Tennis Match Predictions using Machine Learning

Namya Bagree, Sohan Kulkarni, Aishwarya Ravi, Marianna Anagnostou
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA-15213
{nbagree, sohank, aravi2, managnos}@andrew.cmu.edu

Abstract

The objective of this paper is to investigate the applications of Machine Learning algorithms on predicting the outcomes of tennis matches. Along with the match predictions, this paper aims to use these outcomes to also outperform betting odds of a betting scheme based on historical player wins and losses. This paper also aims to determine who is the greatest tennis player of all time by comparing players in their respective peak performance period. To achieve such goals, a pipeline was created to make use of historical tennis match data, feature engineering and four machine learning methods: Logistic Regression, Random Forests, Support Vector Machines, and Gaussian Discriminant Analysis. This pipeline's accuracy and F1 scores would also be evaluated for 20 tennis players in comparison to prior methods of predictions.

1. Introduction

Tennis is widely regarded as one of the most popular sports, having a hierarchical method of scoring, wherein the players pitted against one another play matches, with each match comprising several sets. In addition to this, the tournaments in tennis are generally categorized based on genders and consist of Men's Singles, Women's Singles, Men's doubles, Women's doubles and Mixed Doubles with the singles comprising one player on either side of the net and the doubles comprising of two players on either side of the net. The Grand Slam Tournaments comprising of US Open, French Open, Wimbledon and Australian Open usually bring in the largest crowds to see the players battle it out on the court to vie for the title of the best player.

With the popularity of Tennis rising immensely, it has led to a large number of people contesting in the online betting market with the betters betting on the outcome, and winning the bet if the tennis match matches their prediction. Record-keeping of tennis match data and the usage of Machine Learning algorithms to predict match outcomes has led to the application of these statistical mathematical models to be applied to aid the betting in their favor. Betting frameworks usually comprise of two types, the first case could be that before a match even begins, the odds are published. And the second case is when a calculation of the odds is done after the betters' betting amounts are decided.

Our main task at hand was to create a parameterized model that predicts the outcome of tennis matches accurately, to determine the greatest player of all time (GOAT) and use those predictions to determine the odds in betting. To create such a model, the raw data set needs to be cleaned and the unnecessary features removed to prepare the data for training. Feature engineering is applied with the appropriate weights being defined, as detailed in the further sections. The Machine Learning algorithms used to test and compare the accuracy of the model are Logistic Regression, Support Vector Machine, Gaussian Discriminant Analysis and Random Forest. By running the model, one of the results that was obtained was the payoff per game for 20 different players, by defining simple betting odds comprising of the players' wins and losses. In addition to this, the peak period for a player or in other words, the period for the player at which they were at their best performance was determined along with the answer to the question as to who is the best player of all-time in men's tennis from 1996-2017.

2. Related Work

Previous studies exist which attempted to create models that accurately predict the outcome of Tennis matches. Multiple machine learning methods have previously been applied to improve predictions of tennis match outcomes and in turn, use the outcomes to improve sports betting models. In particular, as mentioned in [18], supervised learning was implemented to render the training data and extract the necessary features to use in the prediction models. Subsequently, logistic regression and neural networks algorithms were used to optimize the outcomes and therefore, propose a profitable betting scheme. One of the methods implemented in [17] is noteworthy as a network model is created

to find out those players that played well in a certain kind of surface and created an algorithm known as prestige score that would detail the importance of every player. In [11] an algorithm based on fuzzy logic is implemented that uses a flexible statistical model and predicts the winning player during the match and not before it.

Historic data is utilized in [5] to find out the chances of the player obtaining a single point in a match and then elaborating further on that to determine that players' probability to win the whole match with a 6.8 percent ROI for the grand slam matches. For this model, misclassification is studied using the Bayesian error learning rate. In [14] logistic regression was executed for match characteristics and player characteristics on multiple variables and their final model yielded a log loss of 0.61. An accuracy of 75 percent was obtained in [1] when artificial neural networks using first-serve percentage as the main feature was used. Additionally, physical characteristics in a player such as their weight, age and height such as in [13] are found to be the factors that affect a player's performance. To account for the player's skills, the player serve strength is accounted for in [15] which is similar to what we have incorporated in our model. In [10] it is indicated that player success depends more on the player's performance over a large period and not upon the factors that influenced a match on a certain day, such as the impact of weather. An attempt was made to increase the number of variables [19] to enable prior planning and the determinants' range for a winning player in a match.

