Problem Chosen	2024 MCM/ICM Summary Sheet	Team Control Number
C		2407527

The Magical Force Determining Fate in Tennis——"Momentum"

Summary

In tennis, momentum is described as the strength or confidence gained through events, which is the psychological and physiological state of players. The transformation of momentum is a key factor in determining the victory. We first establish a mathematical model of momentum and study the correlation between momentum and outcome. Based on this, we establish a model to measure player performance and predict the transition point of the competition situation. Study of this issue can provide tactical and training guidance for players and coaches to predict the game situation and to performance well .

The **first** model is to measure the performance of tennis players in a match. It is influenced by three factors: athlete's technical ability, fatigue level, and momentum. We first constructed a momentum model and considered the influence of multiple factors on momentum, including scoring factors, serving factors, etc. By setting weight, we quantified the **concept of momentum**. Subsequently, we used **logistic regression** to establish a model for measuring the performance of athletes, with an accuracy of **0.694** was visualized and presented. We conducted a significance test on the eleven performance factors of logistic regression using **SPSS**, and found that five indicators had a significant impact on the scores of athletes, such as leading score, order of hand, running distance, etc. At the same time, the **Bayes model** was used for comparison and it was found that the logistic regression model is more suitable. Finally, we used **Pearson** correlation test and successfully demonstrated a significant correlation between momentum and score.

The second model is to predict turning points in tennis matches. We define the moment when momentum changes sharply as the turning point and use **random forest model** for prediction, achieving an accuracy of **0.686**. We visualized the prediction results and studied the importance of each indicator, and found that running distance, order, and score leadership have the **greatest impact** on the turning point . Therefore, we have put forward some suggestions, including strategies such as Inducing opponents to run more, reducing one's own running distance, and providing differentiated suggestions for specific athletes.

Finally, we conducted model extension testing to verify its **generalization ability**. By using our model on previous male/female singles datasets for prediction, the results show that the model performs well .However, When applied to other fields, some indicators, such as breaking rate, may not exist and the rules may be different. We provide some suggestions, including transforming features, adding more rules, etc.

Keywords: Tennis, Logistic regression, Random forest, Tennis

Contents

1 Introduction

1.1 Problem Background

In the 2023 Wimbledon Gentlemen's final, the young and talented 20-year-old Spanish player, Carlos Alcaraz, achieved a stunning victory over the seasoned 36-year-old Novak Djokovic. The match itself was nothing short of extraordinary with its roller coster-like score between the two players.

The remarkable shifts, occasionally spanning numerous points or even entire games, observed in the player who appeared to be in control are frequently ascribed to the concept of "momentum."In sports, momentum is described as the "strength or force gained by motion or a series of events."

The acquisition of momentum is vital in tennis as it provides the player with a feeling of command and places pressure on the opponent to regain control. When a player gains momentum, they tend to play with increased freedom, exhibiting less fear and inhibition. This often translates into a more aggressive and confident playing style. [1]

Capturing and measuring this force during a match or game can be challenging. Additionally, understanding how various events contribute to the creation or alteration of momentum remains a complex and intriguing aspect of sports analysis.

Now, let's explore how we can analyze momentum in tennis and how to utilize it well!

1.2 Our work

1. For 1, 2 questions, we want to build a regression model to capture the changing situation of the scene and measure the real-time performance of the players. We studied many factors to quantify the concept of "momentum" .Next,we used a logistic regression model .We considered momentum and other factors as independent variables and whether they could be scored as dependent variables. Finally, use the matplotlib library to visualize the player's performance.

2. Then, through Pearson correlation test and visualization processing of momentum and outcome in the model, we proved the correlation between momentum and results of the matches to the tennis coach, thus proving that momentum and outcome played a key role in the match.

3. We believe that the turning point in the game is the huge momentum transition point of the player. Based on the momentum model, the turning point is determined by according to the speed of momentum changing. Then the characteristic results are analyzed by Random Forest, and the game styles of different players are analyzed, and the general suggestions and suggestions for players are put forward.

4. We test the model on other data sets to verify the model's generalization of data. The model extends from special to general, from men's tennis field to women's tennis, table tennis, badminton and so on. The ideas and methods of model expansion for other types of sports competitions are also given.

5. Finally, we summarize a memo report, propose the role of momentum in tennis matches, and give some strategy suggestions for tennis players

2 Assumptions

To simplify our model and reduce complexity, we make the following assumptions in the paper.

1. Server advantage hypothesis: The server is more likely to win points than the receiver, which will affect the outcome.[2]

2. Momentum influence hypothesis: A player's performance is affected by momentum in the race, and past success or failure will affect future performance.

3. Stable condition assumption: the type and condition of the tennis court are fixed and will not change with the weather, temperature, humidity and other factors.[3]

4. The assumption of fairness of the match environment: the referee of the tennis match is fair, and the sponsors and media of the match will not interfere with the players due to commercial interests or public opinion.

