

# Open Source Data Science Pipeline for Developing “Moneyball” Statistics in NBA Basketball

Simon Zou

University of California, Los Angeles  
*simonzou@ucla.edu*

**Abstract**—In recent years there has been a rapid increase in the usage and adoption of rigorous statistical analysis among National Basketball Association (NBA) teams, media members, and fans[1]. Increasingly the results and metrics produced by such analysis drive the decisions and narrative coverage of this multi-billion dollar industry. Currently, however, most of the most advanced tools, methodologies, and most predictive statistics are proprietary, being either a trade secret of individual teams or media organizations such as ESPN. We outline in this paper tools that collect the data, and implement the methodologies (through publicly available information and reverse engineering) to produce a metric called regularized adjusted plus-minus (RAPM), which measures attributes of basketball previously thought unmeasurable such as defensive contribution.

## I. MOTIVATION AND BACKGROUND

### A. “Moneyball”

New York Times author Michael Lewis published a book in 2003 entitled *Moneyball: The Art of Winning an Unfair Game*. The book told the story of Major League Baseball’s (MLB) 2002 Oakland A’s teams, which had a collective payroll of \$44 million dollars compared to bigger market teams like the New York Yankees, who had a collective payroll of \$145 million and yet both teams won the same amount of games that season[2].

Oakland had more financial constraints than bigger market teams and as a result started to rely on more rigorous statistical analysis to determine what was the best predictor of wins. This strategy ran counter to the conventional wisdom of insiders (players, managers, coaches, scouts, and the front office) at the time, who believed that only domain experts such as players and coaches could accurately evaluate talent using subjective observations. The Oakland A’s success proved that this mode of thinking was flawed and outdated.

In 2009, Lewis turned his attention toward basketball and published a magazine article entitled *The No-Stats All-Star*[3]. It profiled player Shane Battier, a player who by traditional observations and numbers looked unimpressive, but more sophisticated analysis proved him to be a big contributor to winning basketball. It also profiled Houston Rockets general manager Daryl Morey, the general manager of the Houston Rockets. Morey’s background was a computer science degree from Northwestern and MBA from MIT and thus did not have the traditional player/coach/scout background.

Similar to the Oakland A’s, he and his front office used advanced statistical tools to identify and acquire undervalued players. Similar to the Oakland A’s, he encountered skepticism from the wider basketball community regarding the predictive value of numbers over the subjective observations of experts, also known colloquially as the “eye test”.

### B. The Story Today

Finally, similar to the Oakland A’s, Morey’s approach and methods have since been validated. Every NBA team now has an analytics department that is used in the decision making process and media partners like ESPN also has staff dedicated to creating and publishing metrics and predictions based on those metrics. They drive decisions and discussions at all levels, from the front offices of teams to fans on social media.

However, while literacy and acceptance of these metrics has become more widespread, the matter in which they are produced and how they work has not. This tends to be because those with technical expertise tend to get hired either by the NBA teams or media organizations. Currently the most accurate predictive stat is called Real Plus/Minus (RPM), created by Jeremias Engelmann, who worked for several NBA teams before being hired by ESPN, whom he currently works for. It is cited by journalists who vote on awards that have multi-million dollar consequences to fans trying to win an argument on social media alike. Despite its influence, RPM is still a black box. From publicly available lectures and articles, we know it is a modified version of another stat called Regularized Adjusted Plus/Minus (RAPM)[4]. This paper will chiefly deal with how to collect the necessary data and then calculate this number, which is not widely available.

## II. A SURVEY OF NBA “ADVANCED” METRICS

This section will summarize in more detail the current landscape of publicly available quantitative work and study in the NBA. This field or general category of work has several names within the community, including “advanced stats”, “advanced analytics”, and “advanced metrics”. The words “stats” and “metrics” in this context refer to specific numbers attributable to a player or team, such as how many points they scored per game. In the community there has developed a distinction between “traditional” (or “box

score") statistics and modern "advanced" statistics. We will begin with defining each and how "advanced" stats are different. We then define and evaluate the leading advanced metrics and present a framework for developing new ones.