Cornman[2] tried to maximize the expected earnings by making the decision problem as a single-shot for every match, by betting a dollar on every player which is similar to the model that we have created. By doing so they were able to obtain a profit of 3.3 percent for every match and an accuracy of 69.6 percent on the test data from 2016-2017 matches included in their data set. [21] identified serve-strength as the dominant feature affecting their model. They obtained an accuracy upwards of 80 percent and combined the outcomes predicted from three models and tried to distribute the probabilities of the betting odds in the form of a curve. This would allow them to compare such results with the odds provided by the betting companies and conclude that the betting companies were also distributing the outcomes in the same fashion. AIC and BIC criteria were utilized in [3] along with logistic regression to obtain the correct predictions of tennis match outcomes for 93 percent of the cases in the test and training data-set and defined the threshold as fifty percent as the criteria for the loss of the first player. [6] utilized a model that predicted the player skills to reach the quarterfinals of a tournament and hence, measure their success rate in that particular tournament. To determine the ranking of the player for that particular tournament, they used 5 different variables through an augmented linear programming approach to select the desired tournaments optimally.

To summarise, our model utilizes features such as aces first serves in, surface type, double faults, sets won, service games won, tournament name, match result such as in [7] with the application of Logistic Regression, SVM models [16], Gaussian Discriminant analysis [12] and Random forest [9] to compare the accuracy and predict the betting odds.

3. Data

Historical tennis data is widely available online. Websites such as atpworldtour.com provide access to information about players, the outcomes of ATP matches and statistics related to player performance in particular matches. The data set used for this project is the ATP World Tour tennis data available for public use on https://github.com/serve-and-volley/atp-world-tour-tennis-data. The initial data set contained numerical data and statistics from more than 99,500 games in Men's tennis from 1991-2017 with more than 60 features. Some of the key features available included match statistics such as first serves in, aces, double faults, service games won, sets won, etc, along with match details such as tournament name, round, surface type, match result, etc. The match details data and the match stats data were combined by matching the match ID to provide one big data set with all the information stored.

The raw data set created contained a fair number of inaccuracies and required a few steps of cleaning before the model could be trained on it. Rows with missing or NaN values were dropped. Along with this, certain games ended prematurely due to player retirement or did not occur due to a walkover. This data was also dropped as it would adversely affect the model during training. The data set also contained unnecessary columns that would not be considered as features in the ML model. Table 1 displays the list of features extracted from the data. Each feature (apart from the Win Margin) was determined for both the winner and the loser for each game. The following features were dropped from the analysis:

Match Duration: This feature was dropped as the historical match duration would not help in any way to predict the winner for a particular game. Match duration can come into play for real-time in-game predictions. Certain players perform better in long matches and historical match duration data can help predict the winner of a particular game after the first or second set, however, for the scope of this project, this feature was dropped.

Total Serve Points: This feature was deemed to be unnecessary as the model already considers the service points won and the service point win percentage.

Total First Serve Points: Similar to total serve points, total first-serve points were dropped as the model considers first serve win percentage.

SNo.	Feature Name	# Features
1	Win Margin	1
2	First Serve In Percentage	2
3	First Serve Win Percentage	2
4	Second Serve Win Percentage	2
5	Service Point Win Percentage	2
6	Ace Percentage	2
7	Double Faults Percentage	2
8	Break Point Conversion Percentage	2
9	Total Break Points	2

Table 1. Results

First and Second Serve Return Points: The data set assumes only aces as unreturned serves. Hence considering ace percentage will also take care of returned and unreturned points.

