Problem Chosen	2025 MCM/ICM Summary Sheet	Team Control Number
C		

Momentum: The Invisible Hand in Tennis Matches

Summary

In the 2023 Wimbledon Gentlemen's final, 20-year-old Spanish rising star Carlos Alcaraz defeated 36-year-old Novak Djokovic. The game witnessed multiple incredible swings, they seem to be controlled by a unseen force which we call "momentum". Thus, we analyzed the tennis matches data.

Firstly, we established a **TEP Evaluation Model** to evaluate the score of player's performance in match "2023-wimbledon-1301". We computed player 1 and 2 's cumulative score as **136.6387** and **128.6715**, then we **visualized** match flow, and concluded that player 1 performed **better**.

Secondly, we define momentum as **the ability of continuously win points**. Then we use the **Mann-Kendall Test** to examine the distribution of swings, **its results are: trend is increasing, P value is 0.0, Statistic is 18024.0**. Therefore, we concluded that the swings include **trend component**, and momentum do have **impact** on matches.

Thirdly, we developed **GSSRF Prediction Model** to predict the swings of matches. We determined the **best parameters** for **Random Forest** through **Grid Search** as **{20,2,200}**. And we computed its **mean squared error (MSE)** as **0.9975687439546586**, **root mean squared error (RMSE)** as **0.9987836322020193**, and χ^2**(Chi-Square Test)** as **0.9587307805304899**, which means our model's predictive performance is **excellent**. Then we identify the **related factors** basing on the contribution of each feature value to the training of the Random Forest Model. **And got the related factors: winning the points, untouchable shot, serve, distance ran during the point , serve next round and opponent making unforced error**. Then, based on the related factors, we provide recommendations for the player's matches.**At least, We have constructed a the following quantitative formula for "momentum" based onthe indicator data within the green dashed box.**

Finally, we applied GSSRF Prediction Model to other matches, and obtained the predictive results and **the model's evaluation score: accuracy is 0.91, macro avg is 0.91 and weighted avg is 0.91** indicatiing the prediction performance is good and our prediction model is a high model **generalization**.

Keywords: Momentum, TEP Evaluation Model, Mann-Kendall Test, GSSRF Prediction Model

Figure 1: Caption

Contents

1 Introductionnnn

1.1 Problem Background

"The invisible hand," originally proposed by Adam Smith, fundamentally describes an unseen force or factor that influences market phenomena. In tennis matches, we also see the incredible swings, for example, in the 2023 Wimbledon Gentlemen's final, the fifth and final set started with Djokovic carrying the edge from the fourth set, but again a change of direction occurred and Alcaraz gained control and the victory 6 – 4. It seems that the swings of match is controlled by unseen force, and people always attribute it to "momentum".

If player can grasp this "invisible hand", the flow of match will change a lot. Therefore, making tactical adjustments based on the changing dynamics on the court is crucial. Especially when the opponent is scoring consecutively, it's important to contemplate the factors contributing to it and whether "momentum" truly exists. If it does, how does it impact the course of the match, and what advice should coaches give to players to break the opponent's momentum?

Hence, through in-depth analysis of tennis match data and the establishment of models, we aim to analyze the influence of "momentum," predict the match progression, and provide coaches with more tactical recommendations. This, in turn, contributes to making tennis matches more intense and exciting.

1.2 Restatement of the Problem

The purpose of the task is to use the provided data to establish a model that captures the flow of the game, analyze the presence of "momentum," predict turning points in the game, and provide some suggestions to the coach.

Therefore we need to:

1. Develop a model to identify which player's performing better and how much better at a given time in the match. And provide a visualization based on your model to depict the match flow.
2. Use model to check the trend of swings in play.
3. Establish a match flow prediction model to identify factors that cause fluctuations in the odds of winning, thereby deriving recommendations for the competition.
4. Apply our model to other matches to test the precision of our model's predictions.Identify some factors that might need to be included in future models. Then, apply our model to different types of competitions to assess the generalizability of our prediction model.

1.3 Our Work

Our work is illustrated in Figure 2:

