Ensemble Methods for NBA Salary Prediction

UW SPA DRP Spring 2022

Pranav Natarajan

June 9 2022

Description of Supervised Learning Problem

Given:-

- Features:- Rookie information, age, and their NBA game Statistics *per season* from 1986-2021
  - Note that rookies from seasons before continue on to play, so are included in the dataset.
- Labels:- Player salary earnt for the season in question

To predict:-

the optimal Salaries (normalised by yearly salary cap) of players, given their age, information and game statistics.

Data Sources, and Programming Language

- nbastatR, maintained by Abe Resler, to extract rookie draft information from the NBA, and player statistics & salaries by season from Basketball Reference.
- Yearly salary cap data from Basketball Reference.
- Model training, feature selection, hyperparameter tuning, and plotting done using packages loaded on R version 4.1.2 (2021-11-01) ‘Bird Hippie’.

Unpacking the Features

Before we delve into the statistics of the players, it is important to talk about the selection ranges of the players themselves, and why they were chosen.

- Players chosen from draft lists as that is the primary way to gain entry into an NBA team[1], even for international players from foreign leagues.
- Earliest rookies chosen from 1985-1986 to the 2020-2021 season to ensure that the stats reflect the introduction of the 3 point line by the NCAA[2].

[1] Stein, The NBA Draft Process for Dummies, Jun 21 2018, Forbes.com

[2] Wood. *The History of the 3-pointer.* Jun 15 2011. USA Basketball

Unpacking the Features - contd.

Now, we can talk about the statistics themselves. In interest of time, I am not listing all of the 40+ features loaded in from Basketball Reference, rather the 23 features chosen upon removing duplicates and redundancies (especially upon examining strong relationships between the advanced metrics and most of the per game and all of the per minute stats).

- Position
- Team
- age
- No. of Games (Started, and not started)

Unpacking the Features - contd.

- Advanced Metric percentages:-
  - Effective Field goals
  - True Shooting, 3 point Shooting, Free Throw
  - Total Rebounds
  - Assists, Steals, Blocks, Turnovers
  - Usage — player's effectiveness to team structure
- Ratios:-
  - Win Shares
  - Box Plus/Minus
  - Value Over Replacement Player (VORP) – box score estimate of pts/100 possessions that a player contributed above replacement level (-2.0) player[3]

[3] Basketball Reference. *2021-22 Player Stats: Advanced*

Preprocessing of Data

- 80-20 Train-Test Split. 7704 observations in train set, 1924 observations in test set.
- Team and Age would be one-hot encoded, as they are categorical features
- The rest of the features were standard scaled, using mean and standard deviation values from the training set
- Salary normalised by yearly salary cap to prevent effects of price inflation and other economic conditions.

What is an Ensemble Model?

*A model that aggregates a set of estimators' predictions to predict on the **same feature set** is an ensemble model.*

- Usually obtain predictions on bootstrap samples from training set (aka, bagging), or from samples without replacement from training sets (aka, pasting).
- For Regression, predictions from *each regressor model on each training set* are averaged to provide an estimate from the given observation.

Models Used

- Elastic Net (as a baseline linear regularisation model).
- Ensemble Models:-
    - Random Forest Regressor
    - XGBoost Tree Regressor
    - XGBoost Linear Regressor
    - Stochastic Gradient Boosted Regressor

Why Cross Validation?

- Validation set used to protect models against overfitting, and evaluate out of sample performance
- CV creates K such random validation sets from the training set (we choose $K = 5$ in the interest of minimsing execution time).
- Models with fit hyperparameters evaluated against these validation set to provide best validation RMSE across folds
- Effective method for hyperparameter tuning.

Elastic Net

$$\text{Loss function} = \mathit{MSE} + \lambda \left(\alpha ||\vec{\beta}||_1 + \frac{(1-\alpha)}{2} ||\vec{\beta}||_2^2 \right)$$

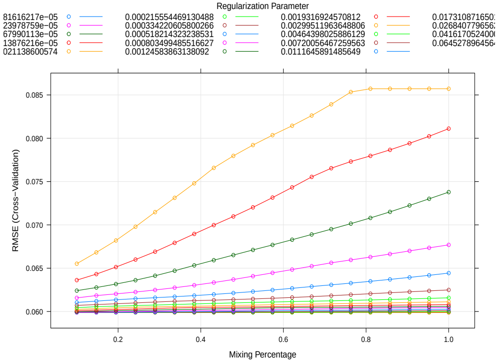
where $\vec{\beta}$ is the vector of coefficients, α is the penalty hyperparameter and λ is the parameter denoting the 'strength of bias/variance tradeoff'.

