# Incorporating High School Recruiting Ratings and Statistics in Predictive Models for Collegiate Basketball Success

Ted Henson and Mario Giacomazzo

18 April 2020

## Introduction

Predicting collegiate win shares from ESPN recruiting ratings [1] and high school statistics provided by PrepCircuit [2] and AAUStats [3] could yield a competitive advantage for collegiate and professional basketball organizations. As it stands, there is little research on predicting individual player's collegiate performance. In 2010, Jamie McNeilly used recruiting ranking quartiles to predict Player Efficiency Rating (PER) and other metrics [3]; however, the models presented did not consider high school statistics as an input, nor did they consider predicting win shares, which is a more authentic measurement of a player's contribution to team success as shown by basketball reference [5].

Accusations of many NCAA coaches paying high profile recruits hundreds of thousands of dollars catalyzed this analysis. If one player can cause decorated coaches to potentially resort to unethical methods, then programs should explore every possible avenue of predicting college performance, especially programs with smaller budgets and less recognition.

In addition to benefiting collegiate programs, the methods presented could benefit NBA front office decision making. The "one and done" climate has evolved into a "none and done" climate. Some high-profile recruits have hardly any collegiate data due to injuries, opting

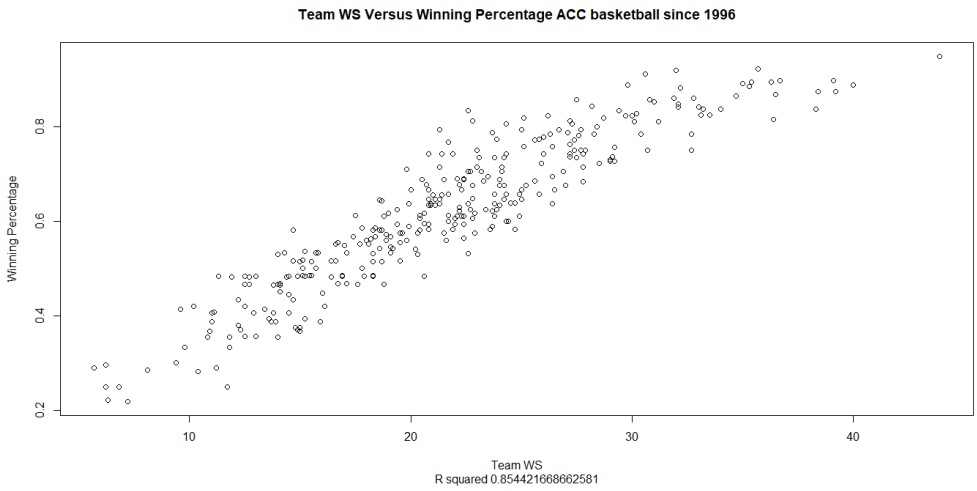
to play overseas, or simply preparing for the draft, and are still selected in the first round based on their high school evaluations. As NBA teams are investing millions of dollars on players with little to no collegiate data, the methods and data presented could be used to model NBA performance in conjunction with their collegiate performance.

The model with the best out-of-sample performance incorporated ESPN ratings and the two sources of high school statistics. This is a significant result as the data has only existed for a few years whereas the ESPN rating has been developing for over a decade. Future research adjusting for player's strength of schedule and teammates could significantly improve the models.

# Data

## Basketball Reference

Every statistic listed on a player's college basketball reference page was collected; however, only a player's first season playing in the NCAA was used in the modeling process in order to fairly evaluate a player's true production out of high school. Due to its all-encompassing nature, win shares represents the dependent variable. Win shares is calculated by taking the points produced by a player, and reduced on the defensive end, and scaling those based on their team. Below is a graph of ACC teams' sums of player win shares plotted against their season winning percentage.