Service and Return Games Played: This feature was dropped as the service and return games played has a direct linear correlation to the difference in points won or difference in sets won. Hence, only one of these features was considered.

With the data cleaning and pre-processing executed as mentioned, the number of games taken into consideration reduced to about 92,000 games and the features considered reduced to 18 features. This cleaned data ensured that the ML model will not be overfitted due to excessive features and the computation time was reduced without affecting the model accuracy.

4. Methods

To make accurate tennis match predictions and subsequently, use such predictions to define betting odds and the best player of all time, we decided to create a parametrized model. This model would use four machine learning algorithms that would take into account player performance features from the cleaned historical tennis match data and predefined weights to tune their predictions.

Initially, the model was designed to be run on the entirety of the data set, however, we soon realized that this approach was not producing desirable results. Instead, we decided to select individual players, pit them against each other and run the model to predict which player would win or lose. Following this alternative method provided results that were indeed accurate and thus, allowed us to use the model and solve the problems that this paper aimed to address.

4.1. Logistic Regression

Logistic Regression was used to classify the players into winners and losers. The hypothesis model is the following:

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}} \tag{1}$$

where θ^T are the weights and x are the features.

The hypothesis gives out values ranging from 0 to 1, probabilities to be used in classification. In order to get the best predictions for match outcomes, we need to use the Maximum Likelihood Estimation method to optimize θ which results in the following function:

$$L(\theta) = \sum_{i=1}^{N} y^{(i)} log(h(x^{(i)})) + (1 - y^{(i)}) log(1 - h(x^{(i)})) \tag{2}$$

where N is the number of training samples and $y^{(i)}$ is the predicted probability of a win[18].

4.2. Random Forests

Random forests are a supervised learning method for classification, regression and other tasks. They make use of multiple decision trees during training and output the class that is predicted. Using classification or mean prediction of decision trees, random forests prevent overfitting from occurring to the training set and allow for better predictions[4].

4.3. Support Vector Machines

A Support Vector Machine (SVM) model is another supervised learning method algorithm that represents data as points in space. Such data points, which are ultimately classified, are plotted in space divided by a gap as wide as possible. An unseen example can then be "mapped into that same space and predicted to belong to a category based on which side of the gap it falls" [20]. Therefore, SVM was used to make classifications of tennis matches and predict their outcome[20].

4.4. Gaussian Discriminant Analysis

"Gaussian discriminant analysis (GDA) is a generative model for classification where the distribution of each class is modeled as a multivariate Gaussian"[8]. The Gaussian Distribution is defined below:

$$P(z) = \frac{1}{(2\pi)^{1/2}|\Sigma|^{1/2}} \times exp(\frac{-1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)) \tag{3}$$

Although generative models such as GDA need less data and add flexibility to decision boundaries, they can often make non-accurate predictions in cases where the data being mapped does not follow a Gaussian Distribution.

4.5. Weights Definition

Three weights were defined to aid our feature engineering by accounting for time discounting, head to head games and surface differences.

Surface Type	Hard	Clay	Grass
Hard	1	0.28	0.24
Clay	0.28	1	0.14
Grass	0.24	0.14	1

Table 2. Surface Type Weight Correlations

4.5.1 Time Discounting Weights

We want to make use of a player's current form and predict results based on that and not just all of their historic games. More recent matches represent the player's current performance more accurately. Due to this observation, we used weights pertaining to time discounting, meaning that tennis matches played more recently were given a higher weight. We assumed an exponential decay of this weight. The earlier a game was played, the lower a value it would have. The factor W_T can be modified to increase or decrease the importance of a game. $T_{current}$ is the year the match takes place and T_{game} is the year the match being used to create the features took place. The equation for this weight follows an exponential function as shown in Equation 4.

$$W(T) = e^{-(T_{current} - T_{game})/W_T} \tag{4}$$

4.5.2 Head to Head Weights

It often happens that two players will have played each other in the past. For that reason, it is important that such a factor be taken into consideration. Players who have played each other in the past will receive a different weight than players who are playing each other for the first time as shown in Equation 5.

$$W(H2H) = \begin{cases} W_{H2H}, & \text{if Head to Head game} \\ 1, & \text{otherwise} \end{cases} \tag{5}$$

4.5.3 Surface Weights

Tennis courts in ATP tournaments have four unique surfaces: Hard, Clay, Carpet, and Grass. Surfaces have an impact on a player's performance due to the playing style adjustments one needs to make. Therefore, when attempting to predict the outcome of a player's game, it is important to take into account that player's performance history pertaining to a particular surface. Table 2 was used to define the surface correlation weights $W(Surface)$ between the unique surfaces.

