Problem Chosen	2024 MCM/ICM Summary Sheet	Team Control Number
C		2421874

Witness How Charismatic the Momentum is

Summary

The defeat of the 36-year-old Djokovic in the 2023 Wimbledon Gentlemen's final has once again propelled tennis into the limelight. To evaluate how tennis players' **Momentum** has been affected, we are expected to accomplish these tasks in this paper: identify the match flow and set up a model to evaluate it; explain how **Momentum** will affect the match; explore, formulate recommendations for players. To solve these problems, we set up several assumptions to simplify the questions and pre-process the data. Several models are established: Model I: **Markov Chain Model (MC)** Model II: **Random Forests Model (RF)**.

For Question 1: To analyze the match flow in a game, we utilize the **MC** based on continuity of the player's score within current game to capture dynamic fluctuations and turning points, thereby demonstrating that a player's ups and downs.

For Question 2: We continue to persuade coaches that **Momentum** influence palyers' performance, not arbitrarily. We prove that some given factors relevant to **Momentum** impact players' performance by **Factor Analysis(FA)**. Besides, we use **Linear Regression(LR)** to emphasize that our **MC** is suitable because of high correlation between the winning percentage predicted by **MC** and real scores within current time.

For Question 3 and 4: We employ **RF**, on which take into account 18 factors that could potentially influence games' flow 2, discovering that four of these factors, namely players' distances ran during point, unforced error, number of shots, server of the point, play a relatively significant role(10% account) among them.The specifics of our recommendations can be summarized as follows: Coaches must notify players when they engage in extended runs, implement strategies to compel opponents to cover longer distances. The players should strive to enhance the frequency of shot attempts while minimizing unforced errors.

For Question 5: To evaluate the precision of the model, a combination of the **Analytic Hierarchy Process (AHP)** and **Entropy Weight Method(EW)** was employed for conducting the **sensitivity analysis**. The results indicate that the tennis model exhibits a high level of fitness, while adding or distracting 5% of the match flow, the deviation is acceptable within range. The model can be considered stable. And we put forward that table tennis matches demonstrating a similarly high degree of approbation.

Our model can effectively predict the trend of the game by **Momentum** in general. The conclusions of the paper can provide advice for players and coaches, and also give fans a more thorough analysis of the game.

Keywords: Momentum, Markov Chain, Random Forest, Factor Analysis, Linear Regression,Analytic Hierarchy Process, Entropy Weight, Sensitivity Analysis

Contents

1 Introduction

1.1 Problem background

The 2023 Wimbledon Gentlemen's final, featuring rising star Carlos Alcaraz and tennis legend Novak Djokovic, marked a historic moment. Alcaraz, a 20-year-old Spanish player, ended Djokovic's Wimbledon dominance since 2013 with a remarkable victory. The match witnessed intense shifts, with Djokovic initially dominating but Alcaraz ultimately securing the win in a tight battle.

The given dataset "2023-wimbledon-1701" holds crucial match data, highlighting the challenges of measuring and understanding momentum fluctuations in tennis.

Momentum, defined as the strength or force gained by motion or a series of events, is a concept that is often felt by teams or players during a match or game. However, measuring and understanding this phenomenon is challenging. It is also not clear how various events during a match contribute to the creation or change of momentum, if it exists.

1.2 Restatement of the Problem

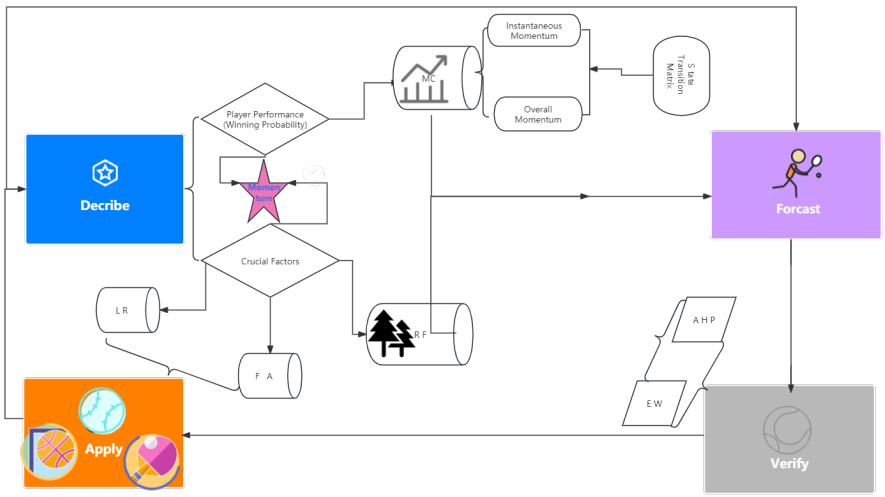
Problem 1: Develop a tennis match flow model incorporating serving probabilities, apply it to matches, and visualize the flow.

Problem 2: Use the model to verify the momentum decides performance of players instead of randomness.

Problem 3 and Problem 4: Build a predictive model for match swings, identify related factors, and provide advice based on past momentum differentials.

Problem 5: Test the model on other matches, assess performance, identify additional factors, and evaluate its generalizability.

1.3 Our work



2 General Assumptions and Notations

2.1 Assumptions

- It is assumed by us that the data used is accurate, complete and representative of the real situation. This includes player statistics, match situations, environmental factors, etc
- In the first model we built, to make Markov chain models accord with tennis player's performance more suitably, we regard that The difference between the scores of the two games next to each other and the ratio of runners scored in each game can be combined to reflect scoring continuity, which could be used to generate our Markov Transition Matrix(MTM).
- During the process of the LR, we use winning probability within each game generated by Markove Transition Probability(MTP) calculated by MTM to simulate the probability of winning the game, with the comparison between the real victor and other factors included.
- During the RF's establishment, we assume that trees in a random forest are typically built based on a random subset of features[4], which means that the features are independent of each other. For example,if a player has missed serving, this player will stay a positive heart to continue his long-distance running.
- The fourth assumption's weakness could be fixed a lot by the combination of MC and RF as our solution to Question 5 explained, because the second assumption of MC and the third assumption of LR have explained the correlation between each factor.

2.2 Notations

All the symbols will be introduced once they are used. Because of our paper's limited space,the exact location of every symbol is on the last paragraph of the corresponding formula instead of being listed there specifically.

3 Data Pre-processing

- There are missing values in some columns, and we need to deal with them first. For categorical variables——speed mph, considering their actual situation, we choose to fill the blanks with the mode. For quantitative variables——serve width,serve depth,return depth, considering their actual situation, we choose to fill the blanks with the 3σ method. The 3σ method assumes that the data follows a normal distribution. In the case of normal distribution, about 68% of the data fall within one standard deviation of the mean, about 95% of the data fall within two standard deviations, and about 99.7% of the data fall within three standard deviations.If a data point is beyond the range of the average $\pm$ 3σ, it can usually be regarded as an outlier.

- Some parameters are actually categorical variables, and 0 or 1 only represents yes or no, so we convert them into categorical variables.
- We use 50 to replace "AD", use 0 to replace "LOVE"

4 Question 1 and 2: Use Markov Chain Model to Present Players' Performance

4.1 The Establishment of Model 1

A Markov chain model is a mathematical tool used to describe a stochastic process based on the Markov property that future states depend only on the present state and not on past states. In sports competitions, especially tennis matches, Markov chain models can be used to capture dynamic changes and turning points in matches and help us understand strategies and decisions in matches.

$$P(X_{n+1} = j \mid X_n = i) = P_{ij} \tag{1}$$

$$P(X_{n+1} = j \mid X_n = i, X_{n-1} = i_{n-1}, ..., X_0 = i_0) = P(X_{n+1} = j \mid X_n = i) \tag{2}$$

In this modeling, we define the state space as two performance states: player_1 wins and player_2 wins, and use the given data to determine the initial probability distribution. Using the m_1= player_1's pointon the last time/ player_1's point on this time, m_2 = player_2's pointon the last time/ player_2's point to define our transfer probability (m > 1 and m < 1 corresponding allocation, and m=1 equal distribution into the probability of winning or losing) generation cost problem transfer matrix, to solve the transfer matrix eigenvector (since each value of multiple eigenvectors is very close, So we take the average of each element of the feature vector as the player1's skill (a constant value not greater than 1) for each of the two players and plug it into the relevant Tennis Match Markov Chain in python as our final visualization.

