploratory Data Analysis techniques applied to NBA data, showcasing proficiency in R and RMarkdown for data manipulation, analysis, and visualization. Feel free to explore the code, visualizations, and insights gained from this EDA endeavor.

Import Libraries

```r
####
library(knitr)
library(ggplot2)
library(htmlTable)
library(tidyverse)
require(reshape2)
```

```
library(corrplot)
library(FactoMineR)
library(factoextra)
library(DataCombine)
library(cowplot)
library(jtools)
library(MASS)
library(Metrics)
library(randomForest)
library(caret)
library(dplyr)
library(plotly)
```

Import data

```
#Import data
#salaries data
sal <- read.csv("Data for EDA/2018-19_nba_player-salaries.csv")
#Payroll
payr <- read.csv("Data for EDA/2019-20_nba_team-payroll.csv")
#Player statistics
Pstats <- read.csv("Data for EDA/2018-19_nba_player-statistics.csv")
#Team statistics 1
Tstats1 <- read.csv("Data for EDA/2018-19_nba_team-statistics_1.csv")
#Team statistics 2
Tstats2 <- read.csv("Data for EDA/2018-19_nba_team-statistics_2.csv")
```

Merging the salaries and player statistics

```
df <- merge(Pstats, sal)
#rename variables
names(df) <- c("player_name", "Pos", "Age", "Tm", "G", "GS", "MP", "FG", "FGA", "FGP", "X3P",
"X3PA", "X3PP", "X2P", "X2PA", "X2PP", "eFG", "FT", "FTA", "FTP", "ORB", "DRB",
"TRB", "AST", "STL", "BLK", "TOV", "PF", "PTS", "player_id", "salary")
```

Exploratory analysis

Checking the distribution of variables

This subsection involves examining the data for errors and missing observations. The following table provides an overview of the number of missing observations in each column

```
miss <- data.frame(colSums(is.na(df)))
miss <- cbind(Variable = rownames(miss), miss)
rownames(miss) <- 1:nrow(miss)
names(miss) <- c("Variable","Number of missing observations")
htmlTable(miss)
```

Variable

Number of missing observations

1

player_name

0

2

Pos

0

3

Age

0

4

Tm

0

5

G

0

6

GS

0

7

MP

0

8

FG

0

9

FGA

0

10

FGP

5

11

X3P

0

12

X3PA

0

13

X3PP

45

14

X2P

0

15

X2PA

0

16

X2PP

13

17

eFG

5

18

FT

0

19

FTA

0

20

FTP

41

21

ORB

0

22

DRB

0

23

TRB

0

24

AST

0

25

STL

0

26

BLK

0

27

TOV

0

28

PF

0

29

PTS

0

30

player_id

0

31

salary

0

Some of the attributes contained missing observations. To handle the problem of missing values, the data as shown in the following table was imputed using the mean of the corresponding variable.

Impute missing with the median

```
dfx <- df

dfx$player_name <- NULL
dfx$Pos <- NULL
dfx$Tm <- NULL

#Drop salary since it has a different scale
#dfx$salary <- NULL
# using colMeans()
mean_val <- colMeans(dfx,na.rm = TRUE)

# replacing NA with mean value of each column
for(i in colnames(dfx))
  dfx[,i][is.na(dfx[,i])] <- mean_val[i]

dfl <- dfx
```

```r
df1 <- dfx
df1$player_name <- df$player_name
df1$Pos <- df$Pos
df1$salary <- df$salary
miss <- data.frame(colSums(is.na(df1)))
miss <- cbind(Variable = rownames(miss), miss)
rownames(miss) <- 1:nrow(miss)
names(miss) <- c("Variable","Number of missing observations")
htmlTable(miss)
```

