

Ensemble Methods for NBA Salary Prediction

UW SPA DRP Spring 2022

Pranav Natarajan



BE BOUNDLESS



PROBLEM STATEMENT

Predicting the Salary of an NBA player given their intrinsic information and in game statistics through the season.



DATA QUERYING & PROGRAMMING

- R v(4.1.2)
- nbastatR (maintained by Abe Resler)
- BasketballReference



R PACKAGES USED

- **tidyr v(1.2.0) & dplyr v(1.0.8)**
- **caret v(6.0-92)**
- **ranger v(0.13.1)**
- **Boruta v(7.0.0)**
- **xgboost v(1.6.0.1)**
- **glmnet v(4.1-4)**



DATA PREPROCESSING

- Rookies from 1985-86 to 2020-21
- Salary normalised by yearly salary cap
- Pertinent Features:-
 - Usage %
 - Player Efficiency Rating
 - VORP
 - Win Shares
 - Age
 - Position
 - Team(s)
 - Count of Games
- 80-20 stratified train-test split
- 5 fold CV for hyperparameter tuning



WHAT IS AN ENSEMBLE MODEL?

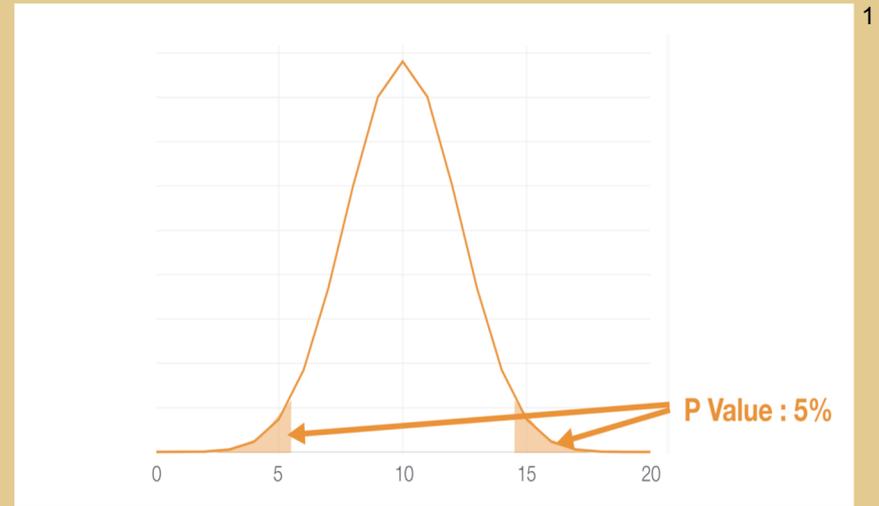
- A Model that “aggregates” estimates from a (large) number of other models (weak learners) to provide a final estimate for the supervised learning problem.
- Training sets are bagged or pasted



BORUTA ALGORITHM

Central Idea:-

Feature is useful *iff* it performs better than the best randomised feature



¹ (Nishida, *Finding variable importance with Random Forest & Boruta*, 2019)

BASELINE MODEL

- Elastic Net
- Optimal Hyperparameters
 - $\alpha = 0.1$
 - $\lambda = 0.0003342206$