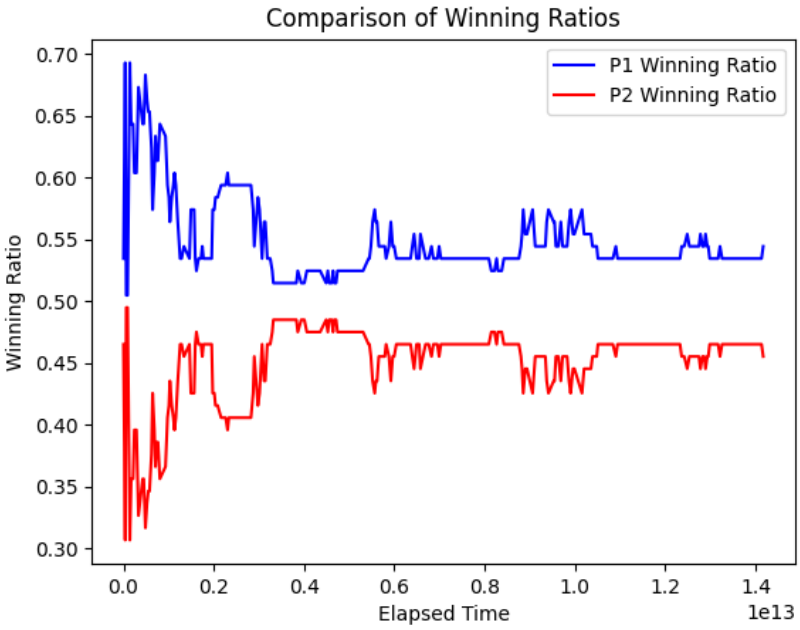


Figure 1: 1301 "momentum"

First, Markov chain models can help us analyze Momentum changes in a race. In tennis, when one side's players score consecutive points, it often creates a momentum or Momentum that can affect their performance and confidence, thereby further increasing their chances of winning. Markov chain models can help us quantify and measure changes in Momentum by analyzing the outcomes and transitions of every point in a match. By looking at how each point is scored in a game and the alternating control between players, we can build a state transition matrix where each state represents a different situation in the game, such as one player controlling the situation or two players balancing. By analyzing the state transition matrix, we can calculate the probability of stability for each state and thus understand changes in Momentum and turning points in the game.

Second, Markov chain models can help us predict the outcome of a match. By looking at the scoring situation and state transition of each point in the game, we can calculate the transition probability and stability probability of each state. These probabilities can be used to predict the outcome of a match, such as which player is more likely to win or which player is more likely to control the situation. By analyzing the state transition matrix and the stability probability, we can get the probability distribution of the match and predict the result of the match.

All in all, Markov chain models can help us analyze strategies and decisions in matches. By observing the scoring situation and state transfer of each point in the game, we can analyze the strategic choice and decision-making process of players in different situations. For example, when one player is in control, they may choose a more conservative strategy to maintain their advantage; And when the other player is in control, they may choose a

more offensive strategy to turn the tide. By analyzing the state transition matrix and the stability probability, we can understand the impact of different strategies and decisions on the outcome of a match, thereby helping players and coaches develop more effective tactics and strategies.

4.2 The Solution of Model 1

Since the momentum is the incredible movements of the players at the moment of an instant, and the instant set of these actions can be reflected as the score of any player at each point in time, the "point victor" in the data set can be comprehensively described

In that case, we use the ratio of each player to generate our model's transfer matrix.

$$\text{General Transfer Matrix } A = \begin{bmatrix} 0.542519 & 0.457481 \\ 0.477005 & 0.522995 \end{bmatrix} \tag{3}$$

When Ax = x,

$$\mathbf{x} = \begin{bmatrix} 0.554455 \\ 0.445545 \end{bmatrix} \tag{4}$$

, which means that player_1 skill = 0.5544555, player_2 skill = 0.445545, then, we could use this to print a picture(Figure1) to describe overall tendency of the momentum.

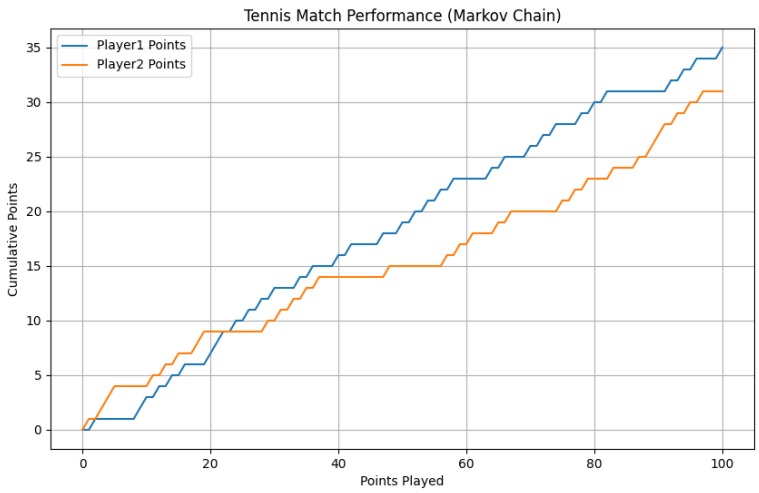


Figure 2: random tennismatchmarkovchain

In order to make the momentum accurate to each point in time to describe the momentum more precisely, we also generate every transform matrix at every time point. In fact, this step sets the stage for the second question to confirm that a player's tournament performance is affected by momentum.

4.3 Use Linear Regression and Factor Analysis to Demonstrate the Important Influence of Momentum on Player Performance

To show this coach that a player's performance is affected by "momentum", we use Linear Regression and Factor Analysis to demonstrate the important influence of momentum on player performance.

The tennis coach is skeptical about the role of "momentum" in a match, suggesting that swings in play and runs of success are random. To address this claim, we will utilize factor analysis as a mathematical modeling approach. Factor analysis allows us to identify underlying factors or dimensions within a set of observed variables. In this case, we have 18 potential factors that can contribute to an "untouchable winner" serve.

These potential factors include serving speed, serving direction, serve rotation, variations and combinations, forehand and backhand shots, service errors, unforced errors, ball in net, winning points at the net, and various scenarios involving player 1 and player 2 serving. Additionally, we have factors such as the distance run by player 1 within a point and the number of shots taken during a point. These potential factors provide insights into the player's activity level, mobility, aggressiveness, and shooting frequency.[3]

By applying FA, we can determine the weight or importance of each factor in describing momentum. The stable performance of players, psychological quality, and the impact of outliers can be comprehensively reflected through factor loadings. Factor loadings indicate the strength of the relationship between each factor and momentum. Higher factor loadings suggest a greater influence on momentum.

To further analyze the relationship between each moment's performance and the flatness of non-outlier data, we will employ linear regression. Linear regression allows us to examine the correlation between two variables and determine the strength and direction of the relationship. In this case, we will assess how the performance of each moment relates to the flatness of non-outlier data. A strong positive correlation would support the existence of momentum.

By utilizing factor analysis and linear regression, we aim to evaluate the claim made by the skeptical tennis coach.

In order to assess whether a player's tournament performance is affected by "momentum", we divide the data in the given dataset into non-outliers and full values.

We analyzed all the data, using the most typical 1301 innings for each detailed moment as an example to describe and further analyze.