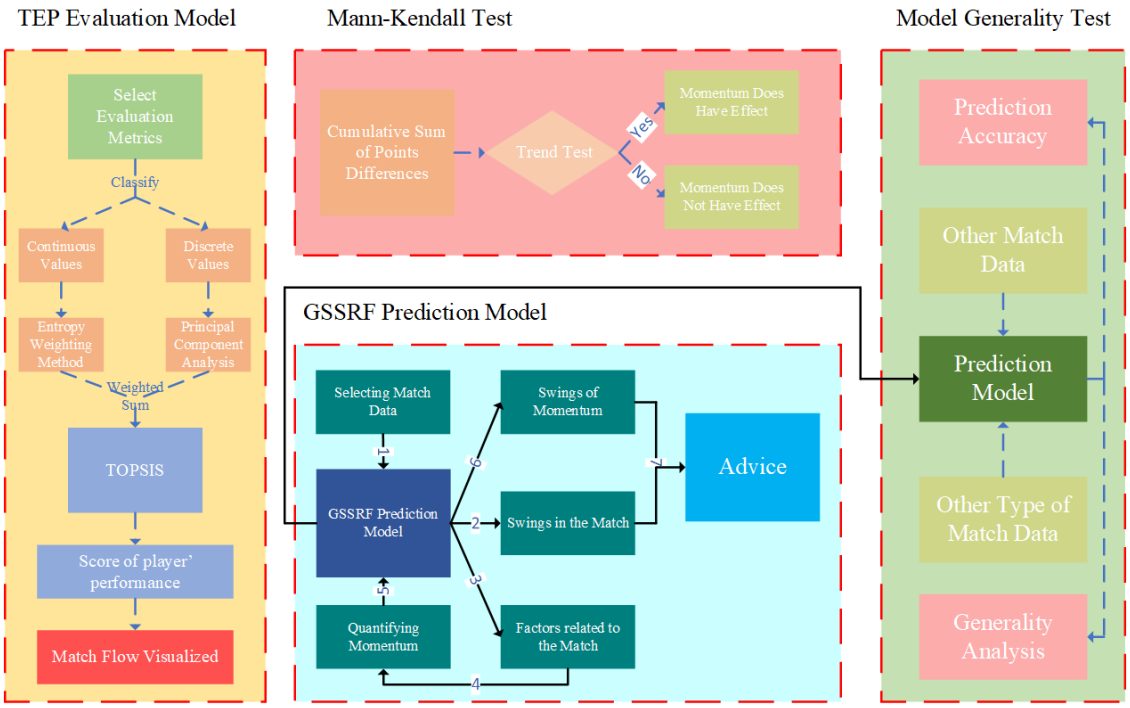

Figure 2: Our Work

2 Assumptions

To simplify our modeling, we make the following assumptions:

- **Assumption 1** We assume that the data we receive is authentic, and the missing data is unrelated to the actual data values. This assumption leads to more accurate data analysis.
- **Assumption 2** There is no correlation between the performance of competitors in different matches, allowing us to focus on the analysis of each individual match.
- **Assumption 3** Both players are of equal skill level, which aids in our analysis of momentum.

3 Notations

Notations we used in this paper are listed in Table 1:

4 Data Preprocessing

We found that some of the data contains missing values, as shown in Table 2:

Where:

- speed_mph denotes speed of serve (miles per hour; mph)

Symbols	Description
$Score_t^i$	Score of player i's performance in the t-th point in match
M_t	Momentum
t	The t-th point in the match.
i	player ($i = 1$ means player 1, $i = 2$ means player 2)

Table 1: Notations

Data containing missing values	Quantities of missing values
speed_mph	752
serve_width	54
serve_depth	54
return_depth	1309

Table 2: Missing Values

- serve_width denotes direction of serve
- serve_depth denotes depth of serve
- return_depth denotes depth of return

Observing these values, we learned that "speed_mph" is continuous values, but others are discrete values. Therefore, for the missing values in "speed_mph", we used the k-NN algorithm for imputation. For the others, we used the mode filling method.

4.1 k-NN algorithm

4.1.1 Description of k-NN algorithm

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. KNN can also be used to fill in missing values, that working off the assumption that similar points can be found near one another.[1]

Its flow is shown in Figure 3:

Similar to predictions problems, we can use the k-NN algorithm to handle missing data values, based on the principle of proximity, which posits that similar data points in the dataset should have similar attribute values. This method involves finding the k nearest neighbors (i.e., the closest k data points) to the data point with missing values within

Figure 3: k-NN algorithm[1]

the dataset, and then estimating or imputing the missing values based on the known attribute values of these neighbors.

4.1.2 Distance Measurement

First, we need to define the distance between each data point.Due to the reason that Euclidean distance is one of the most commonly used metrics, we will now proceed to use it. Its formula is as follows:

$$D(x, y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2} \tag{1}$$

Where:

- x represents the data point with missing values.
- y represents other points in the dataset.
- x_i represents the i-th data point with missing values.
- y_i represents the i-th other points in the dataset.
- n represents dimensionality of data objects.

4.1.3 Estimate the missing values.

Given that the data contains many outliers and to avoid classification ambiguity, we set the k value to 5.

The formula for estimating missing values in the k-NN (k-nearest neighbors) algorithm using the weighted average of k neighbors is as follows:

$$\hat{x}_{\text{missing}} = \frac{\sum_{i=1}^{k} w_i \cdot x_i}{\sum_{i=1}^{k} w_i} \tag{2}$$

Where:

- $\hat{x}_{\text{missing}}$ represents the estimated value of the missing data.
- x_i is the known value of the i-th neighbor among the k nearest neighbors.
- w_i represents the weight associated with the i-th neighbor, typically determined based on distance or similarity.

This formula illustrates that the estimation of the missing value is based on the known values from k nearest neighbors, with each known value weighted according to its respective weight. Usually, closer neighbors have higher weights, while distant neighbors have lower weights. This weighted averaging method is effective in estimating missing values, taking into account the similarity between neighbors.

4.2 Mode filling method

The "mode filling method" is a technique for imputing or filling missing values in data, particularly applicable to handling missing values in categorical or discrete data. In this method, missing values are filled with the mode.

