Maastricht University

Machine Learning

ATP Match Predictions Using Tree-based Methods

Name:	Barbora Rakovanová
ID:	i6243774
Programme:	MSc Econometrics and Operations Research
Tutor:	Rui Jorge Almeida
Date:	June 14, 2024

School of Business and Economics

Master

Contents

1 Introduction

With the advancement of statistical models and availability of detailed data, sports analytical have become inseparable part of the spectator experience. Not only do such analytical features make sports more interesting, they also allow fans to gain more understanding of different aspects of the game. With tennis being one of the most popular sports worldwide, it is not an exception. In the game of tennis, two different forecasting approaches can be taken. While point-based forecasts focus on predicting the winner of a given point, match predictions expand the analysis to the outcome of the whole match.

This paper utilized 30 years of historical ATP tennis data to assess the match-predictive performance of tree-based methods, as well as assess feature importance.

2 Dataset

2.1 Data Cleaning

The initial dataset used for the analysis in this paper was created by Jeff Sackmann and retrieved from https://www.kaggle.com/datasets/guillemservera/tennis. This data consists of 188 161 observations, representing Association of Tennis Professionals (ATP) matches spanning from 1968 to 2022, as well as numerous tournament, player and match-specific information.

Given the objective of assessing match-predictive models, matches resulting in retirement[1] or walkover[2] do not provide useful information. Therefore, 4 891 such match are removed from the set of observations.

While the full dataset spans the whole Open Era[3] up to year 2022, many match statistics (such as ace and double-fault information) are only available for matches after 1991. To utilize the full potential of the dataset, all of the data is used in

[1] Retirement is a result of player's inability to finish a match due to an illness or an injury (Association of Tennis Professionals, n.d.).

[2] Walkover results from a Code of Conduct penalty-based disqualification of a player or an unbegun match due to player's illness (Association of Tennis Professionals, n.d.).

[3] Era of professional tennis as it is known today, established in 1968.

the feature extraction process, resulting in accurate calculation of player-specific measurements. However, due to the lacking statistics in the pre-1991 period, only matches played in 1991 or later are considered for the main part of the analysis.

2.2 Feature Extraction

Out of the 49 variables directly available in the initial dataset, the following 24 are used for the analysis in this paper:

1. **Age Difference**

   Given the trade-off between gained experience and physical deterioration, Gorgi, Koopman, and Lit (2019) finds that male tennists reach the highest performance at the age of 25. Moreover, instead of considering players' ages directly, the difference in ages can be of interest. According to del Corral and Prieto-Rodríguez (2010), age difference has a significant effect on the outcome of male tennis matches, with the probability of the higher-ranked player winning monotonically decreasing when playing against younger players. Given these findings, the effect of age difference between the two players is considered in this paper.

2. **Height Difference**

   Taller players are better servers, hitting the serve faster and more effectively compared to the shorter players (Sackmann, 2017). According to Ovaska and Sumell (2014), the greater the height difference between the players is, the more likely is the taller player to win. However, Sackmann (2017) and Ovaska and Sumell (2014) also note the inverse relationship between height and return effectiveness, as well as the fact that taller players may struggle with mobility. By including the height difference (in centimeters) between the opponents as an explanatory variable in the analysis, the effects of height are examined.

3. **Difference in Ranking Points**

   In every match on the ATP tour, players earn a certain amount of points depending on how far they get in the tournament. The more prestigious a tournament is, the more points are awarded. As a result, ranking point are representative measure of player's performance in a given season and the difference in ranking points of the opponents is used in this paper.

4. **Rank Difference**

   The rank of a payer indicates their standing in the ATP ranking and is directly determined by the amount of ranking points a player has compared to other players. The difference in opponents ranks is used to analyse it's effect on the match outcome.

5. **Seed Difference**

   The seed of a player represents their rank at a given tournament. In a case when all players joint the same tournament, the seed and rank are the same.

6. **Tournament Year**

   In the past, serve-and-volley[4] was a popular style of play, requiring strong serve and volley skills. In contrast, the rallies nowadays are longer, requiring patience, strong baseline game and endurance. To assess the effect of time-related factors on the outcome of a match, the year the tournament was played in is included as an explanatory variable.

7. **Best of**

   Depending on the level of the tournament, male tennis matches are played according to a best-of-3 or best-of-5 format. In the first case, a player has to win two sets in order to win a match (meaning at most three sets are played). In comparison, the latter format consists of at most five sets, requiring a player to win three sets in order to win the match. The *Best_of* variable indicates the two respective formats by an integer number 3 and 5.

[4] A dynamic style of play where the serving player hits the serve and immediately approaches the net to play a volley.

8. **Surface**

   There are four possible surfaces a tennis match can be played on - clay, grass, hard court and a carpet. According to Hamingson (2024), grass courts, known for their fast balls and low bounce, favor players with a strong serve and net game. In contrast, clay courts generate slower balls with higher bounces, which are advantageous for baseline players. Additionally, hard courts exhibit varying speeds and bounces, but they generally provide more consistent bounces compared to clay and grass courts (Hamingson, 2024). Given the strengths and weaknesses of a player, their performance can vary significantly across surfaces. Using dummy variables indicating each of the surfaces, court-related effects can be assessed[5].