(1) The flatness of non-outliers reflects the steady performance of players. We used linear regression to analyze the correlation between momentum at every moment and all the above mentioned factors that can describe momentum of quantitative non-outliers.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{\text{true}_i} - y_{\text{predict}_i})^2 \tag{5}$$

For momentum, We use the transfer matrix of the first model, the Markov model, at different times to solve the winning probability of the two players at different times (the solution of this winning probability was explained above because the properties contained in the solution process can accurately depict the momentum). Firstly, the fitting degree of the two players is similar to that of the linear regression model. (Figure), by analyzing the F values in the two linear regression models of the two players, it can be found that they can significantly reject the null hypothesis that the overall regression coefficient is 0 (P<0.05), indicating that there is a linear relationship between them. The fitting of the model is analyzed by the R^2 value, and the VIF value is analyzed at the same time. The model presents collinearity and the significance of X is analyzed. Combined with the regression coefficient B value, the linear regression model requires that the overall regression coefficient is not 0, that is, there is a regression relationship between variables. Therefore, our results show that:

For player 1, set victor(cofficient=7.1), p1 ace(cofficient= 8.7), p1 double fault(cofficient= -4.8), game victor(cofficient= -5.6), point victor(cofficient= 4.5), p1 break pt missed(-7.1), p1 sets, really count, Both p1 break pt won and p1 distance run have a strong correlation with the momentum expressed by the instantaneous probability.

For player 2, p2 games(-9.7), p2 ace(4.4), set victor(8.0), p2 unf err(-6.66), serve no(-8.3), p2 break pt won(8.9) and momentum expressed in instantaneous probability.

(2) The poor psychological quality reflected by outliers or the jedi counterattack or endurance brought about by the explosion. In order to reflect or even highlight the great effect of outliers on the momentum, we used the original data to intuitively perceive the important impact of outliers on the momentum. This time, we first carried out factor analysis among all quantitative variables; Then, we added the analysis of certain variables, but because the presentation of player 1 and player 2 in certain variables is very similar, with little difference, this result presents the use of player 1's certain variables as a concise and concise description of the situation, we use these certain variables as index labels to analyze the factor relationship, the weight (factor load) of these factor analyses, are factors that can be synthesised to describe momentum.

player1:

$$
\begin{aligned}
F_1 = {} & 0.222 \cdot \text{set\_no} + 0.006 \cdot \text{p1\_net\_pt} + 0.057 \cdot \text{game\_no} + 0.052 \cdot \text{server} + 0.01 \cdot \text{serve\_no} \\
& + 0.004 \cdot \text{p1\_score} + 0.025 \cdot \text{p1\_games} - 0.145 \cdot \text{p1\_wins} + 0.005 \cdot \text{point\_victor} + 0.003 \cdot \text{game\_victor} \\
& + 0.001 \cdot \text{set\_victor} + 0.023 \cdot \text{p1\_break\_pt} - 0.0 \cdot \text{p1\_net\_pt\_won} + 0.028 \cdot \text{p1\_break\_pt\_won} + \\
& 0.046 \cdot \text{p1\_distance\_run} + 0.001 \cdot \text{speed\_mph} + 0.036 \cdot \text{rally\_count} + 0.041 \cdot \text{p1\_break\_pt\_missed} \\
& - 0.052 \cdot \text{p1\_ace} - 0.019 \cdot \text{p1\_winner} - 0.002 \cdot \text{p1\_unf\_err} \\
& + 0.223 \cdot \text{p1\_points\_won} + 0.221 \cdot \text{p1\_sets} + 0.225 \cdot \text{point\_no}
\end{aligned}
\tag{6}
$$

$$F_2 = -0.008 \cdot \text{set\_no} + 0.213 \cdot \text{p1\_net\_pt} + 0.194 \cdot \text{game\_no} - 0.199 \cdot \text{server} - 0.053 \cdot \text{serve\_no} - 0.052 \cdot \text{p1\_score} + 0.2 \cdot \text{p1\_games} + 0.049 \cdot \text{p1\_wins} - 0.212 \cdot \text{point\_victor} - 0.027 \cdot \text{game\_victor} + 0.079 \cdot \text{set\_victor} - 0.387 \cdot \text{p1\_break\_pt} + 0.213 \cdot \text{p1\_net\_pt\_won} + 0.085 \cdot \text{p1\_break\_pt\_won} - 0.081 \cdot \text{p1\_distance\_run} + 0.004 \cdot \text{speed\_mph} - 0.049 \cdot \text{rally\_count} + 0.024 \cdot \text{p1\_break\_pt\_missed} + 0.125 \cdot \text{p1\_ace} + 0.186 \cdot \text{p1\_winner} - 0.079 \cdot \text{p1\_unf\_err} + 0.051 \cdot \text{p1\_points\_won} - 0.014 \cdot \text{p1\_sets} + 0.046 \cdot \text{point\_no} \tag{7}$$

$$F_3 = 0.005 \cdot \text{set\_no} - 0.017 \cdot \text{p1\_net\_pt} + 0.024 \cdot \text{game\_no} + 0.051 \cdot \text{server} - 0.278 \cdot \text{serve\_no} + 0.069 \cdot \text{p1\_score} + 0.043 \cdot \text{p1\_games} + 0.061 \cdot \text{p1\_wins} + 0.009 \cdot \text{point\_victor} + 0.068 \cdot \text{game\_victor} + 0.054 \cdot \text{set\_victor} + 0.374 \cdot \text{p1\_break\_pt} - 0.017 \cdot \text{p1\_net\_pt\_won} - 0.01 \cdot \text{p1\_break\_pt\_won} - 0.267 \cdot \text{p1\_distance\_run} + 0.261 \cdot \text{speed\_mph} - 0.285 \cdot \text{rally\_count} + 0.009 \cdot \text{p1\_break\_pt\_missed} + 0.156 \cdot \text{p1\_ace} + 0.074 \cdot \text{p1\_winner} - 0.111 \cdot \text{p1\_unf\_err} + 0.019 \cdot \text{p1\_points\_won} - 0.004 \cdot \text{p1\_sets} + 0.017 \cdot \text{point\_no} \tag{8}$$

$$F = \frac{0.178}{0.381} \cdot F_1 + \frac{0.107}{0.381} \cdot F_2 + \frac{0.097}{0.381} \cdot F_3 \tag{9}$$

player2:

$$F_1 = 0.229 \cdot \text{set\_no} + 0.017 \cdot \text{p2\_winner} + 0.011 \cdot \text{p2\_unf\_err} - 0.001 \cdot \text{point\_victor} + 0.03 \cdot \text{p2\_net\_pt\_w} + 0.003 \cdot \text{p2\_break\_pt\_won} + 0.004 \cdot \text{rally\_count} + 0.003 \cdot \text{p2\_distance\_run} + 0.02 \cdot \text{p2\_break\_pt\_misse} + 0.136 \cdot \text{p2\_wins} + 0.009 \cdot \text{game\_victor} + 0.024 \cdot \text{speed\_mph} + 0.011 \cdot \text{p2\_break\_pt} + 0.029 \cdot \text{p2\_net\_pt} + 0.001 \cdot \text{p2\_ace} - 0.018 \cdot \text{serve\_no} + 0.007 \cdot \text{set\_victor} + 0.237 \cdot \text{p2\_points\_won} + 0.077 \cdot \text{p2\_games} + 0.048 \cdot \text{server} + 0.017 \cdot \text{p2\_score} + 0.062 \cdot \text{game\_no} + 0.203 \cdot \text{p2\_sets} + 0.236 \cdot \text{point\_no} \tag{10}$$

$$F_2 = 0.01 \cdot \text{set\_no} - 0.166 \cdot \text{p2\_winner} + 0.139 \cdot \text{p2\_unf\_err} - 0.173 \cdot \text{point\_victor} - 0.025 \cdot \text{p2\_net\_pt\_won} + 0.139 \cdot \text{p2\_break\_pt\_won} + 0.238 \cdot \text{rally\_count} + 0.245 \cdot \text{p2\_distance\_run} + 0.176 \cdot \text{p2\_break\_pt\_missed} + 0.022 \cdot \text{p2\_wins} - 0.006 \cdot \text{game\_victor} - 0.177 \cdot \text{speed\_mph} + 0.072 \cdot \text{p2\_break\_pt} + 0.051 \cdot \text{p2\_net\_pt} - 0.198 \cdot \text{p2\_ace} + 0.199 \cdot \text{serve\_no} - 0.026 \cdot \text{set\_victor} - 0.005 \cdot \text{p2\_points\_won} - 0.032 \cdot \text{p2\_games} - 0.127 \cdot \text{server} + 0.05 \cdot \text{p2\_score} - 0.059 \cdot \text{game\_no} + 0.019 \cdot \text{p2\_sets} - 0.006 \cdot \text{point\_no} \tag{11}$$