The mode is the value that appears most frequently in a set of data. When dealing with data that has multiple categories or discrete values, and some data points have missing values, the mode filling method is used to replace those missing values. The specific steps are as follows:

1. Calculate the mode for each feature (column), which is the value that occurs with the highest frequency in that feature.
2. For each data point that has missing values, replace the missing values in the corresponding features with their respective modes.

5 TEP Evaluation Model

The situation on the field changes rapidly, and with each scoring event, various data indicators for each player (such as the number of hits, running distance, etc.) fluctuated.To identify which player's performing is better and how much better at a given time in the match, we developed TEP Evaluation model. And provide a visualization based on TEP Evaluation model to depict the match flow.

5.1 Description of TEP algorithm

TEP algorithm is TOPSIS algorithm based on Entropy Weight Method and Principal Component Analysis(PCA).

The Entropy Weight Method is a multi-criteria decision analysis technique used to determine the weights of multiple decision factors or indicators in the decision-making process.Principal Component Analysis (PCA) is a multivariate statistical analysis method

used to reduce the dimensionality of a dataset and identify the main directions of variation within the data.The TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) algorithm is a multi-attribute decision-making method used to select the best solution or object.

Unlike the traditional TOPSIS algorithm where weights for each criterion are subjectively set, in our TEP algorithm, we did not manually assign weights to the criteria. Instead, we employed a combined algorithm that utilizes the Entropy Weight Method and Principal Component Analysis. However, since some criteria are continuous numerical values while others are discrete binary values (0 and 1), using the Entropy Weight Method alone may result in an unfair weight distribution. Similarly, using Principal Component Analysis alone may allocate excessive weights to continuous criteria, which is not the desired outcome. Therefore, we used a combination of these two algorithms to calculate and allocate weights, aiming to achieve a more reasonable and equitable weight distribution among the criteria.

Its operation is illustrated in Figure 4:

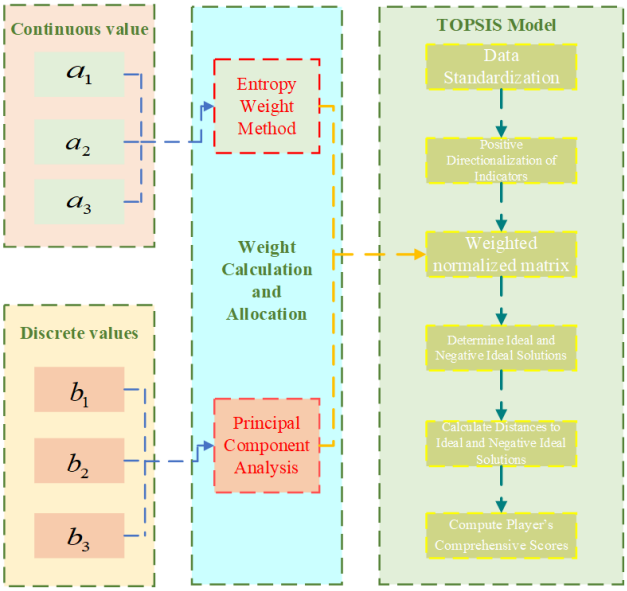

Figure 4: TEP Evaluation Model

5.2 Performance Assessment

To construct the model, we only utilize the data of **player 1's** various indicators in all matchs.

After reading many papers and consulting with experts, we believe that the following indicators determine the performance of players:[2]

- Hitting an untouchable winning shot

- Missing both serves and lost the point
- Making an unforced error
- Winning a point
- Distance ran per shot
- Winning the point while at the net
- Hitting an untouchable winning serve
- Speed of serve

And then, we define the symbols for these indicators as Table 3 shows:

Symbols	Description
w	player hit an untouchable winning shot
df	player missed both serves and lost the point
ue	player made an unforced error
p	player won the point
drp	player's distance ran
npw	player won the point while at the net
a	player hit an untouchable winning serve
s	Speed of player's serve

Table 3: Notations

After observing these values above, we learned that drp and s is continuous value, while others are discrete values. Continuous value has a wider range of values and a greater degree of variation, it contains more information, making the Entropy Weight Method more effective. So, we used the Entropy Weight Method to calculate the weight of drp and s, but applied PCA to the others. The results of discrete values' weight calculations are shown in Table 4:

Indicators	w	df	ue	p	npw	a
Weight	0.36294185	-0.06368468	-0.2566273	0.87618668	0.1325054	-0.11445136

Table 4: PCA Weight

Then we calculated the entropy weights for continuous numerical values, **where the entropy weight is 0.78841932**, with drp and s accounting for **0.709605 %** and **0.290395 %** of the entropy weight, respectively.

After calculating the weights of all the indicators, we proceed with standardizing all of the data indicators to ensure comparability through Min-Max normalization. Next, we performed the process of directionalizing the data, **specially we think winning point while serving is "Cost-type" indicator** (in tennis, the player serving has a much higher probability of winning the point/game), **so we need to reduce the positive impact of the scoring indicator on the ratings if the player is server**.Then We calculate the weighted normalized matrix using the following formula:

$$\boldsymbol{C} = \omega_1 \boldsymbol{f}_E \boldsymbol{NDD}_E + \omega_2 \boldsymbol{f}_P \boldsymbol{NDD}_P \tag{3}$$

Where:

- $\boldsymbol{C}$ denote weighted normalized matrix
- ω_1 and ω_2 denote Weight coefficients.And we set ω_1 as 0.2 and ω_2 as 0.8.
- $\boldsymbol{f}_E$ and $\boldsymbol{f}_P$ denote weight matrix of entropy weight and PCA.
- $\boldsymbol{NDD}_E$ and $\boldsymbol{NDD}_P$ denote standardization feature matrix for continuous values and discrete values.