College win shares have a weaker relationship with wins than Wins Above Replacement in baseball and in the NBA partially due to large differences in league competition; nonetheless, it is a strong predictor of team success as shown by the above R-squared. Other popular basketball metrics such as PER, a per minute efficiency rating, and Box Plus Minus (BPM), a box score estimate of the points contributed by a player per 100 possessions on an average team, were considered as dependent variables; however, they had a much lower correlation value with wins than win shares. Additionally, the top players in terms of win shares aligned with the consensus best players over the past few seasons more so than those with the top PER or BPM. Calculations and analyses for win shares, PER, and BPM can be found on basketball reference [5, 6, 7]. Below are the top 10 players in terms of win shares in the data.

| Group | Player | Win Shares | College Season |
|---|---|---|---|
| Prep and AAU | Zion Williamson | 8.3 | 2019 |
| Prep and AAU | Deandre Ayton | 7.6 | 2018 |
| Only Prep | Marvin Bagley III | 6.9 | 2018 |
| Only Prep | Lonzo Ball | 6.8 | 2017 |
| Prep and AAU | Wendell Carter Jr | 5.9 | 2018 |
| Only Prep | Malik Monk | 5.8 | 2017 |
| Only Prep | TJ Leaf | 5.8 | 2017 |
| Prep and AAU | Trae Young | 5.7 | 2018 |
| Prep and AAU | Tyler Herro | 5.4 | 2019 |
| Neither | Omari Spellman | 5.2 | 2018 |

From a basketball perspective, these players had some of the best freshman seasons over the past few years. Zion in particular has been widely regarded as having the best season from a statistical and basketball perspective in decades. This gives more confidence and validity to win shares as an overall barometer of success.

## ESPN

The ESPN data gathered contained players' overall ratings from 55 to 100. Only the classes from 2016 to 2018 were used in this analysis due to the lack of PrepCircuit and AAU data before the 2016 high school season. In terms of grabbing the basketball reference data, the ESPN data played a critical role. There was no feasible or swift way to accurately gather a high school player's collegiate win shares without knowing where he went to college, which was not in the PrepCircuit or AAU data. Also gathered from ESPN were player's height, weight, and position.

## PrepCircuit

The high school statistics gathered from PrepCircuit contained regular season averages and totals from box score statistics such as points, points per game, assists, etc. The data is fairly encompassing; however, there appear to be some inaccuracies in the data. For example, Lonzo Ball had 31 games where points were tracked, 4 games for minutes, 22 games for assists steals and turnovers, and 21 games for rebounds. One explanation is that PrepCircuit does not keep track of all statistics for every game. The other hypothesis was that if a player did not log a statistic in a given table, PrepCircuit did not count that towards your game total for that statistic. Upon further inspection, it appeared that the most reliable statistics were the given per game statistics.

## AAUStats

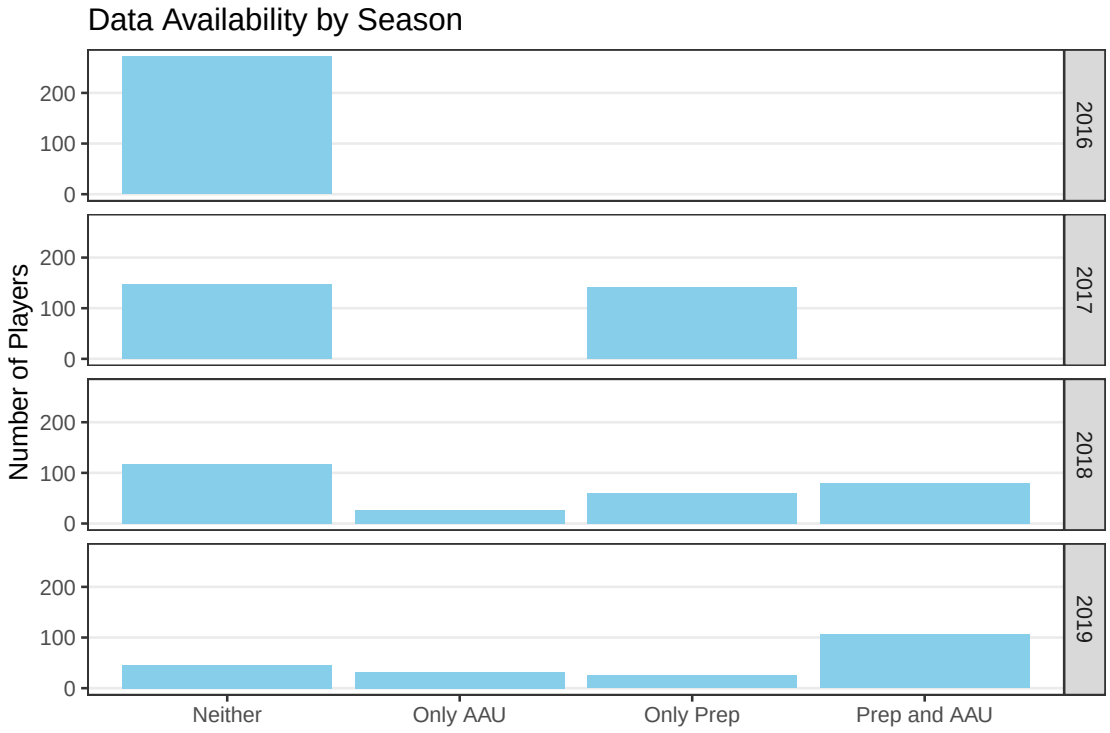
AAU data was gathered from AAUStats.com [3]. The data contained box scores from Nike, Adidas, and Under Armour circuits from the 2017 to 2019 seasons. There is potential to create more advanced metrics with these box scores, or scaling based on the quality of a player's team or opponent; however, for this analysis, common per game box score statistics akin to PrepCircuit's data were created based on player's final season in all AAU circuits.