$$F_3 = 0.03 \cdot \text{set\_no} + 0.16 \cdot \text{p2\_winner} - 0.007 \cdot \text{p2\_unf\_err} + 0.139 \cdot \text{point\_victor} + 0.252 \cdot \text{p2\_net\_pt\_won} + 0.132 \cdot \text{p2\_break\_pt\_won} + 0.12 \cdot \text{rally\_count} + 0.104 \cdot \text{p2\_distance\_run} + 0.078 \cdot \text{p2\_break\_pt\_missed} + 0.01 \cdot \text{p2\_wins} + 0.087 \cdot \text{game\_victor} + 0.021 \cdot \text{speed\_mph} - 0.013 \cdot \text{p2\_break\_pt} + 0.231 \cdot \text{p2\_net\_pt} + 0.004 \cdot \text{p2\_ace} + 0.014 \cdot \text{serve\_no} - 0.067 \cdot \text{set\_victor} - 0.028 \cdot \text{p2\_points\_won} - 0.223 \cdot \text{p2\_games} + 0.157 \cdot \text{server} + 0.201 \cdot \text{p2\_score} - 0.238 \cdot \text{game\_no} + 0.058 \cdot \text{p2\_sets} - 0.03 \cdot \text{point\_no} \tag{12}$$

$$F = \frac{0.174}{0.394} \cdot F_1 + \frac{0.112}{0.394} \cdot F_2 + \frac{0.108}{0.394} \cdot F_3 \tag{13}$$

Figure 3: player1 linear regression

Figure 4: player2 linear regression

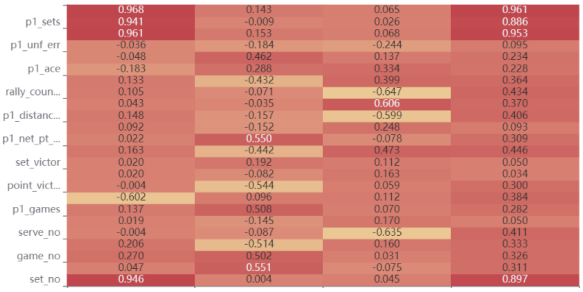


Figure 5: Categorical variable:shoot types

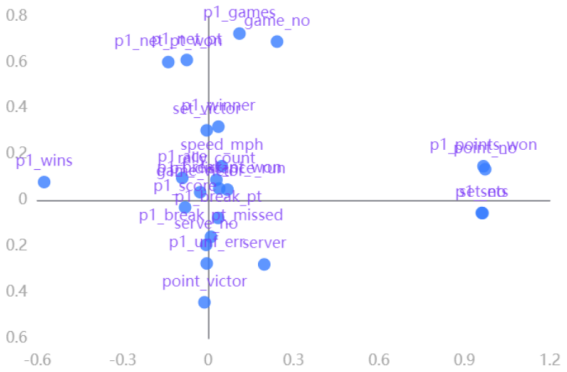


Figure 6: Categorical variable:return...

5 Question 3: Predict the swings in the match and identify the most relevant factors

Random Forest(RF) is a machine learning algorithm that makes final predictions through the comprehensive judgments of multiple trees. Therefore, its robustness is relatively high, and it is suitable for dealing with large data sets, high-dimensional features, and scenarios that require improvement of classification accuracy. However, there are 18 factors in Wimbledon_featured_matches.csv that may affect the flow of play. Therefore, we choose the RF model to predict the swings in the match and identify the most relevant factors.

5.1 the Establishment of Model 2

The basic steps of the random forest algorithm include the following three steps:

1. Random sampling of training sets: This is done to ensure that the input samples for each tree are not the entire data set during training, preventing the decision trees in the forest(following RF majority voting mechanism) from reaching local optimal solutions.

$$\hat{y}_{\text{RF}} = \text{mode}(y_1, y_2, \ldots, y_k) \tag{14}$$

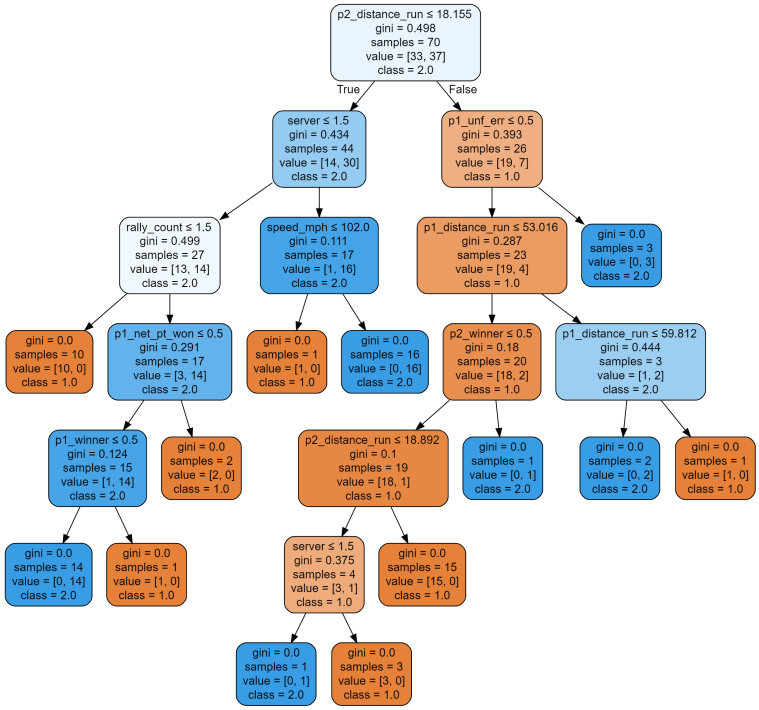
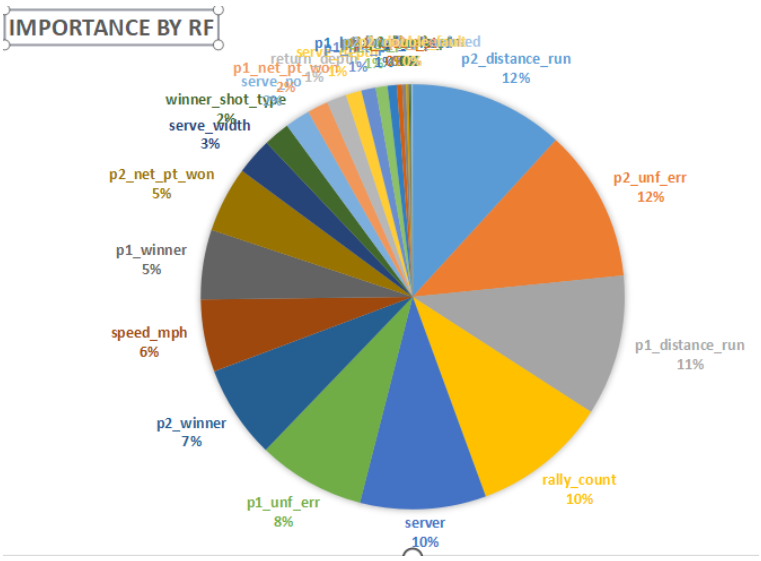


Figure 7: one example tree of the whole RF

2. Building decision trees: For each training subset, a decision tree is established, generating N decision trees to form a forest. Each decision tree does not require pruning. To reduce the correlation between each decision tree and improve the classification accuracy of each tree, randomness is introduced in the node splitting process.

3. Formation of the forest: The final output result is obtained by classifying a test sample based on the N decision subtrees randomly constructed. The results from each subtree are summarized, and the majority class among the obtained results is considered as the class of the sample.