It is worth noting that we set ω_1 as 0.2 and ω_2 as 0.8 because we considered PCA to be more important, as it encompasses many crucial evaluation criteria (e.g., winning points).

With the weighted normalized matrix, we calculated their positive and negative ideal solution. After we calculated their Euclidean distances to the positive and negative ideal solutions, we got $Score_t$ using the following formula:

$$C_i = \frac{d^-(i)}{d^-(i) + d^+(i)}, \quad i = 1, 2, \ldots, n \tag{4}$$

Where:

- $d^+(i) = \sqrt{\sum_{j=1}^{m}(x_{ij} - POS_j)^2}, \quad i = 1, 2, \ldots, n$
- $d^-(i) = \sqrt{\sum_{j=1}^{m}(x_{ij} - NEG_j)^2}, \quad i = 1, 2, \ldots, n$
- $NEG_j = \min_i x_{ij}, \quad j = 1, 2, \ldots, m$
- $POS_j = \max_i x_{ij}, \quad j = 1, 2, \ldots, m$
- x_{ij} represents the score or value of the j-th evaluation criterion for the i-th alternative in the set of alternatives.
- n represents the number of alternative solutions.
- m represents the number of evaluation criteria.

5.3 Visualization of the Competition Process

To obtain a player's score in the game, we summed up the player's scores for each point, as follows:

$$Score = \sum_{t=1}^{n} Score_t \tag{5}$$

where n means the total sum of points in match.

At Point Number 300, the scores are 0.667121 and 0.267115, with cumulative scores of 136.6387 and 128.6715 respectively. The data for score are presented in the AppendixA.

We define "match flow" as the cumulative sum of points differences, and "performing flow" as the cumulative sum of score performance differences. In order to visualize the current status of the players in the match and to have a clear view of the match process, we took match "2023-wimbledon-1301" and "2023-wimbledon-1302" for example and performed visualization of players' scores, as shown in Figure 4 and Figure 5:

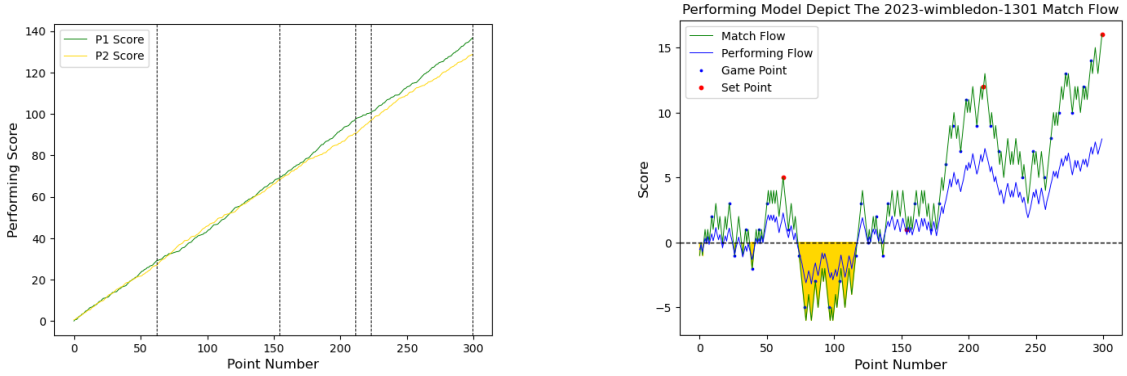

Figure 5: 2023-wimbledon-1301

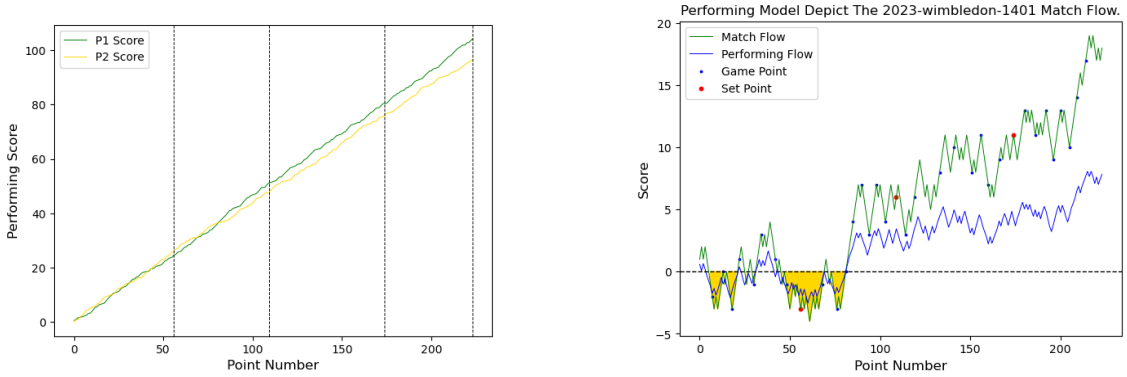

Figure 6: 2023-wimbledon-1302

The line chart on the left shows the cumulative scores of two players, with the black dashed line representing the appearance of the set winner. The line chart on the right

displays the match flow and performing flow. It can be observed that when the match flow favors P1, P1's performance is superior to P2's; when the match flow favors P2, P2's performance is inferior to P1's. Therefore, our TEP evaluation model is able to depict the match flow very effectively.