9. **Player Hand**

   While the vast majority of players is right-handed, there are a few left-handers on the tour. To indicate the dominant hand of each of the contestants in a match, dummy variables are used.

In addition to the fifteen variables described above, 9 other features (including the variable of interest) are considered in this paper. These variables, however, are not directly obtainable from the initial dataset, and instead, had to be computed using the available information.

1. **Head-to-Head**

   Given the full history of matches between a fixed pair of players, the head-to-head measure, representing the number of matches that either of the players has won against the fixed opponent, is be calculated for both of the players.

2. **Home Advantage**

   Whether it is due to a greater support from the crowd or more familiarity with the environment, the notion of home advantage is recognised across majority of sporting events, both group and individual ones. To assess the importance

[5]Note that although tree-based methods can handle qualitative predictors, my dataset already included these categories in terms of dummy variables so I proceeded with the data in this format.

of home advantage on the outcome of a match, two dummy variables are considered in the proceeding analysis, each indicating if the given player's country of origin coincides with the tournament location.

3. **Average of Aces per Match**

   For each of the post-1991 matches in the initial dataset, a statistic indicating the amount of aces a player has served during the given match was recorded. Given that this information is naturally not available before the match, it cannot be used for match-predicting purposes. However, the average value of aces scored by each player in past matches can be used to assess the long-term performance of a player with regard to this measure.

4. **Average of Double Faults per Match**

   Analogously as in the case of aces, the average amount of double faults scored by each player in their respective past matches is calculated and included as an explanatory variable in the preceding analysis.

5. **Target**

   Lastly, a binary target variable is created and can be represented according to the form in equation (1) below. To identify Player 1 in a given match, the last names of the contestants are compared, and the one that comes first alphabetically is designated as Player 1.

$$Target = \begin{cases} 1 & \text{when Player 1 wins} \\ 0 & \text{when Player 1 loses} \end{cases} \tag{1}$$

2.3 Final Dataset

Having utilized as much of the information in the initial dataset as possible, several further reductions are applied. Firstly, the observations corresponding to matches played before 1991 are excluded. Secondly, matches played as part of the Davis Cup, a tournament where players represent their country in a team event, are omitted as they do not have a stable format. Lastly, all rows containing an NA value are

removed.

The resulting final dataset consists of 7 810 observations and 24 variables (including the target variable). In order to train the machine learning models in the proceeding analysis, a training set needs to be selected. For this purpose, 5 467 (70%) observations are drawn at random to comprise the training set. The remaining 2 343 (30%) observations are as a test set for performance comparison of the different models and ensemble methods used in this paper.

3 Methodology

3.1 Decision Trees

Given the aim to analyse the performance of tree-based methods in tennis match prediction and the binary nature of the target variable, the initial considered model is a classification decision tree. Using the the *tree* function from the *tree* package, a decision tree is fitted on the training set (Ripley, 2023). To determine optimal cut-point for splitting the predictor space, both the cross-entropy (*split* set to deviance) and Gini index (*split* set to gini) are applied. Equations (2) and (3) below present the mathematical formulations of the two respective measures in the case of binary classification, where $\hat{p}_{mk}$ represents the proportion of training observations in the m^{th} region that are from the k^{th} class (where k $\in$ {0,1}) (James, Witten, Hastie, & Tibshirani, 2021).

$$D = -\hat{p}_{m1} \log \hat{p}_{m1} - (1 - \hat{p}_{m1}) \log(1 - \hat{p}_{m1}) \tag{2}$$

$$G = \hat{p}_{m1}(1 - \hat{p}_{m1}) + (1 - \hat{p}_{m1})\hat{p}_{m1} = 2\hat{p}_{m1}(1 - \hat{p}_{m1}) \tag{3}$$

Given the tendency of a decision tree to become too complex and, hence, overfit the data, pruning is used to reduce the amount of splits in the initial trees. Using the *cv.tree* function from the *tree* package, a 10-fold cross-validation is performed, and the optimal tree complexity is determined based on the number of misclassifi-

cations.

3.2 Boosting

One of the main disadvantages of decision trees is that they suffer from high variance. In comparison to low variance methods, which produce similar results when applied to different datasets, high variance causes strong sensitivity to training data. In order to reduce the variance and obtain improved predictive accuracy, several ensemble methods can be used. The first such method is boosting. In boosting, the trees are grown sequentially, each subsequent tree being fitted of a modified version of the original dataset (James et al., 2021). In order to find an appropriate boosting model, one that picks up patterns in the data but does not overfit, suitable values of parameters controlling the learning need to be determined. In the case of boosting, the hyperparameters considered in the tuning process, and well as their associated values, are presented in table 1 below.

Parameter name	**Description**	**Considered Values**
n.trees	Integer specifying the total number of trees to fit.	400, 800, 1500
interaction.depth	Integer specifying the maximum depth of each tree.	1, 2, 4, 8
n.minobsinnode	Integer specifying the minimum number of observations in the terminal nodes of the trees.	1, 4, 8

Table 1: Boosting Parameter Tuning

Given the considered values presented in table 1 above, together with a shrinkage set to 0.01, a grid of all possible value combinations is considered in the tuning process. To perform the tuning, the *caret* package is utilised (Kuhn & Max, 2008). More specifically, the *trainControl* function, defining a 5-fold cross-validation, as well as the *train* function, specifying a gbm method, accuracy metric and Bernoulli distribution are used. Given that the gbm is chosen, a Gradient Boosting is per-

formed.