5.2 Problem Solving

We used 70% of the data from Wimbledon_featured_matches.csv as the training set and the remaining 30% as the test set for our RF model.The remaining model parameters are in the table.

predicted results Y	point_victor	Predicted probability of test results_1.0	Predicted probability of test results_2.0	server	serve_no	winner_shot_type
1.0	1.0	0.8057288476229651	0.1942711523770347	2	1	3
1.0	1.0	0.9406884416924666	0.05931155830753353	2	1	1
1.0	1.0	0.6852257835044213	0.31477421649557874	1	2	1
1.0	1.0	0.95125	0.04875	1	2	1
2.0	2.0	0.24476473234367965	0.7552352676563204	1	1	1
1.0	1.0	0.6539963914009967	0.34600360859900325	1	1	1
1.0	1.0	0.8891568627450981	0.11084313725490196	2	2	1
1.0	2.0	0.5941536824574445	0.40584631754255535	2	2	1
2.0	2.0	0.051790170613700025	0.9482098293862999	2	1	1
1.0	1.0	0.8470993626471567	0.1529006373528432	1	1	1

	accuracy	recall	precision	F1
training set	0.994	0.994	0.994	0.994
test set	0.88	0.88	0.88	0.88

The bar chart shows the proportion of importance of each feature (independent vari-

able). Feature importance is calculated through the average impurity reduction of the random forest algorithm, which associates the importance of each feature with its average purity reduction in the tree splits. The higher the importance score, the greater the contribution of the feature to the model.

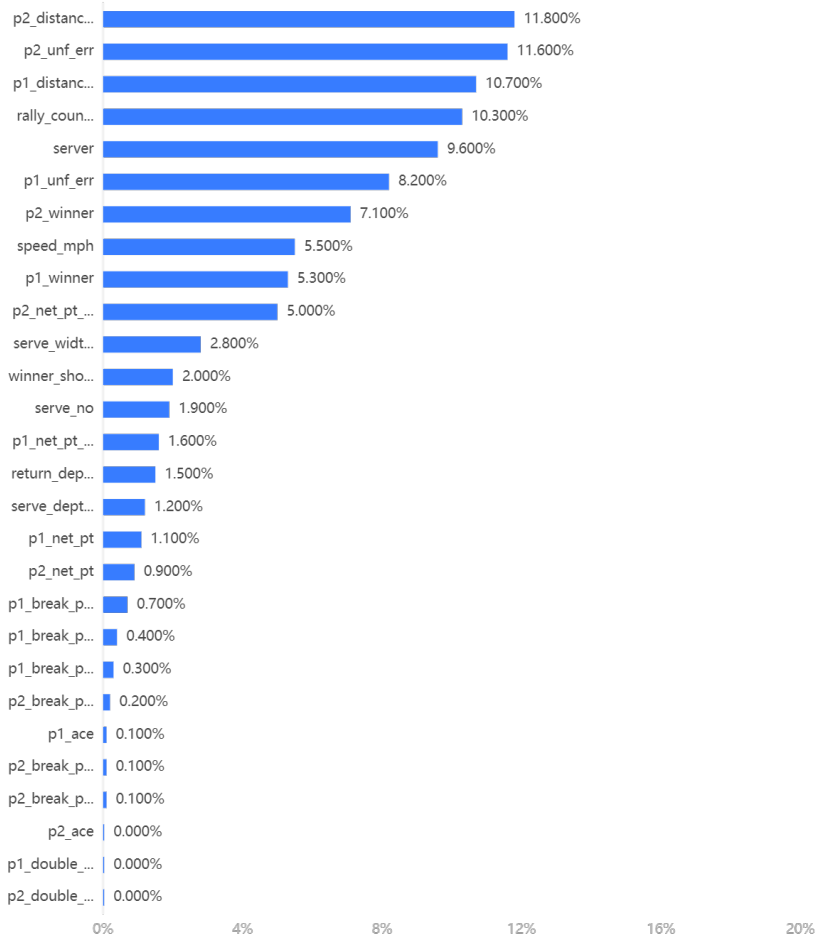


Figure 8: bar chart

Each decision tree independently predicts the test set and generates different classification results. The final classification result is determined through voting.The following table shows the first 10 classification results of the random forest model on the test data.(Only partial variables are presented.)

By calculating metrics such as accuracy, recall, and F1 score on the test set, we evaluate the predictive performance of the random forest algorithm.It seems that accuracy(0.88) is high, meaning the model could successfully grasp the relation between multiple factors and point_victor.

F1 is the harmonic mean of precision and recall. Precision and recall are mutually influential. Although high precision and recall are an ideal situation, in practice, it is

often the case that high precision leads to low recall or vice versa. If we need to consider both precision and recall, then we can use the F1 metric.

6 Question 4: Suggestions for players to make new plays based on momentum changes

According to the chart, the four primary influencing elements, namely players' distances ran during the point, unforced errors, the number of shots, server of the point, all exceed 10%. Additionally, there are 14 other relatively minor elements, including depth of return, serving speed, serving direction, serve rotation, variations and combinations, forehand, backhand, ball in net, serving direction, serving speed, player 1 has a chance to score but Player 2 is serving, player 1 wins but Player 2 is serving, win this point at the net, and player 1 loses the chance to win the match while Player 2 is serving. The following provides an in-depth analysis of the effects generated by the four primary influencers.

Server of the point: The server of the point plays a crucial strategic role in tennis matches. They can control the pace of the game and establish an offensive position by choosing different serving strategies. A successful serve can boost the server's confidence and lay the foundation for gaining an advantage in the match.

Unforced errors: Unforced errors occur when a player makes mistakes without significant pressure from the opponent. These errors can disrupt the player's momentum and give the opponent an opportunity to gain an advantage.

Players' distances ran during the point: The distance covered by a player during a point reflects their activity level and mobility. Higher activity levels can indicate a player's determination and can positively impact their momentum.

Number of shots: The total number of shots taken by a player during a point can reflect their aggressiveness and shooting frequency. More shots can put pressure on the opponent and potentially shift the momentum.

To suggest a new match point for one player to play another player, it is crucial to consider these factors and analyze the strengths and weaknesses of both players. By understanding the impact of each factor on the momentum of the match, a player can strategize and utilize their strengths to exploit the opponent's weaknesses. Additionally, coaches should notify players when they engage in extended runs and implement strategies to compel opponents to cover longer distances. The players should strive to enhance the frequency of shot attempts while minimizing unforced errors. By carefully considering these factors, a player can increase their chances of winning and maintaining momentum throughout the match.

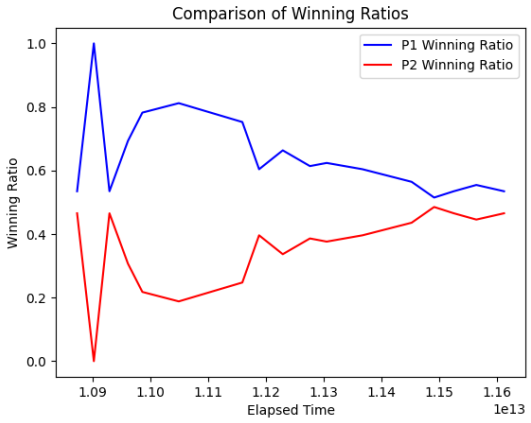
7 Question 5: Use AHP+ Entropy Weight Method with Visual drawing to Let you Trust our Comprehensive Model

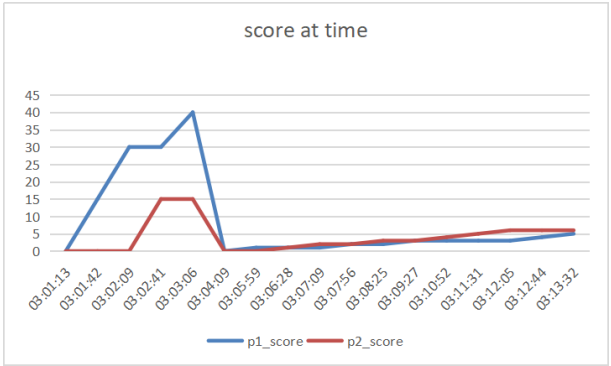
The use of Analytic Hierarchy Process (AHP) and entropy weight method in the comprehensive evaluation model is aimed at improving the accuracy and reliability of the

predictions made by the model. These techniques are widely used in decision making and comprehensive evaluation because they provide a systematic and objective approach to analyzing complex problems. In order to optimize the parameters in the model and compare different models, an evaluation metric is needed. A good degree Quantities should be easy to calculate and available in every model.[5]

In this situation,we interpret the predictive ability of the evaluation model mentioned in the question as a combination of two models, namely the Markov chain and random forest.

