Problem Chosen	2024 MCM Summary Sheet	Team Control Number
C		2406028

A study of in-game momentum based on XGBoost regression modeling

Summary

Momentum has always played an important role in the game.

Momentum has always played an important role in matches, but there are few quantitative studies on momentum and its evaluation and prediction. The purpose of this study is to quantify and define the momentum of a player in a game by analyzing the game data in depth, and to propose a model to predict the fluctuation of momentum in a game with momentum as the core concept.

Question 1: In order to quantify the performance of players, we first cleaned the data, then created several new evaluation metrics based on the match data and weighted them using hierarchical analysis (AHP); finally, we scored and ranked the performance of two athletes at a given moment based on the method of the distance between superiority and inferiority solving (TOPSIS), so that we can compare which one of them performed better at what moment. better. The result is a table of the performance evaluation scores for all moments for all athletes, which is only partially shown in the main text due to space constraints.

Question 2: In order to prove that players' performance fluctuations and streaks are non-random, we will perform the Kolmogorov-Smirnov test (K-S test) on the number of streaks and performance fluctuations (we regard changes in the rate of scoring as match fluctuations), which results in a p-value of less than 0.05, which indicates that when the difference between the cumulative distribution function and the theoretical distribution function turns out to be significant. The original hypothesis is rejected, a certain theoretical distribution is not met, or some other non-random pattern exists. Their performance in the game will then be analyzed to find possible influencing factors, combining the definition of "Momentum" and performance to define the finite difference of the TOPSIS evaluation scores as "Potential".

Question 3: In order to provide effective advice to coaches, we build a model based on Extreme Gradient Boosting (XGBoost) to predict the "momentum" in the game, firstly, we analyze the relevant influencing factors, and then we train the model. -The model we trained has a good fit: R2 = 0.742. Finally, we visualize the predicted values with the influencing factors, in order to find out in which way each factor affects the Momentum. Finally, based on their influence, suggestions are made to improve the momentum. Through the analysis, we found that:

Past point percentage, service breaks, service sides, winners, and past win streaks are positively correlated with momentum.

Break serve errors, past number of consecutive losses are negatively correlated with momentum.

Question 4: In order to improve the generalization ability of the momentum prediction model and to test its generalizability, we first downsize the independent variables of the original model based on exploratory factor analysis (EFA) to obtain high-dimensional features in order to simplify the model and improve the transferability; and then we conduct "momentum" prediction of the new model in other tennis datasets and table tennis datasets, and the prediction results show that "momentum" is a positively correlated with momentum, and "momentum" is a negatively correlated with the number of past wins and the number of past consecutive defeats. " prediction, and the prediction results show that our model has better goodness-of-fit on similar events and can be migrated on similar sports events.

Keywords: momentum, XGBoost model, tennis tournament, K-S test, superiority distance solution method, hierarchical analysis, exploratory factor analysis

Contents

1. Introduction

1.1 Background of the problem

As time goes by, data analytics in sports has become an important tool for athletes to strategize their game, and certain patterns are expected to be found in large amounts of data. In the 2023 Wimbledon Gentlemen's final between Carlos Alcaraz and Novak Djokovic, there was a remarkable fluctuation, which some attributed to "momentum", but the quantification and actual impact of "momentum" is not clear. However, the quantification and actual impact of "momentum" is unclear.

Against this backdrop, we will analyze the data from the 2023 Wimbledon Gentlemen's final in-depth to analyze the changes in momentum during the match, and further investigate its impact on athletes' performance to help coaches provide more effective coaching to their players. This will help coaches to provide more effective guidance to their players.

1.2 restatement

Combining the PROBLEM with the ANNEX, we will focus on analyzing the following questions:

1. construct a model that can quantify a player's performance at a specific time.