3.3 Bagging

The second ensemble method used to improve the predictive performance is bagging. In comparison to boosting, bagging utilizes the power of bootstrapping to decrease the variance by aggregating predictions obtained from several different training subsamples. More specifically, bagging consists of repeatedly sampling observations from the training set, which are then used to fit the model and predict the outcome class. Given the whole set of predicted values from each bootstrapped subsample, the final prediction is obtained by applying the majority vote.

Similarly as in the case of boosting, several parameters are tuned to appropriately control the learning process. In table 2 below, parameters and associated values considered in the tuning of bagging model are shown.

Parameter name	**Description**	**Considered Values**
ntree	Number of trees to grow.	800, 1000, 3000, 5000
sampsize	Size(s) of sample to draw.	500, 800, 1200, 2000
nodesize	Minimum size of terminal nodes.	1, 3, 5, 10
maxnodes	Maximum number of terminal nodes trees in the forest can have.	8, 15, 30, 500

Table 2: Bagging Parameter Tuning

Similarly as before, all possible combinations of the considered variables are assessed, using the *randomForest* function from the *randomForest* package to train the bagging models (specifying sampling with replacement) (Liaw & Wiener, 2002). However, in contrast to boosting procedure, bagging uses the out-of-bag measure to compare performance of the different model specifications.

3.4 Random Forests

Given the objective to decrease variance in tree-based methods, it is important to consider how correlation in predictors can affect the extend of variance reduction

resulting from bagging. According to James et al. (2021), in case of highly correlated variables, bagging does not significantly diminish the variance compared to an approach using a single tree. Therefore, decorrelation of trees

Therefore, to determine a suitable random forest model, the hyperparameter tuning process follows the bagging structure summarised in table 2 above, but includes an additional parameter *mtry*, which represents the number of variables randomly sampled as candidates at each split. In the tuning process, the following values are considered: 2, 4, 7 and 10.

3.5 Evaluation Methods

Given the best performing specifications determined by boosting, bagging and random forest, the test set performance of these ensemble methods can be compared using several measures. Firstly, the accuracy, indicating the proportion of correctly specified matches out of the total amount of (test set) matches, is a natural way to assess performance of classification models. However, in cases when the cost of false positives in high, the precision measure, calculated as the proportion of true positive predictions out of all positive predictions, might be preferable. In contrast, when the cost of false negatives is high, proportion of true positive predictions out of all actual positives, given by the recall measure, might be more suitable. Alternatively, to balances the trade-off between precision and recall, the F1 score can be used. The last measure for comparison of the three ensemble methods is the ROC curve (together with the value of the area under the curve), plotting the true positive rate (recall) against the false positive rate.

3.6 Variable Importance

Several measures can be used to assess the importance of variables given by each model. Firstly, the SHapley Additive exPlanations (SHAP) value, based on the notion of Shapley value in coalitional game theory, can be uses. According to Molnar (2022), the SHAP measure aims to explain prediction of a given instance by computing contributions of every feature to the prediction. As a result, features with

higher SHAP values contribute to the obtained prediction more, indicating a higher importance. To calculate and visualize the SHAP values, the *kernelshap* and *shapviz* functions are used (Mayer, 2024; Mayer & Watson, 2024).

Alternatively, the *importance* function can be used to assess the variable importance in random forest models.

4 Results

4.1 Decision Trees

Firstly, the results of fitting a simple decision tree are discussed. In Figure 1 below, a decision tree with splits determined by cross-entropy is displayed. In the construction of this tree, in total four variables were used, all corresponding to serving statistics (p1_df_avg, p1_ace_avg, p2_df_avg, p2_ace_avg). In comparison, the decision tree with splits based on the Gini Index includes 14 additional variables in it's construction. The plot, however, cannot be displayed due to it's complexity.

