Problem Chosen

ABCDEF

2024
MCM/ICM
Summary Sheet

Team Control Number

2400609

Shortest Path Algorithms: Taxonomy and Advance in Research

Every athlete understand the importance of practicing skills and strategies to win a match. But in most sports, they are not the only determining factors of the outcome of a match. A young and fearless athlete may surprisingly defeat a veteran, or even the most acknowledged and well-known athlete may lose a match due to a momentary lapse in concentration and the tide it raises. Sometimes the outcome seems destined to be altered by a force, even multiple times. We call it momentum.[1] Such is the charm of sports, and such is the charm of tennis. Out of curiosity, we are interested in studying the momentum, a mysterious force in tennis with modern methods.

As for problem 1, we select factors that reflect player's performance and use Analytic Hierarchy Process (AHP) method to calculate the weights. We find that the three most influential factor is score difference, whether scored in the last point and running distance. The sum of the factors reflecting the performance is referred as momentum.

In problem 2, we perform autocorrelation test on momentum, and we find that it has strong first-order autocorrelation (0.5149). Then we quantify the future scores and then determine the correlation between momentum and future scores, they are highly correlated. Momentum has a maximum correlation coefficient of 0.7934 with next score.

In problem 3, to predict the swings in the match, we established model based on Gated Recurrent Unit (GRU) network. We first define the swings and classify them into four patterns, then we processed data that's suitable for training. We trained different model out of different purposes, most importantly data with and without recent scores. We found that the accuracy on test data is around 0.4274. Better than random guess of the four patterns.

To identify the most related factors, we use a novel algorithm called Permutation Feature Importance algorithm. It can determine the importance of features of input in neural network. We find that the most important feature while with recent score is still the score difference, and without, whether win within 3 rally count, from which we inferred that recent scores has significant impact on momentum, improving our understanding of problem 2.

As for the advice for athletes, we trained model on their matches to identify their feature importance, so we can provide them with customized advice.

In problem 4, we chose model trained on half of the data and tested on the other half, and the accuracy is around 0.4115. As for women's tennis and table tennis, sadly we failed to find abundant data to train the model. But the path forward after finding the data is clear, we reassess AHP and train the GRU network.

The letter including advice for coaches and athletes is attached in the appendix.

Key words: Momentum in Tennis, AHP, correlation analysis, GRU, Permutation Feature Importance

Contents

1 Introduction

1.1 Overview

Tennis more than any other sport, is a game of momentum. The absence of a clock to do the dirty work of finishing off an opponent, and a scoring system based on units used, makes the flow of the match much more important than any lead that has been established.–Chuck Kriese
In the realm of tennis, the concept of "momentum" stands as a pivotal factor, exerting a direct influence on players' performance and match outcomes. But momentum is a subjective feeling, it is hard to measure and quantify. So make a precise definition of momentum and find its relationship with other events become an interesting topic.

1.2 Restatement of the Problem

First Problem Momentum is "strength or force gained by motion or by a series of events." It directly shows the player's current performance. To assess the players' performance, it is crucial to have a clear understanding of "momentum." We will focus on the following tasks:

- determine the influencing factors of "momentum"

- Quantify the variations in "momentum"

- Visualize the "momentum" function

Second Problem "momentum's role in the match" means the level of momentum affects the future scores of the match. The coach may subscribe to the idea that each point is an independent event and governed by probability. In this view, consecutive success and momentum changes (swings) are seen as more random than influenced by previous events. To judge this autocorrelation and to use our model, we

- perform autocorrelation test on momentum

- perform correlation test between current momentum and future scores.

Third Problem The goal of the third problem is to predict the swings of the play. Considering the extra tasks, we will split the problem into five parts:

- define swing of the play

- decide what data the prediction is based on

- develop a model to predict the swings of the play

- decide what among the data are the most decisive to the prediction

- give advice to player based on the weights of the data

1

Fourth Problem   Testing the model on other matches is a process of generalization ability test. We will split the problem into five parts:

- predict the swings on test data

- compare the prediction with the momentum function in Problem 1

- figure out reasons for the poor prediction

- generalize all our models to Women's matches or other sports

Fifth Problem   The memo of advice for coaches on momentum, and players on preparation for potential momentum swings will be placed in the Appendix.

1.3   Assumptions

To simplify the problem, we made the following assumptions:

- **Assumption 1:** The `px_unf_err` column of the data only counts those unforced errors that occurred when the player was hitting in baseline.
  Justification: Usually when a player is at net, the point will end in a few strikes, and there's little probability that the player will hit an unforced error within that few strikes. What's more, the `px_net_point` and `px_net_point_won` columns of the data can predominantly reflect the player's ability at net, therefore reducing the impact of counting the unforced errors while at net.

- **Assumption 2:** The "current performance" we usually refer to on a certain aspect of a player can be reflected by the player's 3 latest shots of that aspect.
  E.g. The current performance can be reflected by a combination of, the proportion of aces in the 3 latest **serves** of the player, the proportion of winners in the 3 latest **shots** of the player, the return depth of the 3 latest **returns** of the player, etc.
  Justification: The current performance of a player consists of the average performance and the status of the player at the moment, which can be comprehensively reflected in the player's performance on recent shots. For convenience, we specified that the 3 latest shots can reflect the player's current performance.

- **Assumption 3:** We only consider factors mentioned in data, other factors such as the court and the audience are neglected. These factors are hard to quantify. Also, these objective factors can't be changed, they are not effective in giving advice to atheletes.

- **Assumption 4:** The given data is correct after our process. **Justification:** The data about tennis is very likely to be collected by human, and the data is not always accurate. We have to assume the data is correct to make our model work.

- **Assumption 5:** We simply believe that qualitative factors such as return depth do not have variability in one category, for instance, we in the process of data, return depth is calculated in 3 types, D, ND and NA, regardless of the real number. **Justification:** Though this may be not reasonable when the quantitative values are close but qualitative values are different, but it indeed corresponds to the given data.

2

2 Momentum Evaluation Model

2.1 Model Introduction

To determine which player is performing better at a specific time, we create a indicator "Momentum" using the Analytic Hierarchy Process (AHP) to give a quantitative and overall evaluation.

To investigate the reasons behind "momentum", we first need to provide a preliminary definition for "momentum". The magnitude of "momentum" is defined as

$$f_{ijk} = \boldsymbol{\omega} \cdot \boldsymbol{x_{ijk}}$$

where:

1. f_{ijk} represents the "momentum" of player k before the jth point number in the ith match (in the order given by the table).

2. $\boldsymbol{x_{ijk}}$ is an n-dimensional column vector representing some influencing factors at the corresponding moment. Specific details will be provided later.

3. ω is an n-dimensional row vector indicating the specific weights of the influencing factors, which will be obtained through the Analytic Hierarchy Process (AHP).

4. In this formula, there are two different calculation methods, one representing rounds where the player serves and the other representing rounds where the opponent serves. We can express it as

$$\boldsymbol{\omega} = \boldsymbol{\omega_0} \circ \boldsymbol{\delta} = (\omega_0^{(0)} \delta^{(0)}, \omega_0^{(1)} \delta^{(1)}, \ldots, \omega_0^{(n)} \delta^{(n)})$$

representing a vector formed by element-wise multiplication of two vectors of the same dimension. Here, $\boldsymbol{\delta}$ is a 0, 1 vector indicating whether it is the player's serving round. In the specific calculation, we will consider two cases separately.