### A. Traditional Box Score Numbers

Like baseball, basketball has had a long tradition and history of recording numbers. If nothing else, the score has to be measured and counted accurately to know who won the games. The following are some traditional box score statistics and their definition:

- Field Goal** - A basket scored on any shot or tap other than a free throw, worth two or three points depending on the distance of the attempt from the basket.
- Point** - Number value tallied when a player successfully throws the ball into his basket
- Rebound** - A recovery of the ball after a missed field goal attempt
- Assist** - A pass leading directly to a made basket and points
- Block** - A deflection a field goal attempt from an offensive player to prevent a score
- Steal** - Occurs when a defensive player (legally) causes a change in possession by some action, such as deflecting and controlling, or by catching the opponent's pass or dribble of an offensive player.

NBA FINALS - GAME 7, CLE wins series 4-3														
Final														
	1	2	3	4	T	CLE	23	19	33	18	93			
	GS	22	27	27	13	GS	73-9	73	9	89	89			
Starters	MIN	FG	3PT	FT	OREB	DREB	REB	AST	STL	BLK	TO	PF	+/-	PTS
L. James SF	47	9-24	1-5	8-10	1	10	11	11	2	3	5	1	+4	27
K. Love F	30	3-9	0-3	3-4	4	10	14	3	2	0	1	2	+19	9
T. Thompson C	32	3-3	0-0	3-4	0	3	3	0	0	2	0	4	+2	9
K. Irving PG	43	10-23	2-5	4-4	3	3	6	1	1	1	2	3	+10	26
J. Smith SG	39	5-13	2-8	0-0	0	4	4	2	1	0	2	3	+7	12
Bench	MIN	FG	3PT	FT	OREB	DREB	REB	AST	STL	BLK	TO	PF	+/-	PTS
R. Jefferson SF	26	1-4	0-0	0-0	1	8	9	0	1	0	0	1	-8	2
M. Williams PG	5	1-3	0-1	0-0	0	0	0	0	0	0	1	-5	2	
I. Shumpert SG	19	1-3	1-3	3-3	0	1	1	0	0	0	0	-9	6	
C. Frye PF													DNP-COACH'S DECISION	
T. Mozgov C													DNP-COACH'S DECISION	
M. Dellavedova PG													DNP-COACH'S DECISION	
D. Jones SG													DNP-COACH'S DECISION	
J. Jones SG													DNP-COACH'S DECISION	
Team	33.82	6.25	21.25	9	39	48	17	7	6	11	15	93		
	40.2%	24.0%	84.0%											

Fig. 1. A Sample Box Score from the 2016 NBA Finals via ESPN.com

For each game, these numbers are recorded into a table called the box score. However, these measurements only track specific, discrete events. Any other kind of contribution, for example a player being so skilled or so large that he commands the attention of two defenders or screens, a blocking maneuver that allows teammates to get open for shots is not counted. Something that has proven particularly difficult to measure is how good a player is on defense, or how good he is at preventing the other team from scoring, in contrast to the relatively easy task of measuring how

good someone is at scoring, which are discrete tasks largely attributable to one player, whereas good defense requires more group coordination and teamwork.

Early naive attempts at measuring and ranking how good one player is compared to another, such as the NBA's "Efficiency" statistic [5] would be to sum up their positive box score contributions on a per game basis and subtract their negative ones (e.g. If a player averages 27 points, 7 rebounds, 7 assists, 2 steals, and 2 blocks per game, summing those up would result in a value of 45, which would then be subtracted from for every missed shot).

### B. Advanced Stats - The NBA Learns More Math

Another limitation of traditional numbers is that they did not adjust for how long the player is in the game or how many opportunities they have with the ball. Much of the initial work in developing better, more predictive statistics was to simply do some division with the traditional numbers, and rate-adjusting those numbers so there's a sense of how productive a player is per minute or per possession. A **possession** is defined as the time a team gains offensive control of the ball until it scores, loses the ball, or commits a violation or foul.

The current most comprehensive source for NBA statistics, both traditional and modern is basketball-reference.com. We list three metrics under their "Advanced" tab that have gained acceptance within the community as outperforming traditional statistics in measuring performance.[6].