For simplicity purposes, we classified Carpet as a Grass surface due to similarities in ball bounce and speeds.

4.5.4 Final Weight Definition

To define our final weight for every past game for the player, we will multiply all the weight defined here as shown in equation 6.

$$W = W(T) \times W(H2H) \times W(Surface) \tag{6}$$

4.6. Feature Extraction

We made use of the cleaned data as described earlier.

4.7. Feature Difference

Each of the features extracted from the data as shown in Table 1 does not imply much if used on its own. It would instead make more sense to either take a ratio or difference of the values. As most of the features are already in a percentage format, it made more sense to take the difference of each feature for a player and their opponent. Hence for each game, the features (other than Win Margin) were takes as shown in Equation 7.

$$\text{Feature(Diff)} = \text{Feature(Winner)} - \text{Feature(Loser)} \tag{7}$$

By virtue of selecting the difference of the winner and loser, we appropriately define the sign based on whether the player being searched for won or lost that match.

4.8. Historic Feature Grouping

For every game, we searched through the data for each player's prior game stats and appended them into a table. Based on whether the player won or lost that match, the features would be multiplied by 1 or -1 respectively (to take into account the Feature Difference definition). The weights defined earlier were then multiplied with the features from every game. This created a new set of features that we can make use of as part of our test set (the current game to be predicted).

4.9. Learning Workflow

In prior literature, the models have been defined on entire datasets and not with respect to a certain player. This model formulation tends to take away personal player features that could help predict the winner based on the additional feature engineering used. We defined a model that where the features made use of each player's historic record and stats, weighted according to our prior definitions. As per prior literature, this model should perform better than a model with random prediction. The model then trains itself based on all matches irrespective of the players and then would test itself on a future match based on the features created for the match. This model ended up giving us very poor results. We had an average accuracy of 0.502 while the accuracy of a random prediction model would be 0.5. Modification of