6 Mann-Kendall Test

In tennis matches, we often witness instances where players score consecutively, seemingly harnessing a certain power (referred to as 'momentum'). However, there are times when incredible fluctuations occur even when in an advantageous position. Therefore, we test the match fluctuations to determine if they follow a random distribution.

6.1 Definition of Momentum

According to the Oxford Dictionary's definition, momentum is the ability to keep increasing or developing[3]. This definition also applies to tennis, where we can consider momentum as the ability to continuously win points.

In order to see the swings of the match, we created cumulative score difference charts for the first and second matches, as shown in the Figure 7 and Figure 8:

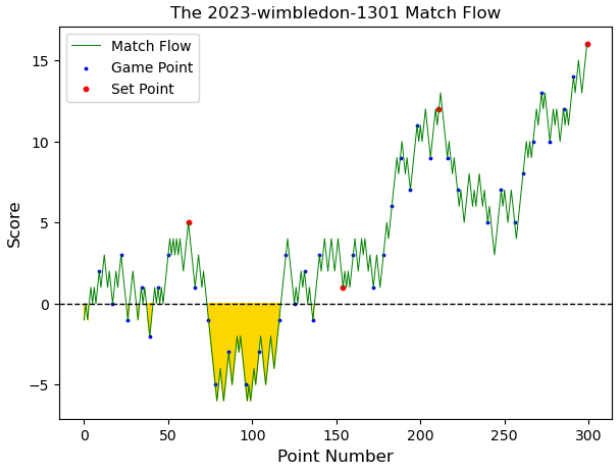

Figure 7: 2023-wimbledon-1301

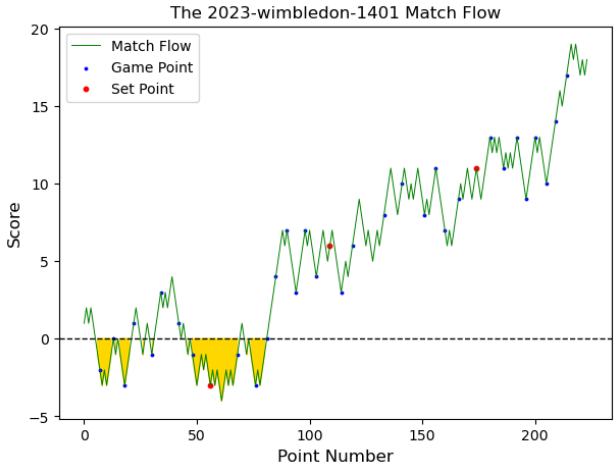

Figure 8: 2023-wimbledon-1302

From the analysis of the above figure, it is not difficult to see that the growth of the points difference has **inertia** (the same applies to a decrease), rather than random fluctuations. If a player's momentum declines, then in the next few points, their momentum will continue to decrease due to this inertia.

6.2 Test the Impact of Momentum On Matches

In the previous section, we preliminarily considered that momentum has an impact on the outcome of the match, because we observed a relationship between momentum fluctuations, we hypothesize that the points difference time series contains a trend component. If we can demonstrate the existence of this trend component, we can conclude

that the swings in the game are not random.

Therefore, we conduct a Mann-Kendall test on our momentum model.And the result of the test are listed on Table 5:

Trend	P Value	Statistic
increasing	0.0	18024.0

Table 5: Mann-Kendall test

From the data above, it is evident that the trend component of the points different is pronounced and the swings in match do not follow white noise distribution. Therefore, we can consider the impact of "momentum" on the match to be highly significant.

7 GSSRF Prediction Model

After data preprocessing, we retained 48 features that may be useful for predicting the outcome of the matches.

As the swings of match is a time series data, theoretically, the swings of match at a certain time point should be related to the data preceding that time point. Therefore, we chose the GS-SRF algorithm as the optimal strategy to predict the match outcome.

Using existing data to train the model, we obtained the optimal strategy and predicted the swings of match“2023-wimbledon-1301”, achieving favorable predictive results.

7.1 Descriptions of GSSRF Prediction Model

Grid search is a method used in machine learning to find the optimal hyperparameters. It involves traversing a predefined grid of parameters and evaluating the model performance using cross-validation for each parameter combination to find the best set of parameters.

Sliding window is a commonly used algorithm technique that processes data by maintaining a fixed-size window and performing corresponding operations as the window slides. In time series forecasting, sliding window is a common technique that helps construct appropriate feature sets and effectively utilizes the temporal patterns and trends in the time series data, thereby improving the accuracy and generalization capability of the predictive model.