2. Demonstrate that player fluctuations and winning streaks during a game are non-random.

3. Build a model that predicts "momentum" in a game, analyze the model, and make recommendations to players.

4. Improve and test the model and study its scalability and generalization.

1.3 Problem analysis

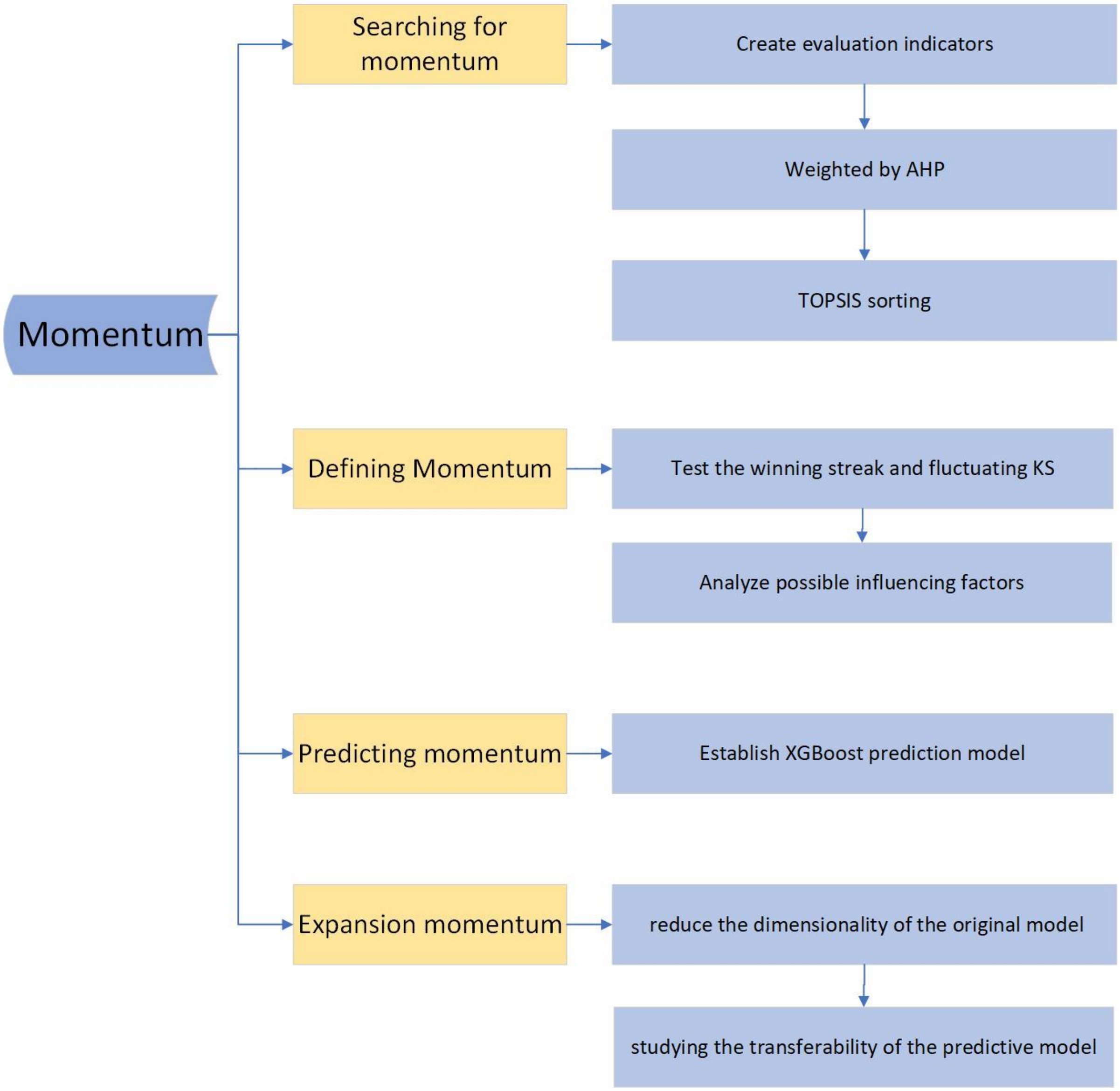
For Problem 1: In order to quantify the players' performance, an evaluation metric is created based on the flow of scores during the game and weighted using **the hierarchical analysis method (AHP)**; then the scores of both players are rated and ranked based on **the method of the Distance of Superiority and Inferiority Solving (TOPSIS**) for their performances at a given moment, which in turn compares which one of them performs better.

For Problem 2: In order to prove that players' fluctuations and streaks are non-random, we will perform **the Kolmogorov-Smirnov test (K-S test)** on streaks and fluctuations, and then we will analyze their performances during the game to find possible influences and define them as "momentum".

For Problem 3: In order to provide effective advice to the players, we will build a model based on **Extreme Gradient Boosting (XGBoost)[1]** to predict the "momentum" in the game and visualize the graphs to conclude which factors will affect the "momentum" in a certain way. ". Make recommendations to improve the momentum based on how it is affected.

For Problem 4: In order to investigate the generalization ability of the predictive model for Momentum and to improve it, we will reduce the dimensionality of the original model's independent variables based on **Exploratory Factor Analysis (EFA)** to obtain highly important and transferable features; and finally, we will migrate the model from one task to another related task and observe its performance on the new task. Task

1.4 Analyze the flowchart


Momentum
Searching for momentum
Create evaluation indicators
Weighted by AHP
TOPSIS sorting
Defining Momentum
Test the winning streak and fluctuating KS
Analyze possible influencing factors
Predicting momentum
Establish XGBoost prediction model
Expansion momentum
reduce the dimensionality of the original model
studying the transferability of the predictive model

2. Model Assumptions and Notation

2.1 Basic assumptions of the model

- assumes that the data set of Wimbledon 2023 Gentlemen's has good reliability or consistency and can consistently measure what it is intended to measure.
- It is assumed that the data set of Wimbledon 2023 Gentlemen's has good practicality in measuring what it is intended to measure.
- It is assumed that the model constructed has some migration between different sports and data sets.
- Assumes that athletes' performance can be quantified

2.2 Description of special symbols

In this work, we use the nomenclature in the table below in our model construction. Other less commonly used symbols will be additionally described once they are used:

Table 1: Illustrative table of mathematical symbols

Sign	Significance
$\boldsymbol{w_j}$	Weight of the jth indicator
$\boldsymbol{r_{ij}}$	Score of the jth indicator for the ith object
$\boldsymbol{PIS_j}$	Positive ideal solution for the jth indicator
$\boldsymbol{NIS_j}$	Negative ideal solution for the jth indicator
$\boldsymbol{v_{ij}}$	Weighted canonical matrix variables
$\boldsymbol{D_i^+}$	Distance of the ith object from the positive ideal solution
$\boldsymbol{D_i^-}$	Distance of the ith object from the negative ideal solution
$\boldsymbol{C_i}$	Composite evaluation index for the ith object
$\boldsymbol{Obj}$	Scoring Functions for Evaluation Models

3. Player performance analysis based on AHP and TOPSIS

3.1. Data preprocessing

In order to ensure the accuracy of the data, we will pre-process the data before data analysis as follows:3.1.1 Data cleansing

- **Vacant value filling**: check whether there are any vacant values in the data, the way of dealing with the vacant values will have an impact on the performance and generalization ability of the model, etc. We select the filling methods such as linear difference or mean interpolation according to the distribution of the data.

- **Outlier handling**: detecting outliers in the data, data outliers can cause bias or overfitting phenomenon in the model, etc. For variables with fewer outliers, take a case-by-case removal, and for more outliers, take interpolation and other processing methods.

3.1.2 dimensionless

Due to the differences in the magnitude of different data, it will cause problems such as low model generalization and data features are not obvious when model building and data comparison, we use the Min-Max Scaling method to process the data without the magnitude of the data, and the Min-Max Scaling method is as follows:

$$\frac{x - Min(x)}{Max(x) - Min(x)} \tag{1}$$

3.2 Definition and correlation analysis of evaluation indicators

In order to better evaluate the performance of the athletes, we extend the basic data to obtain indicators that can be used for quantification, as shown in the table below:

Table 2: Illustrative table of index

Index name	Indicator interpretation	Index calculation
last_time	How long the round lasts	elapsed_time's difference
get_point_speed	The number of points scored per unit of time	1/last_time
rate_score	The percentage of the current score	p_points_won/point_no
runs_won	Winning streak	Winning Streak Accumulation
runs_fail	Losing streak	Losing Streak Accumulation

After exported by SPSS, the same Min-Max Scaling process was performed.