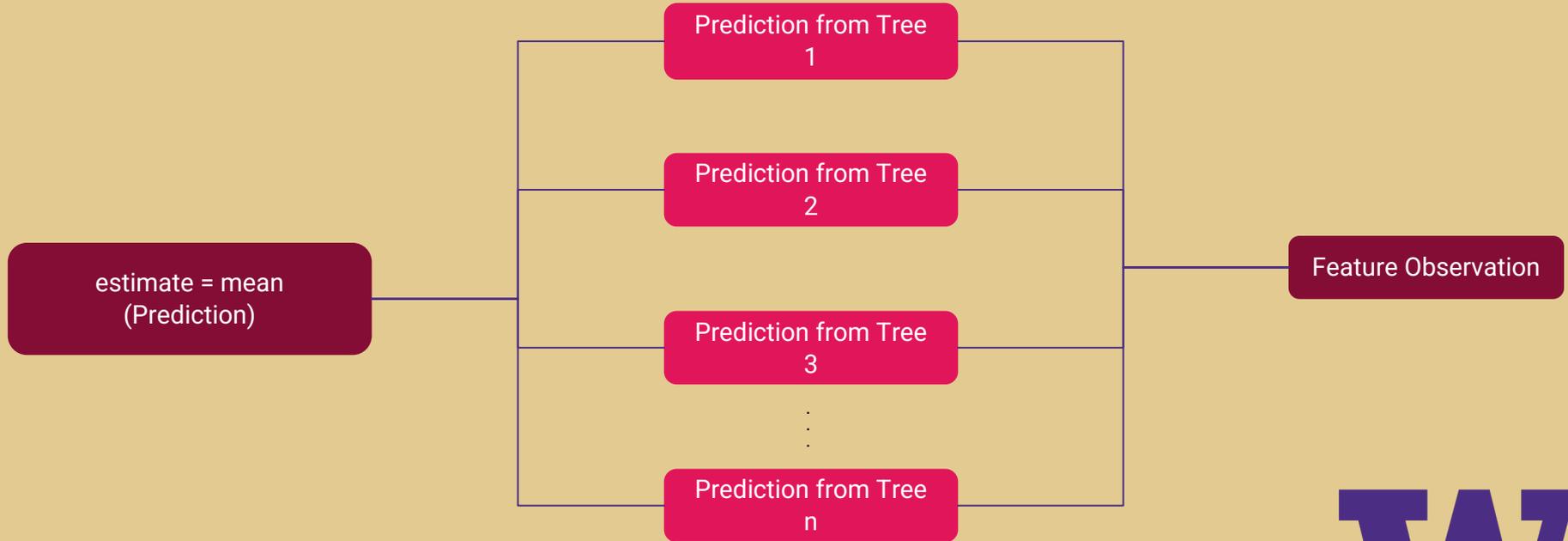


RESULTS

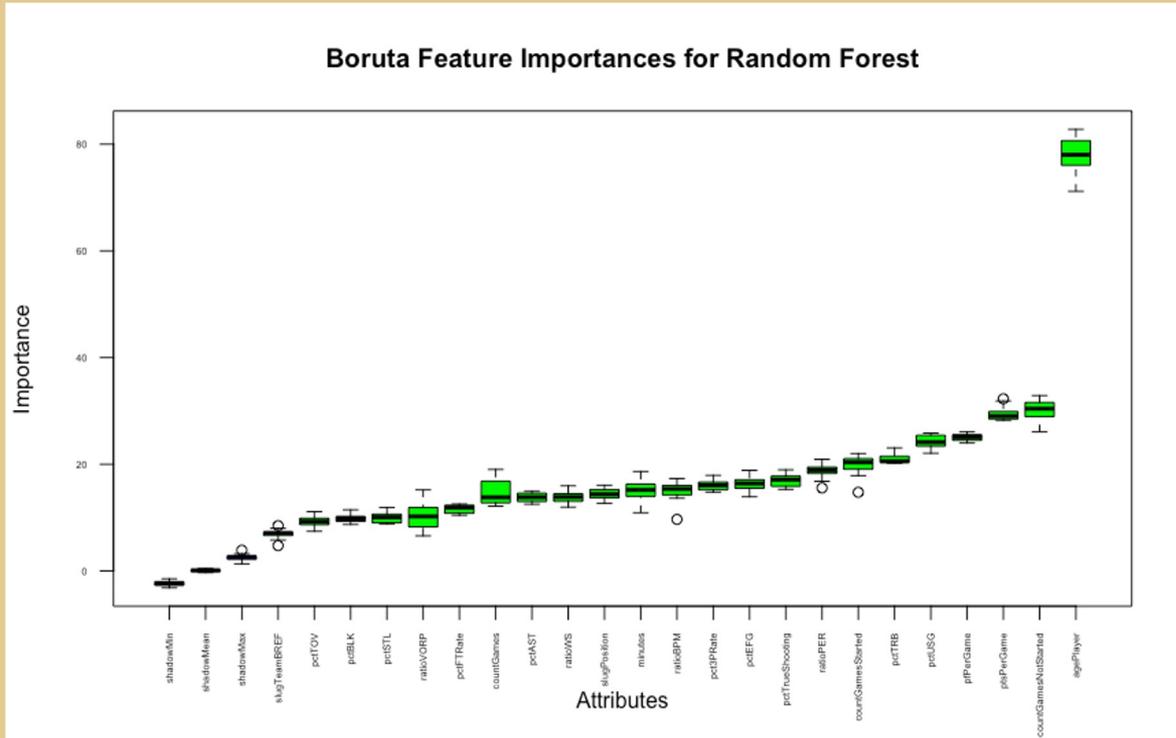
MODEL	OUT OF SAMPLE RMSE	ERROR VALUATION IN 2021 SALARY (\$)
Random Forest	0.05375	6,041,710
XGBoost-Linear	0.05485	6,165,897
Stochastic XGBoost - Tree	0.05486	6,167,435
XGBoost - Tree	0.05491	6,172,589
Elastic Net	0.05945	6,683,473