Out of our own interpretation of AHP, we will break down the problems into three parts:

1. Factor Normalization and Data Cleaning

2. Collinearity Detection

3. Analytic Hierarchy Process (AHP)

2.1.1 Notations

Symbols	Description
$player$	the current player we are considering (e.g. while calculating momentum)
$point_i$	the ith point of the match, a vector consists of fields stated in the given dictionary
cur	the current index of the point, i.e. the match is currently at the cur^{th} point
H_i	denotes the set $\{point_{cur}, point_{cur-1}, \ldots, point_{cur-i+1}\}$
S_i	the set of latest i points where $player$ serves
R_i	the set of latest i points where $player$ returns
P_{ace}	current probability of hitting an ace by $player$
P_{df}	current probability of double-faulting by $player$
P_{1st}	current first serve goal rate by $player$
P_{fw}	current probability of $player$ winning a served point within 3 rallies
rd	current return depth of $player$
P_{win}	current probability of hitting a winner by $player$
P_{net}	current net win rate of $player$
$dist$	$player$'s running distance on the point
P_{unf}	current probability of hitting an unforced error by $player$
$scored$	whether $player$ scored the current point
$diff$	the score diffrence for $player$ in the current game (by number of points)
M	the current momentum of $player$ after a point

To access a certain field in a point, we simply use the field name stated in the given dictionary as index, i.e. for a point $point$, we use $point_{ace}$ to denote the binary variable that shows whether $player$ hits an ace ball in the point.

2.1.2 Factor Normalization and Data Cleaning

For the specific definition of x_{ij}^n, we believe that, in addition to whether the player is serving, many other factors can have an impact, including the player's skills, fatigue level, and real-time mental state of the games. Based on these three main aspects, we have organized 12 factors as preliminary influencing factors, as follows:

$$P_{ace} = \frac{\sum_{p \in S_3} p_{ace}}{3} \tag{1}$$

$$P_{df} = -\frac{\sum_{p \in S_3} p_{double\_fault}}{3} \tag{2}$$

$$P_{1st} = \frac{\sum_{p \in S_3} [p_{serve\_no} = 1]}{3} \tag{3}$$

$$P_{fw} = \frac{\sum_{p \in S_3} [p_{rally\_count} \le 3][p_{point\_victor} = player]}{3} \tag{4}$$

$$rd = \frac{\sum_{p \in R_3} \begin{cases} 0, & p_{return\_depth} = ND \\ 1, & p_{return\_depth} = D \\ -1, & p_{return\_depth} = NA \end{cases}}{3} \quad (5)$$

$$P_{win} = \frac{\sum_{p \in H_3} p_{winner}}{3} \quad (6)$$

$$P_{net} = \frac{\sum_{p \in H_3} p_{net\_pt\_won}}{\sum_{p \in H_3} p_{net\_pt}} \quad (7)$$

$$dist = \begin{cases} 0, & point_{cur,distance\_run} < 5 \\ -1, & point_{cur,distance\_run} > 45 \\ \dfrac{5 - point_{cur,distance\_run}}{40}, & otherwise \end{cases} \quad (8)$$

$$P_{unf} = -\frac{\sum_{p \in H_3} p_{unf\_err}}{3} \quad (9)$$

$$scored = [point_{cur,point\_victor} = player] \quad (10)$$

$$diff = \frac{\sum_{p \in point}[p_{set\_no} = point_{cur,set\_no}][p_{game\_no} = point_{cur,game\_no}](2[p_{point\_victor} = player] - 1)}{\min\{3, \sum_{p \in point}[p_{set\_no} = point_{cur,set\_no}][p_{game\_no} = point_{cur,game\_no}]\}} \quad (11)$$

In order to normalize the data processed, we convert the original data to limit them in $[-1, 1]$. For those factors that negatively influence the momentum, such as P_{df}, we made sure it's in $[-1, 0]$. For those factors that positively influence the momentum, such as P_{win}, we made sure it's in $[0, 1]$. For those factors that influence the momentum in both ways, such as $diff$, we made sure it's in $[-1, 1]$.

2.1.3 Collinearity Detection

After processing the data, considering the potential collinearity among factors within the same category, such as serving aces, first-serve scoring rate, and whether the previous point was scored may be correlated, as well as running distance and the number of strokes possibly being related, we conducted collinearity detection using Stata. The results of the detection indicate a significant variance inflation factor between running distance and the number of strokes. Therefore, we decided to exclude one of them, choosing to retain the remaining 11 variables for the Analytic Hierarchy Process (AHP).

2.1.4 Analytic Hierarchy Process

We have previously decomposed the included factors from top to bottom into several levels, where factors within the same level are subordinate to factors in the level above or influence factors in the level above. They also dominate factors in the next level or are influenced by factors in the next level. Starting from the second level of the hierarchy, we construct comparison matrices for each factor influencing the factor in the level above, until reaching the bottom level. Each element in the matrix indicates the preference level between factor i and factor j at the same level. It is essential to note that we have

separately established a series of such comparison matrices for two different serving types (serving by oneself and serving by the opponent). Here, we illustrate the matrix using serving by oneself as an example:

influe	ability	degre	manta
ability	1	1	1/3
degre	1/1	1	1
manta	3	1/1	1

ability	serve_	winne	net_wi
serve_	1	5	7
winne	1/5	1	3
net_wi	1/7	1/3	1

Figure 1: Comparison matrix for influencing factors and ability

degre	distan	unforc
distan	1	3
unforc	1/3	1

manta	scored	score_
scored	1	1/5
score_	5	1

Figure 2: Comparison matrix for degree of fatigue and mantality

serve_	ace	doubl	first_s	fast_w
ace	1	1	1/3	1/3
doubl	1/1	1	1/3	1/3
first_s	3	3	1	1/3
fast_w	3	3	3	1

Figure 3: Comparison matrix for serving

We obtain the weights for each component by calculating the maximum eigenvalue and normalizing its corresponding eigenvector. Certainly, for each matrix, we first need to test consistency using the Consistency Ratio (CR), where $CR = \frac{CI}{RI}$, $CI = \frac{\lambda_{max}-n}{n-1}$, $RI = 0.0, 0.58, 0.9$ (for matrices of size 2, 3, 4). The computed Consistency Ratios for the matrices are 0.076, 0.037, 0.0, 0.0, 0.046. Since they are all less than 0.1, it confirms the consistency of the matrices.