- PER (Player Efficiency Rating)** - It calculates a weighted sum, giving each box score contribution a weight based on how much of a possession they are worth (e.g. a steal is worth 1 possession, a defensive rebound was determined to be worth approximately 0.7 possessions and so on). It adjusted for minutes played and for each NBA season normalized the metric for 15.0 to be the average [7]. Developed by John Hollinger, former ESPN contributor and current VP of Basketball Operations for the NBA team Memphis Grizzlies and outlined in his articles and books [7].
- WS/48 (Win Shares / 48 Minutes)** - Using how many points produced relative to the league average and a player's team wins, Win Shares is an attempt to give credit for team success to individual players. Developed by Daniel Myers, from the concept of Win Shares in baseball developed by Bill James as well as the concepts from the book *Basketball On Paper* by Dean Oliver.
- TS% (True Shooting Percentage)** - The traditional measure of scoring efficiency was field goal percentage, which is field goals made divided by field goals attempted. True Shooting percentage incorporates how well players shoot on free throws as well as the fact that some field goals are worth three points as compared to two. Put another way, if multiplied by 2, it is an estimation of how many points a player is likely to score everytime they take a shot.  
The formula is given below, where  $PTS$  is points scored,  $FGA$  is Field Goals Attempted, and  $FTA$  is

Free Throws Attempted.

$$TS\% = \frac{PTS * 100}{2(FGA + (0.44 * FTA))} \quad (1)$$

### C. Plus/Minus Statistics

We now discuss the another grouping of rate adjusted stats that use something other than traditional box score statistics.

Taking a leaf from hockey[8], NBA box scores have begun incorporating a statistic called plus/minus, which measures the difference in scoring margin that happens between the two teams when a player is on the court. If Lebron James enters the game while his team is trailing by 2, and leaves five minutes later with his team up by 6, his plus/minus for that stint would be a +8. The biggest draw back of this measurement is collinearity, or “What if there’s another player that plays all his minutes with Lebron James and thus gets credit simply for being on the court with good players?” To deal with this problem, more sophisticated versions of plus/minus have been developed.

We will first discuss adjusted plus/minus (APM) and regularized adjusted plus/minus (RAPM). We then discuss the relationship of these stats with box plus/minus (BPM), another metric listed in the “Advanced” section of basketball-reference’s player pages. Finally, we discuss ESPN’s industry leading Real Plus/Minus.

1) *Adjusted Plus/Minus (APM)*: The basic idea of APM is to adjust plus/minus to account for other variables, namely who else is on the court at the time on both sides. To do this, we apply a linear regression, treating all the players as independent variables and using possessions in a game (a possession is defined as a continuous block of time where one team has control of the ball) as observations to set up a matrix or system of linear equations, with the outcome of the possession being the result (i.e. an integer in the range [0, 4] signifying the number of points scored).[4]

Figure 2 shows a simplified version of a matrix designed and created from a play-by-play log (assuming only three players on the court per team instead of the usual five). The play-by-play is parsed so that possessions are grouped together into a single row and the players that are on the court at the time are known. Each row of the matrix  $A$  is then a possession in a game and the columns are indicator variables for whether a player was on the court on offense or defense. We include each player twice in every observation so that we are able to note and separate their impact on defense and offense when the system of equations is solved. Thus to calculate APM, we solve the equation

$$Ax = b \quad (2)$$

using Ordinary Least Squares (OLS):

$$x = (A^T A)^{-1} A^T b \quad (3)$$

2) *Regularized Adjusted Plus/Minus*: APM, while descriptive, suffers from the overfitting of ordinary least squares. To counter overfitting and reduce noise, we calculate

Regularized Adjusted Plus/Minus, or RAPM, by using a regularized, or ridge, regression by modifying the equation with a penalty term  $\alpha$ , found by cross validation.

$$x = (A^T A + \alpha I)^{-1} A^T b \quad (4)$$

3) *Box Plus/Minus (BPM)*: Box Plus/Minus (or BPM) is not technically a plus/minus stat. It is the result a regression against large data set of RAPM values with weights assigned to various box score statistics [9]. It is thus a method used to estimate RAPM using a combination of traditional and more advanced numbers. The methodology is public and final results are public on basketball-reference.com, but neither the code nor the dataset used to generate it are available anymore.

4) *ESPN’s Real Plus/Minus (RPM)*: ESPN’s Real Plus/Minus (or RPM) was developed by Jeremias Engelmann, and is currently used by the media company for its forecasting and prediction articles. It has gained acceptance within statistics-minded media and fans and the current best “one-number” metric for evaluating player performance, particular with respect to defense. It is proprietary, but publicly available documents and discussions show that it is an RAPM regression, but with a prior that includes factors such as an aging curve, player height, and box score statistics.

## III. WORK, EXPERIMENTS, AND RESULTS

Currently we have several classes of NBA “Moneyball” statistics as described below that are public in some way, as summarized by 3.