- The L1 penalty performs automatic feature selection,
- L2 norm prevents overfitting

A Randomised Search method using 5 fold CV and 20 fits used to tune $\alpha \in [0, 1]$.

Elastic Net Performance

- optimal $(\alpha, \lambda) = (0.1, 0.0003342206)$



- Out of Sample RMSE $= 0.0594540980893561$

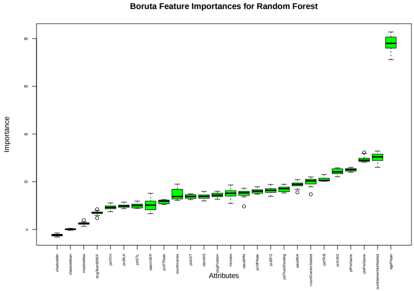
Boruta Algorithm

- Central idea:-
  feature is useful only if it is capable of doing better than the best randomised feature.[4]
- Theory of the Binomial Distribution – many runs to figure out number of accepted, tentative and rejected variables based on number of times their feature importance scored higher than the best randomised feature's.
- Useful for any algorithm containing a *feature importance* metric

[4] Mazzanti. Published on TowardsDataScience

Random Forest Regressor

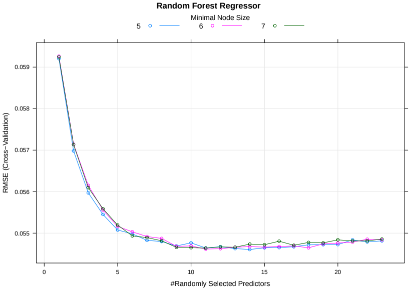
- Uses Decision Tree ensemble,
- Training sets obtained using bagging
- number of observations in *each* bagged training set = number of observations in actual training set
- 250 runs of Boruta Algorithm on the training set to evince important features. None rejected.



Random Forest - contd.

(Optimal) hyperparameters for 5 fold grid Search CV

1. `mtry` - no. of features to use in each decision tree split $\in [1, 23]$. (14)
2. `splitrule` - defining metric of split at each tree node $=$ `(variance)`
3. `min.node.size` - implicit setting of tree depth based on min. no. of obs. in leaf nodes $\in [5, 6, 7]$. (5)



Out of sample RMSE $= 0.0537451743828409$

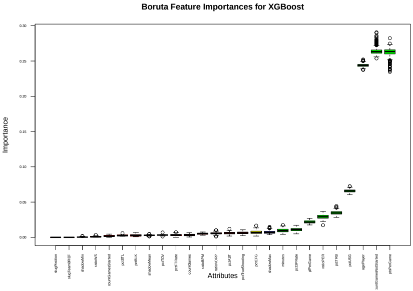
Extreme Gradient Boosted Regressors

- Sequentially add models to the ensemble
- From second model onwards, each model predicts the *residuals* of the previous model ensemble.
- XGBoost is a highly efficient algorithm – decreases computation time[5]
- Different types of models can be added in the ensemble – we will focus on Decision Trees and Regressors.

[5] Tianqi Chen and Carlos Guestrin, "XGBoost: A Scalable Tree Boosting System", 22nd SIGKDD Conference on Knowledge Discovery and Data Mining, 2016, https://arxiv.org/abs/1603.02754

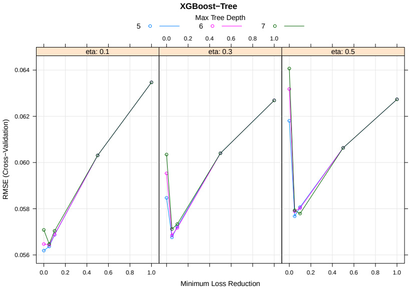
Boruta Feature Selection for XGBoost Models:-

- 250 iterations, reduced 9 feature set contains:-
  - age
  - No. of games not started
  - personal fouls and points per game
  - minutes played
  - Player Efficiency Rating
  - Percentages of 3 Pt Shooting success, Total Rebounds, and Usage