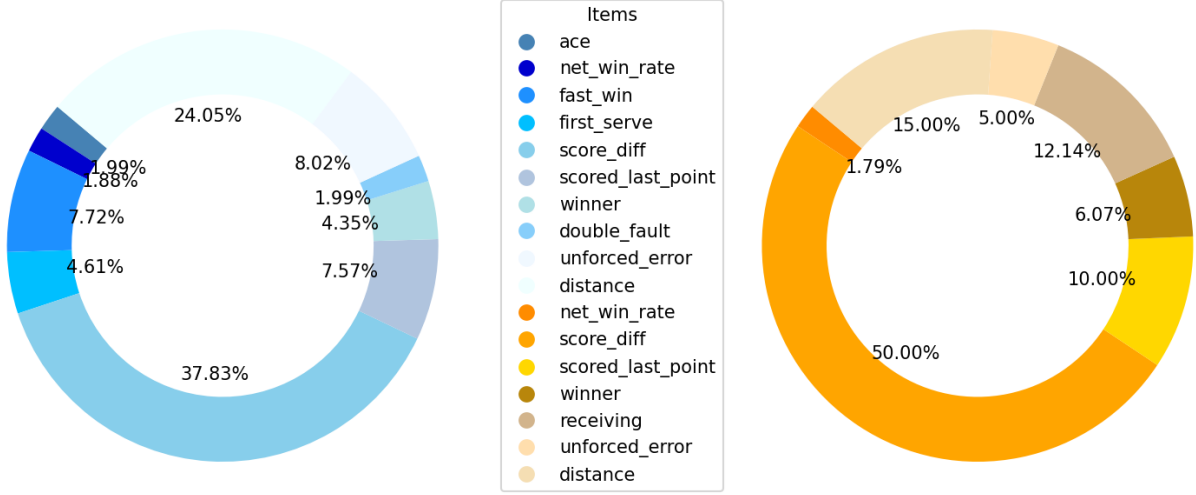Therefore, the weights for our model are as follows:

Figure 4: Weights in two different situations

Analyzing the various factors in the chart, it is evident that the most impactful factor is whether the previous point was scored. Following closely is the distance covered during the play, which aligns well with common intuition.

Thus, our final momentum is defined as:(need to change symbol)

$$
momentum = \begin{cases} \sum_{n=1,n\neq5}^{11} \omega_n x_n, & \text{if the player serves} \\ \sum_{n=5}^{11} \omega_n x_n, & \text{if the opponent serves} \end{cases}
$$

Where $\omega_n(n = 1, \ldots, 11)$ represent the weight of the factors, which is listed in Figure 4.

2.2 Visualization and Analysis

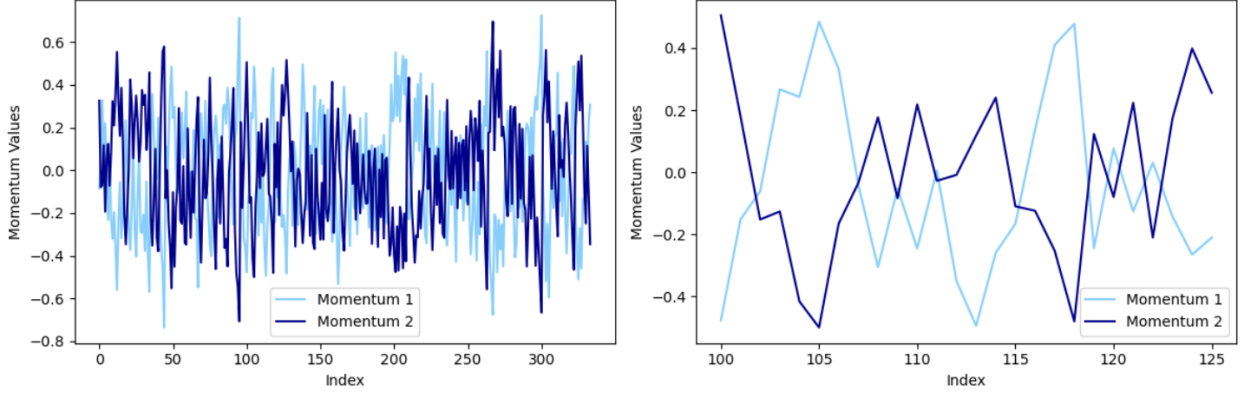Now, we illustrate the graph of the "momentum" in the first match:

Figure 5: Momentum change in the first competition(global and local)

It can be observed that the variation in "momentum" is a process of give and take. In photo 2, when momentum 1 is above momentum 2, it means that the player 1 is performing better than the opponent.

2.3   momentum autocorrelation and correlation with runs of success

To answer the coach's doubt, we need to perform autocorrelation test on momentum, and perform correlation test between current momentum and future scores in this section.

If the momentum has a high autocorrelation, it means that the momentum at this moment has a high impact on future performance. And if the correlation between momentum and future scores is high, it means that the player with higher momentum has a higher chance to win the next multiple round.
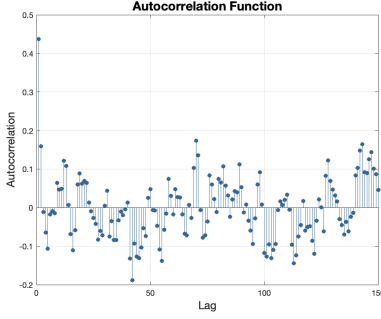
2.3.1   momentum autocorrelation

To check if sequence of momentum is self-related, we calculate the Pearson correlation between momentum and that with a time lag.

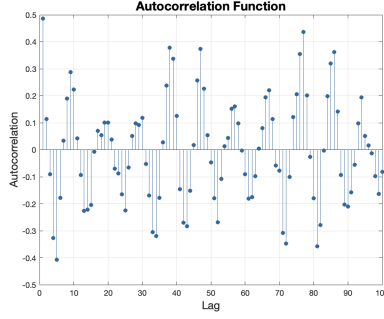**Algorithm 1** Calculate Autocorrelation Function

---

for $i = 1$ to 31 **do**
    $time\_series \leftarrow$ momentum(i^{th} match_index : $i + 1^{th}$ match_index $- 1$)
    $max\_lag \leftarrow \lfloor$length($time\_series$)/2$\rfloor$ ▷ Consider lags up to half of the length of the time series
    $autocorrelation \leftarrow$ zeros(1, max_lag)
    for $shift = 1$ to max_lag **do**
        $correlation \leftarrow$ corrcoef($time_series(1 : $ end $- shift), time_series(shift + 1 : $ end$))$
        $autocorrelation(shift) \leftarrow correlation(1, 2)$
    end for
                          ▷ Further processing or visualization can be performed here
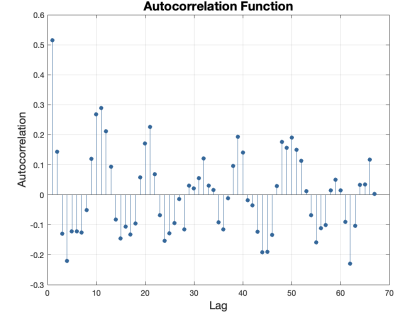**end for**

Here we display the autocorrelation of momentum of the first player in first three games. There are similar results for the second player and for the momentum difference.

8

(a) autocorrelation in match 1   (b) autocorrelation in match 2   (c) autocorrelation in match 3

Figure 6: Momentum autocorrelation

The corrcoef of lag 1 in match one is 0.4546, match two 0.5149, match three 0.5342. It can be seen that the autocorrelation of lag one is high, which means that the momentum at this moment has a strong relationship with the momentum in the next round. And the autocorrelation decreases to random as the lag increases.

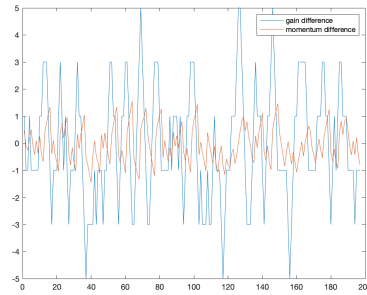### 2.3.2   correlation with runs of success

To give a quantitative evaluation of "future scores", we count points gain in future multiple rounds, and derive the difference by minus that of the opponent. For example, if the player gains 3 points in the next 5 rounds, and the opponent gains 2 points, the difference is 1. The difference indicates how much better the player is performing than the opponent.