3 Notations

Table 1: significance test

symbol	illustrate
P_t	Point winner
S_t	Server
n	total number of rounds of the match
I(cdot)	Indicates
diff	Game difference between players
$\text{score}_{\text{diff}}$	Score difference between players
rally_factor	Normalized number of rounds
distance_factor	normalized running distance
break_point_value	The basic momentum score increase value
S_{pw}	consecutive scoring
S_{gw}	the number of consecutive winning games
BS_{diff}	a large score difference
SS_{diff}	a small score difference
BP	The impact of a break
R_{factor}	the number of beats
I(rally_count > 10)	multi-goal scoring
I(ACE)	ACE score
I(unforced_error)	The impact of unforced errors

4 Model I:Model for Evaluating Player Performance

4.1 Data Preprocessing

1.**Categorical Values:**
The dataset contains numerous categorical variables in string format. To facilitate regression modeling, we encoded and assigned the values of the last three columns of tennis match.We unified all categorical variables into numerical values for subsequent processing. For instance, converting "AD" in the score to 50, replacing "ND" and "D" with "5" and "15" in the returning depth, and replacing "CTL" and "CT" with "6" and "3" in the serving depth.[2]

2.**Abnormal Value:**
Abnormal Value can have a detrimental impact on the model. Following the 3-standard deviation rule, the range of outliers is determined, marked, and addressed using mean imputation.We ensure the accuracy and consistency of data, preventing its impact on subsequent analyses and the model.

4.2 Model for Evaluating Player Performance Construction

We reviewed literature investigating the determining factors in tennis matches, categorizing them into three main types: the overall performance of the player, the overall performance of the opponent, and the decisions made by the umpire. Among these, the overall performance of the opponent and the umpire's decisions are beyond the player's control. In contrast, the player's own overall performance is the only aspect they can genuinely influence and control, making it the core factor in ultimately securing victory.

The current determining factors are primarily classified into three types: fatigue level, individual technical ability, and player's state . We summarized the state as "momentum" .[4]

- For individual technical ability, we can calculate it using past or real-time player scoring situations.
- For player fatigue level, we can calculate it using the player's running distance.
- Momentum is a force or power gained through motion or a series of events. We use events that can influence player's state such as serving errors, consecutive wins, and direct serving points, to quantify momentum.

Table 2: Variable

Variable Name	Description
X_1	Score Lead Progress: When the player is in the lead
X_2	Whether the previous point was scored
X_3	Whether the serve resulted in a point
X_4	Whether the return resulted in a point
X_5	Whether there were double faults in this game
X_6	Whether there were unforced errors in this game
X_7	Net Approaches and Net Points Ratio
X_8	Total Running Distance in the Last Three Points
X_9	Serve and Return Win Percentage
X_{10}	Whether it is a serving game
X_{11}	Momentum

4.3 Build the "momentum" Model Construction

In order to comprehensively measure the state of players in a tennis match, we consider the impact of multiple key events on momentum.

The current scoring situation and the base momentum score determined by the server reflect the key scoring situation in the match. Consecutive points scored and consecutive small game wins reflect the additional motivation a player may generate while maintaining a winning streak in a match. The breaks and missed breaks reflected key moments in the match. The number of rounds and distance run can assess the physical condition of the player. Multi-ball points and ACE points provide bonus points and emphasize performance in tight rallies. Unforced errors reflect a player's consistency at a crucial moment. For the relevant weight and coefficient setting of each factor, we refer to the existing literature on tennis winning factors. [4][5][6][7]

$$\text{momentum\_score} = \text{base\_momentum} + \text{S}_{\text{pw}} + \text{S}_{\text{gw}} + \text{BS}_{\text{diff}} + \text{SS}_{\text{diff}}$$

$$+\text{BP} + \text{R}_{\text{factor}} + \text{I}(\text{rally\_count} > 10) + \text{I}(\text{ACE}) - \text{I}(\text{unforced\_error}) \tag{1}$$

- Scoring event:

  Base momentum score increase is equal to the current scorer (P_t) multiplied by the server weight (S_t). When scoring consecutively, additional points are added for each consecutive win: momentum_score is increased by $0.03 \times$ consecutive_point_wins.

- Serve event:

  If it is the server, increase the weight of the serve score: the server's momentum score weight (S_t) is 1.2, otherwise, it is 1.0.

- Small Game win event :

  When consecutive games are won, additional points are added for each consecutive win: momentum_score is increased by $0.2 \times$ consecutive_game_wins.

- Big score gap :

  Calculate big margin compensation: momentum_score is increased by $0.1 \times \text{diff}^2$, where the diff is the difference between the current winning player and the opponent.

- Small score gap :

  Calculate small margin correction: momentum_score is increased by $0.02 \times$ score_diff $\times P_t$.

- Missed break point events:

  The momentum point added value of the break is reduced by missed break points: the break_point_value is reduced by 0.1.

- Number of runs and distance events:

  Increase the momentum score by normalizing the number of turns and distance run: momentum_score is increased by $2.0 \times$ rally_factor $\times$ distance_factor $\times P_t$.

- Multiple ball scoring events (rally_count):

  If the number of turns exceeds 10, additional points are added: momentum_score is increased by 0.5.

- ACE score event (p1_ace, p2_ace):

  If there is an ACE score, add additional points: momentum_score is increased by 0.02.

- Unforced error events (p1_unf_err, p2_unf_err):

  In the case of unforced errors, reduced score: momentum_score is decreased by 0.05.

4.4 Logistic Regression Model Construction

We choose to use statistical logistic method to test the significance of the relationship between the 11 indicators and whether the score can be scored.

The dependent variables of the model are score (1) and no score (0), which is a classification problem, and the output of Logistic regression is a binary classification label, which is an ideal tool for this problem.

Secondly, Logistic regression provides the probability estimation of scores for each indicator, so that we can understand the degree of influence of various factors on the outcome of the game. Finally, through Logistic regression, we can conduct a significance test to determine whether each indicator has a significant impact on the score. This helps to eliminate non-significant factors, thereby refining the model and improving the ability to accurately predict winning.