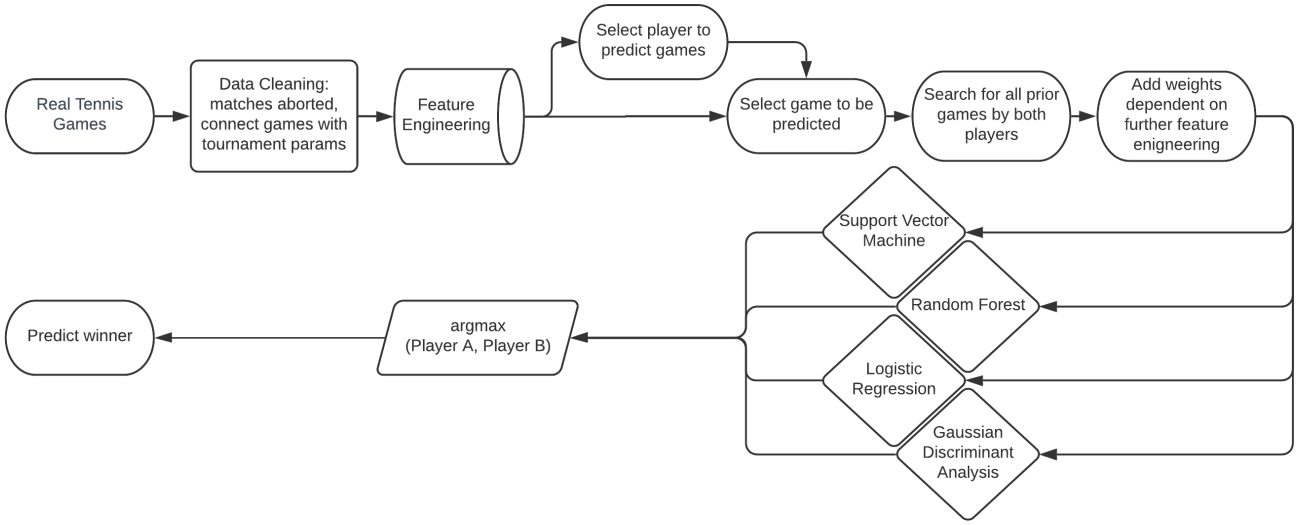

Figure 1. Novel Tennis Prediction Pipeline

weights did not improve our model either. A novel pipeline was defined and made use of to help the model make better use of personal player features. This would help the model understand a player's weaknesses and strengths and predict based on the player's performance against the opponent's historic features.

Once both the player and the opponent's historic features have been collected and weighted, the difference of the features is taken (other than Win Margin) and this will form the features for the current game. We now create two models, one for each player based on solely their respective historic games. These models are trained using sample weights for each game defined in the same manner as described earlier. We then use the test features for the current game on the trained Player Model. For the Opponent model, we multiply the features with -1 (to get the weights as per the opponent being the "Player") and use these features on the trained Opponent Model. Both models will now output a classification on whether the player or opponent should win and may also predict opposing results. We will hence make use of how confident each model is with their prediction and the model with a higher confidence level will predict the winner. In this manner, we convert two classification models into a single model. This formulation helps us understand the features that personally affect a player's game and search for their weaknesses.

We have now created a pipeline where we can predict any game by specifically creating the models for that particular game and making a prediction. Although this method is more time consuming as two models are trained for every game, it has the added advantages of the personal player features incorporation, having every available game as a test set, having flexibility in setting weights and also allowing us to create our own game between any player from any year on any surface, allowing us to potentially predict the greatest tennis player.

The models used in our pipeline are Logistic Regression, Random Forests, Support Vector Machines, and Gaussian Discriminant Analysis. These classification models made the confidence prediction of which player would be the winner/loser with the argmax function. These predictions were then benchmarked and used to establish the return on betting odds and the greatest player of all time.

The pipeline has also been summarized and shown in Figure 1.

5. Models and Results

We made use of four machine learning models to predict the winning player - Logistic Regression, Random Forest, Support Vector Classifier and Gaussian Discriminant Analysis. Along with this we also had the weight hyper-parameters that will influence the model predictions. We divided the hyper-parameters into 3 set of values - no weights, moderate weights and high weights. The pipeline was run for each model and weight to accurately predict the winner and has been discussed in the following sections.

5.1. Machine Learning Models

Once the model was run and we obtained results pertaining to its accuracy, we gathered data to formulate the return of a one dollar bet placed. It can be observed in Figure 2 that the Logistic Regression and Support Vector Machines algorithms had the highest bet returns, demonstrating the