In intuition, the player with higher momentum should have a higher chance to win the next round. And momentum at this moment should have less impact on the future rounds as time extends. The correlation between momentum and future scores verifies our intuition.
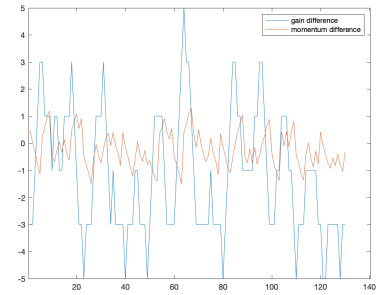
We calculate points gain difference in future one to five rounds at each time of all matches. Here we display five points gain difference and momentum difference in first three games.
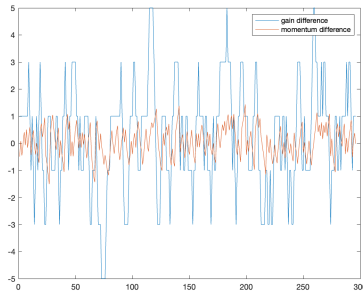
(a) 2023-wimbledon-1301   (b) 2023-wimbledon-1302   (c) 2023-wimbledon-1303

Figure 7: Gain Difference and Momentum in First Three Games

And we derive the correlation between gain difference from one to five rounds and momentum difference of all matches. Here we display the three of them.

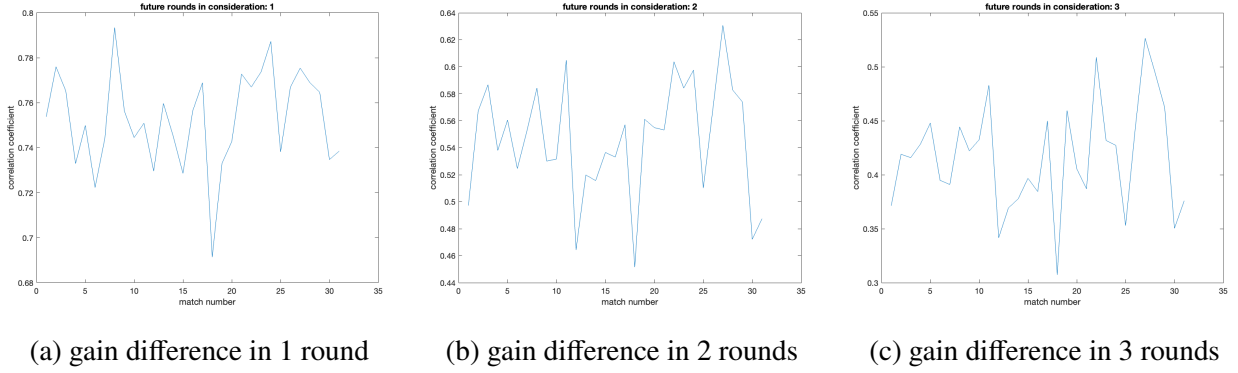(a) gain difference in 1 round   (b) gain difference in 2 rounds   (c) gain difference in 3 rounds

Figure 8: Correlation Between Gain Difference and Momentum Difference

Here we display the max and min correlation in different rounds. (hint. the max and min correlation means maximum and minimum of all matches.)

Rounds	1	2	3	4	5
Max	0.7934	0.6307	0.5264	0.4678	0.3910
Min	0.6914	0.4516	0.3074	0.1824	0.0627

Table 1: Maximum and Minimum Correlation of all matches in different rounds

As we can see from the table, the correlation between momentum and future scores is bigger than 0.5 considering the next 1 round, it implies that momentum has a substantial impact on the next round. And the correlation decreases as the rounds extend, which verifies our intuition, that the momentum at this moment has less impact on the future rounds as time extends.

Now, we have finished problem 1 and 2.

3 Swing Prediction Model

3.1 Model Introduction

To predict the swings, we need to find potential factors that may affect future momentum. In problem 1, we established a standard to judge the player's performance; in problem 2, we found the player's performance, "momentum", has a one order of autocorrelation, which indicates that some factors of the momentum have impact on the future momentum.

To find what and how factors affect the future momentum, we can take advantage of Machine Learning. In this section, we will primarily focus on establishing a supervised learning classification model using the Gated Recurrent Unit (GRU) network to predict momentum swings. We will break down the problems into three parts:

- definition of the swing of the play

- data (potential factors) and label used to train GRU

10

- Utilizing Gated Recurrent Unit

3.1.1 Definition of "Swings of the Play"

We first give the definition of the swings of the play. Based on our previous definition of "momentum," the significant changes of the game largely depend on the "momentum" of the two players. Therefore, we choose "momentum" to represent the swings of the play. The specific definition is as follows:

We use $\Delta f(t)$ to represent the difference in "momentum" between the two players., so it can be easily seen that if $\Delta f(t)$ and $\Delta f(t+1)$ have different signs, it indicates a "swing" in the game's momentum. In this way, we can define four states of "momentum" at time t:

$$
states = \begin{cases} state1, & \text{if } \Delta f(t) > 0 \text{ and } \Delta f(t+1) > 0, \text{ which means stay positive} \\ state2, & \text{if } \Delta f(t) < 0 \text{ and } \Delta f(t+1) > 0, \text{ which means rise from negative to positive} \\ state3, & \text{if } \Delta f(t) < 0 \text{ and } \Delta f(t+1) < 0, \text{ which means stay negative} \\ state4, & \text{if } \Delta f(t) > 0 \text{ and } \Delta f(t+1) < 0, \text{ which means decrease from positive to negative} \end{cases}
$$

Obviously, if state 2 or state 4 appears, we can determine that a swing has occurred.

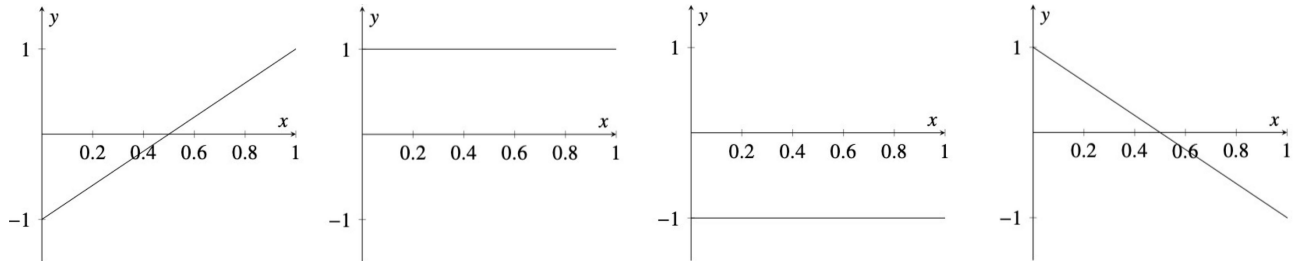To visualize the swing, we choose four shapes of lines to represent the four states of swings at time t:

Figure 9: visualization of swings, from state1 to state4

Here we display the momentum swings drived from AHP of first three matches.
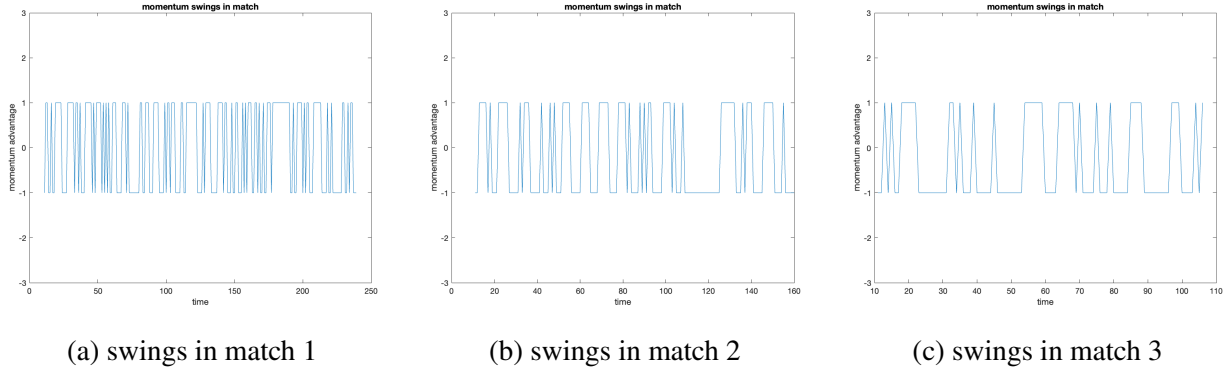
(a) swings in match 1   (b) swings in match 2   (c) swings in match 3

Figure 10: swings in first three matches

3.1.2 Data and Label used to train GRU

To verify all potential factors, meanwhile avoid excessive calculation due to large number of features, we keep track of the following features in each point:

- `ace`: represents whether the player hit an ace in the point. This feature is a straightforward rendering of the player's serving ability.

- `double_fault`: represents whether the player committed a double fault in the point. By taking this factor into account, we can find out whether the player is stable at serving currently.

- `first_serve`: represents whether the player hit the goal in his first serve in the point. This is also a factor reflecting the serving stability of the player.

- `fast_win`: represents whether the player won the point after at most 3 rallies. This reflects the player's ability to gain advantage after his serve/return.

- `return_depth`: represents the return depth of the player in the point. This factor is key to breaking the opponent's serve. (Who can break a serve if he can't properly return the opponent's serve?)

- `winner`: represents whether the player scored by hitting a winner in the point. This shows the player's power at baseline, both forehand and backhand winners are counted.

- `net_pt_won`: whether the player scored by coming to the net, reflecting the capability of the player while at net.

- `distance`: the running distance of the player in the point. This is an important indicator of the player's energy and stamina.

- `unf_err`: whether the player committed an unforced error in the point. Also a indicator of the player's stamina.

- `rally`: the rally count in the point. Represents the player's baseline hitting stability as well as his fitness.

12

- `scored_last_point`: whether the player scored the last point or not.

- `score_diff`: score difference in the current game. Along with last factor, these factors depict the current situation of the match.

- `break_point_diff`: represents the current breakpoint conversion rate of the player. This feature can reflect the player's performance on clutch points.

- `speed`: represents the player's serve speed in the point. Yet another feature to show the player's serving ability.

- `game_victor`: whether the player won a game in this point.

- `set_victor`: whether the player won a set in this point. Along with `game_victor`, these feature also represents the players performance on clutch points.

What's more, we want to find out can swings merely be predicted by the technical data, so that we can give advice to players on skills, therefore we delete `scored_last_point`, `score_diff`, `game_victor`, `set_victor` before we train a second model.

The label we use is the state of the swing derived from model of AHP.

we use different sets of data to train and derive different models.

- First, we will train the model with data from all matches available.

- Then, we will train a new model with data from a single player.

- Finally, we will train the model with part of available match statistics

From the first model, we can find the important factors that affect all players' momentum swings. From the second model, we can find the important factors that affect especially well on that player's momentum swings. From the third model, we can examine the generalization ability to accurately predict the swings in irrelevant players and matches.

3.1.3 Utilizing the Gated Recurrent Unit (GRU) Network

Gated Recurrent Units (GRUs) are a type of recurrent neural network (RNN) architecture that has gained popularity for sequential data processing. And it is appropriate for our many(10)-to-many(5) prediction model.

We will focus on a single unit of recurrent neural network to interpret the hidden mathematical principles:
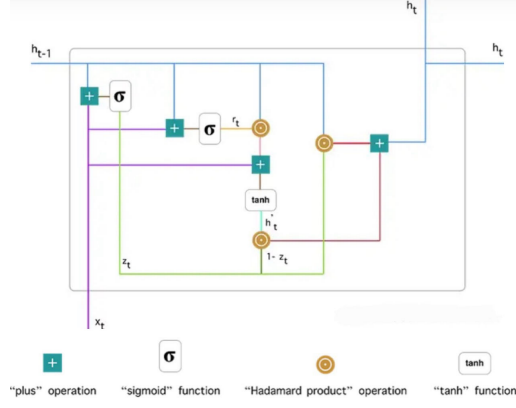
13

Figure 11: Structure of GRU

1. The update gate at time step t is computed using the following formula:
$$z_t = \sigma(W_z \cdot x_t + U_z \cdot h_{t-1})$$

Here, when x_t is input to the network unit, it is multiplied by its own weight W_z. Similarly, h_{t-1}, which holds the information from the previous $t-1$ units, is multiplied by its own weight U_z. These two results are then summed together and passed through a sigmoid activation function to compress the result between 0 and 1.

2. The essence of the reset gate is for the model to decide how much past information to forget. To compute it, we use:
$$r_t = \sigma(W_r \cdot x_t + U_r \cdot h_{t-1})$$

3. The computation of the new memory content using the reset gate is as follows:
$$h'_t = tanh(Wx_t + r_t \odot Uh_{t-1})$$

4. The computation for the new memory content h_t using the update gate is as follows:
$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h'_t$$

Moreover, we need the network to produce 5 outputs to predict the 5 points. To do this, we add 5 fully-connected layers to the output of the GRU, each of them producing one output we need.
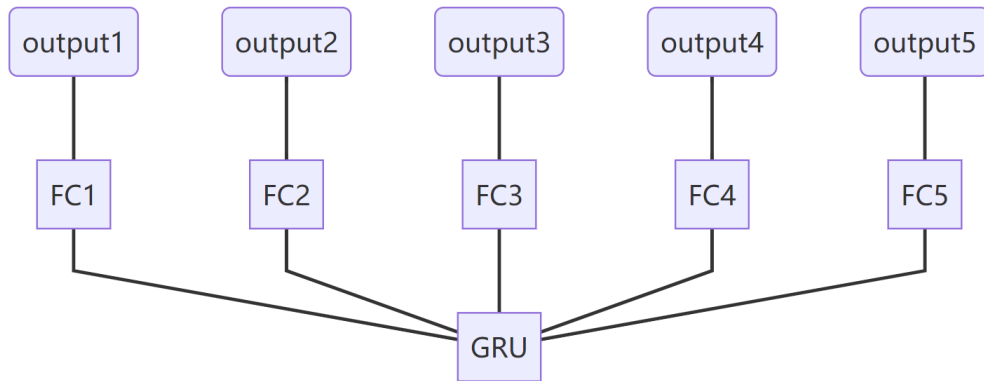

Figure 12: Structure of the whole network

The effectiveness of the cross entropy loss function in classification problems has been proven by pratice. In our model, we also choose this function as our loss function. To tackle the problem that our model has multiple tasks (namely, it has to predict the possible momentum swings in the 5 future points), we slightly modified the function to cater to our need. The definition of the function is as follows:

$$loss(\boldsymbol{p}, \boldsymbol{t}) := \sum_{i=1}^{5} H(p_i, t_i) \tag{12}$$

Where $\boldsymbol{p} = [p_1, p_2, p_3, p_4, p_5]$ is the predicted probability distribution of each of the 5 future points, and $\boldsymbol{t} = [t_1, t_2, t_3, t_4, t_5]$ is the states that the momentum at those points are in. H is the cross entropy function:

$$\begin{aligned} H(p, tar) &= -\ln(\frac{\exp p(tar)}{\sum_{j=1}^{4} \exp p(j)}) \\ &= -p(tar) + \ln(\sum_{j=1}^{4} \exp p(j)) \end{aligned} \tag{13}$$

By observing this function, we can see it gives proper bonus on predicting the right result and penalty on predicting the wrong ones.