Variable

Number of missing observations

1

Age

0

2

G

0

3

GS

0

4

MP

0

5

FG

0

6

FGA

0

7

FGP

0

8

X3P

0

9

X3PA

0

10

X3PP

0

11

X2P

0

12

X2PA

0

13

X2PP

0

14

eFG

0

15

FT

0

16

FTA

0

17

FTP

0

18

ORB

0

19

DRB

0

20

TRB

0

21

AST

0

22

STL

0

23

BLK

0

24

TOV

0

25

PF

0

26

PTS

0

27

player_id

0

28

salary

0

29

player_name

0

30

Pos

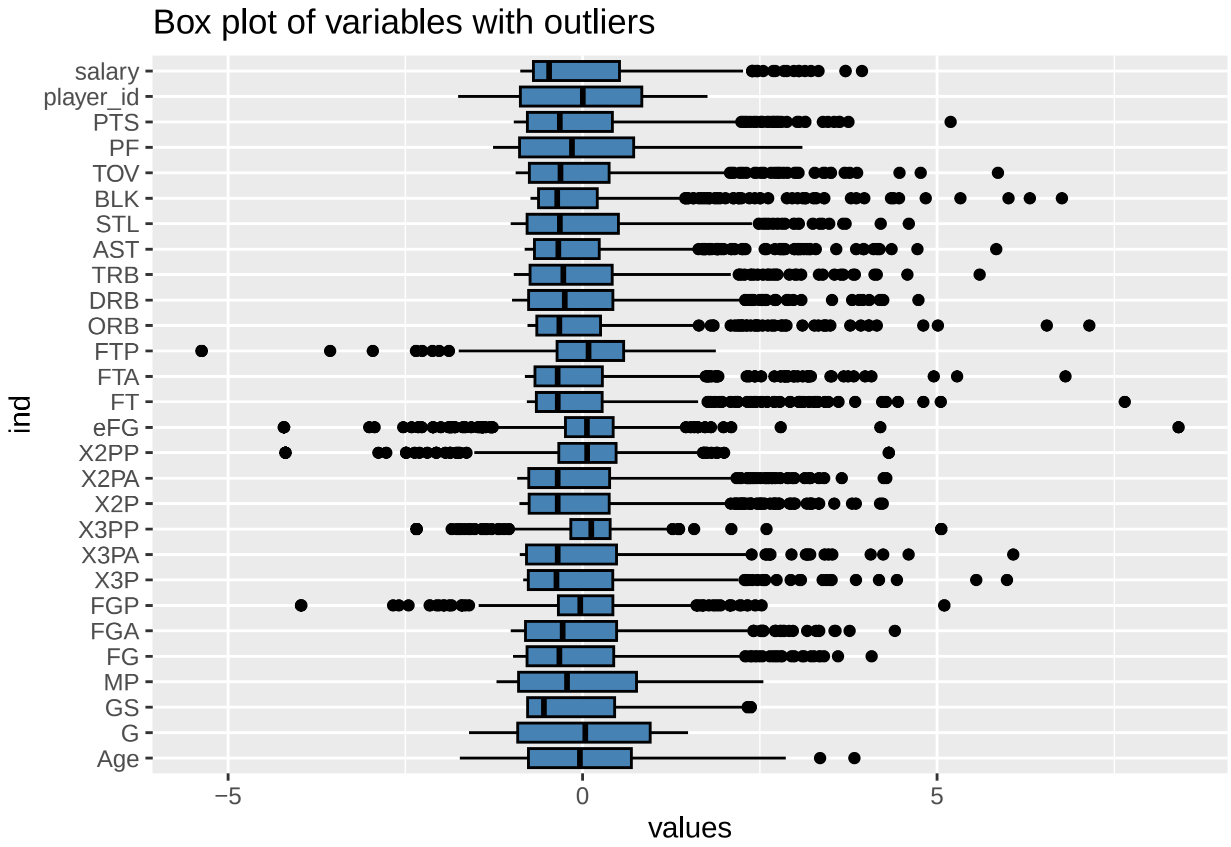
0

Remove duplicates

```r
df <- df %>% distinct()
```

Exploring if there are any outliers (errors) in the data

```r
dfx <- data.frame(scale(dfx))
ggplot(stack(dfx), aes(x = ind, y = values))+
  geom_boxplot(fill='steelblue', color="black") +
  coord_flip()+ggtitle("Box plot of variables with outliers")
```