4.4.1 The significance of each index

The score of each point in each match is labeled (score 1, no score 0).Each indicator of the player when they scoring are variables. The data were imported into **SPSS** for binary logistic regression analysis to determine the significance of these indicators. The results are shown in the figure below.

On the other hand, for the variable analysis in the equation. When the P-value is less than 0.05, the variables are statistically significant, while when the P-value is greater than 0.05, the variables are not statistically significant. It is found that half of the variables are significant to the prediction of the model, including x1,x4,x6,x9,x10. User

Score Lead Progress, Return Point, Unforced Errors, Serve and Return Win Percentage, Serving Game, these 5 technical indicators can significantly affect the player's score, and the other indicators have little impact on whether the score can be scored.

Table 3: significance test

feature	feature weight	significance
x1	-0.003	0.220
x2	-0.046	0.609
x3	-0.107	0.195
x4	-0.714	<0.001
x6	0.657	<0.001
x7	0.292	<0.001
x8	0.000	0.966
x9	-1.368	<0.001
x10	0.104	0.078
x11	0.141	<0.001

4.4.2 Player performance visualization

We visualize the models that measure a player's performance in real time.

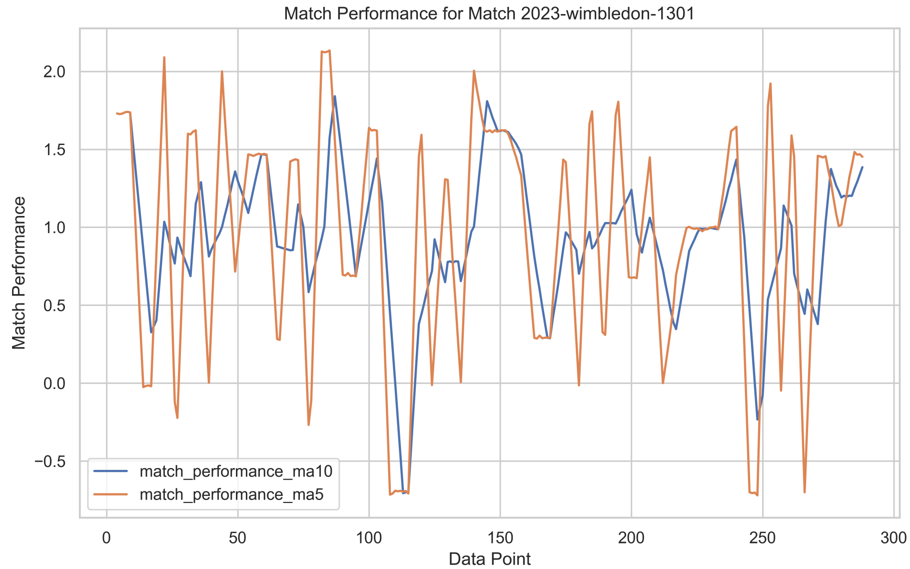

Figure 1: match performance 2023-wimbledon-1301

4.4.3 Accuracy of the model

The accuracy of binary logistic regression is 0.6954. This has a good prediction effect for predicting whether a player can score in the actual competition, indicating that this model can evaluate the performance ability of players. However, The accuracy of prediction of gain score is 0.78, and the accuracy of successful prediction of lost score is 0.58.The model prefers to classify samples with label 1.

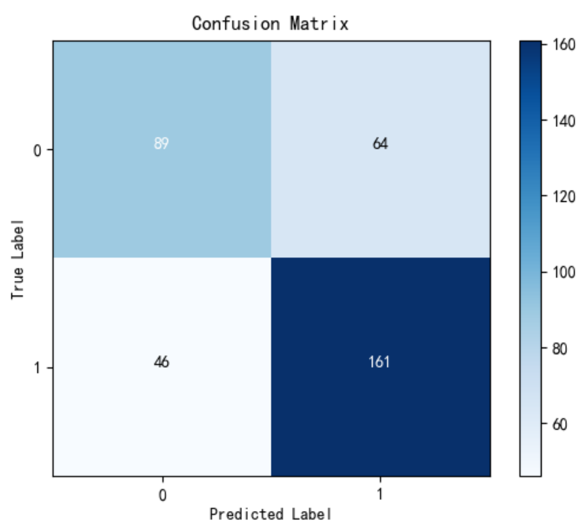

Figure 2: LR model Confusion Matrix

4.5 Comparison with other models

We also trained other models to compare with logistic regression models.We used Gaussian NB Bayes algorithm for comparison, evaluated auc indicators, and verified with the Five-Fold Cross-Validation. The results are as follows: ROC diagram shows that LR regression has better effect.