As for momentum present in instantaneous winning percentage, which could be reflected well in our Model 1(MCTTMP), we use the correlation between points scored in each moment and the probability of winning in each moment to prove its degree of accuracy. However, given that this paper space is limited, we will only show you one of the most classical example, which is the 2023-wimbledon-1302 's 12th set and 13th set. Not only did it happened during a long time tennis game, in which present two players' psychological quality , but also relate to two tennis players' common adversary , about whom people especially tennis fans always talk a lot. Besides, based on Knottenbelt[6]'s Common Adversary Model, when judging based on a common opponent, it becomes more meaningful to compare the statistics between two players. (In fact, in the section of our model 1 of this paper, we have referred to the whole correlation in order to check the degree of the success rate calculated by the model recognizes the player's state.)





Specifically, we utilize the factor analysis conducted with the instantaneous transition matrix of the Markov chain model to calculate the proportion of observed variables, which could be described as "q".

$$F_n = m_{i_1 j_1} q_1 + m_{i_2 j_2} q_2 + \ldots + m_{i_n j_n} q_n \tag{15}$$

$$F = n_1 F_1 + n_2 F_2 + n_e F_e \tag{16}$$

We have known that:

$$\begin{aligned} F_1 = {} & 0.005 \cdot \text{speed\_mph\_ Missing Value Handling\_Outlier Handling} - 0.035 \cdot \text{p1\_score\_Outlier Handli} \\ & - 0.014 \cdot \text{serve\_no\_Outlier Handling} + 0.232 \cdot \text{point\_victor\_Outlier Handling} - 0.006 \cdot \text{p1\_points\_} \\ & + 0.0 \cdot \text{set\_victor\_ Outlier Handling} - 0.102 \cdot \text{p1\_ace\_ Outlier Handling} + 0.105 \cdot \text{p2\_ace\_ Outlier H} \\ & + 0.005 \cdot \text{p1\_double\_fault\_Outlier Handling} - 0.01 \cdot \text{p2\_double\_fault\_ Outlier Handling} + 0.034 \cdot \\ & + 0.149 \cdot \text{p2\_net\_pt\_ Outlier Handling} - 0.157 \cdot \text{p1\_net\_pt\_won\_Outlier Handling} + 0.174 \cdot \text{p2\_net} \\ & - 0.213 \cdot \text{p2\_break\_pt\_ outlier handling} - 0.014 \cdot \text{p1\_break\_pt\_won\_ outlier handling} + 0.06 \cdot \text{p2\_b} \\ & - 0.006 \cdot \text{p2\_break\_pt\_missed\_ outlier handling} - 0.005 \cdot \text{p1\_distance\_run\_ outlier handling} - 0.0 \end{aligned} \tag{17}$$

$$\begin{aligned} F_2 = {} & -0.134 \cdot \text{speed\_mph\_ Missing Value Handling\_Outlier Handling} - 0.059 \cdot \text{p1\_score\_Outlier Hand} \\ & + 0.129 \cdot \text{serve\_no\_Outlier Handling} + 0.047 \cdot \text{point\_victor\_Outlier Handling} - 0.014 \cdot \text{p1\_points\_} \\ & - 0.003 \cdot \text{set\_victor\_ Outlier Handling} - 0.094 \cdot \text{p1\_ace\_ Outlier Handling} - 0.06 \cdot \text{p2\_ace\_ Outlier} \\ & + 0.027 \cdot \text{p1\_double\_fault\_Outlier Handling} - 0.007 \cdot \text{p2\_double\_fault\_ Outlier Handling} + 0.077 \\ & + 0.021 \cdot \text{p2\_net\_pt\_Outlier Handling} - 0.003 \cdot \text{p1\_net\_pt\_won\_Outlier Handling} + 0.019 \cdot \text{p2\_net} \\ & + 0.062 \cdot \text{p2\_break\_pt\_ outlier handling} - 0.052 \cdot \text{p1\_break\_pt\_won\_ outlier handling} + 0.037 \cdot \text{p2\_} \\ & - 0.033 \cdot \text{p2\_break\_pt\_missed\_ outlier handling} + 0.328 \cdot \text{p1\_distance\_run\_ outlier handling} + 0.3 \end{aligned} \tag{18}$$

$$\begin{aligned} F_3 = {} & 0.03 \cdot \text{speed\_mph\_ Missing Value Handling\_Outlier Handling} + 0.207 \cdot \text{p1\_score\_Outlier Handlin} \\ & - 0.019 \cdot \text{serve\_no\_Outlier Handling} + 0.039 \cdot \text{point\_victor\_Outlier Handling} - 0.042 \cdot \text{p1\_points\_} \\ & + 0.076 \cdot \text{set\_victor\_ Outlier Handling} - 0.004 \cdot \text{p1\_ace\_ Outlier Handling} + 0.024 \cdot \text{p2\_ace\_ Outlie} \\ & + 0.002 \cdot \text{p1\_double\_fault\_ Outlier Handling} - 0.021 \cdot \text{p2\_double\_fault\_ Outlier Handling} + 0.022 \\ & - 0.027 \cdot \text{p2\_net\_pt\_ Outlier Handling} + 0.013 \cdot \text{p1\_net\_pt\_won\_ Outlier Handling} - 0.011 \cdot \text{p2\_ne} \\ & + 0.132 \cdot \text{p2\_break\_pt\_ Outlier Handling} + 0.097 \cdot \text{p1\_break\_pt\_won\_ Outlier Handling} + 0.33 \cdot \text{p2} \\ & + 0.385 \cdot \text{p2\_break\_pt\_missed\_ outlier handling} + 0.006 \cdot \text{p1\_distance\_run\_ outlier handling} + 0.0 \end{aligned} \tag{19}$$

$$F = \frac{0.103}{0.261} \cdot F_1 + \frac{0.081}{0.261} \cdot F_2 + \frac{0.077}{0.261} \cdot F_3 \tag{20}$$