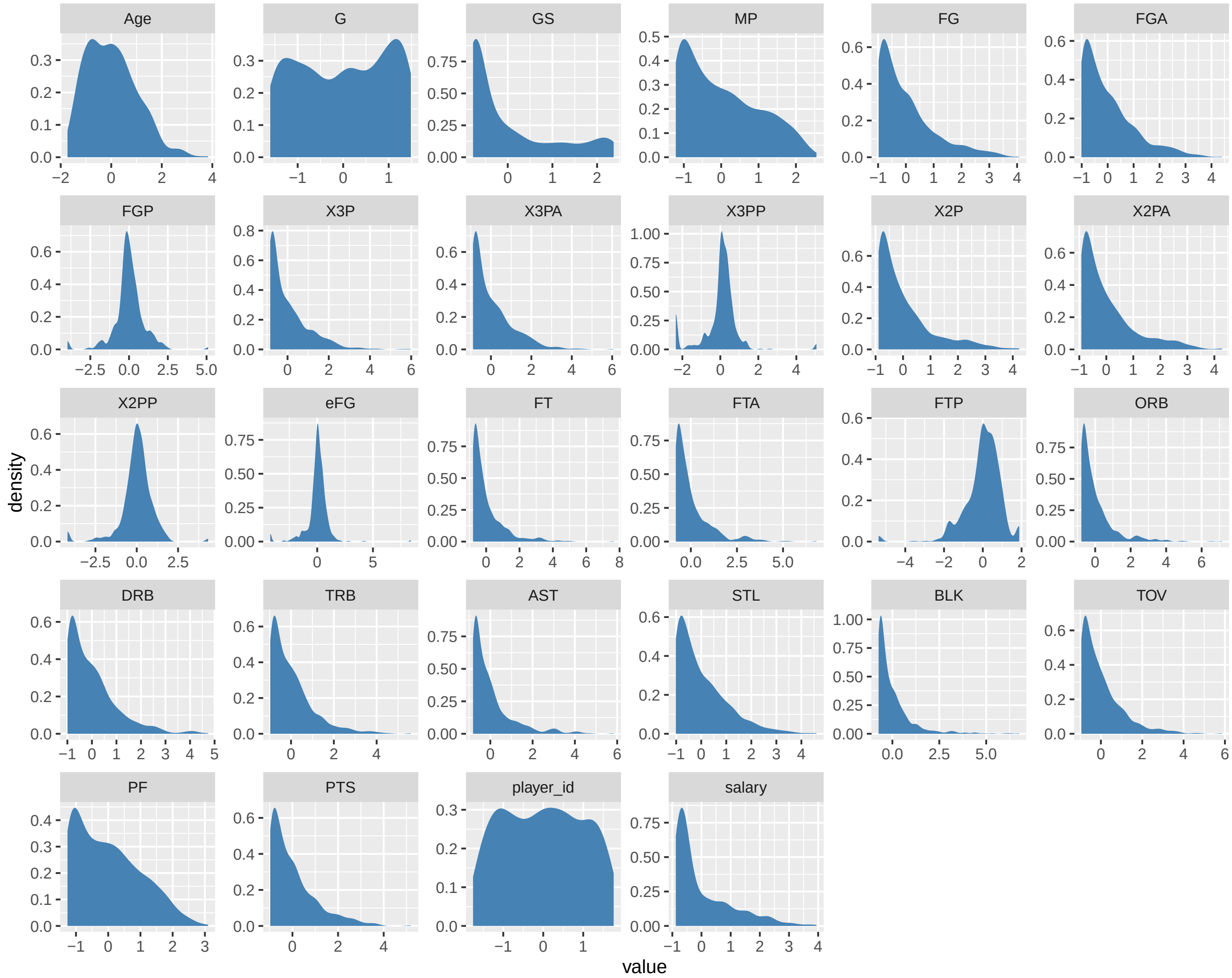


Some of the attributes such as PTS, salary, etcetera as noted in the figure above have got outliers which indicate that some of the players have higher scores and salaries compared to others. In most cases, while these are flagged as outliers, they may be legitimate entries since player salaries and the performance of the players tend to differ depending on factors such as the positions played. As such, the data was retained as it is i.e., no outlier removal was conducted.

Checking the distribution of variables

The following figure shows the distribution of the various attributes in the data

```r
meltdf <- melt(dfx)
ggplot(data = meltdf, aes(x = value)) +
stat_density(fill='steelblue') +
facet_wrap(~variable, scales = "free")
```