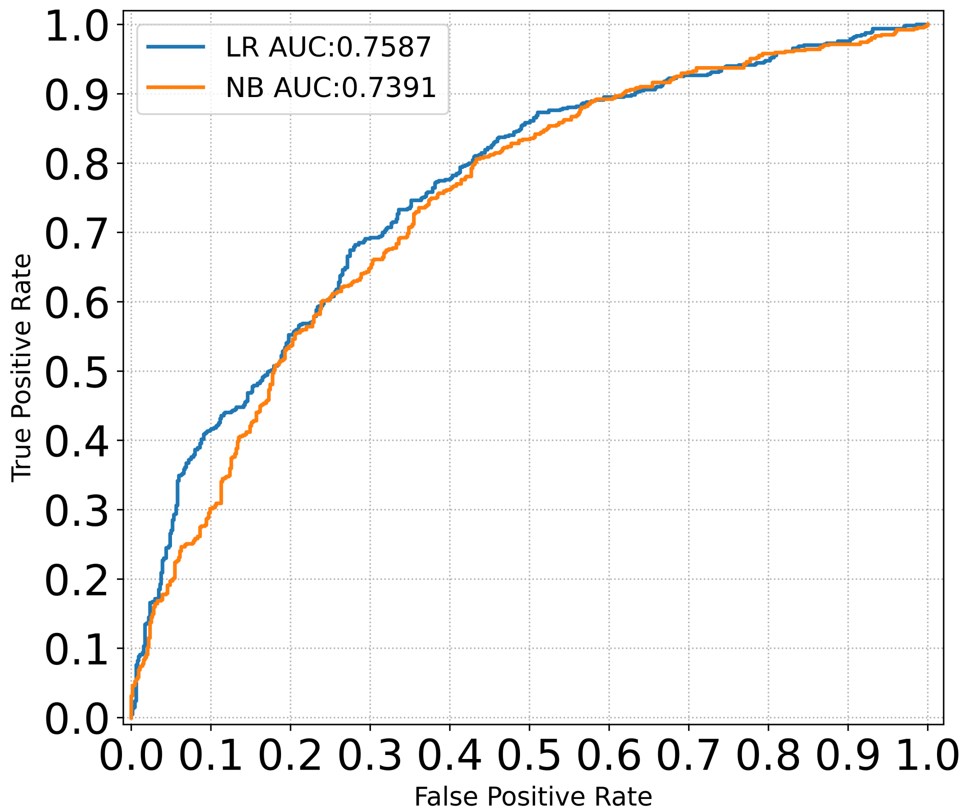

Figure 3: Comparsion with GaussianNB

5 Correlation analysis of momentum and victory

The coach thinks the players' momentum changes randomly during the game, and the role of momentum in the game is questioned. We will use the knowledge of statistics and visualization to verify the authenticity of the coach's view.

5.1 Pearson Correlation test

Null hypothesis (H0): There is no statistically significant relationship between a player's score in a match and momentum.

Alternative hypothesis (H1): There is a statistically significant relationship between a player's score in a match and momentum.

Significance Level: We set the significance level at 0.05, that is, we are willing to make the mistake of rejecting the null hypothesis with a probability of 5 percent.

The p-value represents the observed correlation coefficient or the probability of more extreme cases occurring,if no linear correlation is assumed between the two variables.

- Pearson correlation coefficient: 0.389
- P value: 2.1e-188

According to the calculated results, the null hypothesis can be rejected, and there is a significant correlation between the quantified momentum data and the score

5.2 visualization test

Visualize the momentum with the score difference between the two players, as shown in the figure, and observe that the trend in the figure shows a high degree of similarity. This indicates that the change in momentum is not random, but has a high correlation with how well the players score.

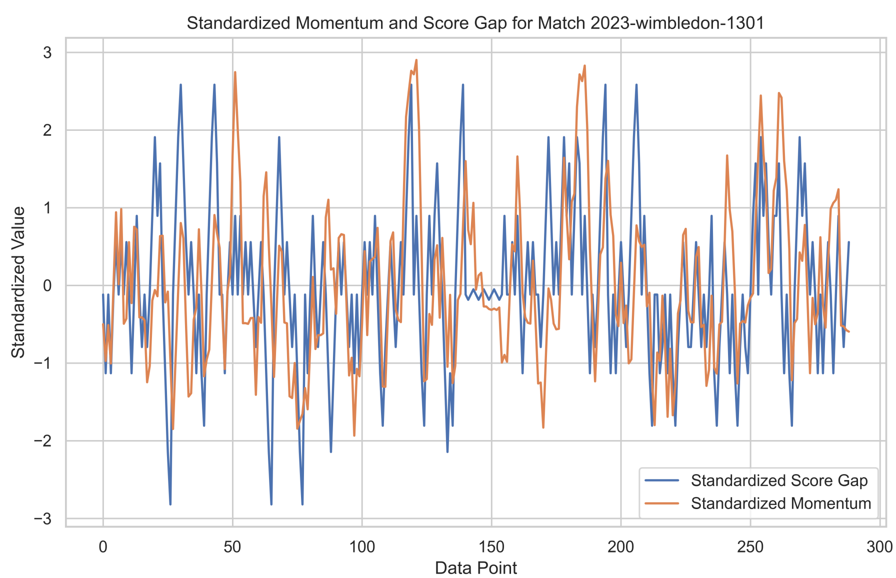

Figure 4: standardized momentum scoregap 2023-wimbledon-1301

6 Model II:Models to predict the shifts of match trends

6.1 Seek transition points

The momentum curve itself is a barometer of the game, which can show that when the situation of the game shifts from one player to another at a certain moment.Thereforemwe define the time when momentum change fast as the transition point. Then the predicted transition points were obtained by random forest model.