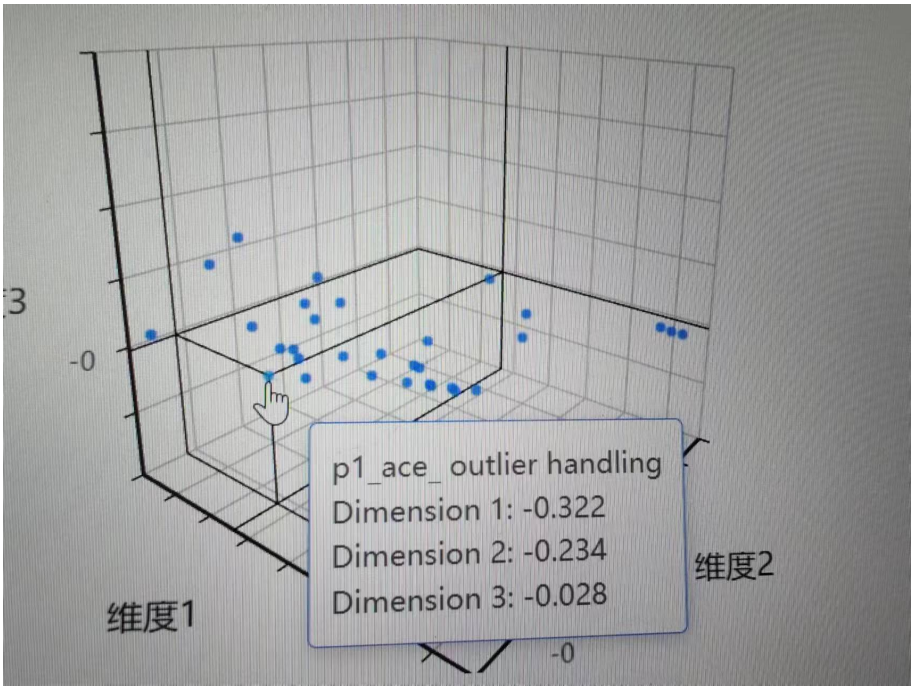
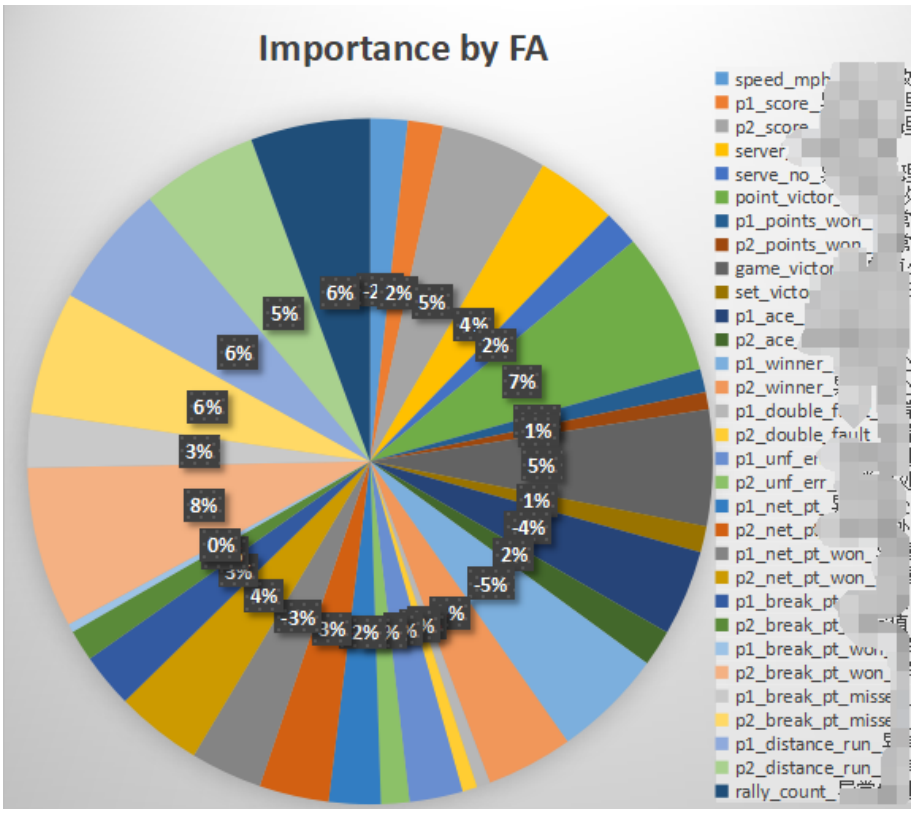


Figure 9: 3D FA



Then, we also have the weights generated within the random forest model, which

could be described as “k”. We then take a weighted average of these proportions(“q” and “k”)to generate a number to describe the importance of a parameter, which could be described as “u”. This weighting allows us to balance the importance of the significant factors in our model.

We define that the importance(FA) of the given parameters could be described as this formula:

$$q_n = (|m_{i_1 j_1}| + |m_{i_2 j_2}| + |m_{i_3 j_3}|) \cdot n_e \tag{21}$$

for example, as for the importance(FA) set_no, it could be described as

$$q_1 = (|m_{11}| + |m_{21}| + |m_{31}|) \cdot n_1 \tag{22}$$

AHP is a multi-criteria decision-making method that allows decision-makers to evaluate and prioritize different factors based on their relative importance. It involves breaking down a complex problem into a hierarchy of criteria and sub-criteria, and then comparing and weighting these criteria based on pairwise comparisons. This helps in capturing the subjective judgment of decision-makers and incorporating it into the evaluation model.

Entropy weight method, on the other hand, is an objective technique that calculates the weights of different factors based on the information entropy theory. It measures the uncertainty and diversity of the data and assigns higher weights to factors that have more information content. This method helps in considering the objective aspects of the evaluation and reducing the bias that may be introduced by subjective judgments.

	p1_distanc	p1_unf_err	p1_winner	p1_net_pt	p1_net_pt	p1_break_	p1_break_	p1_break_	p1_ace	p1_double	rally_count	server	speed_mp	serve_widt	winner_sh	serve_no	return_dep	serve_depth
p1_distanc	1	0.574713	0.293255	0.263852	0.19305	0.277778	0.265957	0.050891	0.374532	0.061387	0.909091	0.763359	0.050075	0.280899	0.189753	0.238095	0.134048	0.104167
p1_unf_err	1.74	1	0.507614	0.458716	0.334448	0.483092	0.460829	0.088028	0.680272	0.106157	1.694915	1.315789	0.086881	0.487805	0.330033	0.414938	0.2331	0.181159
p1_winner	3.41	1.97	1	0.900901	0.657895	0.952381	0.909091	0.17331	1.333333	0.209205	3.333333	2.631579	0.171233	0.961538	0.649351	0.813008	0.458716	0.357143
p1_net_pt	3.79	2.18	1.11	1	0.729927	1.052632	1	0.191939	1.470588	0.231481	3.703704	2.857143	0.189394	1.06383	0.719424	0.900901	0.507614	0.395257
p1_net_pt	5.18	2.99	1.52	1.37	1	1.449275	1.369863	0.262467	2	0.316456	5	4	0.259067	1.470588	0.990099	1.25	0.699301	0.543478
p1_break_	3.6	2.07	1.05	0.95	0.69	1	0.952381	0.182482	1.408451	0.220264	3.448276	2.702703	0.18018	1.010101	0.684932	0.854701	0.483092	0.374532
p1_break_	3.76	2.17	1.1	1	0.73	1.05	1	0.19084	1.470588	0.230415	3.571429	2.857143	0.188324	1.052632	0.714286	0.900901	0.505051	0.392157
p1_break_	19.65	11.36	5.77	5.21	3.81	5.48	5.24	1	7.692308	1.204819	19.23077	14.92537	0.983284	5.555556	3.703704	4.761905	2.631579	2.040816
p1_ace	2.67	1.47	0.75	0.68	0.5	0.71	0.68	0.13	1	0.15674	2.5	1.960784	0.12837	0.719424	0.487805	0.609756	0.343643	0.26738
p1_double	16.29	9.42	4.78	4.32	3.16	4.54	4.34	0.83	6.38	1	16.66667	12.5	0.813008	4.545455	3.125	3.846154	2.222222	1.694915
rally_count	1.1	0.59	0.3	0.27	0.2	0.29	0.28	0.052	0.4	0.06	1	0.793651	0.052301	0.294118	0.198807	0.249377	0.140252	0.109051
server	1.31	0.76	0.38	0.35	0.25	0.37	0.35	0.067	0.51	0.08	1.26	1	0.065703	0.369004	0.249377	0.31348	0.176056	0.136799
speed_mp	19.97	11.51	5.84	5.28	3.86	5.55	5.31	1.017	7.79	1.23	19.12	15.22	1	4.545455	3.125	3.846154	2.173913	1.694915
serve_widt	3.56	2.05	1.04	0.94	0.68	0.99	0.95	0.18	1.39	0.22	3.4	2.71	0.22	1	0.675676	0.847458	0.478469	0.37037
winner_sh	5.27	3.03	1.54	1.39	1.01	1.46	1.4	0.27	2.05	0.32	5.03	4.01	0.32	1.48	1	1.25	0.70922	0.549451
serve_no	4.2	2.41	1.23	1.11	0.8	1.17	1.11	0.21	1.64	0.26	4.01	3.19	0.26	1.18	0.8	1	0.561798	0.436681
return_dep	7.46	4.29	2.18	1.97	1.43	2.07	1.98	0.38	2.91	0.45	7.13	5.68	0.46	2.09	1.41	1.78	1	0.775194
serve_dep	9.6	5.52	2.8	2.53	1.84	2.67	2.55	0.49	3.74	0.59	9.17	7.31	0.59	2.7	1.82	2.29	1.29	1