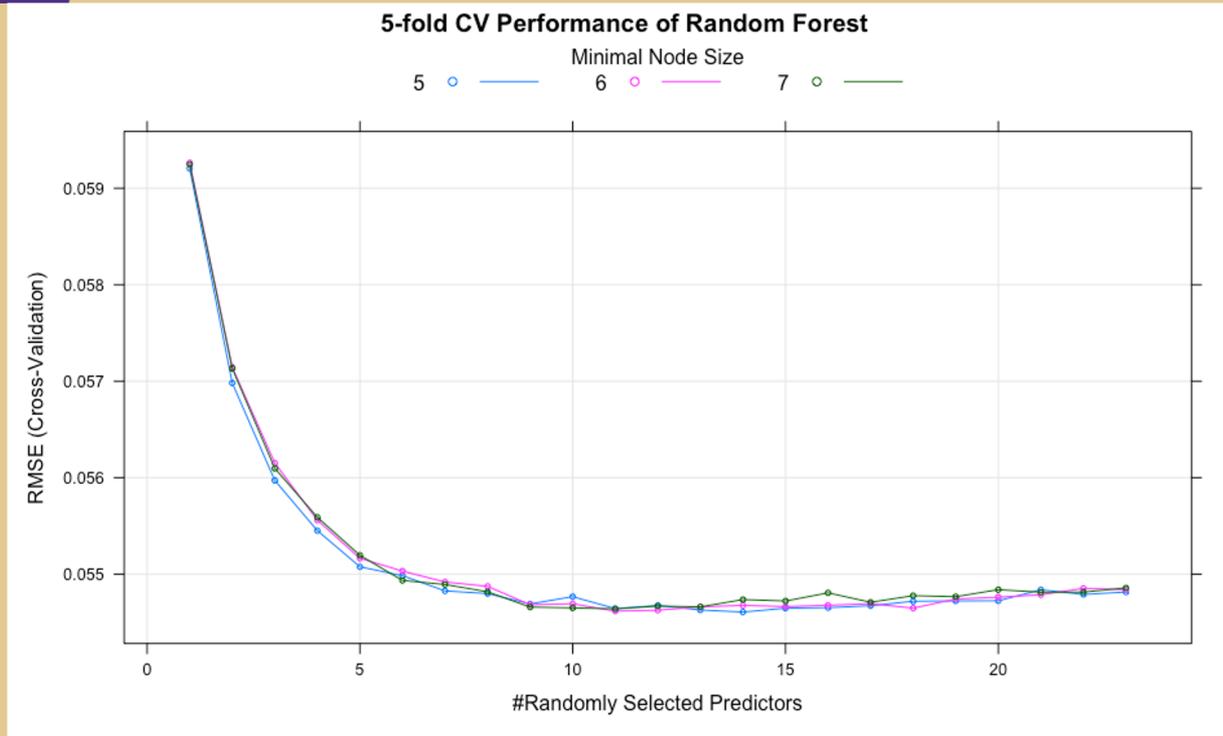
RANDOM FOREST



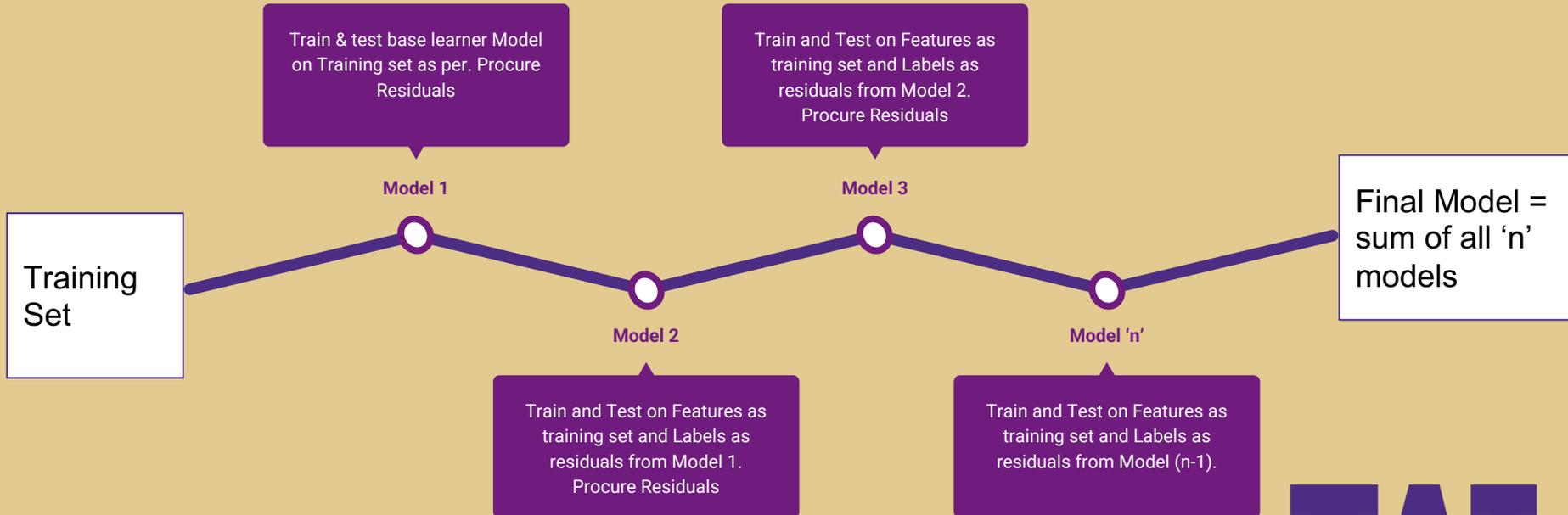
RANDOM FOREST FEATURE SELECTION



RANDOM FOREST HYPERPARAMETER TUNING