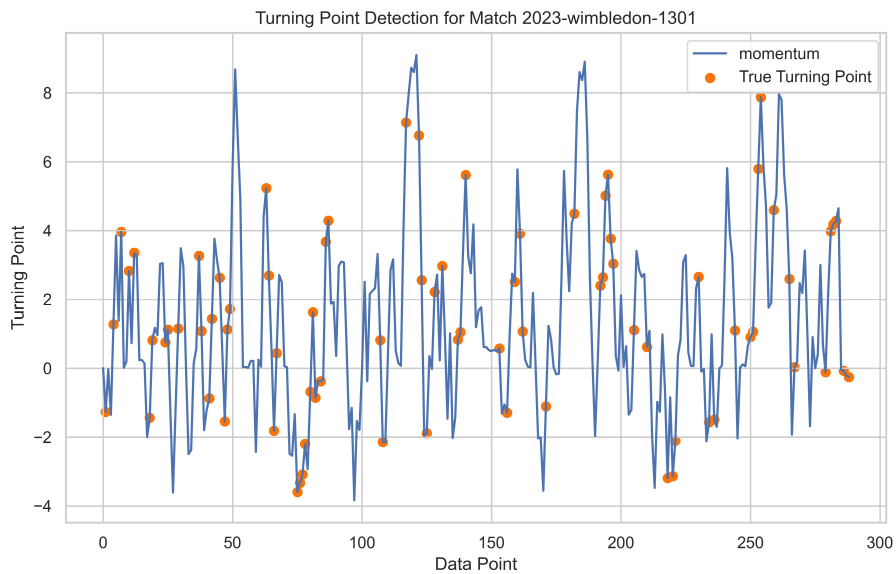

Figure 5: turning point detection 2023-wimbledon-1301

6.2 Prediction of transition points by random forest model

We use the random forest model for prediction, which is based on the synthesis of multiple decision trees. It is robust to noise and outliers, and can better deal with complex data in real tennis matches. Second, our data contains multiple features, and random forests can efficiently process high-dimensional data without the need for feature selection. Finally, the Random Forest provides a ranking of the importance of each feature, indicating which features are most influential in predicting turning points.

We create a random forest classification with 100 decision trees, divide the data set into a training set and a test set, train the model on the training set, make predictions on the test set, and calculate the model performance index.

Next, we visualized prediction results,marking the real turning point and the predicted turning point.

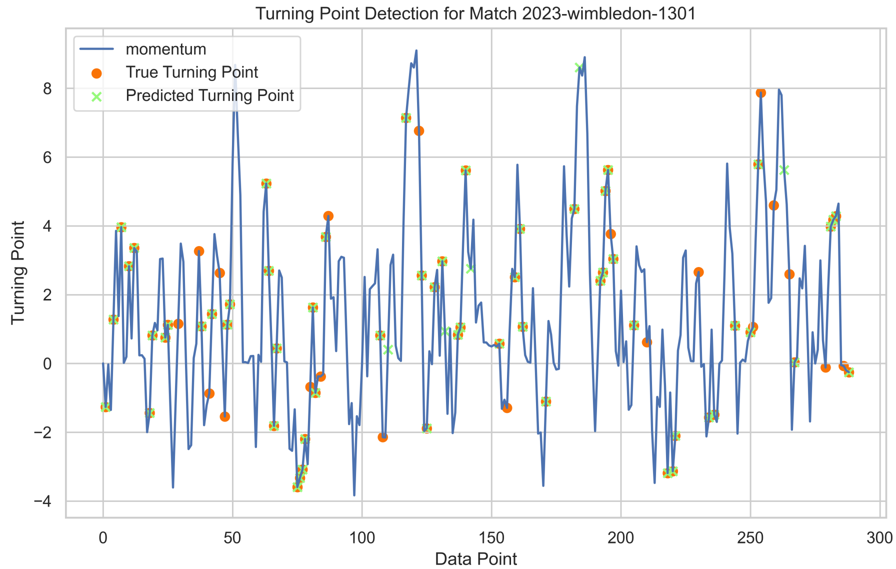

Figure 6: turning point detection 2023-wimbledon-1301 prediction

6.3 The importance of factors affecting the transition points

After the specific moment of situation transformation is obtained from the original data, the importance of each indicator can be ranked in the current time period, and the indicator with the greatest correlation (positive correlation or negative correlation) is the indicator with the greatest impact on the situation.

Output the importance of 10 variables in the random forest model. From the figure below, we can see that X8, the total chart run distance within the last three points, has the greatest influence on the transition point. The score of X1 ,the extent of lead score also influcence a lot .Other factors have few importance.

