OKC Thunder Data Science Application Report

**Previous Project**

In my Major Qualifying Project (WPI senior capstone), I was tasked with ascertaining the relative expected value of each pick in the NBA draft. To accomplish this, my team had to first define what ‘value’ meant. After analyzing the available options, we identified four commonly used metrics (Player Efficiency Rating, Value Over Replacement Player, Win Shares and fantasy points) in addition to 2 custom metrics, to compare the metrics’ accuracy. We analyzed these metrics in detail, to discover what counting stats each metric valued highly, and used that analysis to contextualize the below graph. This graph presents a framework for ascertaining the true ‘value’ of each pick in the NBA Draft. As you can see, different metrics present a different story about the talent level, and none of them quite line up with the NBA rookie salary scale.

**Model Information**

The data I used was scraped from the basketball-reference website. I used the player box score data for seasons, as well as the team standings in order to construct my model. The website tracks players with a team, but unfortunately players which are traded mid-season are assigned a team value ‘TOT’, reflecting their total contributions. I used python (specifically pandas) for assembling the metrics, and Weka for analysis. The primary python document I worked in was main.py, and my results from Weka are shown in the ‘Weka LinReg Output.png’ image.

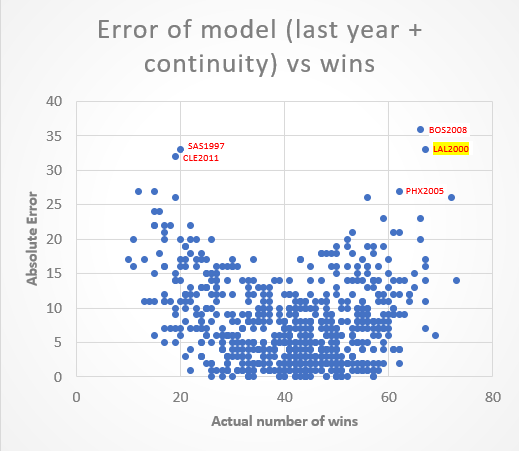
**Model Description**

Upon reading the project brief, my mind immediately jumped to Dean Oliver’s continuity metric in his book, *Basketball on Paper.* This statistic looks at the players returning to the roster and outputs a proportion of the minutes played by those players. For instance, a team with all players returning to the team has a continuity of 1, and a team with all new players would have 0 continuity. The all-time best offenses generally have high continuity, and similarly the low-ranked defenses have low continuity.

I then needed a complimentary statistic that provided insights into the talent level of the players. I opted for PER because it is normalized to the number of minutes played. I constructed a linear regression using these variables, which had a mean absolute error of just under ten wins. Unsatisfied, I incorporated the previous year’s win totals to mitigate under or overachieving teams. This lowered the mean absolute error to 8.19, which is a reasonably accurate model.

Upon reviewing my visualization (more below), I realized a large error in my dataset; the lockout seasons. Of course, the 2012 and 1999 seasons were shortened to 66 and 50 games respectively, and the error in predicting those seasons is extremely high. I considered changing the predicting factor to win%, but instead opted to remove the data I had for those seasons, and the season after. This lowered the MAE to 7.85, but also eliminated the PER statistic from the regression entirely.

Still unsatisfied, I needed a metric which would essentially evaluate clutch performance of a team. Intuitively, I thought that younger teams would struggle to win close games, which would lower their win totals. Additionally, younger teams are more prone to fatigue over the course of a long season. A model including average age was initially worse than the one without it, so it was excluded.

The regression model with the lowest error included all four of my metrics, with one surprising change. The coefficient for average age was negative (-0.4617), which means the model found a correlation between older teams and less wins. This goes against my original hypothesis and is an interesting insight. With more time, I would normalize age by minutes played to get a more accurate picture of the ages of the players who are playing.

The above visualization displays the absolute error of the basic model, before I removed the lockout seasons from the data. After the regression model didn’t immediately consider the average age, I sought a visualization that would provide more context into why the model was making poor predictions. By plotting absolute error against number of wins, I immediately saw two things: firstly, the model was relatively good for middle-of-the-pack teams. Secondly, there were four data points that were particularly egregious. This visualization is what lead me to the decision to remove the lockout-affected predictions.

Biggest Overpredictions

**CLE2011** – LeBron leaves Cleveland, along with Zydrunas Ilgauskas, Shaquille O’Neal, and Danny Green. The model predicts a 51 win season after last year’s 61 wins, but the Cavs lost 27 consecutive games this year on the way to just 19 wins.

**SAS1997** – David Robinson goes down for the season, and the Spurs blow up the team for a year. Infamously, they receive Tim Duncan with their draft pick, who ---with new coach Gregg Popovich‑-- leads the Spurs to 21 consecutive playoff appearances. Because David Robinson suited up for more than one game (he played six) and was thus on the Spurs’ roster, the continuity metric isn’t accurate. The model predicts 53 wins, 33 off their actual total of just 20 wins.

Biggest Underpredictions

**BOS2008** – Boston posts the biggest single-season turnaround in NBA history, after signing Ray Allen and Kevin Garnett through trades. They go on to win the championship against Kobe’s Lakers. As low continuity implies an impending poor season, the 0.391 continuity didn’t accurately reflect the massive talent acquisition. The model expected the Celtics to win 30 games, when they instead won 66.

**PHX2005** – Phoenix re-sign Steve Nash and acquire Quentin Richardson. This year’s roster employs coach Mike D’Antoni’s seven seconds or less offense to unbelievable success, winning 62 games and earning Steve Nash the MVP award.

Here is the same chart for my final model.

Overall, we can see from the visualization that the model does worse predicting win totals for very good teams and very bad teams. Furthermore, teams with superstar departures or acquisitions aren’t predicted well by the model. To counteract this, I’d experiment with ΔPER from off-season acquisitions.

**Trade Proposal**

1. **The more favorable unprotected 1st round pick from either the Los Angeles Clippers or Detroit Pistons in the 2019 NBA Draft.**

Predicting LAC2019:

Last year wins: 42 Continuity: 0.4080 Average age: 27.24

Model: 0.6061 \* last\_year\_wins + 11.3776 \* continuity + -0.4617 \* avg\_age + 21.7312

Predicted wins = 39

Predicting DET2019:

Last year wins: 39 Continuity: 0.5363 Average age: 26.29

Predicted wins = 38

The more favorable of the two picks is likely to be the Detroit pick, but both are very similar.

1. **The unprotected 1st round pick from the Cleveland Cavaliers in the 2019 NBA Draft.**

Predicting CLE2019:

Last year wins: 50 Continuity: 0.3866 Average age: 27.13

Predicted wins: 44

I would recommend accepting the trade proposal for the best of the Clippers/Pistons pick for two reasons: firstly, the model predicts that both teams will win less games than the Cavaliers. Secondly, having a best-pick swap is ideal because it mitigates the error of the model by essentially doubling the chance one of the teams has a bad season.

Thank you for this opportunity! I hope you enjoy reading about my model as much as I did constructing it.