Prior analyses used k nearest neighbor imputation to deal with players missing a given statistic, such as rebounds in PrepCircuit. Below is a table of the number of PrepCircuit players that had points per game, but were missing another per game statistic.

| Per Game Statistic | Number of Players with NAs |
|---|---|
| mpg.prep | 126 |
| reb.prep | 147 |
| blk.prep | 58 |
| spg.prep | 58 |
| tov.prep | 58 |

Although this k nearest neighbor imputation was effective for past seasons, it will not be the best method going forward, as the amount and quality of data has improved drastically over the past few years. Below is a table and histogram of the number of players that have each of our data sources by college season.

| Season | Neither | Only AAU | Only Prep | Prep and AAU |
|---|---|---|---|---|
| 2016 | 272 | 0 | 0 | 0 |
| 2017 | 147 | 0 | 141 | 0 |
| 2018 | 117 | 27 | 61 | 79 |
| 2019 | 46 | 32 | 25 | 106 |




As shown by the graph above, in 2016, the data sources presented did not even exist. By 2019, most ESPN rated players had both AAU and PrepCircuit statistics. This includes players that are rated poorly. The data appears to be improving in its robustness and reliability for current high school seniors and this past season's freshman. Their data could be incorporated into the training set or held out as a test set in the future. This past season was not included in the analysis as it was an incomplete season.

Not every model can be considered on every player as not all players have all sources of information. Using all possible players unfairly favored ESPN models since their sample of players contains many players not talented enough to play in the competitive AAU Circuits or high profile high schools in PrepCircuit. Therefore, only player with complete information will be considered. Additionally, since many players only had partial PrepCircuit data, only points per game and the number of games was used in fitting. Other variables were used in some models, but the elimination of some players lacking other statistics increased the error

rates. In total, there were 185 players used in the models.

# Methods

In order to assess the predictive value of high school statistics, models will be constructed using all possible input combinations. The models are the following:

$$M_{ESPN} : \hat{ws} = f_{ESPN}(X_{ESPN}) + \epsilon$$

$$M_{PREP} : \hat{ws} = f_{PREP}(X_{PREP}) + \epsilon$$

$$M_{AAU} : \hat{ws} = f_{AAU}(X_{AAU}) + \epsilon$$

$$M_{FULL} : \hat{ws} = f_{FULL}(X_{AAU}, X_{PREP}, X_{ESPN}) + \epsilon$$

$$M_{AAU.ESPN} : \hat{ws} = f_{AAU.ESPN}(X_{AAU}, X_{ESPN}) + \epsilon$$

$$M_{PREP.ESPN} : \hat{ws} = f_{PREP.ESPN}(X_{PREP}, X_{ESPN}) + \epsilon$$

$$M_{AAU.PREP} : \hat{ws} = f_{AAU.PREP}(X_{AAU}, X_{PREP}) + \epsilon$$

where

- $\hat{ws}$ *is the predicted win shares of a given player*,
- $X_{AAU}$, $X_{PREP}$, *and* $X_{ESPN}$ *are the data matrices for each data source*,
- $f$ *are unique functions that output predicted win shares*,
- $\epsilon$ *is a random error*

All models were trained through leave one out cross validation. Each player's predicted win shares was outputted from a function fitted on all other players. These values were chosen based on the parameters yielding the smallest out of sample error. The methods and subsequent hyper parameters will be discussed briefly below