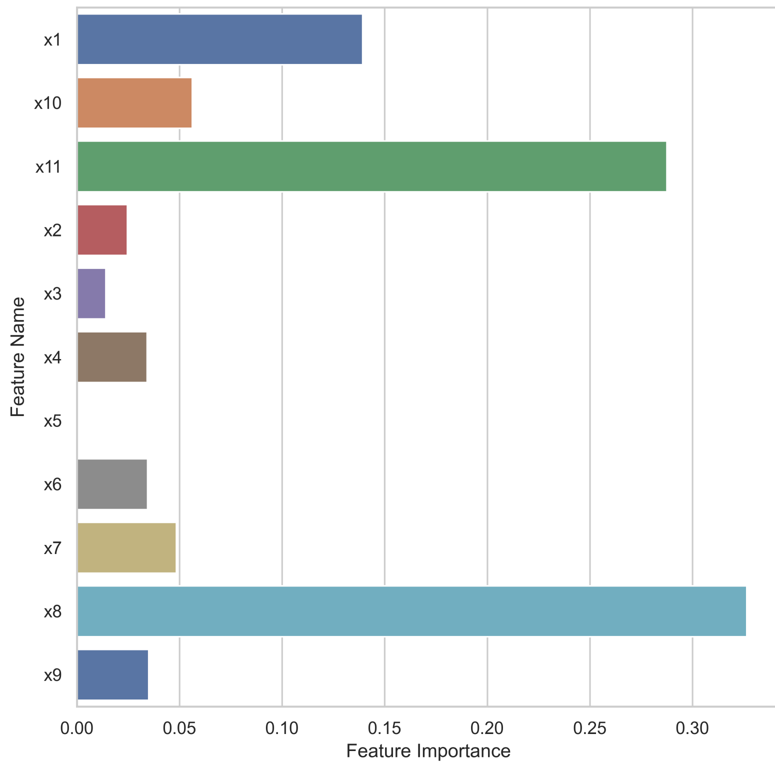

Figure 7: feature importance

6.4 Advise players on transition points

6.4.1 General recommendations

The above analysis shows that the running distance is highly correlated and greatly affects the turning point of the situation. In other words, physical exertion is very important in tennis matches.Players should mobilizing the opponent running distance as much as possible.They should reduce their own running distance,such as constantly switch the near and far ball, left and right ball, in order to retain their own physical strength, consume the opponent's physical strength.

6.4.2 Separate recommendations for specific players

Take Novak Djokovic as an example, we put the data of several matches into the model, obtained the importance of each factor, and found that X8,X11, both running distance and whether it is the service game are the key factors, indicating that the player has a high serving ability, and when facing this player's service game, we should be mentally prepared and alert.

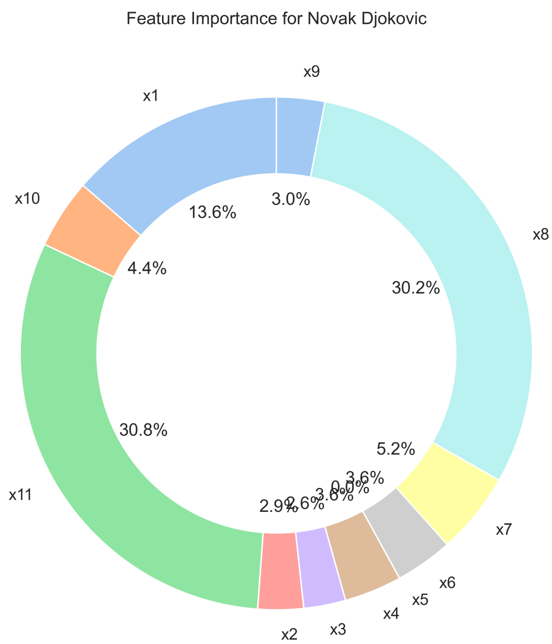

Figure 8: feature importance Novak Djokovic

7 Model generalization evaluation

7.1 Application of the model to other matches

We have completed the model of transition points in the competition, and the prediction result is good. Next, we continue to test the random forest model of Question 3 on other data sets to check the generalization ability of the model. We used the random forest model to predict the male/female single data set, and the prediction effect of the model is as follows It can be seen that the model performs well on other data sets and can be applied to tennis match prediction Below is the Model accuracy index:
Women's singles : 0.683
men's singles : 0.718

7.2 Application of the model to other sports

Our model is based on the analysis of tennis data, some of which, such as break rate and other indicators, may not exist in other sports. Therefore, we believe that when the model is applied to basketball and other fields, variables can be modified as follows:

In terms of basketball, we need to change the scoring system due to a single shot within basketball that is equivalent to a point in tennis can have the result of 1(ex: free throw),2(shot scored within 3-point arc),or 3 points(shot scored outside of 3-point arc). Also rules regarding fouls and penalties are also diverse from tennis and complex in its own way, making us including more rules within the math model of point scoring.

The next factor is that the number of possession rights each team has in the game also helps predict scores, which is not present in tennis because it is not present, since there is no such thing.

A transition encompasses everything that happens between two states. During a transition, 0, 1, 2, or 3 points are scored. This leads us to the set of states: A, B Œ i, s, o, d, f Œ 0, 1, 2, 3. For example, let team B miss a shot and team A rebound, then team A miss a 2pt shot, get an offensive rebound, and dunk for 2pt while being fouled for an extra free throw." The amount of rebound could be seen as a factor for possession in which we need to take into account for the model.

We propose these certain changes should be implemented. Variation in scores should be created based on shots according to the time-varying performance and statistics of players on each team while also looking at strategies of team play. [7]

8 Strengths and Weaknesses

8.1 Strengths

- The model training speed is fast, and simple data processing can quickly model and predict.
- The model has strong generalization, and in different games, the model shows a high accuracy rate. In different games, modified features can also be used to train the model, so as to predict the momentum of different games.
- This model can be correctly and stably deployed on small sample data sets, and can be easily promoted and deployed for downstream tasks.

8.2 Weaknesses

- Although the research on momentum in motion has lasted for a very long time, in fact, most of them are qualitative analysis, and there is a lack of actual quantitative modeling. Therefore, the weight of each parameter in the establishment of momentum model is still to be discussed.
- The model needs to manually select appropriate features first. There are many features in the data, many of which may have no impact on the momentum change, so repeated comparison and verification are required.

References

[1] Z. J. Jesse, "2016 australian open: Novak djokovic, the solitary serbian champion," *Tennis Master*, no. 2, pp. 24–29, 2016.

[2] Z. Lifei, Z. Yang, W. Xuan, and W. Yida, "Analysis of winning factors for chinese elite male singles tennis players under data-driven approach: A case study of zhang zhizhen," in *Abstracts of the 13th National Sports Science Conference - Poster Exchange (Sports Training Branch) (III)*, School of Physical Education, Southwest University; School of Physical Education and Health, Dalian University of Technology; 2023, pp. 306–308.