Grid Search Random Forest Combined with Sliding Window is typically applied in time series forecasting or other problems that require consideration of temporal factors. It aims to enhance the model’s performance and generalization by adapting to changes over time and effectively capturing time-related features. Its flow is shown in Figure 9:

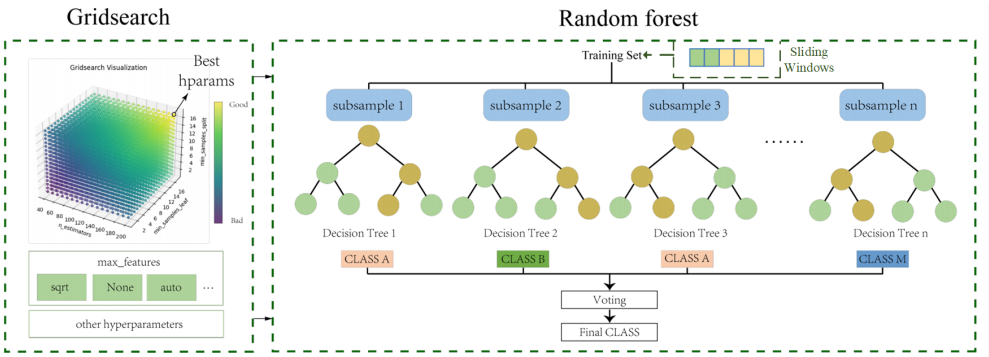

Figure 9: GSSRF

7.2 Prediction on Match and Test of Model

Using data from match "2023-wimbledon-1301", we employed grid search to select the best parameters for the random forest, as illustrated in the Table 6:

max _ depth	min _ samples split	n _ estimators
20	2	200

Table 6: Best Parameters

Where: **n _ estimators** represents the number of decision trees to be used, **max _ depth** controls the maximum depth of each decision tree, **min _ samples split** specifies the minimum number of samples required for a node to split into child nodes during tree growth.

We applied a sliding window to the "match flow" time series data of the first match, resulting in a new dataset. We then used this dataset as the training set for the random forest algorithm with a random seed set to 44. The ratio of the test set to the training set is 0.2. Subsequently, we made predictions using the trained random forest model and compared the results with the actual values, as shown in Figure 10:

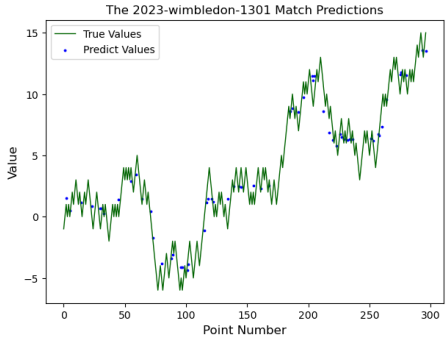

Figure 10: Predictions

By observing Figure 10, we noticed that the predicted values are roughly equal to the actual values. But to further validate the model, we continued to compute its mean squared error (MSE), root mean squared error (RMSE), and χ^2(Chi-Square Test), as are shown in Table 7:

MSE	RMSE	χ^2
0.9975687439546586	0.9987836322020193	0.9587307805304899

Table 7: Test of Model

Therefore, we obtained a small prediction error for the GSSRF prediction model, and there was no significant difference between the predicted values and the actual values. This also indicates the excellence of our prediction model.

We believe that accurately predicting who will win the next point is crucial for tennis matches. Therefore, we proceeded to train a second GSSRF model to predict the winner of the next point. In the end, we obtained the predictive results and **the model's evaluation score : accuracy is 0.98,macro avg is 0.98 and weight avg is 0.98**. Excellent scores prove the outstanding performance of our model.

7.3 Factors influencing the Match Flow and Advice

Based on the contribution of each feature value to the training of the random forest model, we can identify the factors that influence the match flow, as shown on Figure 11:

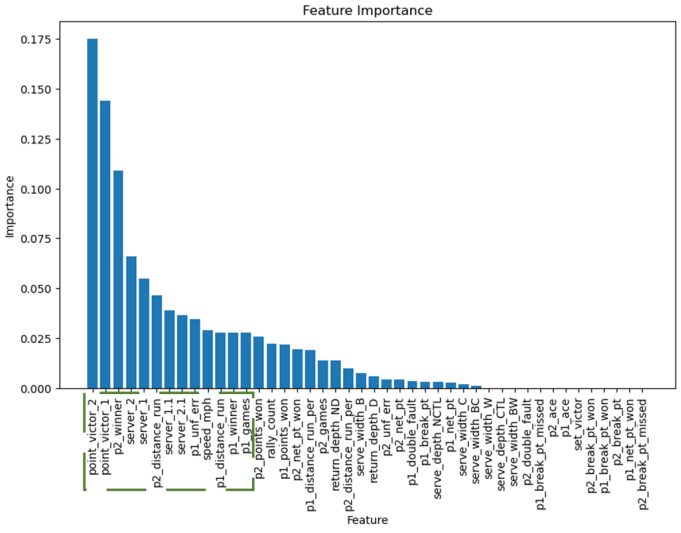

Figure 11: Factors related to match flow

Where: server_1.1 and server_2.1 represents serve in next round.

From Figure 11, we learned that except **winning the points, untouchable shot, serve, distance ran during the point , serve next round and opponent making unforced error**

all have a significant impact on the flow of match, and the factor related to the swings most is **untouchable shot**(except winning the point).