GRADIENT BOOSTING & XGBOOST

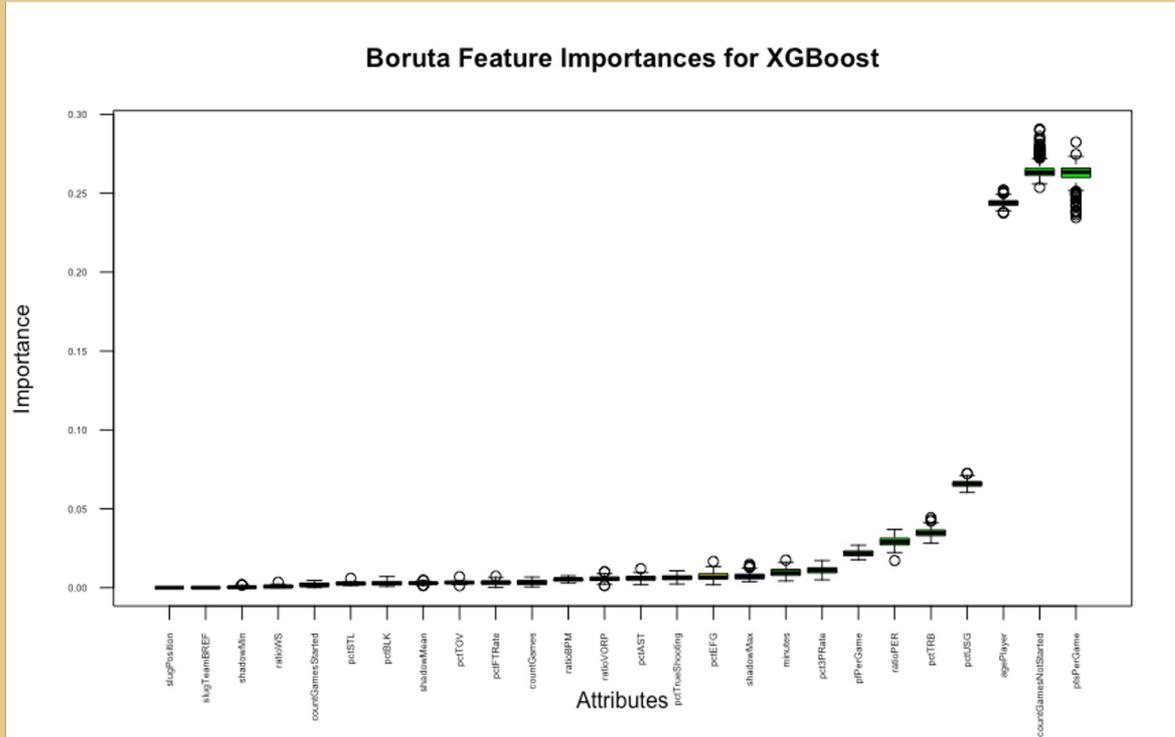


STOCHASTIC GRADIENT DESCENT

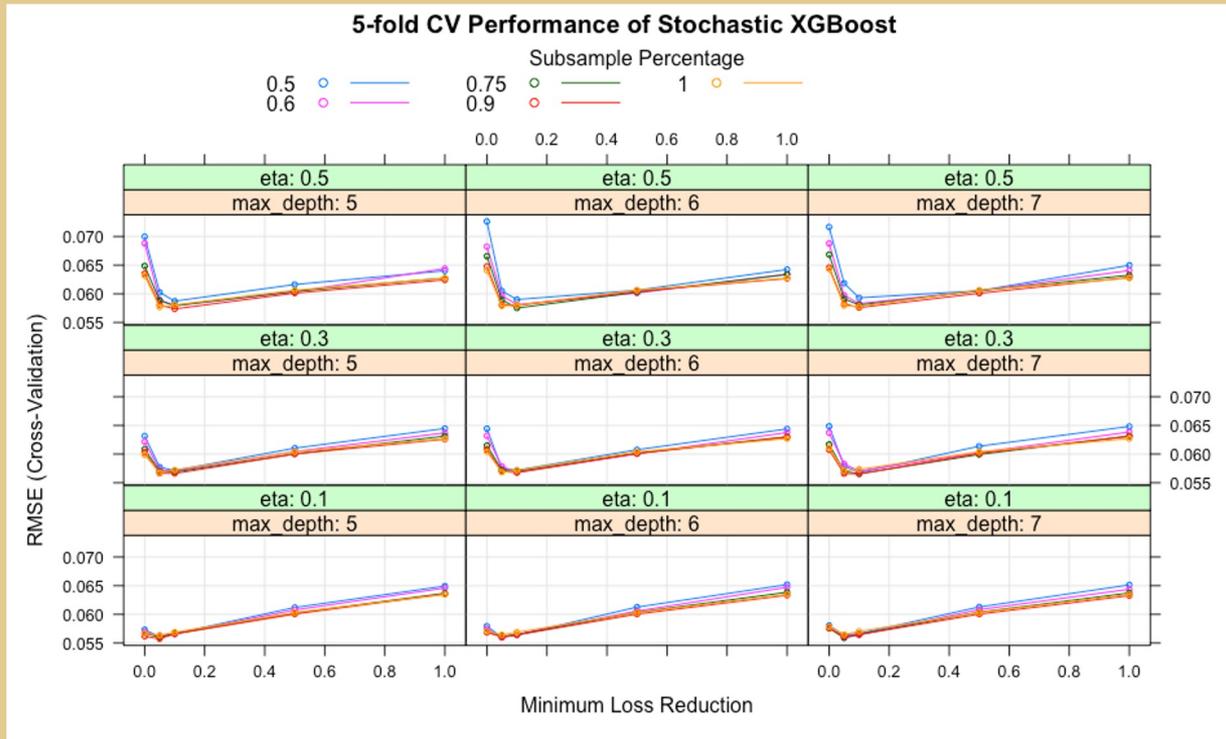
- Random Subsampling (without replacement) of Training set**
- Evades plateaus and local minima in cost function**
- Faster execution with minimal tradeoff**