- Lasso
  - Fits a standard linear regression, but shrinks the sum of the absolute value of the coefficients by a value, $\lambda$, chosen through cross validation.
- Ridge
  - Fits a standard linear regression, but shrinks the sum of squares of the coefficients by a value, $\lambda$, chosen through cross validation.
- Averaged Neural Network
  - Assigns one or more linear weights to each of the variables depending on the number of input layers (number of predictor variables). The hidden layers are comprised of a linear combination of the input variables. The number of hidden layers is arbitrarily chosen, in this case, 1/2 of the number of input layers. The final output is a linear combination of the hidden layers, and is then converted into a predicted value through some function. In this case a linear weight was chosen. 200 networks were constructed as such. The final prediction for a player was an average of all such networks.
- Earth
  - Models non linearaties and interactions by creating one or more hinge functions constructed as h(x-a), where a is the "knot", a cutoff value for the variable x. When x is below a, it is multiplied by a weight $b_1$. When x is greater than a, it is multiplied by a separate weight $b_2$. The number of these hinge functions is controlled by a hyper parameter as is the degree (number of interactions and higher order terms).

- Support Vector Machine
  - Builds a hyper plane, a plane or line with dimension of our predictor matrix, that attempts to minimize the distance from the response value. Rather than minimize the total sum of the errors, the hyper plane is fit so that all errors are less than a specified hyper paramater $\epsilon$. If an error is greater than $\epsilon$, it is penalized proportionally to a cost value, C, found through cross validation. In this case, the non linear hyper plane had a lower out of sample error.
- Random Forest
  - Builds "n" regression trees, in this case n was 200, and creates a prediction based on the average of all of the trees. To reduce correlation between the trees, "m" random predictors are chosen when constructing each tree. M is found through cross validation.
- Gradient Boosted Trees
  - Builds n "weak" regression trees and creates a prediction based on the average of all of the "weak" trees. The trees are pruned by cutting them at a specified "max depth" found through cross validation. For the boosted trees, these hyper-parameters were chosen through 10 fold cross validation as leave one out lead to over fitting.
- Stacked
  - The stacked model creates a combined prediction for each player from the predictions of each of the other methods discussed above. A linear regresssion on the predictions of the other methods is run for each player with leave one out cross

validation. The linear weights assigned to each method for the full model can be found in the appendix.

# Results

There were many variables to consider for these models. Many statistics in the PrepCircuit dataset were missing; moreover, many statistics in both high school datasets were randomly inaccurate, leading to poor out of sample prediction. For example, in PrepCircuit, Zion Williamson had 0 assists per game. This was an obvious example of inaccurate data, but there were many other examples that were more subtle. The variables used in the final models were only the most predictive and accurate statistics. Many methods and combination of inputs were trained to make these decisions. Note that some methods were not used for ESPN as they are tailored to data sets with many predictor variables. In this case the ESPN dataset is only one variable, ESPN rating. A full linear regression on all variables used in the models is shown in the appendix. The most interesting result was that the AAU points per game was not highly significant, but blocks and steals were. This may seem surprising, but many AAU teams are strong teams which may lead to individual star players sharing the ball more. In contrast, these same star players may be the only capable scorer on their school teams, leading to a predictive PrepCircuit points per game. Although the full linear regression performed close to the other methods after implementing leave one out cross validation, it was not used in the stacked model as it was highly correlated with the lasso and ridge models. Below are the cross validated $R^2s$ for the different models.

## Percentage of Variation Explained by the Models

| Method | ESPN (n=185) | PREP (n=185) | AAU (n=185) | FULL (n=185) |
|---|---|---|---|---|
| Linear | 23.4% | **23.2%** | 21.1% | 27.9% |
| Averaged Neural Network | | 19.3% | 18.2% | 23.9% |
| Lasso | | 23% | 21.2% | 27.7% |
| Ridge | | 22.9% | 21.4% | 29.5% |
| Random Forest | | 16.7% | 16.5% | 24.4% |
| Earth | 28.9% | 20.8% | 15% | 30.1% |
| Support Vector Machine | **33.1%** | 18.1% | **21.6%** | 24.9% |
| Gradient Boosted Trees | | 6.9% | 6.8% | 14.8% |
| Stacked | 32% | 15.8% | 17.7% | **37.4%** |

