NBA Player Analysis – Stat 133

Brandon Pearl, Nicolas Min, Ryan Ice, and Tram Pham

11/25/2016

Contents

Abstract

The project involves analyzing data about basketball players from the NBA League in the 2015-2016 season. The motivating question of the analysis is: "In the 2015-2016 season, how do the skills of a player relate to his salary?" We argue that regardless of their positions, most players earn a salary which is directly correlated with their skill levels. We observe players' skill levels by calculating each player's EFF index using his basic individual statistics during the season: points, rebounds, assists, steals, clocks, turnovers, and shot attempts. Principal Components Analysis (PCA) is used to assign weight for each term in the original EFF formula. For the earned salaries during the season, we reference the released data from Basketball Reference's Salary reports for individual players. (From analysis of EFF index and salary of individual players, we indeed find that skill levels and salary have a direct positive relationship among the players in the NBA League).

Introduction

In any sports league, performance of a player during the season determines his/her monetary value in the League and eventually determines his salary. In this sense, it is important to understand how closely a player's performance is related to his salary. Analyzing this relationship provides an understanding of how fairly and reasonably have the NBA teams have paid their players during the 2015-2016 season.

To measure each player's performance during the season, we adopt in this paper, the EFF or efficiency statistics, an index that is widely used to measure the performance of NBA players. EFF is derived by a simple formula: `EFF = (TP + TR + A + S + B - MFG - MFT - T) / GP`, where TP = total points, TR = total rebounds, A = assists, S = steals, b = blocks, MFG = missed field goals, MFT = missed free throws, T = turnovers, and GP = games played.

However, one of the issues with the EFF index is that it favors offense oriented players over defense oriented players, as defense players have less chance to score a goal, or catch a rebound compared to the offensive players. To compensate for this drawback, we use a modified EFF that consider the player's positions. We utilize Principal Components Analysis (PCA) giving weight for each term in the original EFF formula.

The modified efficiency, EFF*, is computed as: `EFF* = (w1/s1)*x1 + ... + (w8/s8)*x8`

We predict that this modified efficiency, EFF* have direct positive relationship with the salary that a player receives during the season. The remainder of the paper is organized as follows. In Data we briefly discuss the data table, its structure, and each variable's significance. In Methodology we describe how we obtained and cleaned our data to fit our purpose of study and illustrates how we computed the modified EFF*. Results investigates the relationship between performance and salary from the perspective of the computed EFF*. Finally, in Conclusions we summarize our findings and point out any outstanding trends.

Data

The National Basketball Association (NBA) is the pre-eminent men’s professional basketball league in North America, and is the premier men’s professional basketball league in the world. It has 30 teams (29 in the United States and 1 in Canada). citation

Basketball Reference.com provides professional basketball statistics updated daily for every season. The site also includes sections for coaches, awards, leaders, and the playoffs. It is one of the sites that is under Sports Reference that provide both basic and sabermetric statistics and resources for sports fans everywhere.

From the raw data files of NBA players played in 2015-2016 season provided by Basket Reference, we have created a single csv file, “roster-salary-stats.csv”, containing all variables from Roster, Totals, and Salary, with only one column for the name of the player (the methodology of data acquisition and cleaning will be discussed in the section that follows).

In the csv file, the table contain the following variables: Player, Team, Number, Position, Height, Weight, Birth Date, Country, Experience, College, Rank Totals (within the team), Age, Games, Games Started, Minutes Played, Field Goals, Field Goal Attempts, Field Goal Percentage, 3-Point Field Goals, 3 Point Field Goal Attempts, 3 Point Field Goal Percentage, 2 Point Field Goals, 2 Point Field Goal Attempts, 2 Point Field Goal Percentage, Effective Field Goal Percentage, Free Throws, Free Throw Attempts, Free Throw Percentage, Offensive Rebounds, Defensive Rebounds, Total Rebounds, Assists, Steals, Blocks, Turnovers, Personal Fouls, Points, Points, Rank Salary (within the team), and Salary.