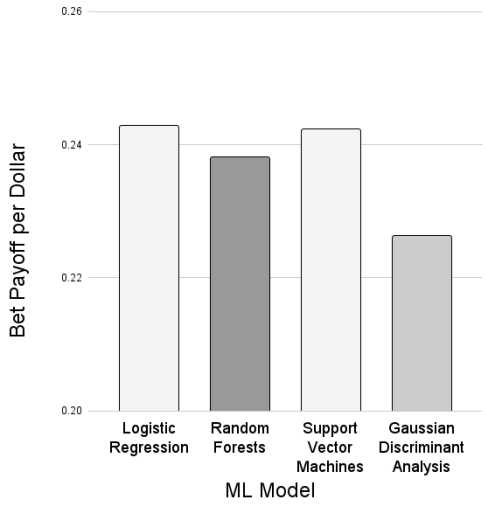

Figure 2. Performance of ML Models on Betting Payoff

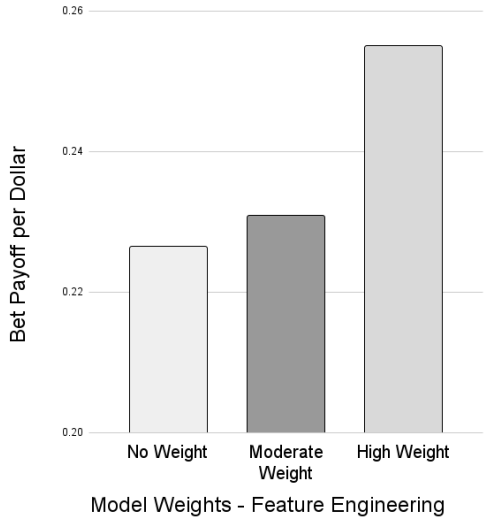

Figure 3. Performance of Weights on Betting Payoff

increased accuracy of these models in classifying the tennis players into winners and losers. In addition, Random Forests was a promising model. However, GDA consistently ended up giving the lowest returns and also had the lowest accuracy on tennis match predictions.

5.2. Weight Hyper-parameters

Similarly to the process followed in the previous section pertaining to the Machine Learning models, we determined the relationship between bet payoff and the weights given as parameters to the model. Three cases were tested: no weight, moderate weights ($W_T = 6, W_{H2H} = 8$), and high/aggressive weights case ($W_T = 2, W_{H2H} = 15$). It can be observed in Figure 3 that the higher the weights, the higher the dollar return, indicating the importance of the weights in our model. The no weight and moderate weight cases demonstrate smaller returns.

6. Experiments and Results

As seen in the prior section, our model formulation does not take into account any common train set and every prediction is a part of the test set. The experiments are created in order to predict three objectives - match prediction against betting odds, the best period for a player and the greatest tennis player. Each of these objectives make use of the same pipeline Method defined.

6.1. Match prediction against betting odds

We ran the following model for all games played by a certain player after the player has completed a total of 75 wins and losses. The payoff was calculated per game bet on for the respective player. We made use of simple betting odds defined by the number of games a player has won or lost. This is defined as follows:

$$\text{Games won = W, games lost = L} \tag{8}$$

$$\text{Winning odd = L/W, losing odd = W/L} \tag{9}$$

A very simple betting strategy has been used and that is to bet a dollar for or against the player at hand based on whether our model predicts that player to win or lose. The payoff for a certain game is as follows:

$$\text{Win prediction, player win: L/W} \tag{10}$$

$$\text{Win prediction, player lose: -1} \tag{11}$$

$$\text{Loss prediction, player lose: W/L} \tag{12}$$

$$\text{Loss prediction, player win: -1} \tag{13}$$

The total payoff is divided by the number of games a player has played so that we get the money earned per game (or dollar spent). A value of 0 signifies a random model whereas a value of 1 signifies a perfect model. The resultant payoff per game for 20 players is shown in Table 3.

6.2. Best period for a player

The best player for every player is found out by using the pipeline to create a proxy match between the same player on 2 different years. For uniformity, each year is counted at the first grand slam game of the year. Starting from the first 2 years of the player, the year that wins is kept and is

Player	#Games	Payoff/game	Accuracy	F1
Djokovic	286	$ 0.10	0.87	0.93
Federer	537	$ 0.17	0.86	0.92
Nadal	258	$ 0.26	0.83	0.90
Cilic	212	$ 0.32	0.69	0.75
Querrey	165	$ 0.23	0.61	0.59
Wawrinka	268	$ 0.29	0.68	0.75
Agassi	144	$ 0.11	0.77	0.87
Murray	278	$ 0.40	0.83	0.89
Sampras	49	$ 0.07	0.77	0.86
Roddick	130	$ 0.23	0.75	0.84
Berdych	355	$ 0.35	0.71	0.79
Haas	248	$ 0.23	0.66	0.76
Ferrer	426	$ 0.30	0.70	0.78
del Potro	91	$ 0.32	0.74	0.82
Hewitt	142	$ 0.25	0.62	0.67
Tsonga	161	$ 0.35	0.74	0.82
Ferrero	141	$ 0.23	0.63	0.69
Isner	163	$ 0.28	0.66	0.73
Moya	202	$ 0.18	0.65	0.74
Nishikori	116	$ 0.35	0.75	0.83
All	4375	$ 0.26	0.74	0.81