[3] M. Jiacai, C. Qingguo, L. Cairong, and Z. Zichen, "Model construction and research analysis of winning factors for professional tennis players," *Bulletin of Sports Science and Technology Literature*, vol. 31, no. 66-68+118, 2023, ISSN: 1005-0256. DOI: 10.19379/j.cnki.issn.1005-0256.2023.04.021.

[4] W. Qianru, Z. Qiuhan, C. Yonghuan, and G. Peizhen, "Analysis of tennis winning techniques and training insights based on the logistic regression model: A case study of men's singles at the 2021 us open," *Youth Sports*, 2022, ISSN: 2095-4581.

[5] J. Ting and L. Qing, "Factors influencing the outcome of men's singles matches in professional tennis tournaments: A study based on the four grand slam events from 2014 to 2018," *Journal of China Sport Science and Technology*, vol. 57, no. 62-68, 2021, ISSN: 1002-9826. DOI: 10.16470/j.csst.2019115.

[6] Y. Yuzhe, "Design and implementation of tennis match tactical analysis system based on data mining," M.S. thesis, Xi'an Shiyou University, 2023.

[7] P. Yujiao, "A study of technical winning factors in cba teams based on logistic analysis," M.S. thesis, Nanjing Normal University, 2017.

Momentum

Aids You in Tennis Matches

Momentum refers to the psychological advantage that one player gains over another by winning a few consecutive points or games. When a player wins a few points in a row, it can impact their opponent's confidence, leading to mistakes and missed opportunities. Gaining momentum in tennis is crucial because it gives the player a sense of control, and it puts pressure on the opponent to try and regain control.

Stay Strong

players should enhance mental resilience, especially when facing strong serving opponents, by effectively dealing with serves, maintaining composure, and staying focused to ensure optimal performance during key moments.

Save Energy

To conserve energy, players should focus on tactics such as varying shot lengths and directions, aiming to move their opponent more while minimizing their own running distance, simultaneously exhausting their opponent's stamina, thereby increasing the chances of securing victory.