In order to verify that the newly defined metrics have a degree of consistency that collectively describes the player's performance, a Kendall's W test was performed and the results of the Kendall's W analysis are shown in the table below:

Table 3：Kendall's W Analysis Results

Idex name	ordinal mean	upper quartile	Kendall's W factor	X^2	P
get_point_speed.1	3.449	1.15	0.037	655.346	0.000***
runs_won	2.971	0			
runs_fail	3.037	1			
rate_score	2.847	0.5			
server	2.696	0.5			

*Note: ***, **, * represent 1%, 5%, and 10% significance levels, respective*

As can be seen from the table, the overall data presents a significance p-value of 0.000*** at the level of significance and the original hypothesis is rejected, therefore the data presents consistency and we believe that they have the ability to explain the performance of the players together.

3.3 Determination of indicator weights based on hierarchical analysis

In order to more cognitively reflect the impact of each indicator on player performance and to combine quantitative and qualitative analysis, we will use the hierarchical analysis method (AHP)[2] to determine the weights of each indicator.

Firstly, the subjective evaluation matrix is obtained by comparing the relative importance of all the factors in this level against a factor in the previous level, as shown in the table below:

Table 4: Subjective evaluation matrix

Index	rate_score	get_point _speed	runs_won	runs_fail	server
rate_score	1	0.667	1	1	2
get_point _speed	1.5	1	1.5	1.5	3
runs_won	1	0.667	1	1	2
runs_fail	1	0.667	1	1	1
server	0.5	0.333	0.5	1	1

Then the approximation of the matrix eigenvectors is calculated based on the square root method and the AHP hierarchical analysis results are obtained as follows:

Table 5: Table of AHP hierarchical analysis results

Index	eigenvector	weighting (%)	Maximum characteristic root	CI value
rate_score	1.004	20.081	5.059	0.015
get_point _speed	1.506	30.121		
runs_won	1.004	20.081		
runs_fail	0.893	17.859		
server	0.593	11.859		

According to the calculation results, the largest characteristic root is 5.059, and according to the RI table, the corresponding RI value is 1.11, so CR=CI/RI=0.013<0.1, which passes the one-time test - therefore, the weights of each index are: "rate_score " has a weight of 20.081%, "get_point _speed" has a weight of 30.121%, "runs_won" has a weight of 20.081%, "runs_fon" has a weight of 20.081%, and "runs_feed" has a weight of 20.081%. "runs_fail" has a weight of 17.859% and "server" has a weight of 11.859%.

3.4 Evaluating player performance based on the superior-inferior solution distance method

In order to make full use of the information of the original data, the results can accurately reflect the gap between the evaluation programs, we based on the distance between superiority and inferiority solution method (TOPSIS) to carry out a comprehensive evaluation within the group, that is, the cosine method is used to find out the optimal program and the worst program in a limited number of programs, and then calculate the distance between each evaluation object and the optimal program and the worst program, respectively, to get the relative proximity of the evaluation objects to the optimal program, which serves as the basis for evaluating the superiority and inferiority.

Step1: Raw data normalization + co-trending

In TOPSIS, indicators are categorized into "positive indicators" and "negative indicators". In this survey, the positive indicators are: "rate_score", "get_point _speed", "runs_won", In this survey, the positive indicators are: "rate_score", "get_point _speed", "runs_won"; in order to accurately and objectively reflect the level of players, the negative indicators are: "server", "runs_fail". Since Min-Max Scaling was performed before the analysis, it is sufficient to take only absolute values for the variables.

Step2: Calculate the optimal and worst solutions:

Suppose there are n evaluation objects and m indicators:

The optimal solution is:

$$PIS_j = \max_{i=1}^{n} v_{ij} \tag{2}$$

The worst solution is:

$$NIS_j = \min_{i=1}^{n} v_{ij} \tag{3}$$

Step3: The gap between each evaluation metric and the optimal and worst vector is:

$$v_{ij} = w_j \cdot r_{ij} \tag{4}$$

$$D_i^+ = \sqrt{\sum_{j=1}^{m} \quad (v_{ij} - PIS_j)^2} \tag{5}$$

$$D_i^- = \sqrt{\sum_{j=1}^{m} \quad (v_{ij} - NIS_j)^2} \tag{6}$$

Where w_j is the weight (importance) of the jth attribute.

Step4: Measure the proximity of the evaluation object to the optimal program

$$C_i = \frac{D_i^-}{D_i^+ + D_i^-} \tag{7}$$

Where, C_i value is the comprehensive score index, and the larger indicates that the evaluation object is more optimal

Due to space constraints, a selection of scoring forms from one match are shown below:

Table 6: Player Percentage Score Sheet

player	match_id	set_no	game_no	TOPSIS_scores

Carlos Alcaraz	2023-wimbledon-1301	2.00	13.00	80.83
Nicolas Jarry	2023-wimbledon-1301	2.00	13.00	80.58
Nicolas Jarry	2023-wimbledon-1301	2.00	13.00	80.53
...	...	...	...	...

