Final Project Very Rough Draft

Grant Naberhaus, University of Oklahoma 04/14/2020

I'll format this to make it look at a paper, and not a homework assignment, for the final draft.

Introduction

- NBA Model of 2017-2018 Season (I intend on having more seasons, I just need to alter some dates on my RScript
- Model uses a variation of Player Efficiency Rating, Opponent Player Efficiency Rating, Home Court, and Days Between Games

Literature Review

- Discuss Player efficiency rating and how it is calculated
  - PER normalized to 15
- Model uses a variation of Player Efficiency Rating, Opponent Player Efficiency Rating, Home Court, and Days Between Games
- Discuss why this statistic was used for model (because it's all encompassing)
- Describe method to obtain "Team" PER (For each player on a team, I took their PER and divided their total minutes of the season by the total team minutes of the season. Total team minutes is the number of minutes played by all players added together. I then multiplied each players unique quotient by his PER, then added all the PER's for a given team together to obtain "team" PER)
  - Team PER ranges from 13.095 to 17.048

Data

- Data obtained from ballr, which is an API that pulls NBA data from basketball-reference.com
- Data was manipulated to produce "Team" PER
- Describe issues with the data (unable to get "team" PER per game, so I had to calculate the "team" PER based on the entire season off data and apply this same value to each game. The issue with this is if a team's best player sits out for a game, the model does not reflect this change for that one game, but rather spreads out that decrease in team PER

over the whole season, since team PER was calculated with season data, not on a per game basis)

Methods

Variables are binary for Home court, TeamPER and OpponentPER as numeric values, and Days between games as a categorical variable. 1Day (back to backs) was omitted from the model below:

- OLS Linear Probability Model
  WIN = 0.40277 + 0.12163(TeamPER) + (-0.12237)(OpponentPER) + 0.15070(Home) + 0.04303(2Days) + 0.03019(3Days) + 0.06796(4Days) + (-0.03348)(5Days) + (-0.03082)(Greater5Days)
  Adjusted R2 = 0.1514

TeamPER, OpponentPER, and Home are the only variables that have a statistically significant relationship with WIN, with all P-values being <2*10-15

- LOGIT Model

Beta values similar to OLS Probability Model
Issues with Models

- The biggest issue with the model is that it uses Player Efficiency Ratings for the entirety of a season. This results in some teams being predicted to win only a handful of games (like the 24-58 Atlanta Hawks or the 21-61 Phoenix Suns in the 2017-2018 season). Ideally, being able to find players' PER for each game of a season would remove this error, as different team PERs could be calculated on a per game basis. However, that data is not available with the package I am utilizing.

Findings

- LOGIT Backtest
  - Model has a prediction accuracy of: 64 - 70 percent depending on the break up of training/testing data, with most tests being around 67 percent accurate. Although I would've preferred to have predictions in the 70+ percent range, this lower accuracy is a result of some season long data being applied on a per game basis.
- Elastic Net

WIN = 0.1566(Home) + 0.1155(TeamPER) + (-0.1146)(OpponentPER) + (-0.0142)(1Day) + (0.0206)(2Day) + (-0.0046)(3Day) + (0.0469)(4Day) + (0.0361)(5Day) + (-0.0963)(More5Day)
RMSE: 0.4589
I have not run a backtest to determine accuracy of elastic net model. Will also include Lasso and Ridge model in final paper, but upon running these models, their RMSE was similar to the elastic net model.

Conclusion

- Model performs better than a coin, but is slightly shy of the desired level of consistently 70 percent.