Also, by adding up the losses of 5 points, which is equivalent to calculating the average loss on the 5 classification tasks, we can efficiently backward the losses to the whole network in a single backward step, which makes it easy for training and testing the model.

With this model, we can predict the momentum swings from the technical data of the past 10 points.

3.2 Visualization and Analysis

Out of simplicity, the model shown in this section is trained merely on technical data, and only the next swing of the next 5 swings is shown.

First, we display the model trained on all matches available to predict the swings of the play on test data. To avoid dazzling figure, we minus the swing from AHP with prediction, showing the difference.

Here's the difference of first three matches. It's worth mentioning that the training data is fisrt 0.8 of the match.
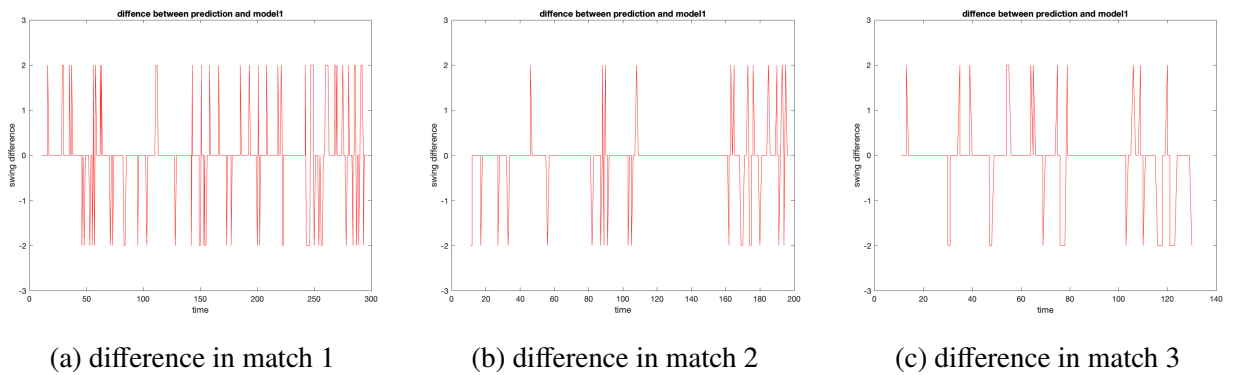

(a) difference in match 1   (b) difference in match 2   (c) difference in match 3

Figure 13: difference between prediction and AHP model

3.2.1 Prediction Error compared with AHP model

we calculate the prediction rate by the number of correct predictions (the flat part of the figure) divided by the total number of points. The fomula is as follows:

$$r_{accurate} = \frac{\sum\limits_{matches} P_i}{\sum\limits_{matches} L_i}$$

where L_i stands for the length of the match.

The predictions can be concluded as the table below:

Step	1	2	3	4	5
Train Set	0.7946	0.8297	0.8396	0.8404	0.8044
Test Set	0.3771	0.3423	0.3203	0.3078	0.2964

Table 2: Accuracy Scores for Each Step

To make sure training the model is useful, we give the prediction rate of untrained model.

Step	1	2	3	4	5
Train Set	0.1827	0.1821	0.1949	0.3174	0.1816
Test Set	0.1768	0.1768	0.1915	0.3199	0.1724

Table 3: Accuracy Scores of Untrained Model

We can see that the prediction rate of the untrained model is around 0.25, namely the probability of random guess. The prediction rate of the trained model is much higher.

Considering the situation of over-fitting, we decrease the epochs and result is as follows:

Step	1	2	3	4	5
Train Set	0.5379	0.4930	0.4693	0.4579	0.4471
Test Set	0.5073	0.4274	0.4057	0.4094	0.4035

Table 4: Accuracy Scores of Fit Model

Now we can see that the prediction rate of the trained model is bigger than random guess, we can conclude that momentum swings can be predicted by the data (technical data if only consider the statistics shown) of the past 10 points.

Although the prediction rate is bigger than random guess, 0.5 is not a big number. Now we analyze the reason.

Factors we may fail to consider

First of all, the AHP model is not considering all possible factors to momentum.

1. The potential shifts in player strategy, during the course of the game, they may become more familiar with the opponent's technical characteristics and make targeted changes to shift "momentum".

2. The deliberate concealment of abilities by players, this is similar to the previous item.

3. The incentive of the audience, this is more likely to have something to do with the difference of home and away, just like football. The influence of the audience on foreign and native players is totally different.

These factors are easily influenced by past events but are challenging to quantify. If these elements play a crucial role in what is commonly referred to as the momentum effect, then our model may not accurately reflect our intuition.

Furthermore, from the conclusion of problem 2, momentum only has one order of autocorrelation. So when we try to predict the far future momentum, there's no reason to be accurate.

3.3 Weight of Factors

If we can derive the derivative of output with respect to input, we can get the weight of factors. Although it's difficult to directly acquire derivative, we can use Permutation Feature Importance theory to avoid getting the derivative.

3.3.1 Permutation Feature Importance theory

Permutation Feature Importance theory is used to measure the importance of a feature by calculating the increase in the model's prediction error after permuting the feature. A feature is "important" if shuffling its values increases the model error, because in this case the model relied on the feature for the prediction. A feature is "unimportant" if shuffling its values leaves the model error unchanged, because in this case the model ignored the feature for the prediction.

Algorithm 2 Permutation Feature Importance

Input: Trained model $\hat{f}$, feature matrix X, target vector y, error measure $L(y, \hat{f})$.
Estimate the original model error $e_{orig} = L(y, \hat{f}(X))$ (e.g. mean squared error)
For each feature $j \in \{1, ..., p\}$ **do:**
    ○ Generate feature matrix X_{perm} by permuting feature j in the data X. This breaks the association between feature j and true outcome y.
    ○ Estimate error $e_{perm} = L(Y, \hat{f}(X_{perm}))$ based on the predictions of the permuted data.
    ○ Calculate permutation feature importance as quotient $FI_j = e_{perm}/e_{orig}$ or difference $FI_j = e_{perm} - e_{orig}$
Sort features by descending FI.