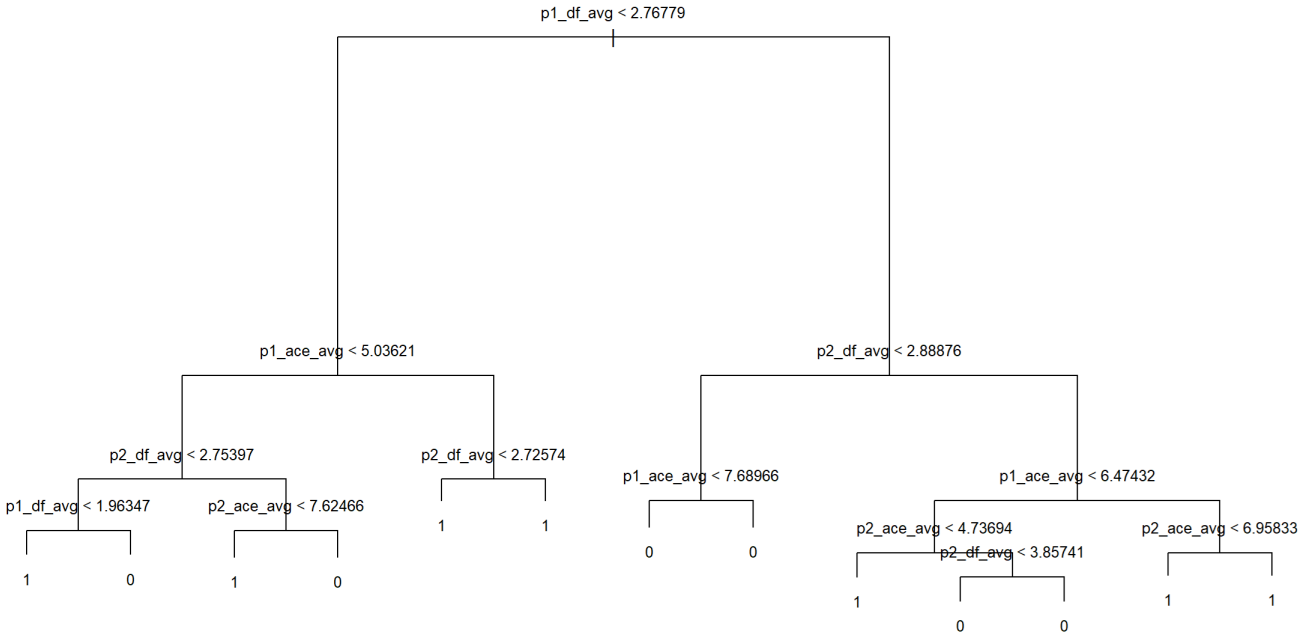

Figure 1: Decision Tree Based on Cross-Entropy

When using the cross-validation to prune the tree (determine the optimal tree complexity), the cross-entropy based tree chooses an optimal size of 8, while the

gini-based tree results in a size of 100. In table 3 below, the training and test set accuracy results for the simple as well as pruned decision trees are summarised. While the gini-based trees outperform the associated cross-entropy-based trees in terms of the training accuracy, they present worse results in when considering the test set predictions. Given the fact that the gini-based trees have a much higher complexity compared to the cross-entropy counterparts, these accuracy results signal overfitting in the gini-based models.

Method	Training Accuracy	Test Accuracy
Decifion tree (cross-entropy)	82.59	80.67
Decifion tree (Gini Index)	89.12	73.67
DT (crross-entropy) pruned	81.95	80.92
DT (Gini Index) pruned	81.34	69.88

Table 3: Training & Test Accuracy Results For Simple Decision Trees

4.2 Ensemble Methods

Table 4 below summarizes results of the hyperparameter tuning for the three ensemble methods. Considering the results, it seems that rather large values

Ensemble Method	Tuned Values
Boosting	n.trees = 1500, interaction.depth = 8, n.minobsinnode = 4
Bagging	ntree = 5000, sampsize = 2000, nodesize = 3, maxnodes = 500
Random Forest	ntree = 5000, sampsize = 2000, nodesize = 1, maxnodes = 500, mrty = 10

Table 4: Summary of Hypeparameter Tuning Results

Using these model specifications, test performance can be calculated and compared using the 5 performance measured described in section 3.5. While in the case of tennis match predictions the cost of false positives or negatives is likely

not too high, meaning the accuracy measure would be sufficient, the remaining measures can provide indication of possible shortcomings of the models in terms of their predictive power of the given classes. In table 5 below, the test performance results are displayed.

Performance measure	Boosting	Bagging	Random Forest
Accuracy	89.80	87.41	87.28
Precision	90.41	88.98	88.57
Recall	90.35	87.05	87.29
F1 score	90.38	88.00	87.93
AUC	0.96	0.95	0.95

Table 5: Test Performance Results of Ensemble Methods

Based on the results in table 5 above, several observations can be noted. Firstly, all three of the ensemble methods outperform the simple decision trees discussed in the previous subsection. Secondly, amongst the ensemble methods, boosting results in the highest predictive accuracy using all five of the performance measures. What is more surprising, however, is the similarity in performance between the bagging and random forest models. Based on the accuracy, precision and F1 score, bagging even slightly outperforms random forest. One reason for this would be an improper tuning of the *mtry* parameter. The selected value of 10 is fairly high compared to the total amount of 20 parameters. However, given that the out-of-bad error of this random forest model specification is approximately 10.97, translating to an accuracy of 89.03, a comparable value to the test set accuracy, it does not seem to be the case that the random forest model is overfitting.

To further assess the performance of these ensemble models, specifically in terms of their ability to match the true likelihood of the event occurring. This can be done by assessing the calibration curves, which are respectively depicted for the boosting, bagging and random forest models in figures 2, 3 and 4 below.