3.5 Performance scoring data visualization

In order to present more intuitively the fluctuation of the players' performance in the competition, we will show the effect graphically based on data visualization. Some of the graphs are shown below:

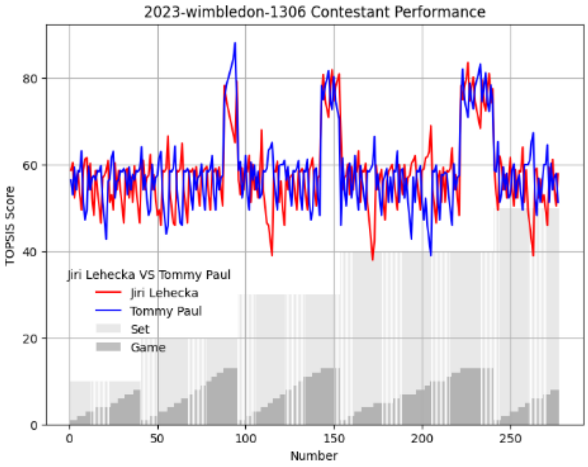


Figure 1

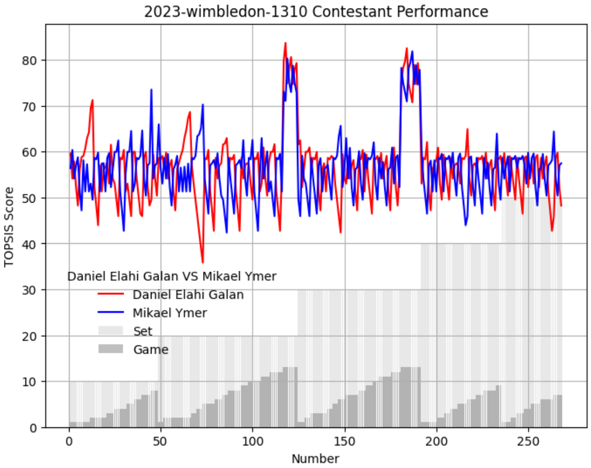


Figure 2

Analyzing the pictures, it can be found that there is a certain pattern in the performance of the two players, which can be clearly reflected in the figure is that they tend to perform at a higher level at the end of each SET, especially in the tiebreak.

4. Proof of non-randomness and definition of momentum based on K-S test

4.1 Proof of non-randomness of data

In order to verify that the hypothesis "swings in play and runs of success by one player are random" is correct, we will use the Kolmogorov-Smirnov test[3] (K-S test) since we do not know the exact distribution of the data. The Kolmogorov-Smirnov test (K-S test) is used to test the randomness of the variables "get_point_speed" and "runs_won".

The Kolmogorov-Smirnov test is a non-parametric statistical method used to test whether a sample conforms to a specific distribution, which is very suitable for situations where we don't know the exact distribution of the data at the moment. Again, to avoid the zero-value problem, kernel density estimates of the actual data are used to generate the expected data. We used ks_2samp in python's scipy to analyze the results as follows:

variable "Runs_of_Won" :

KS Statistic: 0.5028080607862571 p-value: 0.00

variable "Get Point Speed" :

KS Statistic: 0.49702675916749256 p-value: 0.00

The p-values are much less than 0.05, the original hypothesis is rejected and the observed effect is considered significant, which is analyzed in the context of the autocorrelation diagram:

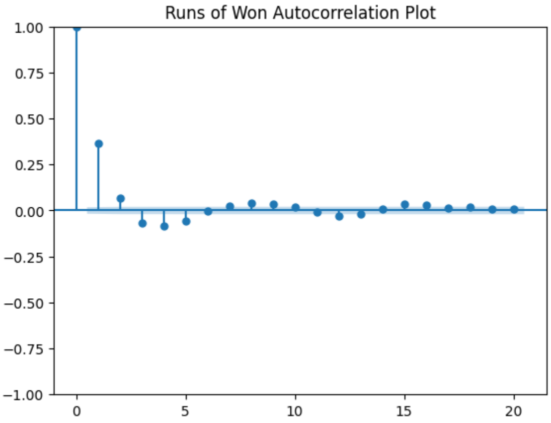


Figure 3

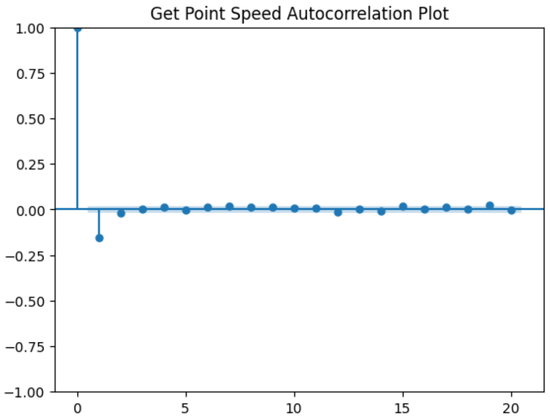


Figure 4

Demonstrates that the distributions of the number of streaks and performance fluctuations are different from the theoretical distribution and are significantly different. This may indicate that the distribution of winning streaks and performance fluctuations does not conform to a theoretical distribution or that there are other non-random patterns.

4.2 Definition of Momentum

4.2.1 Visualization and analysis of potential connections

To further explore what "momentum" is, we visualized the data from the previous question and presented it together with the TOPSIS scores, as shown in the figure (only two matches are shown for lack of space):

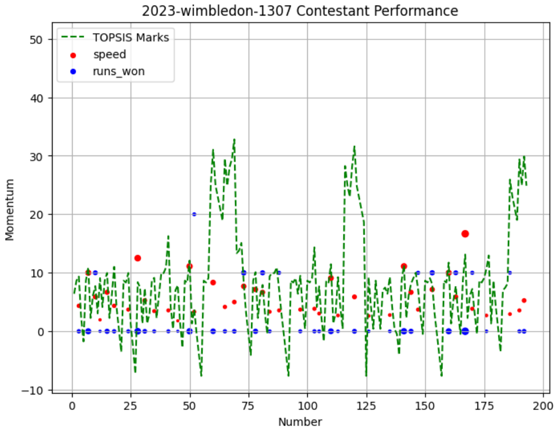


Figure 5

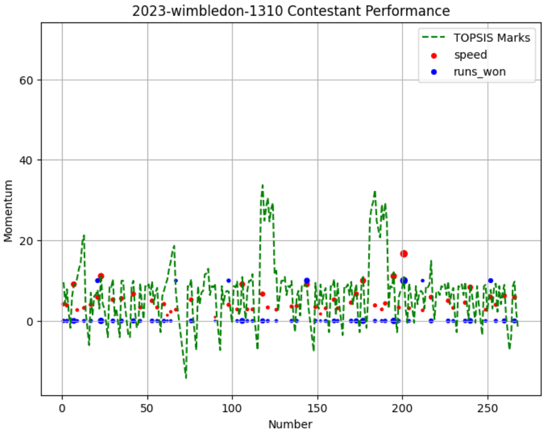


Figure 6

Looking at the picture, we can see that the TOPSIS score tends to increase when the number of wins and scoring rate are non-negative. To explore the deeper connection, we perform finite differences on the TOPSIS score, which is usually the equivalent operation of differential in discrete functions, and can reflect the trend of the score. This can reflect the trend of the scores, as shown in Fig:

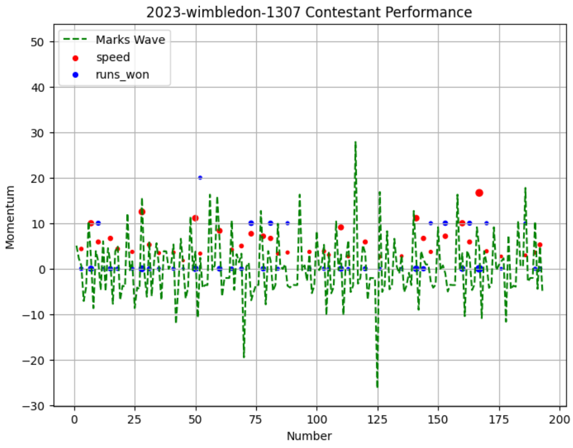


Figure 7

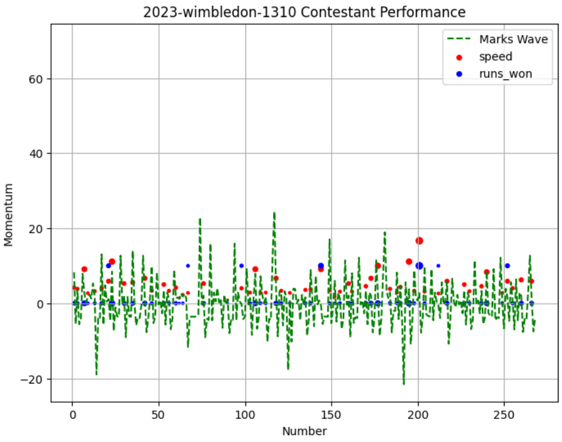


Figure 8

As can be seen from the figure, when the number of consecutive wins and scoring speed are non-negative, the finite differences of TOPSIS scores tend to be non-negative as well. In other words, the finite difference of TOPSIS score can reflect the trend of player performance.

4.2.2 A clear definition of 'momentum'

Since "momentum" usually refers to the trend of performance change in a match, and the finite difference score of TOPSIS score can adequately represent the trend of performance change, we get the following definition:

"Momentum" = finite difference of TOPSIS score

We use the term "mark_wave";

Again, depending on whether the momentum is increasing or decreasing, we can categorize it as:

"Positive momentum" = the finite difference of the TOPSIS score is positive

"Negative momentum" = the finite difference of the TOPSIS score is negative.

Here we have a clear definition of "momentum".

5 XGBoost-based prediction model for Momentum

5.1 Correlation analysis to identify variables associated with momentum

If we want to develop a model to predict the momentum of a game, we need to find the influencing factors from the known variables.

Since our data has a small linear relationship and is prone to extreme values, Spearman's correlation analysis can be used to measure not only the linear relationship between two variables, but also capture non-linear relationships; and since it is based on the rank order rather than the original numerical values, it has relatively little impact on extreme values (outliers).

For this purpose, we use Spearman correlation analysis[4] to analyze the degree of correlation between the possible influencing factors and momentum two by two to get the most relevant factors, and finally the factors with high correlation are shown in the table below:

The above data has strong prior correlation with "momentum".

5.2 Establishment of prediction model based on XGBoost regression analysis

5.2.1 Introduction of XGBoost regression analysis

Table 7: Spearman correlation test

Factor classification		English name	P-value
Performance factors		mark_wave	
Impact factor	Long-term impact factor	runs_won_past	0.381(0.000***)
		rate_of_point	-0.046(0.002***)
		rate_score_past	0.246(0.000***)
		runs_of_fails	-0.329(0.000***)
	Short-term impact factor	p1_ace	-0.044(0.004***)
		p1_net_pt	-0.025(0.095*)
		p1_break_pt	-0.114(0.000***)
		p1_break_pt_won	-0.074(0.000***)
		p1_break_pt_missed	-0.085(0.000***)
		server	-0.145(0.000***)
		set_no	-0.001(0.960)
		game_no	0.044(0.004***)
		is_tie_breakers	0.06(0.000***)

XGBoost is the abbreviation of "Extreme Gradient Boosting" (Extreme Gradient Boosting), XGBoost algorithm is a class of base functions and weights are combined to form a synthetic algorithm that has a good effect on data fitting.

Unlike the traditional Gradient Boosting Decision Tree (GBDT)[5], xgboost adds a regularization term to the loss function, which reduces the occurrence of overfitting. While observing our data, we can find that the degree of data discretization is high, if we use ordinary machine learning models, the pair is easy to cause fitting, and XGBoost is more efficient in dealing with large-scale datasets and complex models, and our dataset is very large, so we chose to use XGBoost for regression analysis.

XGBoost is an algorithm based on gradient boosting tree, so first of all, we need to understand the principle of gradient boosting tree.

For a dataset containing n entries of m dimensions , the XGBoost model can be expressed as:

$$\overset{\wedge}{y_i} = \sum_{k=1}^{K} f_k(x_i), f_k \in F(i = 1,2, \dots n) \tag{8}$$

$$F = \{f(x) = w_{q(x)}\}(q: R^m \rightarrow \{1,2, \dots T\}, w \in R^T) \tag{9}$$

Eq. (9) is the set of CART decision tree structures, q is the tree structure of the sample mapped to leaf nodes, T is the number of leaf nodes, and w is the real fraction of leaf nodes. When constructing the XGBoost model, it is necessary to find the optimal parameters according to the principle of minimizing the objective function to achieve the best results.The objective function of the XGBoost model can be divided into the error function term L and the model complexity function term Ω.