Extreme Gradient Boosted Tree Regressor

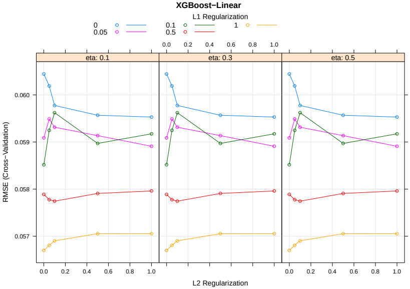
- model = Decision Tree Regressors
- (Optimal) Hyperparameters for 5 fold grid Search CV
  - No. of iterations = 100
  - Maximum Depth of the tree $\in [5, 6, 7]$. (5)
  - shrinkage parameter (i.e., learning rate) $\eta \in [0.1, 0.3, 0.5]$. (0.1)
  - required loss reduction to cause partition in tree $\gamma \in [0, 0.05, 0.1, 0.5, 1]$. (0)



Out of Sample RMSE = 0.0549094322959995

Extreme Gradient Boosted Linear Regressor

- model = Linear Regularised models (i.e, Elastic Nets!)
- (Optimal) Hyperparameters for 5 fold grid Search CV
  - No. of iterations = 100
  - L2 Regularisation $\lambda \in [0, 0.05, 0.1, 0.5, 1]$. (0)
  - L1 Regularisation $\alpha \in [0, 0.05, 0.1, 0.5, 1]$. (1)
  - shrinkage parameter (i.e., learning rate) $\eta \in [0.1, 0.3, 0.5]$. (0.1)



Out of Sample RMSE = 0.0548499062700181

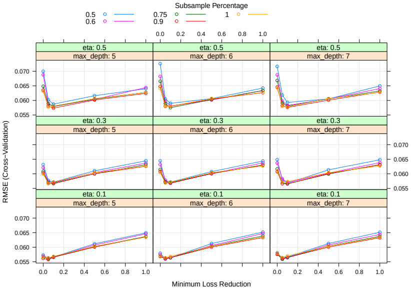
Stochastic Gradient Boosting

- i.e, random subsampling of the training set *without replacement*.
- Varying shapes of objective cost functions
- Random subsampling without replacement helps algorithm avoid local minimas and plateau regions to converge to global minimum[6]
- We can use the optimised xgboost algorithm to perform stochastic gradient boosting in R. The features, thus, are the ones chosen by the Boruta Algorithm on XGBoost feature importances.
- `subsample` , the subsampling ratio on the training set, added to tuning grid.

[6] UC Business Analytics R Programming Guide: Gradient Boosting Machines

Stochastic XGBoost Tree Regressor

- (Optimal) Hyperparameters for 5 fold gridSearchCV
  - No. of iterations = 200.
  - Maximum Depth of the tree `max_depth` $\in [5, 6, 7]$. (5)
  - shrinkage parameter (i.e., learning rate) $\eta \in [0.1, 0.3, 0.5]$. (0.1)
  - required loss reduction $\gamma \in [0, 0.05, 0.1, 0.5, 1]$. (0.05)
  - subsample ratio `subsample` $\in [0.5, 0.6, 0.75, 0.9, 1]$. (0.9)



Out of sample RMSE = 0.0548635873192439

Conclusions - RMSES

	Test RMSE
RandomForest	0.0537452
XGBLinear	0.0548499
SGDTree	0.0548636
XGBTree	0.0549094
ElasticNet	0.0594541

Conclusions - contd.

- **Random Forest** ensemble method provides the **best out of sample performance**.
  - Random Forests (and tree models in general) usually fit the training set quite well, must prevent overfitting by setting a minimum number of observations in a leaf node.
- It is to be noted that the Stochastic XGBoost Regressor did do marginally better and worse respectively to the usual XGBoost Tree and Linear Regressor respectively.
  - Performance suggests non convexity of cost function, and thus rationalises further hyperparameter tuning on smaller neighbourhoods to improve performance.
- Finally, we see that **all of the ensemble methods performed better out of sample than a simple supervised learning algorithm**

References (create.bib file locally using r markdown and finish)

1. Stein, Leighberg. The NBA Draft Process for Dummies. Forbes.com, Jun 2
2. Wood, Ryan.*The History of the 3-pointer*. USA Basketball. Jun 15 2011.
3. Basketball Reference. *2021-22 Player Stats: Advanced.* Accessed Jun 3 202
4. Mazzanti, Samuele. Boruta Explained Exactly How You Wished Someone E
5. Chen,Tianqi and Guestrin, Carlos. *XGBoost: A Scalable Tree Boosting Syst*
6. UC Business Analytics R Programming Guide: Gradient Boosting Machines