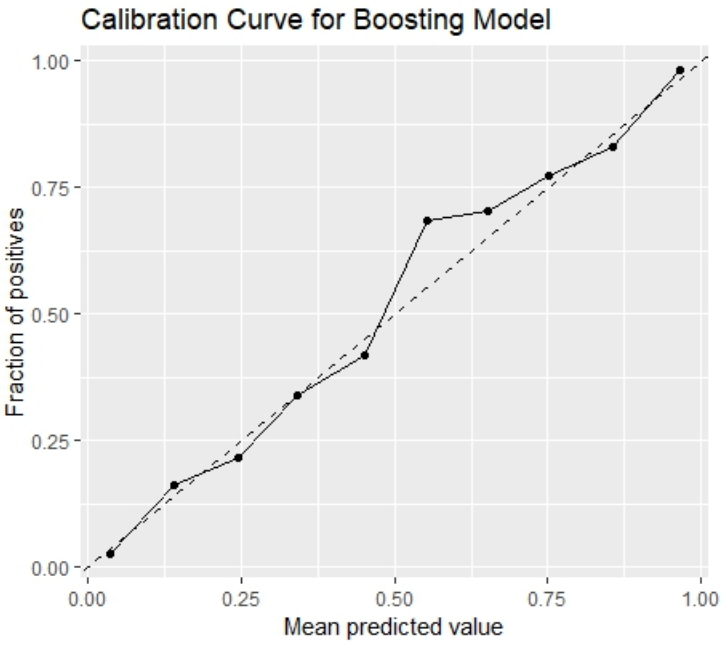

Figure 2: Calibration Curve of Boosting Model

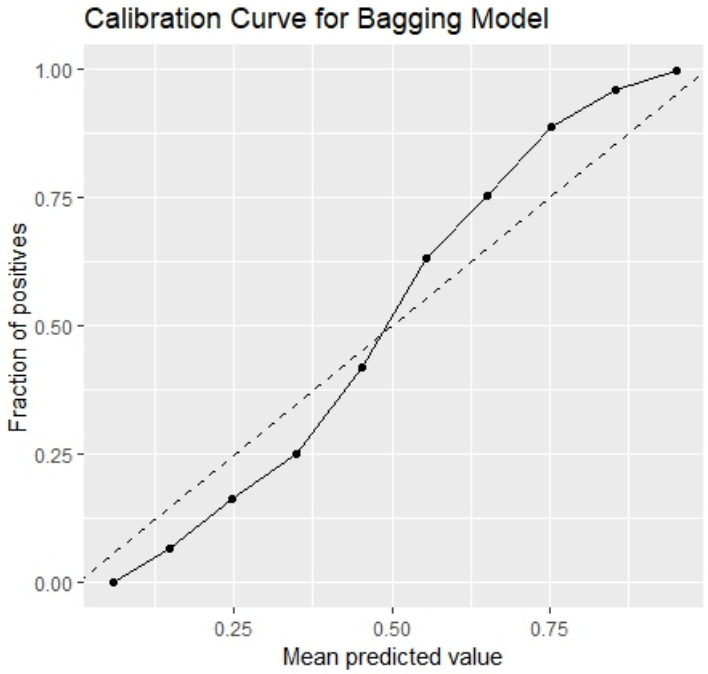

Figure 3: Calibration Curve of Bagging Model

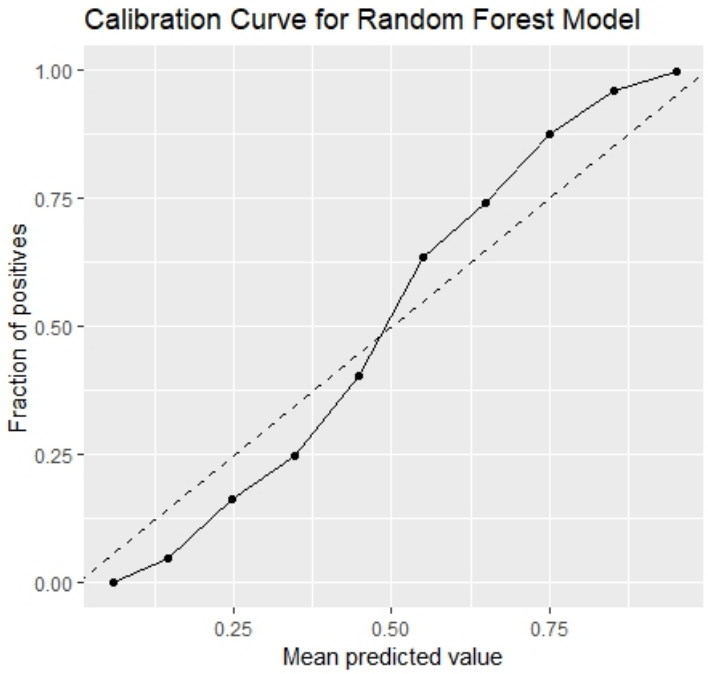

Figure 4: Calibration Curve of Random Forest

When the calibration curves lie directly on the diagonal line, it mean that the model is perfectly calibrated (if a model predicts x% of winning, the player actually wins x% chance of the time). Given the plots above, we can see that the the random boosting model is calibrated almost perfectly in case of the mean predictive value being below a half. However, around a mean predicted value of approximately 0.55, the model tends to underestimate the probability of the positive class (signaled by the curve lying above the diagonal line). In cases of the bagging and boosting models, they tend to symmetrically overestimates the probability of the positive class and underestimate the probability of the positive class.