Using the model trained on all matches available, and the above argorithm, we sort FI and normalize them, then we plot the figure:
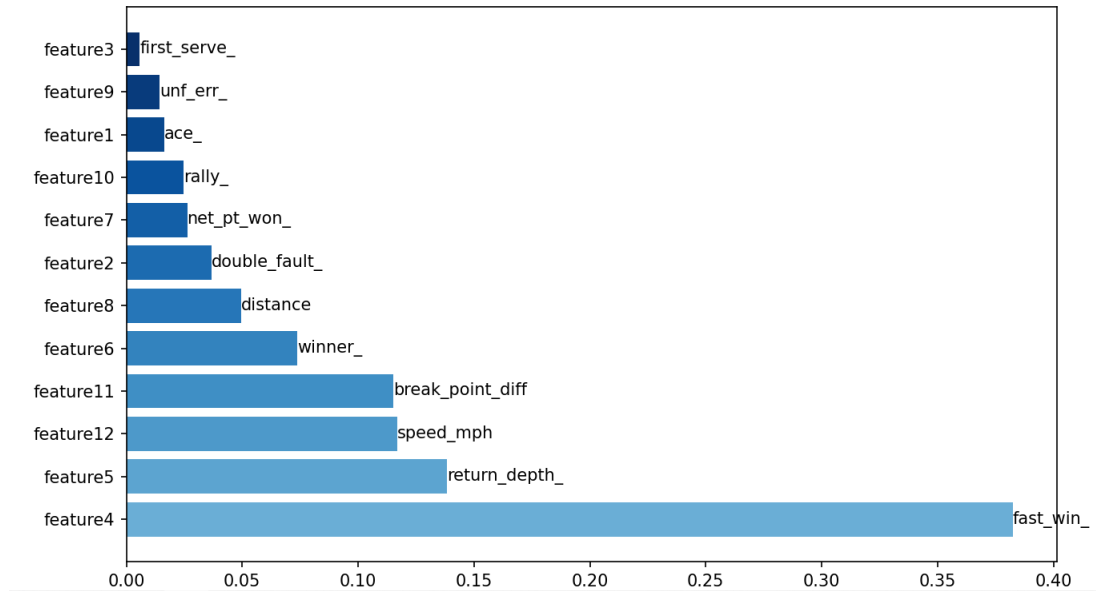

Figure 14: feature importance

We can see that the most important feature is fast win (rally count < 3 and wins the point), and the second most important feature is unforced error. Compared to problem 1 (main skill features are running distance, unforced error and fast win), they coincide. The running distance become insignificant, we think it is because this factor is also related to fast win.

The two factors are the most important factors that affect all players.

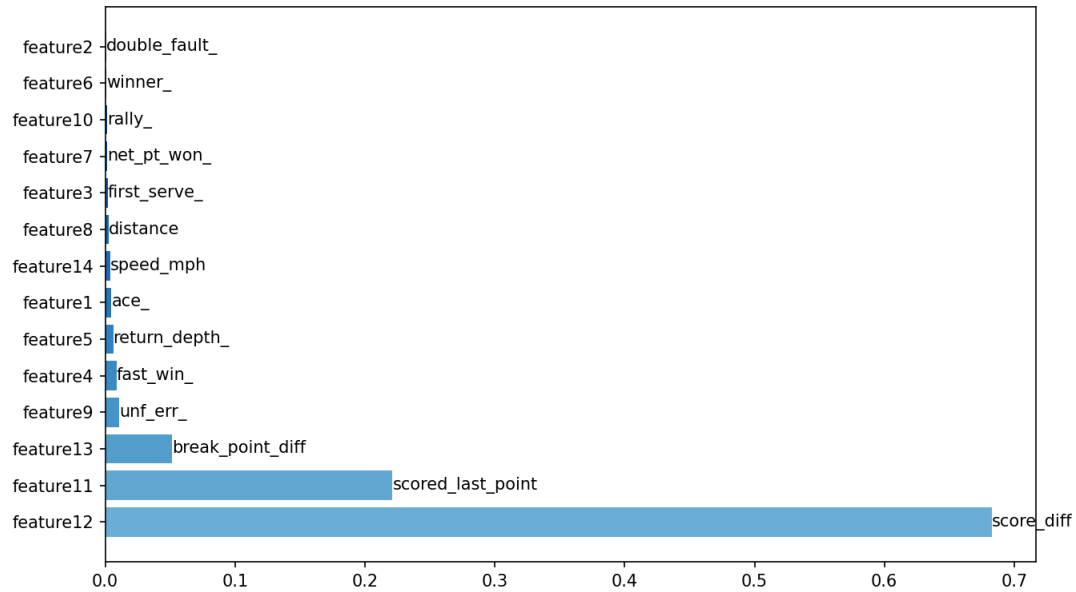Using the model with recent score data, we plot the figure:

Figure 15: feature importance

We can find out that the most important feature is the score difference, way more important than technical features. This is because the recent score condition is the most direct factor to the momentum. Now we have expanded our knowledge to problem 2, that among multiple factors that affect future momentum, the recent score condition is the most important.

3.4 Advice against Opponent Based on Weight of Factors

As said previously, model trained from match data of a single player, the factor with bigger weight is extremely effective to this player.

And to give specific advice against the opponent, we choose the model trained on technical data.

We choose three players:rublev(three matches), sinner(four matches), alcaraz(five matches), and we use the feature analysis method in problem 3 to find their features, the result is as follows:
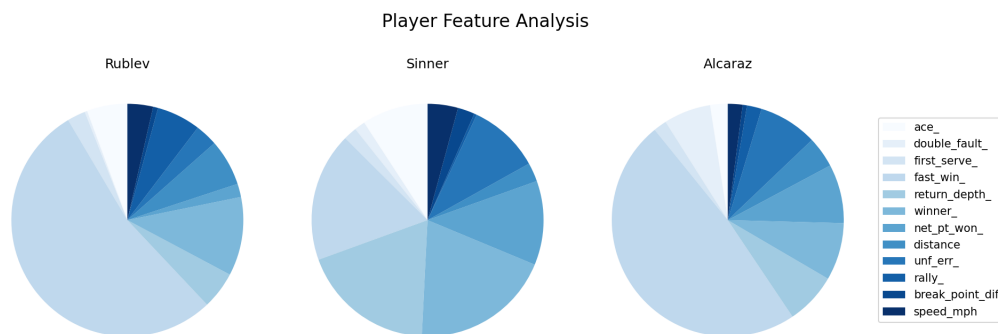

Figure 16: features of three players

19

1. In comparison to general features, both Rublev and Alcaraz phave significant feature fast win. Therefore, when competing against other players, we recommend them to emphasize the rhythm of seizing the early shots. For example, they can focus more on the quality of their serves, progressively achieving a fast win to enhance their momentum.

2. Additionally, Rublev is hardly affected by negative factors such as double fault and unforced error. Hence, in matches, there is no need for him to worry about occasional mistakes; instead, he can play boldly.

3. For Sinner, his technical characteristics are markedly different. Important features for him are winners and return depth. Consequently, we advise Sinner to fully exploit his strong scoring abilities of winner points and utilize his control over return depth to steer the course of the match and establish an advantage.

Now we have finished Problem 3.

4 Robustness Analysis

4.1 Generalization ability

In this section we will focus on the generlization of the model, that is to test model on irrelevant player or match. The model is trained on first 16 matches and test on the rest matches.