We aim with this work to move as much as possible to the center, having created a pipeline and dataset for calculating RAPM, which allow for more other kinds of analysis as well. To that the goal of this work is twofold - to publicize a measurement that rivals the current industry leaders, as well as the methods for calculating it and other measurements.

### A. Evaluation Methodology

Somewhat ironically, the evaluation of these all these different statistics is not very evidence-based. Media publications, rankings, predictions, and player evaluations will cite win shares and ESPN’s RPM as evidence but how the different metrics actually rank against each other is largely subjective. The public acceptance of a statistic relies largely on whether it gets enough obvious things right (“Lebron James and Steph Curry are top players”).

There is, however, a methodology of evaluating the predictive accuracy of each statistic, as developed by Paine and Rosenbaum[10]. For a given team in a given season (for example the 2017 Golden State Warriors), a weighted average is calculated based on the player’s rating from previous years and the current year’s minutes. An  $r^2$  correlation value is then calculated between the weighted average and the team’s wins. Thus we expect high  $r^2$  scores for the current year, and then decreasing over time. “Better” stats will have higher correlations and degrade less quickly though that carries with it an assumption that professional players largely stay the

Play-By-Play Log	Warriors						Cavaliers						
	Offense						Defense						
	S. Curry	D. Green	K. Durant	L. James	K. Love	T. Thompson	S. Curry	D. Green	K. Durant	L. James	K. Love	T. Thompson	Result
Stephen Curry makes 2pt jumper	1	1	0	0	0	0	0	0	0	1	1	0	2
Lebron James misses layup	0	0	0	1	1	0	1	1	0	0	0	0	0
Draymond Green grabs defensive rebound	0	1	1	0	0	0	0	0	0	1	1	0	3
Substitution: Kevin Durant for Stephen Curry													
Kevin Durant makes 3 pt jumper													

Fig. 2. Sample Design Matrix and Results Vector based on Play-by-Play

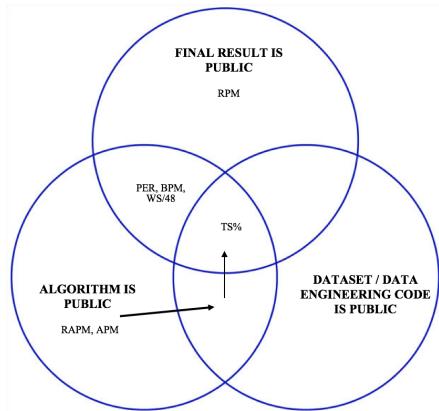


Fig. 3. Venn Diagram of how Public Advanced NBA Statistics Are. Thick arrow demonstrates work done in this report. Thin arrow represents work yet to be done

same year to year. Since this is generally not the case, all stats will degrade fairly quickly in an absolute sense.

We present here results for evaluating a number of metrics with this methodology. For more information on how the data was collected and processed, see the *Data Engineering* section.

### B. Evaluating Current Metrics

Fig. 4 displays both chart and table with the correlation values of the “Moneyball” metrics compared with team wins. Wins itself and the team’s win total from previous seasons is also used here as a control. We see that all metrics, save for True Shooting percentage, which is the most limited and simple stat among the given metrics, correlate reasonably well to winning with the current year’s data. However using previous years’ data to estimate current year’s wins is progressively less accurate the more years removed. Notably, APM is completely uncorrelated with winning outside of the current season, which matches earlier assertions that APM, which uses OLS, suffers from overfitting. We remove it from later figures.

We also specifically draw attention to the performance of RAPM, the metric calculated from scratch by the pipeline created in this work, and find that it compares favorably to current leading public metrics.

Figure 5 and Figure 6 show similar charts and tables over larger data sets. We can see that the stats based purely on box score, PER and Win Shares, do rather poorly by this measure