4.3 Variable Importance

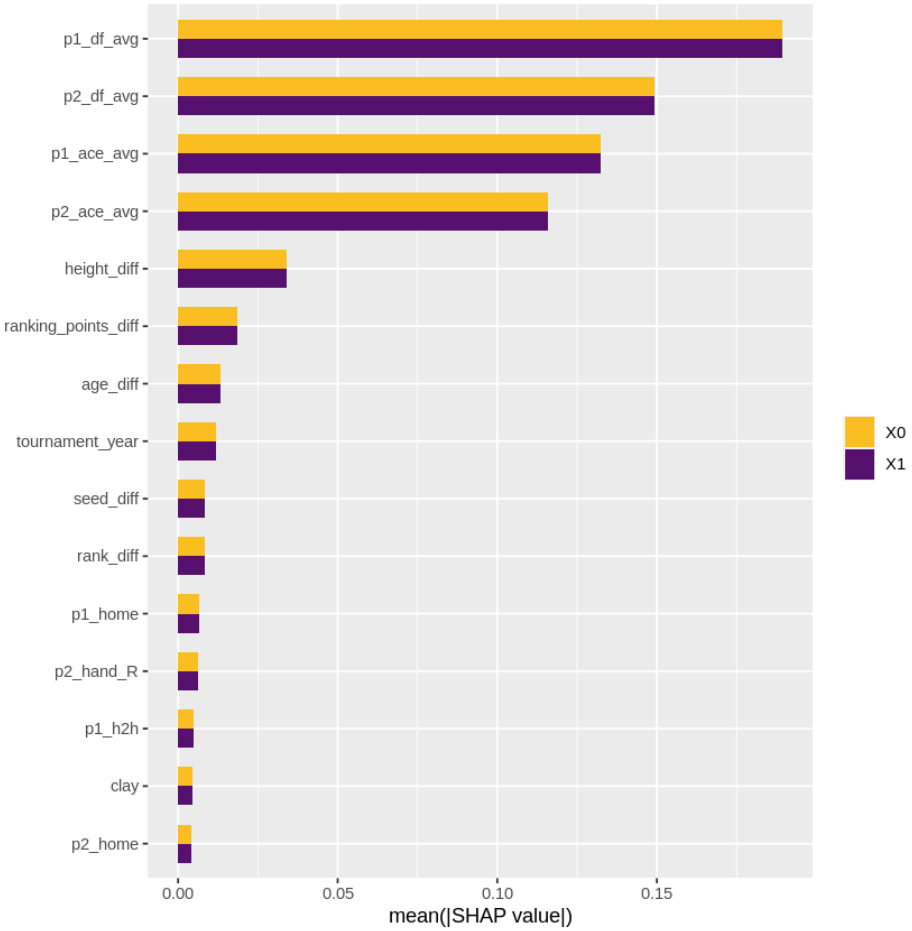

Figure 5: SHAP of the Boosting model

Looking at the SHAP value of the best-predicting model in figure 5 above, it can be seen that all variables play equal role in predicting the positive and negative class. Moreover, the four serving-related statistics (p1_df_avg, p1_ace_avg, p2_df_avg, p2_ace_avg) are identified to contribute the most to the predictions.

Moreover, feature importance in the bagging model can be evaluated based on figure 6 above. Given that the black lines represent the confidence intervals, it can be seen that the bagging model identifies the following six features to have significant importance: p1_df_avg, p1_ace_avg, p2_df_avg, p2_ace_avg, ranking_points_diff and, lastly, tournament_year.