After analyzing the factors influencing match flow as mentioned above, we can delve into the data indicator momentum, which was not provided in the question. We are interested in understanding how momentum affects match fluctuations, in order to provide suggestions that can benefit the players in achieving victory.

We have constructed a the following quantitative formula for "momentum" based on the indicator data within the green dashed box.

$$\boldsymbol{M_t = M_{t-1} + (P_t \times W_t \times S_t \times G_t \times E_t - U_t)} \tag{6}$$

Where:

- P_t denotes winning the point (lose: 0 or win: 1)
- W_t denotes untouchable shot(not hit: 1 or hit: 1.2)
- S_t denotes serve(serve: 1 or no serve: 1.1)
- U_t denotes unforced error(no error: 1 or no error:1.3)
- G_t denotes winning the game(lose:1 or win:1.1)
- E_t denotes winning the set(lose:1 or win:1.2)

And we created the figure of momentum flow, as is shown in Figure 12:

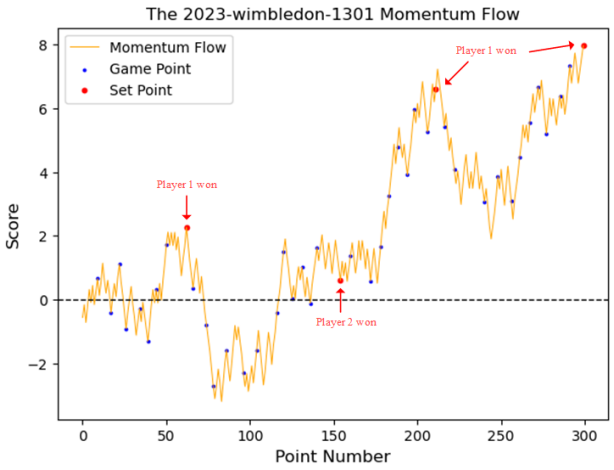

Figure 12: Flow of Momentum

We can learn from the Figure 12 that when the player grasps momentum, the match outcome is often in their favor. On the contrary, when the momentum is in the opponent's control, the match is unfavorable for him.

Therefore, player should grasp the momentum. When the opponent completely controls the momentum, player need to take full advantage of their serving opportunities because the chances of winning a point during serves are higher. This helps break the opponent's winning streak and provides a chance to seize the momentum.

When the opponent begins to control the momentum, player should alter strategy, like attacking the net more frequently or playing more defensively. Also, player can slow down the pace, for example, take your time between points, use your full allowed time for serves, and disrupt their rhythm.

When the player begins to control the momentum, player must keep winning points to fully control the momentum. Player can utilize serving opportunities, and speed up the pace to deny your opponent breathing room, like serving quickly and boost the speed of ball.

When the player fully control the momentum, player should stay focus, stick to the plan and maintain the rhythm to keep controlling the momentum.

8 GSSRF Prediction Model On Other Matches

We used the GSSRF prediction model established with data from matches "2023-wimbledon-1301" and "2023-wimbledon-1302". To validate the predictive performance of our model, we predicted other match data and compared it with the actual values, allowing us to analyze the effectiveness of our prediction model.

In this section, we will predict and analyze the data for match "2023-wimbledon-1401" and the US Open men's singles match. Figure 13 represents the comparison between the actual values and model predictions for match "2023-wimbledon-1401" (with the prediction target being the scores):

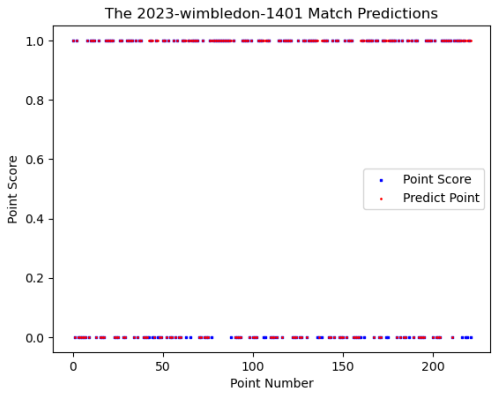

Figure 13: "2023-wimbledon-1401"

From the above figure, it can be observed that the distribution of actual value points is roughly similar to the distribution of predicted value points. Additionally, we obtained the predictive results and **the model's evaluation score:accuracy is 0.91,macro avg is 0.91, and weighted avg is 0.91** indicating that our predictive model performs well.

The complete model scoring table can be found in Appendix A.

9 Strengths and Weaknesses

9.1 Strengths

- **TEP** : The TEP algorithm achieves objective weight allocation by using a combination of entropy weight method and principal component analysis, avoiding the problem of subjectively setting the weight for each criterion in traditional TOPSIS algorithm. Furthermore, by combining these two algorithms, the TEP algorithm aims to achieve a more reasonable and fair distribution of weights among criteria, thus realizing a fair and reasonable weight distribution.
- **GSSRF** : Unlike the regular random forest algorithm, the GSSRF algorithm in section 7 can directly predict time series data and use the best hyperparameter combination to train and predict the random forest model, significantly improving the model's performance and avoiding overfitting or underfitting issues.