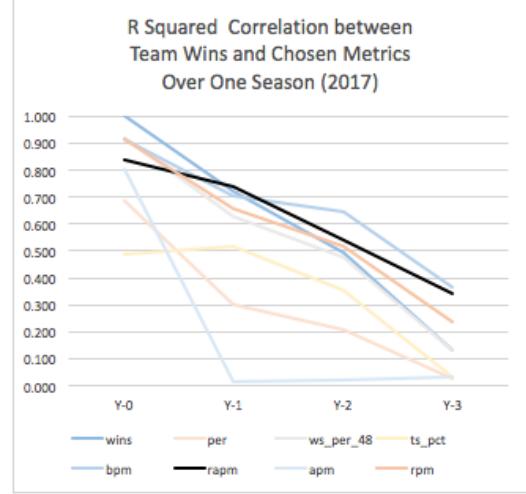


Fig. 4. Chart and Table with R Squared Correlation Values between Team Wins and Chosen Metrics Over One Season (2016-17)

and are not very predictive, even just one season out. This explains the move towards plus/minus type statistics in recent years. Interestingly, BPM tends to perform better than the plus minus stats it is trying to approximate. It’s again notable that RAPM performs well relative the industry standards. All metrics (except for TS% and APM for reasons previously stated) beat the control of simply using the previous years’ wins.

It should be noted that there’s also been advancement in the metrics that measure how strong a team really is. Wins are not always reflective of that and it has been found that other factors are stronger indicators of success. These include margin of victory, net rating, pythagorean wins, and SRS. In an effort to not crowd this report with even more definitions, we use wins as the metric to be correlated against.

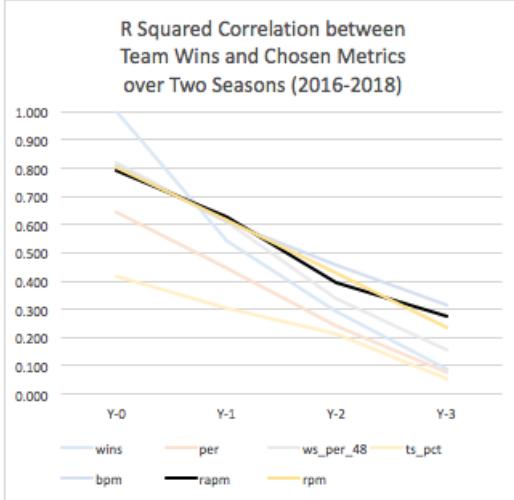


Fig. 5. Chart and Table with R Squared Correlation Values between Team Wins and Chosen Metrics Over Two Seasons (2016-18)

#### IV. DATA ENGINEERING

It's often been said that most data scientists spend only 20 percent of their time on actual data analysis and 80 percent of their time finding, cleaning, and reorganizing huge amounts of data. This was certainly the case with this project and involved many steps including:

- Scraping multiple data sources
- Reconciling the data from multiple sources
- Importing the data into a NoSQL database
- Processing the data into the necessary format
- Converting the processed data into a matrix
- Calculating and storing of results

To hopefully save future parties 80% of the work, and to increase the transparency, development, and research into these numbers, the code for each part of this pipeline is publicly available and open source [11].

Tools used:

- **Anaconda** - python package for data science
  - **requests** - module for http requests
  - **numpy** - module manipulating matrices and vectors
  - **requests** - module for http requests
  - **scipy** - module for doing basic statistics like  $r^2$  values
  - **sklearn** - module for machine learning
- **JsonEditorOnline** - webpage for quickly inspecting JSON data
- **MongoDB** - NoSQL database
- **Robo 3T** and **Compass** - GUI interfaces for MongoDB

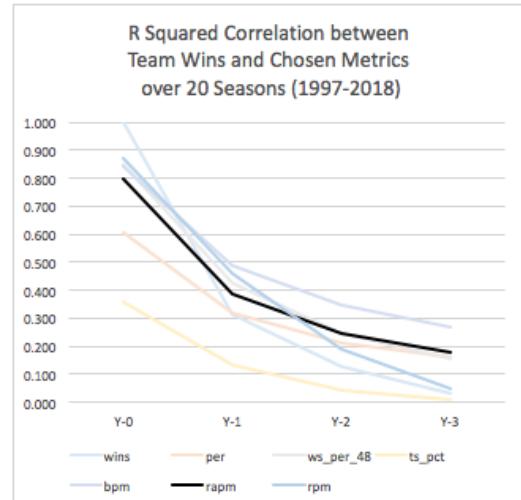


Fig. 6. Chart and Table with R Squared Correlation Values between Team Wins and Chosen Metrics Over Twenty+ Seasons (1997-2018)

- **pickle** - python module used to save data objects to file too big for MongoDB
- **pymongo** - python library for interfacing with MongoDB

##### A. On Small and Medium Sized Data

To hold every possession and all player and team info used in this analysis for the last two decades in a relatively space-inefficient format only takes approximately 1.8 GB, 2.4 GB if you count the pickles too large to store in the database. Big Data, this is not (yet). However, just as small and medium sized businesses have real challenges and help power the economy, so too can small medium sized data still do some interesting things.