Table 3. Results for 20 players after their first 75 wins and losses

put against the next year until the player's final year. This analysis was done on the three top players of the past 15 years (Federer, Nadal, Djokovic). The top year for each of the players was as follows:
Federer: 2009, Nadal: 2011, Djokovic: 2015.
This is a statistical way of predicting when a player was at their peak. This information is also used in the next section to predict which of the 3 players had the best chance of winning had they all reached their peak together.

6.3. The greatest tennis men's player (1996-2017)

Emotionally, this argument is difficult to conclude upon, but we make an attempt to predict the greatest tennis player had they all arrived at their peak performance together. For all the top players, their peak performance year was selected from the earlier explained method. After this, there were proxy games created between players at their given peak period on multiple surfaces (grass, clay, hard). This method would predict the expected winning player through our pipeline. The ranks predicted on various surfaces for the top 3 players was as shown in Table 4 (note that Federer vs Djokovic on grass is given the same rank as our models were split deciding between the two, displaying how close the two players were statistically). Additionally, on Table 5, a more analytical representation of court dominance is demonstrated. Depending on the court surface, predictions were made on which player would win if pitted against one another. On grass, it is noted that both Djokovic and Federer are mentioned. That is because 2 of our models predicted Djokovic to win and the other 2 predicted Federer to win. This also displays how close a game between two such players would have been.

Surface	Federer '09	Nadal '11	Djokovic '15
Clay	3	1	2
Hard	2	3	1
Grass	=1	2	=1

Table 4. Rankings for the greatest men's tennis player

Surface Type	Federer '09	Nadal '11	Djokovic '15	Player
Hard	-	F	D	Federer '09
Clay	-	N	D	Federer '09
Grass	-	F	D/F	Federer '09
Hard	F	-	D	Nadal '11
Clay	N	-	N	Nadal '11
Grass	F	-	D	Nadal '11
Hard	D	D	-	Djokovic '15
Clay	D	N	-	Djokovic '15
Grass	D/F	D	-	Djokovic '15

Table 5. Match outcome played between the greatest men's tennis players on a specific surface

It can be seen that the ranks are not unevenly divided and there are surfaces at which different players are expected to dominate. It is noteworthy to state how Nadal wins every match played on Clay which reaffirms the widely held notion that he is the most dominant player on that specific surface type. However, taking all acquired data into consideration, Djokovic of 2015 was most likely to win and would be the greatest men's player of this duration.

7. Conclusion

The project aimed to develop a model to predict the match-winner of a tennis game based on historical data using shallow Machine Learning techniques. Using this predictive model and a custom betting scheme defined in the previous sections, the model also returned an average payoff of 26% for 20 selected players. A competitive average match prediction accuracy of 0.74 and an average F1 score of 0.81 was achieved, which is a significant upgrade over the preliminary model and is comparable or better than the methods outlined in the reference literature. Using this model, we were also able to analyze the peak performance period for a particular player by pitting the player against itself at different years. A comparison was made between Nadal, Djokovic and Federer at their respective peak performance period and results were evaluated. Consistent with the common belief, Nadal is the favorite to win on clay and Federer is the joint favorite on grass with Djokovic. However, overall the surfaces combined, Djokovic emerges as the outright winner and can be ascertained to be the greatest Men's Singles player over the 1996-2017 period.

8. Future Work

Different aspects of the model can be updated and improved upon in the future. For the features, more features can be included and an in-depth understanding of the importance of features over a larger set of players can be performed. For the weights used, further weights can be incorporated such as rankings of opponents played, the importance of tournament played, stage of the tournament, etc. For the models used, neural network models can also be tested out. Although, based on our pipeline with models being trained on just one player's past games, it will be interesting to note if a neural network would out-perform our machine learning models due to the relatively small training size. It may also be interesting to see if differently sized neural networks can be used based on the size of the data available for a player. On the betting performance, we made use of a very simple strategy and more complex strategies like Kelly's criterion can be integrated with our model. It will also be interesting to note how the betting payoffs would look based on real-life betting odds and how updated weights can perform on such data.