Prediction of Other Matches

A figure of swing difference on train match, and a figure of swing difference on test match are displayed.
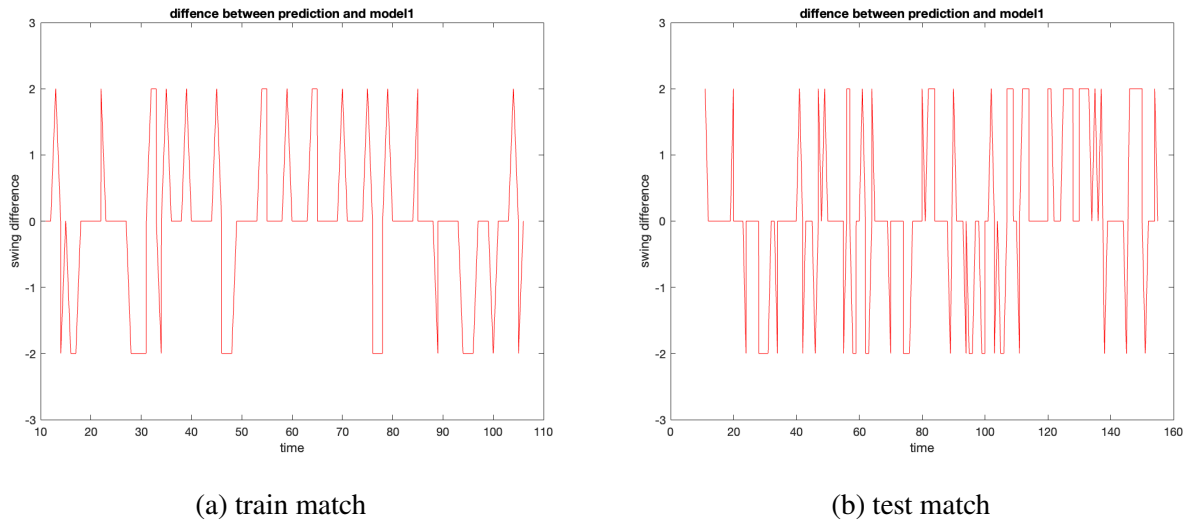

(a) train match (b) test match

Figure 17: difference on train and test match

Table of accuracy on train and test match is displayed.

Step	1	2	3	4	5
Train Set	0.4495	0.4426	0.4374	0.4370	0.4250
Test Set	0.4115	0.4016	0.3983	0.3994	0.3932

Table 5: Predict Rate for Each Step

Analysis on Women's Tennis and Table Tennis

As for women's tennis, it is a pity that we did not find enough data online, but from all men's match predictions, the accuracy rate is stable, so we have every reason to believe that our data can also apply to women's matches.

For table tennis, technical statistics also includes:

1. more serve types, such asforehand, backhand, short, long, or sidespin serves.

2. more receiving techniques,such as forehand push, backhand push, flick, or topspin.

So we need to modify the influencing factors in our model, but the general framework is similar.

Now, we have finished Problem 4.

4.2 Parameters Sensitivity Analysis

Now we focus on the sensitivity of the first problem. We have to do this bacause the judgement of the parameters of influencing factors is subjective to some extent, so our result of parameters is a little different from reality. The function of momentum is a uniformly continuous function of parameters(because variables are normalized), so little difference will not make the outcome of momentum imprecise.

5 Strength and Weaknesses

5.1 Strengths

1. **Visualization in conclusion**: Visualization is used to interpret the result, making it intuitive and concise.

2. **Proper models chosen**: Two models in problem 1 and problem 3 have similar results of the influencing factors, which makes it more reliable.

3. **Stable performance**: Our model has stable prediction in different men's matches.

4. **Classification of different situations**: We evaluate momentum in two different situations, serve and opponent serve.

5.2 Weaknesses

1. **Little subjective**: In problem 1, AHP is a subjective method to some extent, so although we get desired result similar to problem 3, some new completely objective method needs to be carried out.

2. **Limited ability in predicting other kinds of matches**: The technical indexes are different in other matches, so if we want to predict them, we need to do some modification on our model.

References

[1] Alice Axford, Bob Birkin, Charlie Copper, and Danny Dannford. Demostration of bibliography items. *Journal of TEXperts*, 36(7):114–120, Mar 2013.

A   Advice to Tennis Coaches and Players on Momentum

dear coach:

We hope this letter finds you well. We are , a team with a keen interest in the training and development of tennis athletes. We would like to share some discoveries of "momentum" during the match that We believe could be beneficial for athlete training and play spot, using mathematical modeling as an analytical tool. After data analysis and modeling. The following are some suggestions based on our analysis:

We hope these suggestions are helpful for your coaching endeavors. If you have any questions or would like to discuss this further and learn more detailed information, you can read the full text of our model.Also, we would be more than willing to discuss additional thoughts and insights with you.

B report on Use of AI

1.

OpenAI ChatGPT (Feb 2,2024 version,ChatGPT3.5)

Query1: *"swings in play and runs of success by one player are random" can this sentence to be interpreted as swings in play and runs of success by one player are independent?*

Output: *Yes, the sentence "swings in play and runs of success by one player are random" can be interpreted as suggesting that swings in play and runs of success by one player are independent of each other. In statistical terms, independence means that the occurrence or outcome of one event does not influence the occurrence or outcome of another.*

In the context of tennis, this would imply that the fluctuations in the dynamics of the match (swings in play) and a player's success streaks (runs of success) are not connected or dependent on each other. Each event is considered random and not influenced by the occurrence of the other.

2.

OpenAI ChatGPT (Feb 5,2024 version,ChatGPT3.5)

Query2: *what is general technical statistics in table tennis games*

Output: *In table tennis, general technical statistics refer to the analysis and measurement of various technical aspects of a player's performance during a match. These statistics provide valuable insights into a player's strengths, weaknesses, and overall playing style. Here are some general technical statistics commonly used in table tennis games:*

Serve Statistics:

Serve Type: Recording the types of serves a player employs, such as forehand, backhand, short, long, or sidespin serves. Serve Effectiveness: Analyzing the success rate of different types of serves, including direct points and opponent errors. Receiving Statistics: Receiving Techniques: Documenting how a player receives various types of serves, such as forehand push, backhand push, flick, or topspin. Receiving Success Rate: Calculating the player's success rate in returning different types of serves. Stroke Execution Statistics: Forehand and Backhand Shots: Analyzing the usage and effectiveness of forehand and backhand strokes in various situations. Types of Strokes: Recording the diversity of strokes played, including topspin, backspin, drives, and smashes. Rally Statistics: Rally Length: Measuring the average length of rallies in terms of the number of strokes exchanged. Rally Outcome: Tracking the outcomes of rallies, such as points won or lost, and identifying patterns in rally progression. Placement and Direction Statistics: Ball Placement: Analyzing where a player tends to place the ball on the opponent's side of the table. Direction of Shots: Recording the direction of shots, including cross-court, down the line, and shots to the opponent's body. Point Distribution Statistics: Point Sources: Identifying the sources of points, such as winners, opponent errors, or forced errors. Point Distribution: Analyzing how points are distributed throughout the match, helping to identify key moments. Error Analysis: Unforced Errors: Documenting the number of unforced errors made by a player. Forced Errors: Identifying instances where the opponent's play led to errors by the player. These technical statistics are crucial for players, coaches, and analysts to assess performance, strategize for future matches, and tailor training programs to address specific areas of improvement. They provide a comprehensive overview of a player's game and contribute to a more informed and strategic approach to table tennis.