The bulk of the data comes from the possession and play-by-play data. There are over 1k games in a normal NBA season, and each game has on average 200 possessions, resulting in 20-25k possessions for each year of two decades of data. This data is processed into a more compact form but essentially duplicated into a form that can be processed into a design matrix and then the design matrix itself.

##### B. Scraping multiple data sources

Data for this was acquired from three separate sources. Play-by-play data was scraped from stats.nba.com, player and team "advanced" data was scraped from basketball-reference.com, and RPM data was scraped from espn.com. Nba.com has a semi-open API in that it's public and returns JSON but not officially documented and so to scrape multiple

seasons of play-by-play data, a combination of browser developer tools and unofficial documentation was used to find the proper endpoints for querying for all the games in a given season and then the play by plays for that season. Data was scraped from other sites using the python library pandas' ability to read and parse tables on a web page.

### C. Reconciling the data from multiple sources

The main challenge that arises from collecting data from multiple sources is how to combine them as they all use different player ids and also could have potentially different representations of team names and player names. We use basketball-reference's player page as a base and compare other site's names against it. We match on a stub (a player's names with special characters and spaces stripped), team name, and season played to uniquely identify a player's season. Differences in the representation of team codes were detected and hard coded (e.g. the New York Knicks team abbreviation is NY on ESPN is NYK on basketball-reference).

Basketball-reference does not use player suffixes as part of a player's name but the other two sites do, so to resolve against it, any "Jr.", "II", "III", "IV" had to be dropped. Additionally about several dozen other hardcoded cases had to be detected of a player being called one thing on one site and something else on another (e.g. Ike vs Isaac, Vince vs Vincent, Slava vs Stanislav).

The end result of this is a unique one that has the unique player ids for multiple sites.

### D. Importing the data into a NoSQL database

A NoSQL database was used here largely for speed of development in that it facilitated the importing of data from API calls directly, mapped well to python dictionaries, and allowed for easy adding of more columns when appropriate. While no schema is strictly required, collections had to be designed with similar principles in terms of normalization and avoiding duplicate data while providing fast lookups.

### E. Processing the data into the necessary format

The play-by-play data from nba.com's API was massaged into a list of possessions in this format (plus some additional metadata).

```
[  
  {  
    "home_lineup": [p_1, p_2, ..., p_5],  
    "away_lineup": [p_6, p_7, ..., p_10],  
    "scoring_margin": 0-4,  
    "home_team_is_on_offense": T/F  
  }, ...  
]
```

The list contains approximately 20-25k total objects, one for every possession in every game of an NBA season, with fluctuations resulting from variation in individual games, as well as league size in terms of number of teams.

*1) Algorithm for determining the players on the court given play-by-play data:* Scraped play-by-play data describes a game action and the players directly involved, but do not show the full lineup on the court. In order to determine this, passes are made through the data, resetting the lineups at each quarter. A player is added to the lineup whenever they appear in a game log item and then are backfilled to the beginning of the quarter. Player substitutions are also parsed and incorporated.

*2) Algorithm for counting possessions:* We note several domain knowledge specific details about the implementation here. A possession is considered to be ended by a defensive rebound, a turnover, a made field goal, or a final made free throw. The play-by-play data does not distinguish between defensive and offensive rebounds so the surrounding events (which team shot) is used to determine that. Free throw attempts from one trip are aggregated together into one possession. Substitutions that occurred between free throws are reflected in the lineups after that possession.

### F. Converting the processed data into a matrix

The above JSON format is then converted into the matrix described by Fig. 2. SciPy's sparse matrix format is used here and then those pickles are written to disk (they are too large for MongoDB's default configuration) as files.

### G. Calculating and storing of results

Once the design matrices have been prepared, we use sklearn's `linear_model.RidgeCV` to do cross validation to find an appropriate penalty term (which ended up being around 2900) and then `linear_model.Ridge` to calculate the weights for each player on offense and defense. This results in two numbers representing the contribution on offense (ORAPM) and defense (DRAPM), which are summed up to get the final value.

## V. FUTURE WORK

Future work will be focused along three primary goals. One is to improve the analysis and the final results, another is to make the pipeline more efficient, flexible, and adaptable, and the third is pedagogical - informing those interested about these numbers, the methodologies, and presenting the tools do the analysis themselves.