The objective function can be written as:

$$\begin{gathered} Obj = L + \Omega \\ L = \sum_{i=1}^{n} (y_i - y_i^{\wedge})^2 \\ \Omega = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \end{gathered} \tag{10}$$

where: γT is the L1 regular term and$\frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$ is the L2 regular term.

When using the training data to optimize the training of the model, it is necessary to keep the original model unchanged and add a new function f to the model so that the objective function is reduced as much as possible, the specific process is as follows:

$$\begin{gathered} \hat{y}_i^{(0)} = 0 \\ \hat{y}_i^{(1)} = \hat{y}_i^{(0)} + f_1(x_i) \\ \hat{y}_i^{(2)} = \hat{y}_i^{(1)} + f_2(x_i) \\ \dots\dots \\ \hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \end{gathered} \tag{11}$$

At this point the objective function is expressed as:

$$Obj^{(t)} = \sum_{i=1}^{n} (y_i - (\hat{y}_i^{(t-1)} + f_i(x_i)))^2 + \Omega \tag{12}$$

In the XGBoost algorithm, in order to quickly find the parameters that minimize the objective function, a second-order Taylor expansion of the objective function is performed to obtain an approximate objective function:

$$Obj^{(t)} \approx \sum_{i=1}^{n}[(y_i - \hat{y}^{(t-1)})^2 + 2(y_i - \hat{y}_i^{(t-1)})f_t(x_i) - h_i f_t^2(x_i)] + \Omega \tag{13}$$

When the constant term is removed it can be seen that the objective function is only related to the first and second order derivatives of the error function. At this point, the objective function is expressed as:

$$\begin{aligned} Obj^{(t)} &\approx \sum_{i=1}^{n}[g_i w_{q(x_i)} + \frac{1}{2}h_i w_{q(x_i)}^2] + \gamma T + \frac{1}{2}\sum_{j=1}^{T} w_j^2 \\ &= \sum_{j=1}^{T}[(\sum_{i \in I_j} g_i)w_j + \frac{1}{2}(\sum_{i \in I_j} h_i + \lambda)w_j^2] + \gamma T \end{aligned} \tag{14}$$

If the structural part of the tree q is known, the objective function can be used to find the optimal Wj and obtain the optimal objective function value. The essence of this can be categorized as the problem of solving the minimum of a quadratic function. The solution is obtained:

$$\begin{aligned} w_j^* &= \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \\ Obj &= -\frac{1}{2}\sum_{j=1}^{T}\frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \end{aligned} \tag{15}$$

By recursively calling the above tree building method, a large number of regression tree structures can be obtained, and use Obj to search for the most optimal tree structure and put it into the existing model, so as to build the optimal XGBoost model.

5.2.2 Selection of training parameters and code writing

For machine learning, the choice of parameters is crucial; for this training, we made several parameter adjustments, and the final parameter choices are as follows:

The base learner was selected as gbtree[6], the number was 100, and the learning rate was set to 0.1. Since the data was too discrete, we set the sample levy sampling rate, the tree feature sampling rate, and the node feature sampling rate to 0.7 to minimize the overfitting situation. For the canonical terms, L1 was set to 10 and L2 was set to 5 to allow data smoothing.

The evaluation results of the model obtained from the training are shown in the table below:

Table 8: Table of machine learning results

	MSE	RMSE	MAE	MAPE	R^2
training set	10.336	3.215	2.478	567.061	0.742
test set	15.314	4.828	3.673	571.315	0.611

The predicted values are plotted in a coordinate with the original values and the comparison graph is shown below:

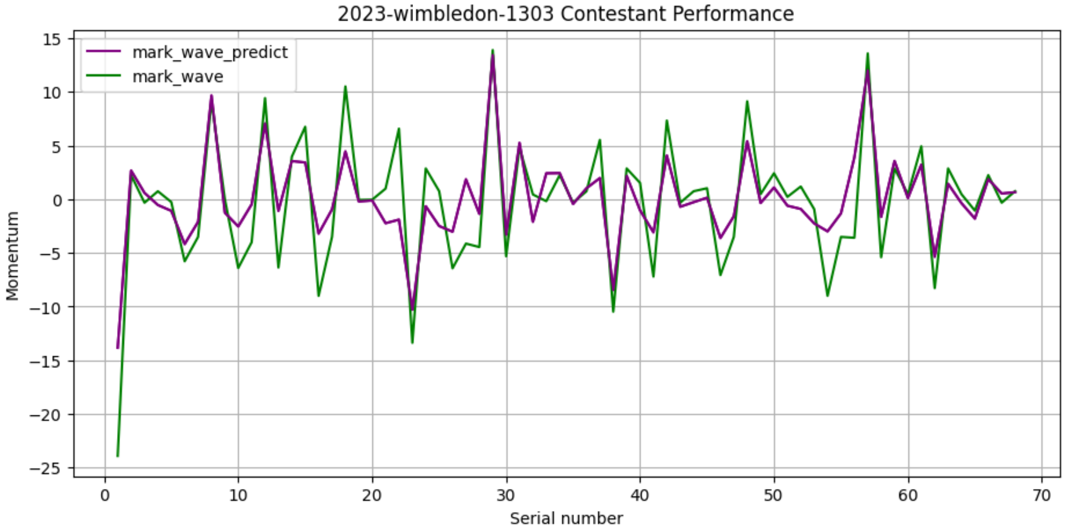


Figure 9

Combining the evaluation results and the comparison charts, it can be concluded that the modeling is more well established with a high confidence level.