9. Contributions

9.1. Namya

I contributed to the feature selection, weight definitions and data cleaning. I worked on the pipeline creation and wrote down the code to implement computationally expensive tasks such as model fitting and historic feature searches efficiently. I also helped with the result metric selections and analysis and implemented the formulation of predicting proxy matches.

9.2. Sohan

I participated in the feature selection decision process and implemented the code for feature extraction and data pre-processing and cleaning. I assisted Namya in defining the pipeline creation. Also assisted in obtaining the result metrics.

9.3. Aishwarya

I contributed to reviewing the literature relevant to the existing model and comparing and analyzing it with the models present in the literature. I also assisted in the debugging process.

9.4. Marianna

I contributed to the analysis and interpretation of the model's results after implementation and the in-class presentation. I helped with evaluating the preliminary model and assisted with the novel pipeline debugging.

10. Resources

GitHub repository - `https://github.com/nambags/MLAI_Tennis`

References

[1] S. P. A. Somboonphokkaphan and C. Lursinsap. Tennis winner prediction based on time series history with neural modeling. *IMECS 2009: International Multi-Conference of Engineers and Computer Scientists, Vols I and II, I:127132*, 2009.

[2] D. W. Andre Cornman, Grant Spellman. Machine learning for professional tennis match prediction and betting. 2011.

[3] M. Bache, K. Lichman. Predictors for winning in men's professional tennis. 2013.

[4] A. C. J. N. Barati Farimani, A. Lecture notes 9. me cmu.

[5] T. Barnett and S. R. Clarke. Combining player statistics to predict outcomes of tennis matches, 2005.

[6] S. R. Clarke and D. Dyte. Using official ratings to simulate major tennis tournaments. *International Transactions in Operational Research*, 2000.

[7] B. D. Knudson. Principles of tennis technique: Using science to improve your strokes. *Vista, CA: Racquet Tech Publishing*, 2006.

[8] A. Z. Farimani, A. A.lecture 15. me notes cmu.

[9] W. Gu and T. Saaty. Predicting the outcome of a tennis tournament: Based on both data and judgments. 2019.

[10] L. S. Hosmer, D. W. Applied logistic regression (2nd ed.). 2000.

[11] M. J. Klaassen, F. Forecasting the winner of a tennis match, 2003.

[12] D. Knudson. Biomechanical principles of tennis technique: Using science to improve your strokes. *Vista, CA: Racquet Tech Publishing, 200*, 2006.

[13] B. L. M. Reid, M. Crespo and J. Berry. Skill acquisition in tennis. *Research and current practice, vol. 10*, 2007.

[14] L. C. T. Y. M. S. Ma, S.M. Winning matches in grand slam men's singles: an analysis of player performance-related variables from 1991 to 2008. *J. Sports Sci. 31(11), 1147–1155 (2013)*, 2006.

[15] B. Pawel and K. Klaudia. Body height and career win percentage in relation to serve and return games effectiveness in elite tennis players. *ci. Rev. Phys. Cult., vol. 4, no. 3, pp. 75–80*, 2015.

[16] R. E. Quandt. Betting and equilibrium. *Q. J. Econ., vol. 101, no. 1, p. 201*, 1986.

[17] F. Radicchi. Who is the best player ever? a complex network analysis of the history of professional tennis, 2011.

[18] M. Sipko. Machine learning for the prediction of professional tennis matches, 2015.

[19] S. Srivastava. Predicting success probability in professional tennis tournaments using a logistic regression model. *Advances in Analytics and Applications (pp.59-65*, 2019.

[20] B. F. Wang, T. A.lecture 6. me notes cmu.

[21] A. K. Zijian Gao1. Random forest model identifies serve strength as a key predictor of tennis match outcome. 2012.