9.2 Weaknesses

- The momentum testing model cannot eliminate the influence of differences in player abilities.
- Most of the indicators used in the "GSSRF" model to predict match flow are unique to tennis matches, so our predictive ability for other types of matches is limited
- The weight allocation in the "TEP Evaluation Model" for the weighted summation of continuous and discrete data is subjectively determined.So there might be situations where the weight allocation is unreasonable or the scoring is slightly subjective.

References

[1] https://www.ibm.com/topics/knn

[2] Han Yuhong. A Comparative Study of Technical and Tactical Application in the Women's Singles Finals of the Australian Open and Wimbledon between Kerber and Williams in 2016 [D]. Capital University of Physical Education and Sports, 2018.

[3] https://www.oxfordlearnersdictionaries.com/definition/english/momentum?q=Momentum

MEMORANDUM

To: Coach

From: Team

Subject: Suggestions on Tennis Match

Date: January 20, 2025

Dear Coach, We are pleased to summarize the results of our research competition on the impact of "momentum" in evaluating current player performance and predicting future winning probabilities. We primarily analyzed various indicators using a random forest model based on grid search and concluded that "momentum" plays a significant role in tennis matches. We will provide recommendations to coaches regarding the role of "momentum" and help players prepare adequately using this information.

We are currently preparing to present the key results and recommendations over the next four days. Results: We initially preprocessed the data using the k-nearest neighbors algorithm with Euclidean distance and filled missing values with the mode. We retained 48 potential features that could be useful in predicting match trends. After conducting a thorough analysis, we decided to use a random forest prediction model based on grid search and sliding window. We utilized grid search to determine the best parameters and employed a sliding window approach due to the time series nature of match trends. The random forest model provided good predictive results for match trends.

Using the TOPSIS algorithm with entropy weighting and principal component analysis (PCA), we identified 8 quantifiable indicators of "momentum", such as delivering a winning shot, making unforced errors on both serves, winning points at the net, distance covered per shot, winning points at the net, and serving with unmatched power and speed. We conducted model validation through cross-validation, splitting the dataset into training and testing sets. We trained the training set using random forest, entropy weighting, and TOPSIS methods, and evaluated their performance on the testing set. The accuracy was approximately 0.87, with a recall rate of 0.90. The random forest model indicated the importance scores of features, and the predicted values were in close alignment with the actual values, demonstrating the model's superiority. Testing across different matches, venues, and sports showed the model's stability and strong generalizability.

The role of momentum in matches has been shown to motivate and propel players forward. This can guide coaches in providing support and encouragement to players during matches, helping them maintain or regain momentum.

Coaching Recommendations: Coaches can help players develop a set of strategies and tactics for different scenarios before matches. When a player performs well, such as delivering a winning shot, winning points at the net, serving with unmatched power, and

achieving consecutive wins, they are in a good momentum. Coaches should continue to encourage and support players to utilize momentum as motivation in matches. Additionally, if a player is making mistakes such as double faults, losing points at the net, or losing a break, they are lacking momentum. Coaches should identify and address these issues promptly, guiding players to turn the tide. Player Preparation during Tennis Matches: Players should understand when they are in a good momentum and when they are experiencing a low-point. Through training and practice, players can enhance their ability to maintain positive momentum and adjust their mindset during challenging times to find momentum again. By reaching the turning point and regaining momentum, players can increase their chances of success in subsequent matches.

In conclusion, we hope that our analysis will assist you in better understanding and utilizing momentum. Players can use this data to evaluate and improve their performance, ultimately increasing their chances of winning.

Sincerely,

Team #

Appendices

Appendix A Model Scoring Table

	precision	recall	f1-score	support
0	0.81	1.00	0.89	83
1	1.00	0.86	0.92	139
accuracy			0.91	222
macro avg	0.90	0.93	0.91	222
weighted avg	0.93	0.91	0.91	222

Appendix B Score Table

Point Number	p1_performance_score	p2_performance_score
1	0.063511424	0.622396994
2	0.67032023	0.267588577
3	0.065044902	0.623775082
4	0.780830847	0.266569697
5	0.781140374	0.260832761
6	0.211396698	0.619966704
7	0.812145305	0.277504784
8	0.211640984	0.813619889
9	0.681733569	0.265007826
10	0.668163391	0.266322224
11	0.215034761	0.748286827
12	0.669644741	0.265673151
13	0.672944589	0.062445176
14	0.212268439	0.741374312
15	0.212107583	0.620120111
16	0.664665871	0.264256348
17	0.214283935	0.624499482
18	0.214044553	0.825424007
19	0.782380792	0.260941953
20	0.666795186	0.267469195
21	0.212361104	0.624380592
22	0.669107383	0.059782459
23	0.667153894	0.265618372
24	0.067950719	0.623268008
25	0.213927657	0.623384863
26	0.210154947	0.737797406
27	0.216382847	0.741638745
28	0.777005157	0.266611692
29	0.668245547	0.268628157
30	0.679581759	0.265418854
31	0.06289857	0.621808495
32	0.21304544	0.625643817
33	0.068917025	0.624790259
34	0.684505674	0.266058745
35	0.685982411	0.269873969
36	0.218290241	0.625303431
37	0.669458674	0.063265674
38	0.21660589	0.624089474
39	0.211425207	0.620637205
40	0.214016177	0.623395965