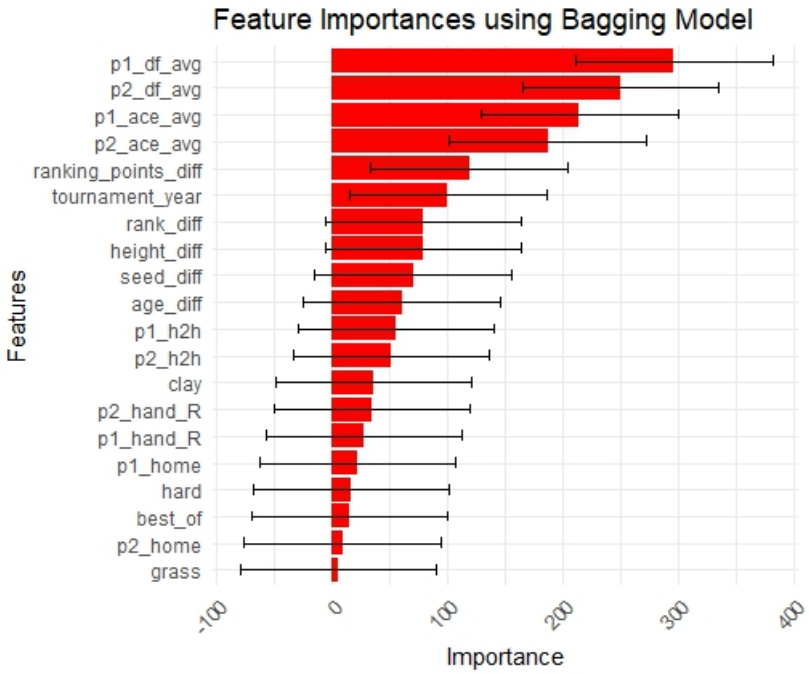

Figure 6: Feature Importance using Bagging Model

5 Limitations

In case of the hyperparameter tuning processes in all three ensemble methods, the considered values of tuned variable were chosen based on consideration of the sample size, the amount of parameters, but also suggestions given by Rooij (2021). Moreover, in cases when maximal value of the considered range was chosen as the best performing one, the given range was expanded and model was re-tuned, but estimation efficiency had to be taken into account. While the considered values were selected carefully, given the sensitivity of the ensemble models to the chosen parameters, this process can play a significant role in the resulting model performance. Therefore, given more computational power, the analysis in this paper would benefit from more thorough tuning process.

6 Conclusion

This paper uses 30 years of ATP match data to analyse the predictive performance of tree-based models, as well as the boosting, bagging and random forest ensemble methods. Based on the comparison of several performance measures, the boosting method provides the most accurate predictions for the outcomes of male tennis matches. Moreover, bagging and boosting perform very similarly to one another, each slightly outperforming the other based on the chosen assessment measure. In terms of variable importance, while the boosting model identifies the ace and double fault serve statistics to have the highest contribution for match prediction, the bagging model also finds the difference in ranking points between the players, as well as the tournament year to be important match-predicting features.

References

Association of Tennis Professionals. (n.d.). *The 2024 atp official rulebook.* Retrieved on May 7, 2024, from https://www.atptour.com/-/media/files/rulebook/2024/2024-rulebook_30apr.pdf

del Corral, J., & Prieto-Rodríguez, J. (2010). Are differences in ranks good predictors for grand slam tennis matches? *International Journal of Forecasting*, *26*(3), 551-563. Retrieved from https://www.sciencedirect.com/science/article/pii/S0169207009002076 (Sports Forecasting) doi: https://doi.org/10.1016/j.ijforecast.2009.12.006

Gorgi, P., Koopman, S. J., & Lit, R. (2019). The analysis and forecasting of tennis matches by using a high dimensional dynamic model. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *182*(4), 1393–1409. doi: 10.1111/rssa.12464

Hamingson, N. (2024). *Tennis court surfaces compared: What are the differences?* Retrieved from https://www.redbull.com/se-en/tennis-court-surfaces-grass-clay-hard-court

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An introduction to statistical learning with applications in R* (2nd ed.). Springer.

Kuhn, & Max. (2008). Building predictive models in r using the caret package. *Journal of Statistical Software*, *28*(5), 1–26. Retrieved from https://www.jstatsoft.org/index.php/jss/article/view/v028i05 doi: 10.18637/jss.v028.i05

Liaw, A., & Wiener, M. (2002). Classification and regression by randomforest. *R News*, *2*(3), 18-22. Retrieved from https://CRAN.R-project.org/doc/Rnews/

Mayer, M. (2024). shapviz: Shap visualizations [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=shapviz (R package version 0.9.3)

Mayer, M., & Watson, D. (2024). kernelshap: Kernel shap [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=kernelshap (R

package version 0.5.0)

Molnar, C. (2022). *5.10 shap (shapley additive explanations) | interpretable machine learning.* Retrieved from https://christophm.github.io/interpretable-ml-book/shap.html ([Online; accessed 14-June-2024])

Ovaska, T., & Sumell, A. J. (2014). Who has the advantage? an economic exploration of winning in men's professional tennis. *The American Economist*, *59*(1), 34-51. Retrieved from https://doi.org/10.1177/056943451405900104 doi: 10.1177/056943451405900104

Ripley, B. (2023). tree: Classification and regression trees [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=tree (R package version 1.0-43)

Rooij, C. V. (2021). *Machine learning in tennis: Predicting the outcome of a tennis match based on match statistics and player characteristics.* https://arno.uvt.nl/show.cgi?fid=158548.

Sackmann, J. (2017). *How much does height matter in men's tennis?* Retrieved from https://www.tennisabstract.com/blog/2017/09/04/how-much-does-height-matter-in-mens-tennis/ ([online] Heavy Topspin. Available at: https://www.tennisabstract.com/blog/2017/09/04/how-much-does-height-matter-in-mens-tennis/)