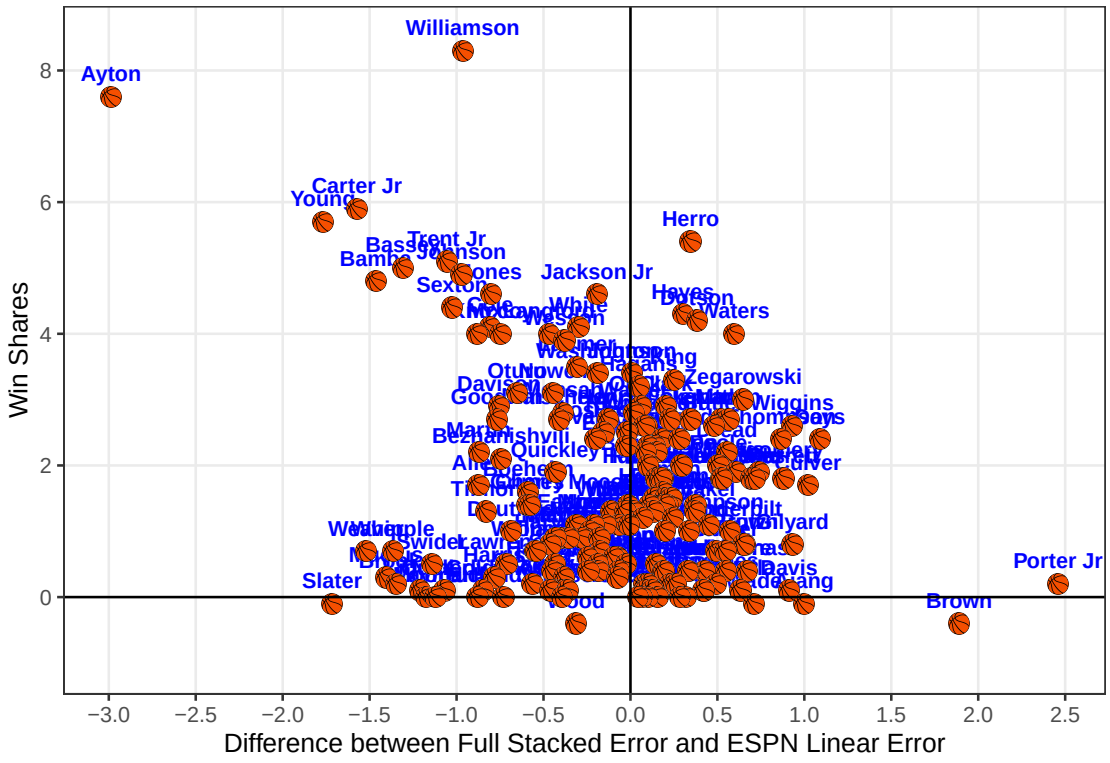
| Method | AAU.ESPN (n=185) | PREP.ESPN (n=185) | AAU.PREP (n=185) |
|---|---|---|---|
| Linear | 25.3% | 27.3% | 26.7% |
| Averaged Neural Network | 14.8% | 18.7% | 22.8% |
| Lasso | 24.6% | 27% | 26.6% |
| Ridge | 26.2% | 27.3% | **27.7%** |
| Random Forest | 23.2% | 25.6% | 21.7% |
| Earth | 28.5% | 30.1% | 14.4% |
| Support Vector Machine | 27.7% | 28.7% | 23% |
| Gradient Boosted Trees | 14.7% | 8.8% | 13.4% |
| Stacked | **31%** | **36.4%** | 21.4% |

Based on these results, the ESPN models performed better than the PrepCircuit and AAU models. The best ESPN model explains almost 10% more variation than the best PrepCircuit model. When forecasting prospect performance, 10% is a huge performance boost. It should be noted, that the best ESPN model is a radial support vector machine, indicating the presence of non linearities. A player rated 10 points higher than another player generates a larger increase in expected wins at the higher end of the rating scale. Scouting and projecting players more accurately at the higher end of the talent spectrum may yield a greater expected number of wins than at the lower end.