5.2.3 Visualization of factor-regression curve relationships to analyze factor effects

In order to analyze the relationship between each influence factor and the "momentum", we plotted each influence factor and the predicted value in the same coordinate system, and observed and summarized the interactions, as follows (for the sake of space, we will only show individual graphs):

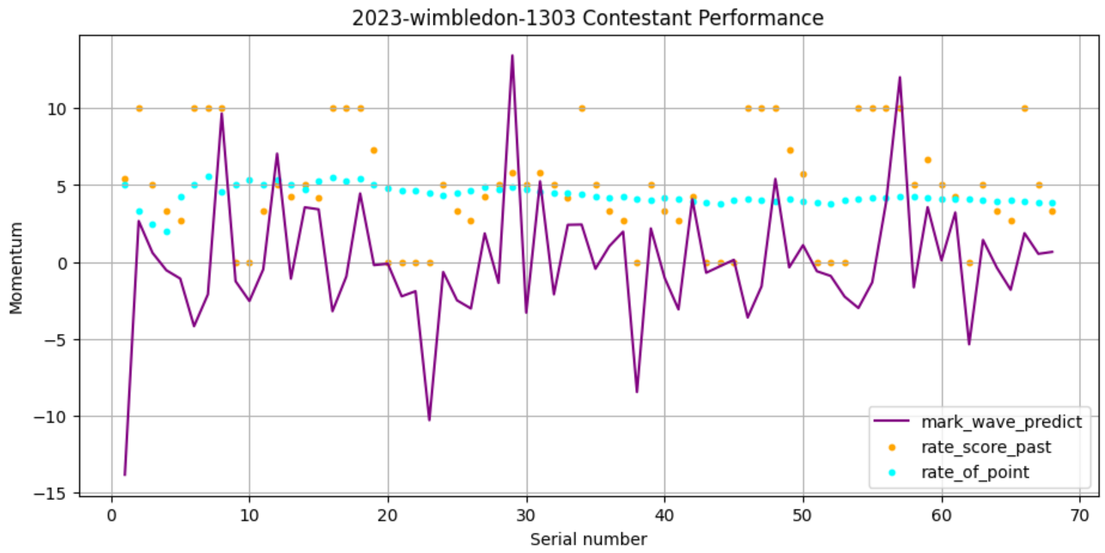


Figure 10

Based on the factor scatter plot and the predictive value line graph, we can get some conclusions to help coaches and students to analyze the "momentum".

5.2.4 Presentation of conclusions and recommendations

According to the above asked chart, the following results can be obtained:

·Past point percentage, break serve successes, service sides, winners, past winning streaks are positively correlated with momentum

·Break serve errors, past losing streaks are negatively correlated with momentum

·Whether or not it's a tiebreaker tends to create huge momentum swings, and it's impossible to tell if it's positive or negative.

Some suggestions for coaches would be to be vigilant when there are too many service break errors and past consecutive losses, and to try to increase their momentum with successful service breaks, etc., and to be careful with service breaks when the player is on serve, and to use the positive momentum to ride the momentum.

6. Model Improvement and Verification of Generalization Effect

6.1 Exploratory Factor Analysis Dimensionality Reduction for Improved Generalizability

Testing the developed model on multiple other matches, we found that the prediction of fluctuations was much less effective than on the original model. After analytical studies, it was found that the more complex the model, the worse the generalization ability, and the variables used for the original model training were not present within certain scenarios.

In order to improve the migration ability of the model, we performed an exploratory factor analysis on the factors of the model to reduce the feature dimensions, and the results of the factor analysis are as follows:

Table 9: Factor analysis results table

	New Factor		
	Historical_Achievement	Skill	Current_Situation
set_no	-0.001	0.004	0.023
game_no	-0.001	0.006	0.574
server	0.047	-0.224	-0.065
p1_ace	0.05	-0.091	-0.049
p1_net_pt	0.007	-0.083	-0.001
p1_break_pt	0.029	0.457	0.003
p1_break_pt_won	0.014	0.264	-0.007
p1_break_pt_missed	0.025	0.003	0.008
is_tie_breakers	-0.008	-0.003	0.573
runs_won_past	0.367	-0.032	0.014
runs_of_fails	-0.358	0.036	-0.013

For the new three factors, we categorized them as historical performance, skill, and current situation. Then re-train them as independent variables again for XGBoost prediction model training, and the results of this newly trained model evaluation are as follows:

Table 10: XGBoost prediction model training results

	MSE	RMSE	MAE	MAPE	R^2
training set	11.721	3.424	2.752	998.6	0.702
test set	10.581	3.708	2.374	979.097	0.731

Comparing the results of previous evaluations, we can find that the model trained after dimensionality reduction has increased the goodness of fit on the test set. It can better predict the "momentum" of other data sets.

6.2 Predictions for other matches

The following prediction curves and evaluations were obtained by predicting data from other tennis matches and table tennis matches, respectively:

Table 11: Predicted results table for other matches

Evaluation indicators	Evaluation results of table tennis tournaments	Evaluation results of other tennis tournaments

MSE	271.90	107.08
RMSE	16.49	10.35
MAE	11.93	7.70
R^2	-0.31	-0.07
MAPE	599.74	1444.61

Tennis:

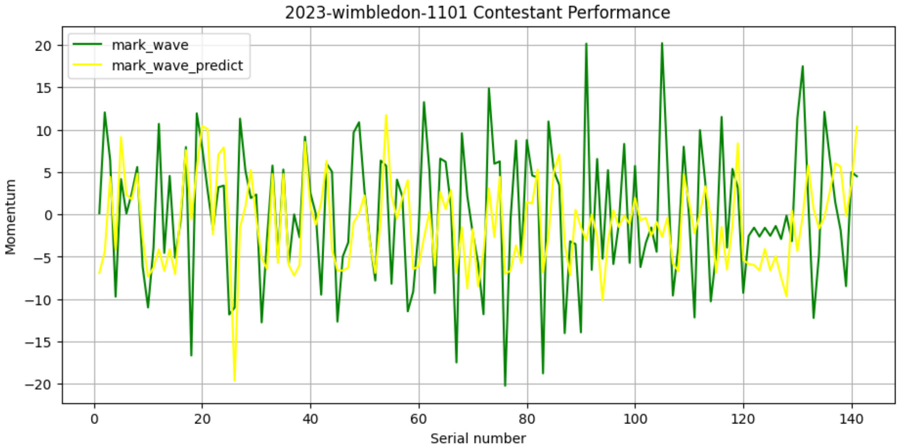


Figure 11

Table tennis:

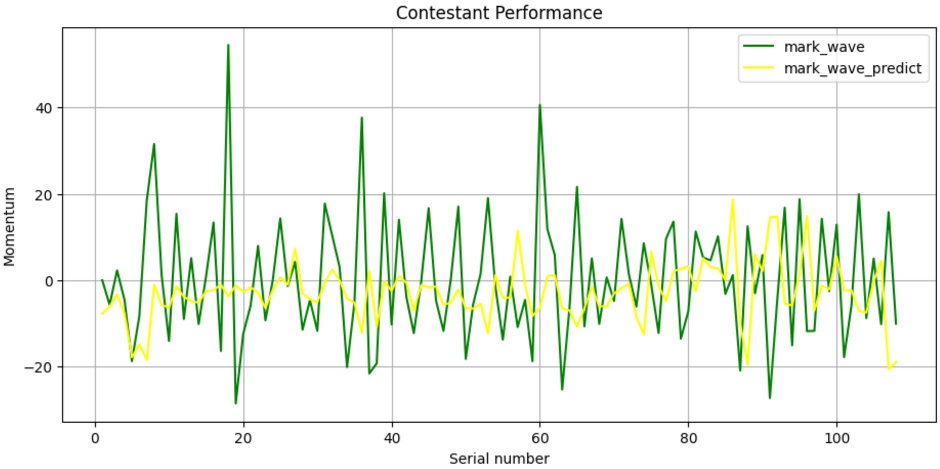


Figure 12

7. Conclusion

7.1 dominance

- Our model is built on XGBoost (Extreme Gradient Boosting), which is more efficient in dealing with large-scale datasets and complex models, and also performs well in preventing overfitting and improving generalization.

• Unlike traditional Gradient Boosting Decision Trees (GBDT), xgboost adds a regularization term to the loss function to more accurately handle new datasets.

• In point 6 we use sex factor analysis to reduce the complexity of the new model trained after dimensionality reduction to greatly reduce the complexity of the new model, allowing for better migration learning and model generalization capabilities.

7.2 Possible improvements

·Accuracy needs to be improved, perhaps the training parameters of the model need to be adjusted to improve the model's goodness of fit and reduce the mean square error

·The model's explanation of periodicity needs to be improved, perhaps by Fourier transforming some of the variables and then using some of the periodicity models, such as wavelet transforms, seasonal decomposition methods, and other models

·It can be combined with other models to better provide reliable advice to athletes or coaches.

8. Memo Summary

From: Team #2406028

Date: February 6, 2024

Topic: Research on momentum in matches based on XGBoost regression model

Dear Sir Or Madam,

Coaches or players are often interested in momentum in a game, but momentum has not been fully quantified and predicted in today's research. So in our work, we provide you with the best evaluation and prediction models based on tennis match data. We are excited to write this memo to show you how our model works, and then we will show you the results of our model. ,

Now I would like to present to you the results of our research on momentum, which is mainly divided into three parts. In the first part, we set up a new evaluation index system based on TOPSIS to measure the performance of players in a match; In the second part, K-S test is used to prove the non-randomness of streak and score fluctuation, and according to his association with momentum, momentum is clearly defined as the finite difference of performance evaluation score.

In the third part, we build a momentum prediction model based on XGBoost model in order to provide effective advice to coaches or athletes. In order to improve the accuracy of prediction, we study various models and algorithms, such as neural networks and traditional gradient lifting decision trees. Finally, we find that the extreme gradient ascent method performs better and has higher accuracy in our prediction work.

In order to enable him to have strong generalization ability in different data sets of multiple games. We use exploratory factor analysis to simplify the quality of the model, which reduces the complexity of the model and improves the generalization ability of the model. Then we tried our model on additional tennis match data and table tennis match data. The results show that our prediction model still performs well on the new data set. I hope our model can help you. Analyze changes in momentum during the game.

We want our model to be instructive. Through this research, we expect to provide coaches and players with innovative tactical analysis tools to help them more fully understand the dynamics of the game in order to optimize tactical decisions."

Finally, thank you for your interest in our strategy and wish you the best in life!

Your Sincerely,

Team#2406028

9. Bibliography

[1] D. Barochiner, R. Lado, L. Carletti and F. Pintar, "A machine learning approach to address 1-week-ahead peak demand forecasting using the XGBoost algorithm," 2022 IEEE Biennial Congress of Argentina (ARGENCON), San Juan, Argentina, 2022, pp. 1-5

[2] Fajwel Fogel, Alexandre d'Aspremont, and Milan Vojnovic. 2016. Spectral ranking using seriation. J. Mach. Learn. Res. 17, 1 (January 2016), 3013–3057.

[3] Imanol Arrieta-Ibarra, Paman Gujral, Jonathan Tannen, Mark Tygert, and Cherie Xu. 2022. Metrics of calibration for probabilistic predictions. J. Mach. Learn. Res. 23, 1, Article 351 (January 2022), 55 pages.

[4] S. Peng, R. Cheng and Z. Dai, "Study on Ethanol Coupling Reaction Based on Regression Algorithm and Spearman Rank Correlation Coefficient," 2022 11th International Conference of Information and Communication Technology (ICTech)), Wuhan, China, 2022

[5] B. N. Kommula, V. R. Kota and N. R. Tummuru, "Maximum Power Point Tracking for Photovoltaic Brushless DC Motor Connected Water Pumping System Based on GBDT-BOA Technique," 2021 IEEE International Power and Renewable Energy Conference (IPRECON), Kollam, India, 2021, pp. 1-6

[6] Philipp Probst, Anne-Laure Boulesteix, and Bernd Bischl. 2019. Tunability: importance of hyperparameters of machine learning algorithms. J. Mach. Learn. Res. 20, 1 (January 2019), 1934–1965.