### A. Technical - Analysis

To fully reverse engineer ESPN's RPM and other proprietary metrics, the RAPM regression can be done with a prior.

The basic idea is that instead of simply using indicator variables the design matrix, a prior based on some box score would be used.

We can additionally add information currently not (known to be) a part of the RPM calculation, including wingspan length. Additionally, all possessions are currently being treated the same, but we want to put more emphasis and weight on actions done late in the game when the score is close. With more time and computing power, other linear regressions can also be used, such as ElasticNet.

BPM's performance in our evaluation indicates that using multiple years of data will also produce more predictive results. The problem of collinearity is counteracted by the player movement of trades and free agency so having the larger sample will help.

As mentioned earlier, other metrics of team success besides wins can be used as the basis for analysis and prediction.

### *B. Technical - Engineering*

For efficiency of storage, the backend database will eventually be converted to a SQL database. The data itself can be stored more efficiently for our analysis as a collection of possessions where the same players are on the floor.

### *C. Pedagogical / Public Interest*

For pedagogical purposes, explanatory python notebooks will be developed demonstrating how to use the whole data pipeline for anyone to run. The code will also be refactored and commented with better documentation.

Additionally, a website will be published to have searchable, sortable, historical RAPM data dating back to 1996-97, the first season for which play-by-play data is available.

Having a standard and open repository for this pipeline will facilitate additional analysis previously not possible, including the impact of coaches on the game or a player's impact on his teammate's shooting percentages.

## VI. SUMMARY AND CONCLUSION

We have presented here a set of open source tools for producing previously private and proprietary analysis of NBA players. In evaluating them as a way to predict future wins, the calculated metric Regular Adjusted Plus/Minus (RAPM) compares favorably with the current state of the art.

Going back to Michael Lewis' example of Shane Battier, we take a look at his 2005 season. By traditional box score numbers out of 464 players he was 144th in points scored per game, 103rd in rebounds per game, and 184th in assists per game. By traditional measures he would be considered at best an average player and as a result was not recognized with any awards from the league and was paid the league's near minimum salary and was the 10th highest paid player on his own team. By RAPM, however, he was 5th in the league, and played the most minutes and on a Memphis Grizzlies team that overperformed expectations and made the playoffs. In particular, he was by this measure the second best defensive player in the league.

As the league began to realize Battier's value, helped by Lewis 2009 article, he was often tasked with being the isolation defender on the league's best players such as Kobe Bryant, and was eventually recruited Lebron James' Miami Heat teams where he was a key contributor to teams that won back-to-back championships.

We hope this work will facilitate more rigorous analysis and decision making as it pertains to evaluation of players by fans, media, and front offices alike.

## REFERENCES

- [1] Ross, Terrance. "Welcome to Smarter Basketball." *The Atlantic*. 25 Jun 2015. <https://www.theatlantic.com/entertainment/archive/2015/06/nba-data-analytics/396776/>
- [2] Lewis, Michael. *Moneyball: The Art of Winning an Unfair Game*. New York: W.W. Norton, 2003. Print.
- [3] Lewis, Michael. "The No-Stats All-Star." *New York Times Magazine*. 3 Feb 2009. <http://www.nytimes.com/2009/02/15/magazine/15Battier-t.html>
- [4] "Efficiency." *NBA.com*. 16 Jun 2015. <http://www.nba.com/statistics/efficiency.html>
- [5] Khan, Ehran. "Advanced NBA Stats for Dummies: How to Understand the New Hoops Math." *Bleacher Report*. 18 Oct 2013. <http://bleacherreport.com/articles/1813902-advanced-nba-stats-for-dummies-how-to-understand-the-new-hoops-math>
- [6] "Calculating PER." *basketball-reference.com*. <https://www.basketball-reference.com/about/per.html>
- [7] Macdonald, Brian. "A Regression-based Adjusted Plus-Minus Statistic for NHL Players." 22 Jun 2010. <https://arxiv.org/abs/1006.4310>
- [8] Myers, Daniel. "About Box Plus/Minus." <https://www.basketball-reference.com/about/bpm.html>
- [9] Paine, Neil. "Is WP a legitimate stat?" *apbr.org*. 26 Mar 2013. Message 16.
- [10] <https://gitlab.com/basketball-analytics/>