However, this difference between the ESPN and high school models should not discount the usefulness of the high school statistics. On their own, the PrepCircuit and AAU models

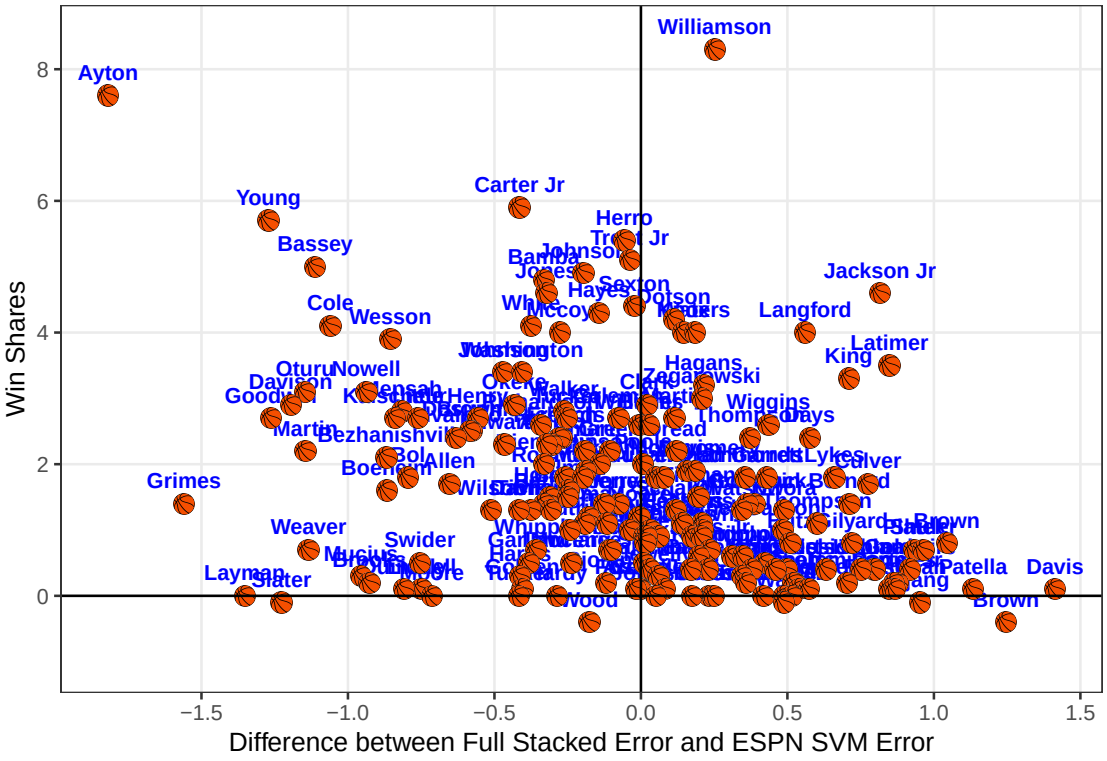
were able to explain 23% and 21% of the variation respectively, on par with the linear ESPN model; moreover, models incorporating the ESPN ratings and high school statistics performed even better, with the best model explaining 37.4% of the variation, 4% better than the non linear ESPN model, and 14% better than the basic linear ESPN model which could be viewed as the conventional approach. One could argue the non linear ESPN model should be the null model since it is well established that a star can have a disproportionate effect on the game; however, if it is so inherent to the game, than the ratings should be scaled as such.

To illustrate how this predictive advantage could be used, the plot below shows the difference in absolute error between the best full model and the linear ESPN model. Negative values mean the full model had a smaller error compared to the ESPN model and positive values mean the ESPN model performed better. In the graph below, values to the left of the vertical line indicate negative values, where the full models performed better. The y axis is the actual win shares.



The plot shows there were several players that the full model out projected the ESPN model by one or more win shares. In the most extreme example, the full model out projected Deandre Ayton by almost 3 win shares compared to the ESPN model. Other large differences favoring the full model are Trae Young, Brandon Slater, and Wendell Carter Jr.

The ESPN model predicted best on Michael Porter Jr. who only played in only 1 college game due to an injury. The ESPN model only predicted better by 1 win share on 3 other players. To further justify the use of the high school data and statistical methods, the best full model will be compared to the support vector ESPN model.




The players favoring the full model were similar here as well, but to a slightly lesser degree: Deandre Ayton, Quentin Grimes, Ryan Layman, and Trae Young. Most of the players that the ESPN model predicted substantially better on (more than 1 win share) ended up having very low win shares and a low ESPN rating. The three players favoring the ESPN model the most: KJ Davis, Shandon Brown, and Bailey Patella all had an ESPN rating of 64. Only 4

players were out predicted by more than 1 win share by this ESPN model, compared to 12 by the full model.