The data table combines three distinct data tables (roster, totals, and salary) and is sorted by the player names. As the purpose of our study is to investigate the relationship between performance and salary, the variables that derive performance index and salary are critical. Among the variables acquired from totals table, the critical variables to compute EFF index (the performance index) are Total Points, Total Rebounds, Assists, Steals, Blocks, Field Goals, Field Goal Attempts, Free Throws, Free Throw Attempts, Games (total # of games). Also, the players’ positions, obtained from the roster table, are also critical in that we later subset the player statistics dataset by positions to calculate separate EFF indices for each position. Needless to say, salary is the most important variable in our study.

All data tables created in this project are in csv format while data summary files are in txt format.

Methodology

Data Acquisition

We obtained our data through webscraping in two phases. First, we accessed the page with all team names and extracted the team abbreviations using the XPath of the desired rows. Then, using the team abbreviations, we concatenated the URL’s to access and we did the following for each page:

1. Find the line containing the table ID we want (e.g. ‘id=“roster” ’).
2. Iterate through the lines following the previously found line until the “
   ″ closing tag is found.
3. Parse the table found between those two bounds (inclusive) into a data.frame to be exported to a CSV file.
4. Repeat for other 2 tables we desire.

In this way, we were able to obtain all of our raw data to be process by our cleaning script.

Data Cleaning

After acquiring the data from Basketball References, we have cleaned three distinct data tables according to the following schema: * Roster table: contains basic information about players for each team (e.g height, birth date, college,etc). We atempted to clean data in which each element in same column obtains the same standard unit or expression. For example, in Experience column, "R" means 0 year experience, so we converted R to 0 so that Experience column contains integers only. Same procedure is applied for the rest. In details, Player and College columns are kept as character. Country is abbreviated in upper case, Height is in feet and Weight is in lbs. Birth Date is in the `YYYY-MM-DD` format and all empty cells are considered as NA. * Stats table: contains all statistical performance data for each player. All of the columns are numeric except for Player which is character. Again, empty cells are filled with NA. * Salary table: contains the salary for each player. We converted the salary columns from `$00000` to `000000`.

After cleaning each table, we then merged the three distinct data tables- roster, totals, salary- of each team and merge them using `merge()` which also sorted them by the player names. We repeated this process for all teams and combined all teams' data using `rbind()`. There were some players who appeared in multiple teams' tables. For these players, we have used "duplicated()" function to only acquire the data from one team.
Data cleaning produces a big data frame, called `roster-salary-stats.csv` which contains roster, stats, salary information for all teams.

EDA

Sink'd Data

For the aggregated dataset, we partitioned the columns into 2 groups:

1. Qualitative data, e.g. Position
2. Quantitative data, e.g. Steals

For the qualitative data, we used dplyr to determine the frequency of each value and printed the resulting 2 column data.frame to `eda-output.txt`.

For the quantitative data, we used R's built-in `summary()` function to generate most of the summary statistics. To get range, we simply subtract `Min.` from `Max.` in the return value from `summary()` and added it to the same result. This was then printed to the same output file, `eda-output.txt`.

Plotted Data

Data Analysis

From the cleaned, "roster-salary-stats.csv" file, first subset the data according to the player's position. Then add extra columns, Missed Free Throws (=Free Throws-Free Throws Attempts), Missed Field Goals (=Field Goals – Field Goal Attempts) and change the number of turnovers into a negative value to match our EFF formula: EFF = (total points + total rebounds + assists + steals + blocks – missed field goals – missed free throws – turnovers) / (games played). From the subset of data table, select variables that are needed to compute EFF. Eliminate the compounding variable, "number of games played" by dividing each variable by the "number of games played". Using "prcomp()", compute PCA1 for each subset. Divide these weights with standard deviation of each variable, re-expressing the weights. Multiply these adjusted weights with each variable (now divided by the number of games). Sum these together to obtain EFF*.

ShinyApp

In order to better visualize and understand the data, we built two shiny apps. The first, titled Salary Statistics by Team, displays a horizontal boxplot that compares the different NBA Teams' salaries. The second, titled Statistic Comparison for Players, presents a scatterplot comparing a range of variables for individual players and displays the variables correlation. These apps serve as an approachable way to further analyze the data; the results of which, will be discussed further in the rest of the report.

RESULTS

TO BE ADDED

CONCLUSIONS

TO BE ADDED