Appendix A

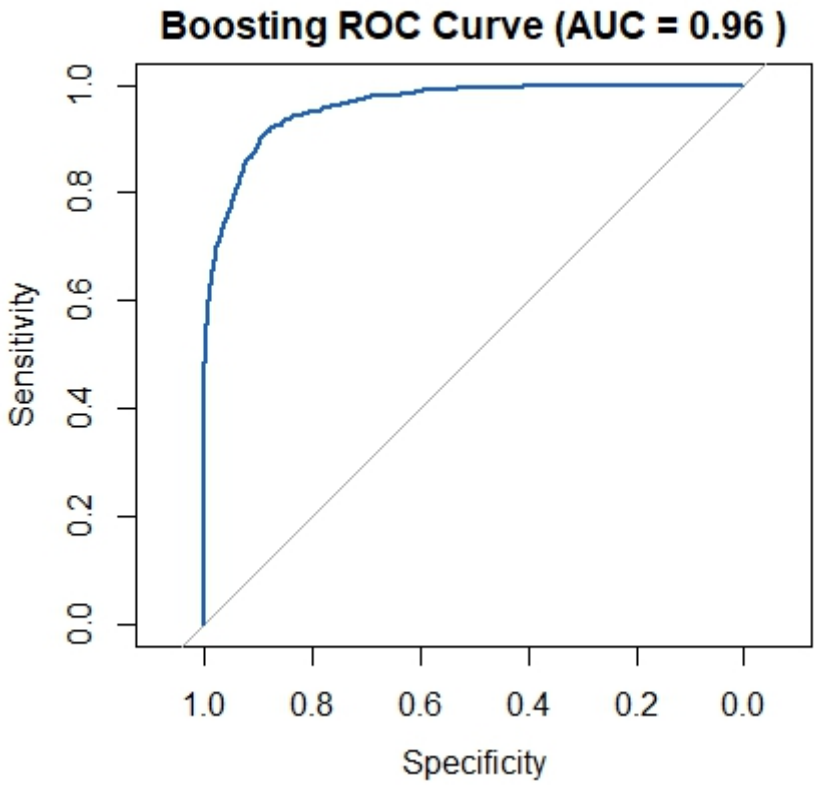

Figure 7: The ROC of boosting model.

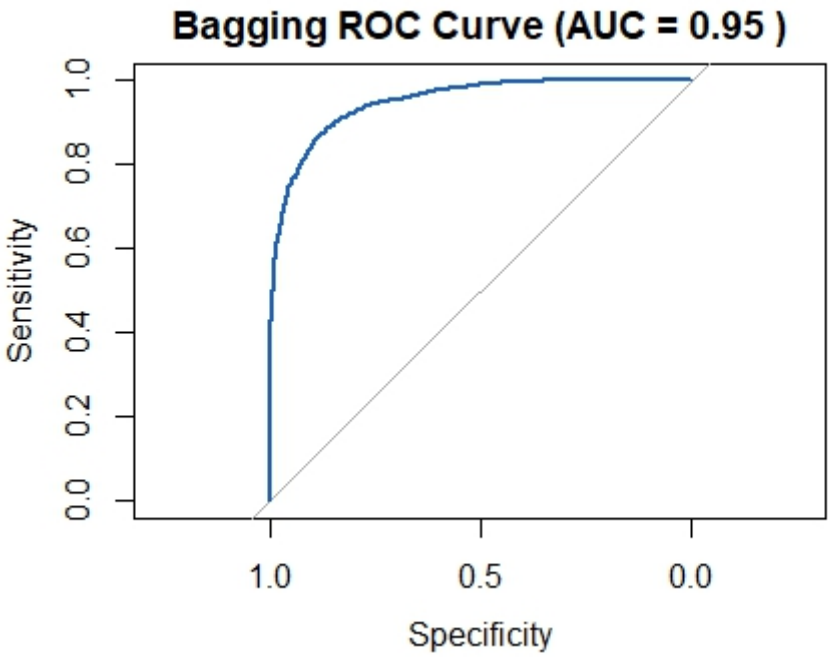

Figure 8: The ROC of bagging model.

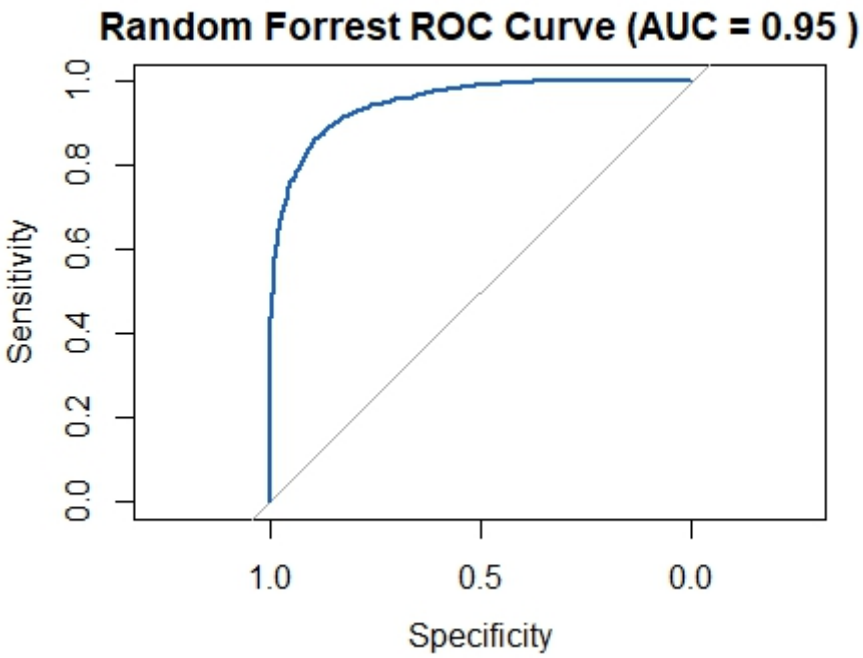

Figure 9: The ROC of random forest model.