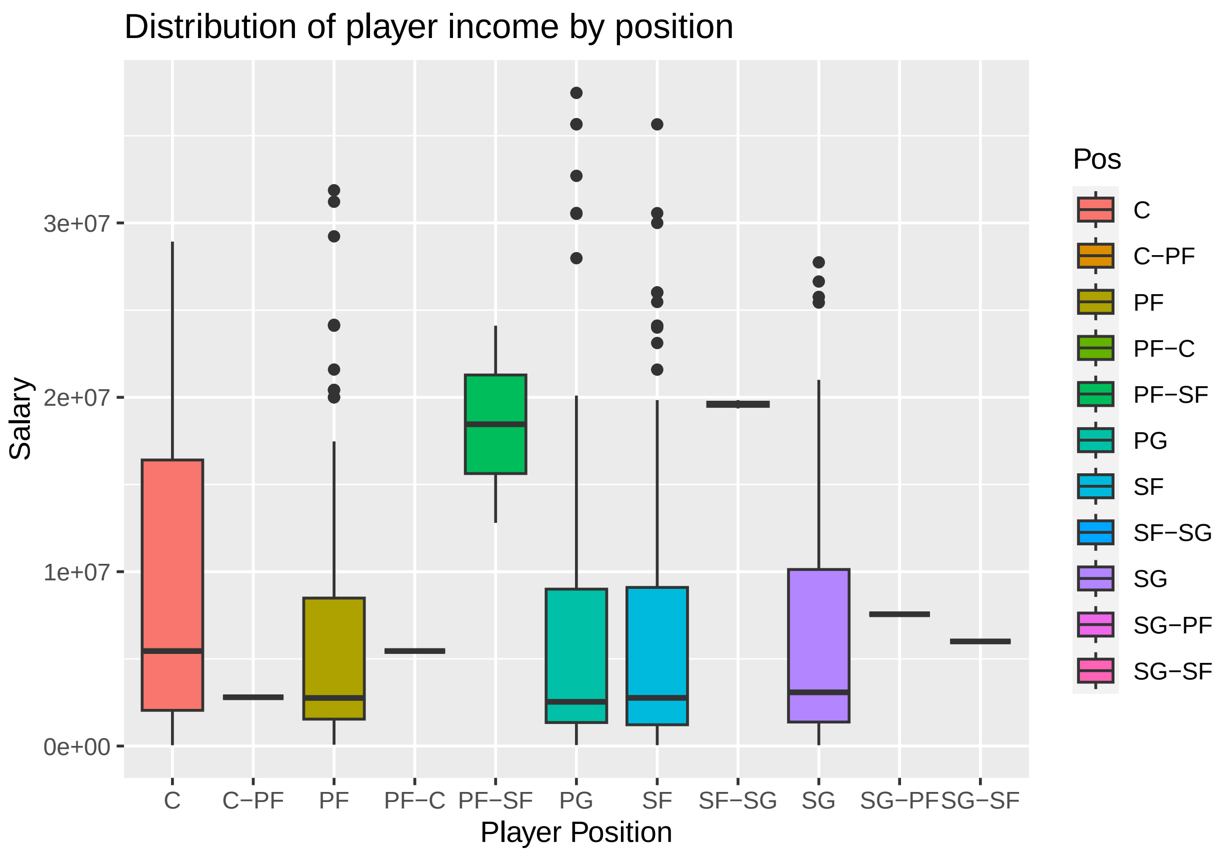


As noted from the preceeding plot, most of the attributes are skewed to the right including salary which indicates that the mean salary is generally Mode < Median < Mean. For instance, this indicates that a few players earn more that the rest which leads to a higher mean salary compared to the median salary.

Distribution of players by position

The following plot shows the distribution of salary by player position. From the following plot, it is noted that the shooting guard (SG)-small forward (SF) and power forward (PF)-small forward (SF) players tend to make higher median salary while Point Guard (PG) and Center (C)-power forward (PF) players tend to earn the least median salary.

```
df %>% ggplot(aes(x =Pos , y = salary, fill = Pos)) +
                geom_boxplot()+
        labs(x="Player Position", y="Salary")+
        ggtitle("Distribution of player income by position")
```



checking for relationships between variables, or differences between groups

Correlation between various player statistics and salary

```
htmlTable(cor(dfx[-27], dfx$salary))
```

Age

0.406829704807545

G

0.311425218166907

GS

0.503186925318159

MP

0.479634217232513

FG

0.530190616855337

FGA

0.522966733274886

FGP

0.150883603049609

X3P

0.384309942259444

X3PA

0.394886436565219

X3PP

0.0601630018119093

X2P

0.509976666280395

X2PA

0.513381153423713

X2PP

0.0976490622712409

eFG

0.131162879072948

FT

0.544757559146876

FTA

0.54608040258682

FTP

0.134283324665561

ORB

0.357498401778281

DRB

0.508423015532346

TRB

0.484454357814877

AST

0.485300965125052

STL

0.470925610938124

BLK

0.330103001392837

TOV

0.520699203943621

PF

0.384630484667724

PTS

0.539437247170817

salary

1

From the correlation table above, it is observed that salary is highly positively associated with Free throws including Free Throws, Free Throws Attempted, and points where, an increase in a players Free Throws, Free Throws Attempted, or points corresponds to an increase in the amount of salary received.