# Conclusion

It is important to judge the data and its predictive accuracy with some perspective. Original modeling with relatively new high school statistics improved upon a rating system that has been bettering itself for 13 years. The information used was only box score statistics which does not paint the full picture of the game. Even in baseball where statistical modeling in sports is an integral part of the process, the best projection systems incorporate scouting and statistics, and the pure statistics based models rarely beat the combined models despite the individual aspect to baseball.

This finding should encourage more analysis and collection of high school data not only for the collegiate level, but at the professional level, where one draft pick can change a franchise for decades. If this data holds predictive value at the collegiate level, there is reason to believe it can assist an NBA projection system, particularly in cases where a high school superstar underperforms, gets injured, or does not play in college. These analyses could also help evaluate NBA or collegiate potential when a low rated prospect has a phenomenal high school or college career. Although the improvements shown by the models and data are small relative to the amount of statistical analysis, in 13 years the reliability and robustness of the data will improve the models substantially.

# References

[1] *ESPN basketball recruiting - player rankings*. ESPN Internet Ventures, 2020 [Online]. Available: http://www.espn.com/college-sports/basketball/recruiting/playerrankings/_/class/2016/order/true

[2] PrepCircuit, *Statistics - 2015-2016 regular season - hs circuit*. SportsEngine, Inc., 2020 [Online]. Available: https://www.prepcircuit.com/stats/league_instance/34558?subseason=245525

[3] *AAUStats*. AAUStats.com, 2020 [Online]. Available: http://aaustats.com/

[4] J. McNeilly, *Examining the relationship between ncaa division i ranked recruits and their ensuing athletic production in college*. Epublications.marquette.edu, 2010 [Online]. Available: https://epublications.marquette.edu/cps_professional/14/

[5] *Calculating win shares*. Sports Reference, 2020 [Online]. Available: https://www.sportsreference.com/cbb/about/ws.html

[6] *Calculating per*. Sports Reference, 2020 [Online]. Available: https://www.basketball-reference.com/about/per.html

[7] *About box plus/minus (bpm)*. Sports Reference, 2020 [Online]. Available: https://www.basketball-reference.com/about/bpm2.htmll

# Appendix

## Full Linear Regression Coefficients

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -2.7482430 | 1.3179819 | -2.0851903 | 0.0385373 |
| espn.rating | 0.0379628 | 0.0181221 | 2.0948334 | 0.0376594 |
| Points.prep | 0.0485737 | 0.0174540 | 2.7829494 | 0.0059927 |
| GamesPlayed.prep | 0.0365939 | 0.0144797 | 2.5272483 | 0.0124029 |
| Minutes.aau | -0.0297244 | 0.0196943 | -1.5092884 | 0.1330706 |
| Rebounds.aau | 0.0985618 | 0.0711822 | 1.3846403 | 0.1679665 |
| Blocks.aau | 0.3359496 | 0.2459050 | 1.3661763 | 0.1736776 |
| Steals.aau | 0.4462566 | 0.2367293 | 1.8850922 | 0.0611137 |
| Points.aau | 0.0109654 | 0.0139632 | 0.7853101 | 0.4333585 |
| GamesPlayed.aau | -0.0270764 | 0.0259706 | -1.0425793 | 0.2986149 |
| Position.BasicPF | -0.5590865 | 0.3582525 | -1.5605934 | 0.1204685 |
| Position.BasicPG | -0.3308614 | 0.5234688 | -0.6320556 | 0.5281947 |
| Position.BasicSF | -0.7849094 | 0.4020491 | -1.9522724 | 0.0525388 |
| Position.BasicSG | -0.3777551 | 0.4560155 | -0.8283821 | 0.4086089 |

## Full Stacked Model Linear Weights

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -1.2432768 | 0.4691364 | -2.6501393 | 0.0087747 |
| nnet | 0.9665427 | 0.4177753 | 2.3135466 | 0.0218420 |
| lasso | -1.4765924 | 0.8689065 | -1.6993686 | 0.0910065 |
| ridge | 2.5443861 | 0.9436695 | 2.6962683 | 0.0076891 |
| rf | -1.2867831 | 0.4415991 | -2.9139173 | 0.0040300 |
| earth | 1.2093529 | 0.1827830 | 6.6163326 | 0.0000000 |
| svm.radial | 0.0952975 | 0.2643152 | 0.3605449 | 0.7188696 |
| xgboost | -0.2605762 | 0.1645476 | -1.5835914 | 0.1150718 |