XGBOOST FEATURE SELECTION



STOCHASTIC XGBOOST TREE HYPERPARAMETER TUNING



CONCLUSIONS & SCOPE FOR FURTHER RESEARCH

- Ensemble methods better for the supervised learning problem
- Best Model:- Random Forest
- Further hyperparameter tuning

- 2021 Salary Cap = \$ 109,140,000

MODEL	OUT OF SAMPLE RMSE	ERROR VALUATION IN 2021 SALARY (\$)
Random Forest	0.05375	6,041,710
XGBoost-Linear	0.05485	6,165,897
Stochastic XGBoost - Tree	0.05486	6,167,435
XGBoost - Tree	0.05491	6,172,589
Elastic Net	0.05945	6,683,473



Thank You

BE BOUNDLESS



REFERENCES

Adams, L. (2021, August 4). *Rookie scale salaries for 2021 NBA first-round picks*. Retrieved June 8, 2022, from <https://www.hoopsrumors.com/2021/08/rookie-scale-salaries-for-2021-nba-first-round-picks.html>

Boehmke, B. (n.d.). *Gradient Boosting Machines*. Gradient Boosting Machines · UC Business Analytics R Programming Guide. Retrieved June 8, 2022, from http://uc-r.github.io/gbm_regression#idea

Chen, T., & Guestrin, C. (2016, June 10). *XGBoost: A scalable tree boosting system*. arXiv.org. Retrieved June 8, 2022, from <https://arxiv.org/abs/1603.02754>

Friedman, J., Hastie, T., & Tibshirani, R. (2010). *Regularization Paths for Generalized Linear Models via Coordinate Descent*. *URL*. Retrieved from <https://www.jstatsoft.org/v33/i01/>



REFERENCES

Kuhn, M. (2022). *caret: Classification and Regression Training. R package version 6.0-92*. Retrieved from <https://CRAN.R-project.org/package=caret>

Kursa, M. B., & Rudnicki, W. R. (2010). *Feature Selection with the Boruta Package*. Retrieved from <http://www.jstatsoft.org/v36/i11/>

Nishida, K. (2019, September 27). *Finding variable importance with Random Forest & Boruta*. Medium. Retrieved June 8, 2022, from <https://blog.exploratory.io/finding-variable-importance-with-random-forest-boruta-28badd116197>

Parr, T., & Howard, J. (n.d.). *The intuition behind gradient boosting*. Gradient Boosting: Distance to Target. Retrieved June 8, 2022, from <https://explained.ai/gradient-boosting/L2-loss.html#sec:2.3>

Resler, A. (2022, May 6). *Abresler/NBASTATR: R's interface to NBA data version 0.1.151 from github*. Retrieved June 8, 2022, from <https://rdrr.io/github/abresler/nbastatR/>



REFERENCES

Sports Reference LLC. (n.d.). Basketball-Reference.com - Basketball Statistics and History. Retrieved June 8, 2022, from <https://www.basketball-reference.com/>

Steinberg, L. (2018, June 28). *The NBA draft process for dummies*. Retrieved June 8, 2022, from <https://www.forbes.com/sites/leighsteinberg/2018/06/21/behind-the-scenes-the-nba-draft-process-for-dummies/?sh=38a3134f6095>

Wickham, H., & Girlish, M. (2022). *tidyr: Tidy Messy Data. R package version 1.2.0.* . Retrieved from <https://CRAN.R-project.org/package=tidyr>

Wickham, H., François, R., Henry, L., & Muller, K. (2022). *dplyr: A Grammar of Data Manipulation. R package version 1.0.8.* . Retrieved from <https://CRAN.R-project.org/package=dplyr>



REFERENCES

Wood, R. (2011, June 15). *The History of the 3-pointer*. The history of the 3-pointer. Retrieved June 8, 2022, from <https://www.usab.com/youth/news/2011/06/the-history-of-the-3-pointer.aspx>

Wright, M. N., & Ziegler, A. (2017). ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. *Journal of Statistical Software* 77(1), 1-17. <https://doi.org/10.18637/jss.v077.i01>

Mazzanti, S. (2021, February 12). *Boruta explained the way I wish someone explained it to me*. Medium. Retrieved June 8, 2022, from <https://towardsdatascience.com/boruta-explained-the-way-i-wish-someone-explained-it-to-me-4489d70e154a>