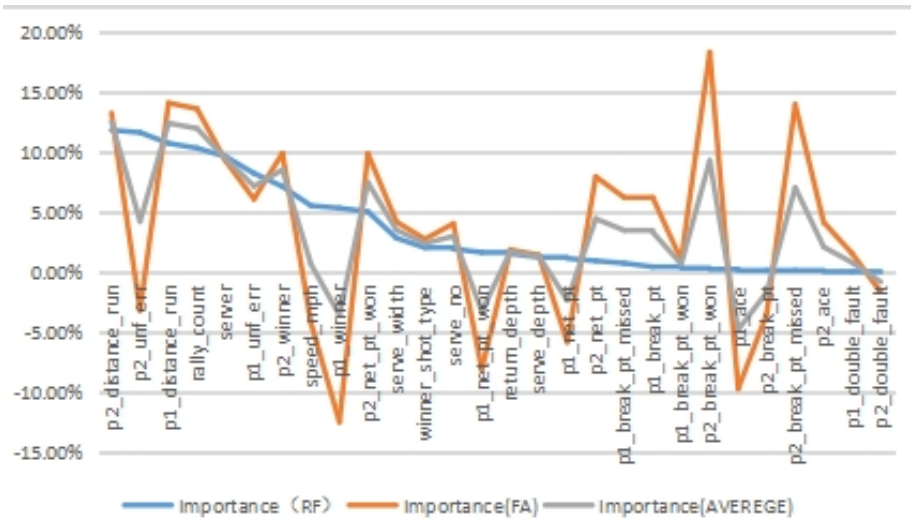
AHP analytic hierarchy results				
item	Eigenvectors	Weight Value (%)	Maximum feature root	CI value
p1_distance_run	0.234	0.882	18.009	0.001
p1_unf_err	0.409	1.541		
p1_winner	0.806	3.035		
p1_net_pt_won	0.892	3.357		
p1_net_pt	1.224	4.607		
p1_break_pt_missed	0.846	3.185		
p1_break_pt	0.887	3.337		
p1_break_pt_won	4.644	17.478		
p1_ace	0.606	2.281		
p1_double_fault	3.861	14.531		
rally_count	0.245	0.921		
server	0.309	1.164		
speed_mph	4.446	16.73		
serve_width	0.848	3.192		
winner_shot_type	1.253	4.716		
serve_no	0.999	3.761		
return_depth	1.774	6.677		
serve_depth	2.287	8.605		

Consistency test results

Maximum feature root	CI value	RI value	CR value	Consistency test results
18.009	0.001	1.613	0	Pass

Figure 10: EW result

By combining AHP and entropy weight method, the comprehensive evaluation model can provide a more balanced and accurate assessment of the factors influencing the outcome of a match. The subjective judgments of decision-makers are considered through AHP, while the objective aspects of the data are taken into account through entropy weight method. This combination helps in reducing the potential biases and uncertainties in the evaluation process.



As for other kinds of ball matches, to evaluate Momentum like this paper having described, we should firstly find the ability reflecting score gaining consistently and secondly classify lots of factors to with highly importance, medium importance and not very importance. For example, ping-pong matches, sports analysts often analyze important factors, our RF could help them do this better more efficiently, some of which are closely related to scoring continuity and mental endurance, and can be solved using our MC the

whole match, Ma used only five backhand twists to deal with short balls, Fan Zhendong used 75 backhand twists. In addition, Malone 73 forehand play, the use of rotation and drop point to contain Fan Zhendong, so that opponents can not return the most powerful ball road. Ma's strategy is to control the pace of the game, he controls Fan Zhendong's path, and then waits for the opportunity to effectively counter attack when the opponent returns.[2] Given that we could not find a whole comprehensive data like this Problem C, because of our MC' s accuracy in matches within each time and RF' s strengths based on adjusting hyperparemeters according to randomness adjustment, classifying and filtering important features while creating new features for feature scaling, adjusting decision tree, leaves and depth, we could also say confidently that our two Model could analyze well.

As for improvement, we may let our RF proceed Gradient Boosting[1].

8 Strengths and weaknesses

8.1 Strengths

- The primary benefits of Markov analysis are simplicity and out-of-sample forecasting accuracy.
- Factor analysis is more interpretable than other dimension reduction techniques and is not as sensitive to the scale of variables.
- Random Forest is a robust algorithm that can handle noisy data and outliers. It is less likely to overfit the data, which means it can generalize well to new data.
- The model is constructed with full consideration of the realistic situation and has good interpretability.

8.2 Weaknesses

- Markov analysis is a valuable tool for making predictions, but it does not provide explanations.
- One assumption is that we assume linear relationships between input variables and bivariate normal between each pair of variables.
- The algorithm requires a sufficient amount of historical data for accurate predictions, which may not always be available.

Appendices 1 of Python Code of MC

```
data = pd.read_csv(
r"D:\2023-2024
   sem1\2024_MCM-ICM_Problems\Wimbledon_featured_matches1.csv",
index_col='elapsed_time')
data.index = pd.to_timedelta(data.index)
```

```
    states = ['P1_Win', 'P2_Win']

    def simulate_match(start_state, num_points,
        transition_probabilities):
    transition_probabilities.fillna(0.5, inplace=True)
    np.random.seed(42)
    current_state = start_state
    results = [current_state]
    for _ in range(num_points):
    current_state = np.random.choice(
    states,
    p=transition_probabilities.loc[current_state]
    )
    results.append(current_state)
    return results

    time_strings = data.index
    for time_string in time_strings:
    start_time = pd.to_timedelta('0:00:00')
    sub = data.loc[start_time:time_string]
    transition_counts = pd.DataFrame(0, index=states, columns=states)
    for i in range(1, len(sub)):
    prev_winner = sub.iloc[i - 1]['point_victor']
    current_winner = sub.iloc[i]['point_victor']
    if prev_winner == current_winner:
    transition_counts.loc[states[int(prev_winner) - 1],
        states[int(current_winner) - 1]] += 1
    else:
    transition_counts.loc[states[int(prev_winner) - 1],
        states[int(current_winner) - 1]] += 1

    transition_probabilities =
        transition_counts.div(transition_counts.sum(axis=1), axis=0)
    start_state = 'P1_Win'
    num_points = 100
    simulation_results = simulate_match(start_state, num_points,
        transition_probabilities)
    p1_wins = simulation_results.count('P1_Win')
    p2_wins = simulation_results.count('P2_Win')
    data.at[time_string,'p1_wins']=p1_wins / len(simulation_results)
    data.at[time_string,'p2_wins']=p2_wins / len(simulation_results)
    print(data[['p1_wins','p2_wins']])
```

Appendices 2 of Report on use of AI

1. Open AI ChatGPT-3.5

Query1: <Ask for suggestion about modifying our codes with mistake.>

Output: <The location of the wrong codes and suggestion of how to improve them.>

Query2: <A more efficient way to understand the rules of tennis and what each factor actually means>

Output: <Introduction about tennis.>

Query3: <Advice about change an overly long sentence to make it easier to understand, which may benefit both our efficiency of bibliographic retrieval and writing skill>

Output: < Practical and useful methods to help us analyze and modify sentences >

References

[1] Zhou Jieying He Pengfei Qiu Rongfa Chen Guoguo Wu Weigang , Intrusion detection based on fusion of random forest and gradient lift tree, School of Data, Science and Computer Science, Sun Yat-sen University Mathematical Society and Addison-Wesley Publishing Company , 1984-1986.

[2] Big data detailed World Table Tennis Championships Ma Long VS Fan Zhendong "textbook" men's singles championship battle, cn.ITTF.com

[3] Gao Huixuan. Applied Multivariate Statistical Analysis [M]. Beijing: Peking University Press,2005.

[4] Christopher M. Bishop with Hugh Bishop,Deep Learning_ Foundations and Concepts

[5] Zhang Rong with tutor Chen Li, Tennis match result forecast and player analysis , Yunnan University

[6] Knottenbelt W J , Spanias D , Madurska A M . A common-opponent stochastic model for predicting the outcome of professional tennis matches[J]. Computers & Mathematics with Applications, 2012, 64(12)