Appendices

Appendix A the code of Momentum Model and Performance Model

data processing **Python source code:**

```
from sklearn.preprocessing import MinMaxScaler
import openpyxl
import pandas as pd
from lightgbm import LGBMClassifier
import matplotlib.pyplot as plt
from sklearn.metrics import accuracy_score
from sklearn.metrics import f1_score
from sklearn.metrics import recall_score
from sklearn.metrics import precision_score
from xgboost import XGBClassifier
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from sklearn.linear_model import LogisticRegression
import re
import json
import numpy as np
import os

from utils.data_set_turning_point_v2 import calculate_turning_points
from model_setup_random_forest import match_turning_point_pred
from model_setup_random_forest import feature_importance_calculation
from utils.win_predict_v2 import win_predict_LR_model
from model_setup_LR_v1 import match_performance_calc


class DataCleanAndGenerate:
    def __init__(self, df_path):
        self.df_path = df_path
        self.df = self.data_clean(df_path)

    # Samples with abnormal scores were eliminated
    # samples with missing pace were deleted.
    def data_clean(self, df_path):

        # Select different reading methods based on the df_path suffix name
        if df_path.endswith('.xlsx'):
            df = pd.read_excel(df_path)
        elif df_path.endswith('.csv'):
            df = pd.read_csv(df_path)

        df.loc[(df.p1_score == 'AD'), 'p1_score'] = 50
        df.loc[(df.p2_score == 'AD'), 'p2_score'] = 50
```

```
        df['p1_score'] = df['p1_score'].astype(int)
        df['p2_score'] = df['p2_score'].astype(int)
        df.dropna(subset=['speed_mph'], inplace=True)

        # Save the cleaned data to a new folder

        data_dir = self.get_new_folder_name()
        new_file_name = self.get_origin_file_name().split('.')[0] + '_edit.csv'
        new_file_name = os.path.join(data_dir, new_file_name)

        df.to_csv(new_file_name, index=False)
        print('file path after clean:', new_file_name)

        return df

    def calculate_comprehensive_momentum(self, player_number, \
                                         window_size=4) -> list:

        data = self.df
        momentum_scores = [0] * len(data)
        consecutive_point_wins = 0  # Track streaks
        consecutive_game_wins = 0  # Track winning streaks
        previous_game_winner = None  # Track the winner of the previous round
        # Base Momentum Score Addition for Break of Serve
        initial_break_point_value = 1

        for i in range(1, len(data)):
            recent_data = data[max(0, i - window_size):i]
            momentum_score = 0

            for _, feature in recent_data.iterrows():
                # Basic Momentum Score Calculation
                P_t = 1 if feature['point_victor'] == player_number else -1
                S_t = 1.2 if feature['server'] == player_number else 1.0
                base_momentum = P_t * S_t
                momentum_score += base_momentum
                 # Reset break point value
                break_point_value = initial_break_point_value

                # Continuous score correction (linear)
                if P_t == 1:
                    consecutive_point_wins += 1
                else:
                    consecutive_point_wins = 0  # Reset when losing points
                momentum_score += 0.03 * consecutive_point_wins

                if feature['game_victor']:
                    current_game_winner = feature['game_victor']
                    if current_game_winner == player_number:
                        if current_game_winner == previous_game_winner:
                            consecutive_game_wins += 1
                        else:
                            consecutive_game_wins = 0
                    previous_game_winner = current_game_winner
```

```
                momentum_score += 0.2 * consecutive_game_wins

            if feature['set_victor']:
                player1_set = feature['p1_sets'] + \
                    1 if feature['set_victor'] == player_number \
                        else feature['p1_sets']
                player2_set = feature['p2_sets'] + \
                    1 if feature['set_victor'] == player_number \
                        else feature['p2_sets']
                diff = (player2_set - player1_set) * \
                    (-1 ** player_number)
                momentum_score += 0.1 * (2 ** diff)

            if feature['game_victor']:
                score_diff = \
                    abs(feature['p1_games'] - feature['p2_games'])
                momentum_score += 0.02 * score_diff * P_t

            if feature['p1_break_pt_missed'] == 1 or \
                feature['p2_break_pt_missed'] == 1:
                break_point_value -= 0.1

            if feature['p1_break_pt_won'] == 1 \
                or feature['p2_break_pt_won'] == 1:
                break_point_value = max(break_point_value, 0.1)
                momentum_score += break_point_value * P_t

            rally_factor = feature['rally_count'] / 30
            distance_factor = (
                feature['p1_distance_run'] + \
                    feature['p2_distance_run']) / 122
            momentum_score += 2.0 * rally_factor * distance_factor * P_t

            if feature['rally_count'] > 10:
                momentum_score += 0.5

            if (player_number == 1 and feature['p1_ace'] > 0) \
                or (player_number == 2 and feature['p2_ace'] > 0):
                momentum_score += 0.02

            if (player_number == 1 and feature['p1_unf_err'] > 0) \
                or (player_number == 2 and feature['p2_unf_err'] > 0):
                momentum_score -= 0.05

        momentum_scores[i] = momentum_score

    return momentum_scores

def generate_data_para_x(self, if_match_id=False) -> pd.DataFrame:

    x1_ls, x2_ls, x3_ls, x4_ls, x5_ls, x6_ls, x7_ls, x8_ls, x9_ls, x10_ls = [
    ], [], [], [], [], [], [], [], [], []

    label_ls = []
```

```
match_ls = []

for match_id, set_no, game_no, point_no in zip(self.df.match_id, \
            self.df.set_no, self.df.game_no, self.df.point_no):
    match = self.df[self.df.match_id == match_id]
    set_ = match[match.set_no == set_no]
    game_ = set_[set_.game_no == game_no]
    point_ = game_[game_.point_no == point_no]

    # The score of this game leads the progress
    x1 = point_['p1_score'].values[0] - point_['p2_score'].values[0]
    # Whether the previous point scored
    x2 = 0 if x1 < 0 else 1
    # Whether ACE
    x3 = 1 if 1 in game_['p1_ace'].values else 0
    # whether to score
    x4 = 1 if 1 in game_['p1_winner'].values else 0
    # Is there a double fault in this game?
    x5 = 1 if 1 in game_['p1_double_fault'].values else 0
    # Are there any unforced errors in this game?
    x6 = 1 if 1 in game_['p1_unf_err'].values else 0
    # Ratio of Internet times and Internet score
    x7 = game_['p1_net_pt_won'].sum(
    )/game_['p1_net_pt'].sum() if game_['p1_net_pt'].sum() != 0 else 0
    # The total mileage in the last three points
    x8 = point_['p1_distance_run'].values[0]
    # Return score rate
    x9 = 1 if 1 in game_['server'].values else 0
    # Is it a service game?
    x10 = 1 if point_['serve_no'].values[0] == 1 else 0

    label = 1 if point_['point_victor'].values[0] == 1 else 0
    label_ls.append(label)

    x1_ls.append(x1)
    x2_ls.append(x2)
    x3_ls.append(x3)
    x4_ls.append(x4)
    x5_ls.append(x5)
    x6_ls.append(x6)
    x7_ls.append(x7)
    x8_ls.append(x8)
    x9_ls.append(x9)
    x10_ls.append(x10)

    data_dir = self.get_new_folder_name()

    if if_match_id:

        match_id = point_['match_id'].values[0]
        match_ls.append(match_id)

        new_file_name = 'Standard_Training_Data_' + \
            self.get_origin_file_name().split('.')[0] + '_match_id.csv'
```

```
        new_file_name = os.path.join(data_dir, new_file_name)

        dataset = pd.DataFrame({'x1': x1_ls, 'x2': x2_ls, 'x3': x3_ls, \
                                'x4': x4_ls, 'x5': x5_ls, 'x6': x6_ls, \
                                'x7': x7_ls, 'x8': x8_ls, 'x9': x9_ls, \
                                'x10': x10_ls, 'label': label_ls, \
                                'match_id': match_ls})
    else:
        new_file_name = 'Standard_Training_Data_' + \
            self.get_origin_file_name().split('.')[0] + '.csv'
        new_file_name = os.path.join(data_dir, new_file_name)

        dataset = pd.DataFrame({'x1': x1_ls, 'x2': x2_ls, 'x3': x3_ls, \
                                'x4': x4_ls, 'x5': x5_ls, 'x6': x6_ls, \
                                'x7': x7_ls, 'x8': x8_ls, 'x9': x9_ls, \
                                'x10': x10_ls, 'label': label_ls})

# Calculate overall momentum score
comprehensive_momentum_1 = self.calculate_comprehensive_momentum(1)
dataset.insert(10, 'x11', comprehensive_momentum_1)

dataset.to_csv(new_file_name, index=False)

return dataset
```