**Using Python to Forecast NBA teams Win Ratios**

**Jasper Wang**

1. **Background**

In the past, the dominant way of judging a team or play’s performance is by watching games. But the new trend is data analysis that with the help of the formulas and figures, the coach can adjust their game strategies by seeing how many three shots their opponent makes per game or how is the most efficient rim protector in their opponent’s team. Even though there are [voices](https://youtu.be/NZf9NFaCQHQ) against this new trend, data analysis has been heavily incorporated into NBA games.

Recently, analysts and sports saw the potential of advanced metrics in measuring values of players and teams which can’t be exclusively reflected from the stats sheet. For instance, a player who focuses on putting pressure on the ball and hustling in every contact will have a big impact on the defense side, but, from traditional statistics like rebounds, it’s really hard to measure this player’s value. With metrics like the opponent’s PTS per 100 possessions, it becomes possible for people to measure the team’s defense when this player is on and off the court.

1. **Research Topic**

The research topic of this final project is combing simple statistics into a new metric **Team Efficiency Rating(TER)** which can be used to measure how good a team is and forecast win ratios of NBA teams based on that metric.

In my model, the core metric is **Player Efficiency Rating(PER)** developed *by John Hollinge*r from ESPN. PER is both a pace-adjusted and per-minute measure, which is derived from basic player statistics. Adjusting for the pace of teams makes sure that players on slow-paced teams are not penalized for having slightly lower numbers. In addition, as a per-minute measure, PER is one of the most effective ways to compare players’ performances who have huge disparities in minutes played in games.

With the individual player’s PER, the teams PER can be calculated through a certain method. By comparing a team’s win ratio and this team’s PER, I can get the correlation between these two metrics. This correlation should be a linear regression that can be used to predict teams’ win ratios. In this project, I’ll calculate every team’s TER in the past 10 regular seasons.

1. **Method** (see definition for each term in *[Glossary](http://stats.nba.com/help/glossary/)*)

Unadjusted PER is calculated as follows:

uPER = (1 / MIN) \*

[ FG3M

+ (2/3) \* AST

+ (2 - factor \* (TEAM\_AST / TEAM\_FGM) \* FGM

+ (FTM \*0.5 \* (1 + (1 - (TEAM\_AST / TEAM\_FGM)) + (2/3) \* (TEAM\_AST / TEAM\_FGM)))

- VOP \* TOV

- VOP \* %DREB \* (FGA - FGM)

- VOP \* 0.44 \* (0.44 + (0.56 \* %DREB)) \* (FTA - FTM)

+ VOP \* (1 – %DREB) \* (REB - OREB)

+ VOP \* %DREB\* OREB

+ VOP \* STL

+ VOP \* %DREB \* BLK

- PF \* ((LG\_FTM / LG\_PF) - 0.44 \* (LG\_FTA / LG\_PF) \* VOP)]

Less obvious terms:

factor = (2 / 3) - (0.5 \* (LG\_AST / LG\_FGM)) / (2 \* (LG\_FGM / LG\_FTM))

VOP = LG\_PTS / (LG\_FGA - LG\_OREB + LG\_TOV + 0.44 \* LG\_FTA)

%DREB = (LG\_RRB - LG\_OREB) / LG\_REB

PACE\_ADJUSTMENT = LG\_PACE/ TEAM\_PACE

After uPER is calculated, an adjustment must be made for the team's pace. The pace adjustment is:

aPER = (PACE\_ADJUSTMENT) \* uPER

The the TER is calculated as:

TER= ( aPER1 \* MIN + aPER2 \* MIN + … + aPERn \* MIN) / TEAM \_MIN

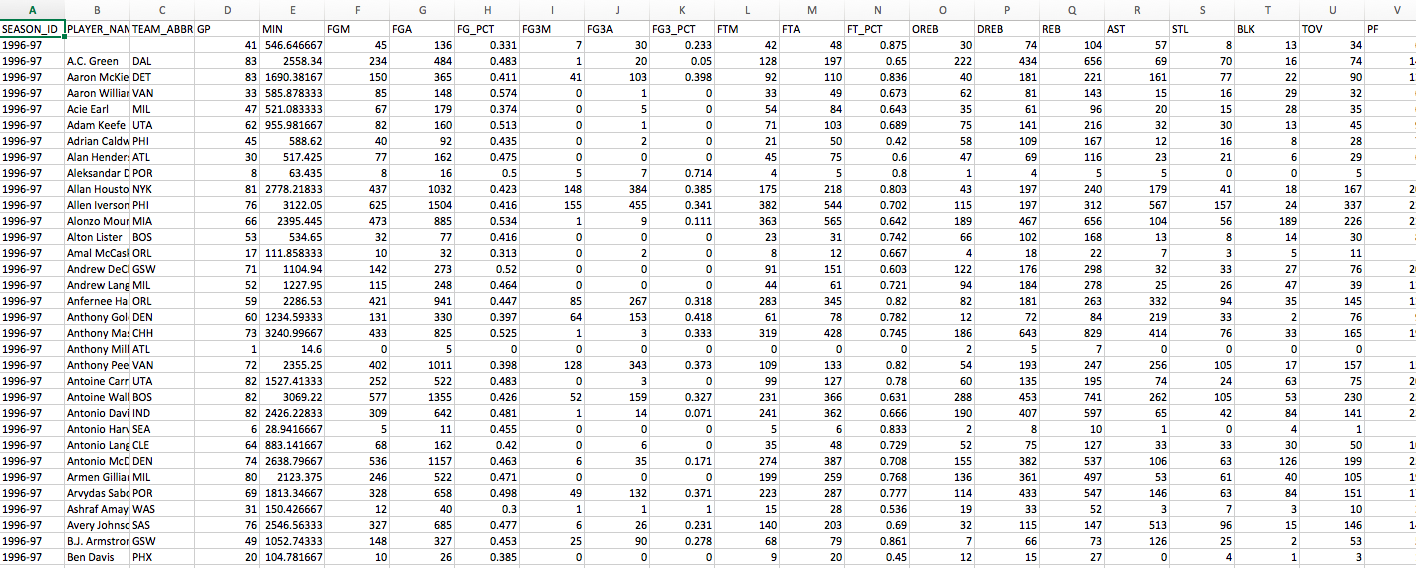
1. **Data Source**

[NBA official stats site](http://stats.nba.com/)

The NBA stats doesn’t provide any API for the public to access their data. Luckily, there is an excellent [blog](http://www.gregreda.com/2015/02/15/web-scraping-finding-the-api/) showing how to explore NBA’s data with python. By using the developer tools embedded in browsers like Firefox and Chrome, I got all the XHR files which contain the data in the format of JSON as well as URLs that I can use to access the data.

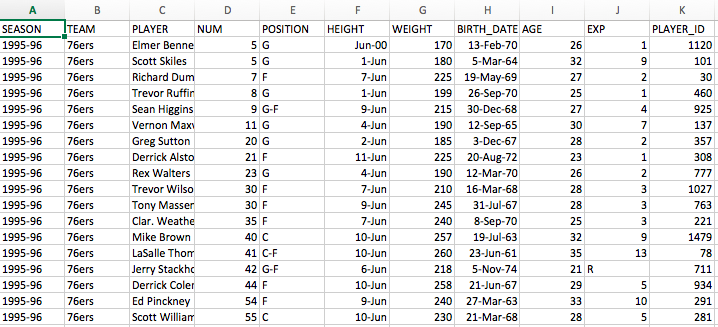
The data used in this project are organized into four categories:

1. Player Stats by Year



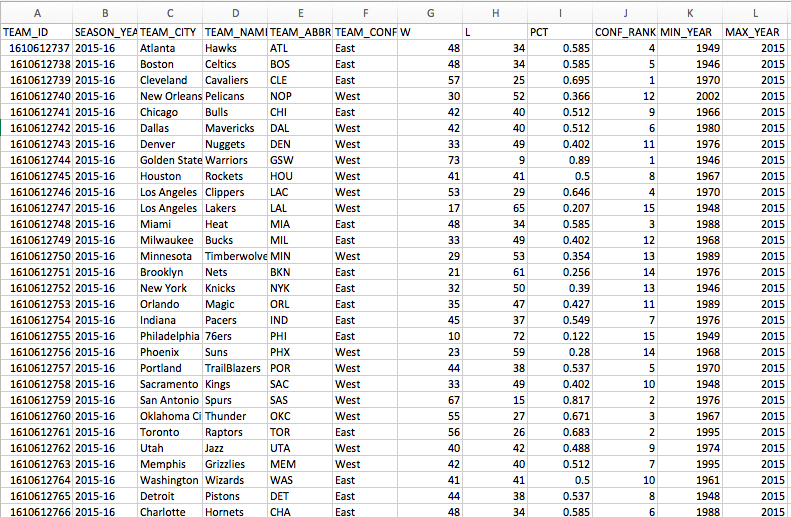
4-1 Player Stats by Year

1. Team Roster by Year and Player ID



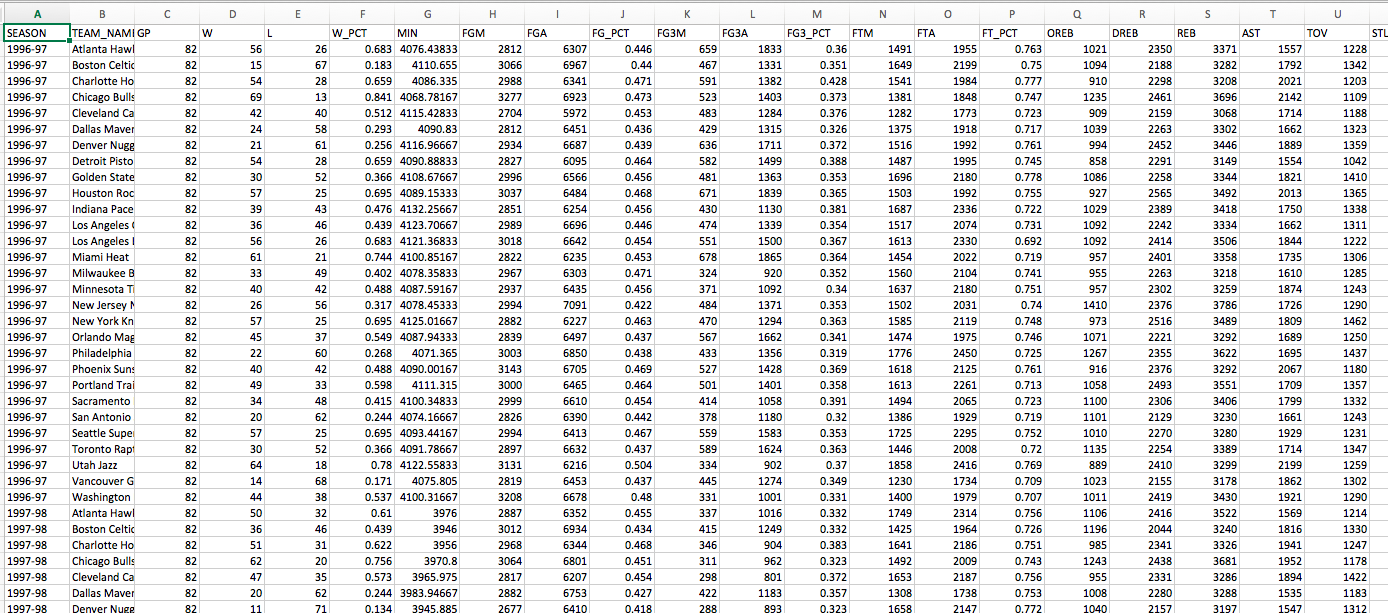
4-2 Team Roster and Player Id

1. Team Name and ID

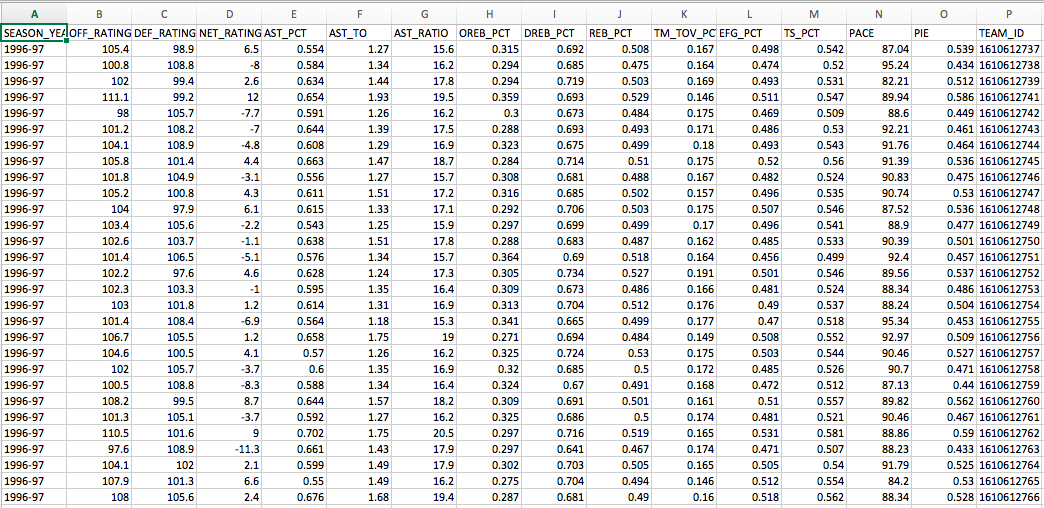


4-3 Team Name and ID

1. Team Basic and Advanced Stats by Year



4-4-1 Team Basic Stats by Year



4-4-2 Team Advanced Stats by Year

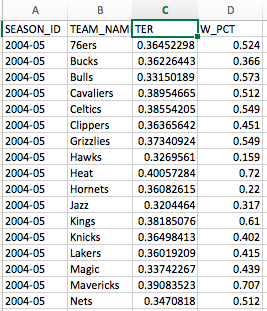
Among all the data mentioned above, two sets of data are very crucial to this project: player id and player names; team id and team names. Only with these two sets of data, I can navigate through the stats of every team and player.

1. **Coding**

I used Requests Library to get the access of the data and the CSV module to write the data into csv files. The specific code can be divided into six processes:

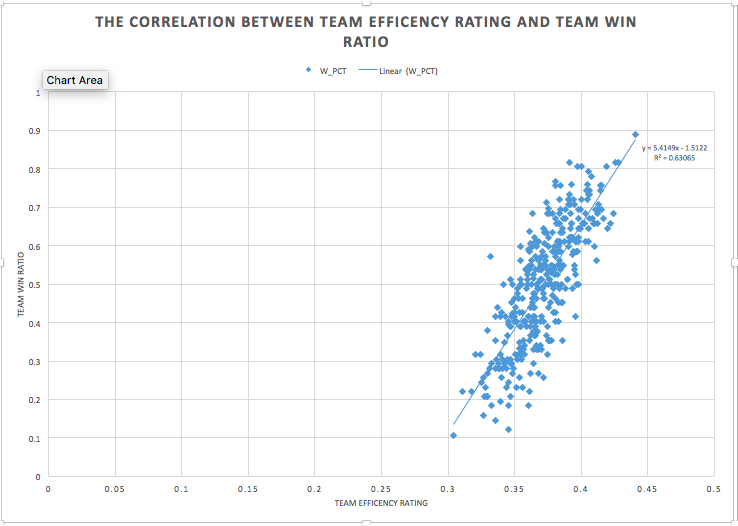
1. Get Team Name and ID
2. Get Team Roster and Player ID
3. Get Player Stats by Year
4. Get Team Basic and Advanced Stats by Year
5. Get Team Efficiency Rating
6. Output
7. **Analysis Result**

The final data are outputted as follows:



6-1 Final Data Set

With this data set, I took TER as the X-axis and W\_PCT as Y-axis and got the scatter plot as follows:



6-2 The Correlation between TER and Team Win Ratio

As we can see in this plot, the team has higher TER tends to have higher win ratio. The linear trend, which yields a strong R square of 0.63065, shows a strong correlation between TER and Team Win Ratio. With this correlation, a team’s win ratio can be forecasted based on this team’s TER.

**Reference**

Yuanhao, Yang (2015): *Predicting Regular Season Results of NBA Teams Based on Regression Analysis of Common Basketball Statistics.* University of California at Berkeley.

HOLLINGER, JOHN: “*What is PER?”* http://sports.espn.go.com/nba/columns/story?columnist=hollinger\_john&id=2850240