MCM/C

February 5, 2024

Contents

1 Background Information

The 2023 Wimbledon Championships marked a significant event in the world of tennis, particularly in the Gentlemen's Singles category. One of the most notable matches was the final between the young Spanish sensation, 20-year-old Carlos Alcaraz, and the seasoned champion, 36-year-old Novak Djokovic. In a surprising turn of events, Alcaraz defeated Djokovic, ending the latter's impressive winning streak at Wimbledon since 2013. This match stood out not just for its outcome but also for its dramatic shifts in momentum, a concept often discussed in sports but hard to quantify. Throughout the match, both players exhibited periods of dominance, with the advantage oscillating between them in an unpredictable manner. This intriguing match, along with other games from the tournament, provides a rich dataset to explore the elusive concept of momentum in tennis and its impact on match outcomes.

Momentum in sports has been studied by many scholars in various fields. Some aim to quantify momentum from a statisitical point of view, while others try to explain it from a sports psychological point of view. All fields are equally valid. However, our team will build a mathematical model to explain and answer the following questions:

- Create a model to capture the flow of play in tennis matches, identifying the player performing better at any given time and the extent of their advantage.
- Provide a visualization based on your model to depict the match flow.
- Use our model to evaluate the claim: "swings in a tennis match are random, as opposed to being influenced by momentum."
- Develop a model that predicts when the flow of play is about to change. Identify factors that might be related to these shifts.
- Test your model on other matches to evaluate its effectiveness and generalizability.
- Produce a report (max 25 pages) with your findings. Include a memo summarizing results and advice for coaches regarding momentum and preparation for matches.

1.1 Some Info on Tennis Scoring

Game

- Each point won counts as one.
- The first to score 4 points wins the game.
- At 3 points each, the score is 'Deuce.' From Deuce, a player must win by two points to win the game.
- In international tennis, the scores of 0, 1, 2, and 3 points are represented by the English words Love, 15, 30, and 40, respectively.

Set

- The first player to win 6 games wins the set.
- If both players win 5 games each, the set is won by the first player to lead by two games.

Tie-break Scoring When the game score in a set reaches 6-all, one of the following tie-break methods is used:

- Long set: One player must lead by two games to win the set.
- Tie-break (or 'short set'): Except in the final set (unless otherwise specified), the following rules apply:

- The first player to score 7 points wins the game and the set (at 6-all, a player must win by two points).
- The first server serves the first point; thereafter, players alternate serving two consecutive points.
- The first point is served from the right court, the second from the left, and the third from the right.
- Players change ends after every six points and at the end of the tie-break.

Best of Five Sets Format The match is won by the player who first wins three out of five sets.

2 Who Has A Bigger Momentum? A Visualisation Approach

Before we begins any real development of our model, we need to visualise. Visualisation provides a clear and graphical approach. But, we need to state our definition for "momentum". In one of the academic papers we found, the authors [1] identified momentum as:

Definition 1 (Momentum by Chen et al.)**.** *Score difference in a given period of time.*

In the paper. Momentum is defined mathematically as:

$$M(t, t_0, \gamma) = \begin{cases} y(t_0 + t) - y(t) & \text{if } y(t_0 + t) - y(t) > \gamma \\ 0 & \text{otherwise} \end{cases},$$

where $y(t)$ represents the score difference between the home and visiting team at time t_0, t denotes an increment of game time and γ represents the threshold value of momentum. Similarly, the momentum of the other team is

$$M(t, t_0, \gamma) = \begin{cases} y(t_0 + t) - y(t) & \text{if } y(t + 0 + t) - y(t) < -\gamma \\ 0 & \text{otherwise} \end{cases}.$$

Equipped with this definition, we now proceed to visualisation.

Data Preprocessing First we need to process some data in the "Wimbledon_featured_matches.csv".

- Convert "match_id" into unique sequential numbering
- Convert all time to seconds
- Standarise all scoring into points(e.g. 1,2,3). Reason being nonuniform scoring could affect calculation

Now, based on your definition of momentum, we have generated the following graphs. Original data are grey dots , and blue trendline is drawn based on original data.

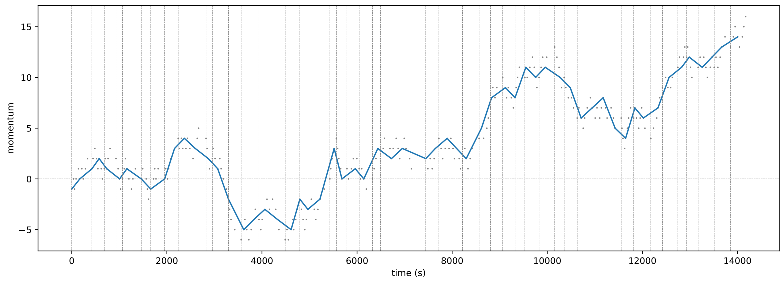


Figure 1: Global/Cumulative momentum(score difference) graph versus time, 1st match. Dotted lines seperates each game.

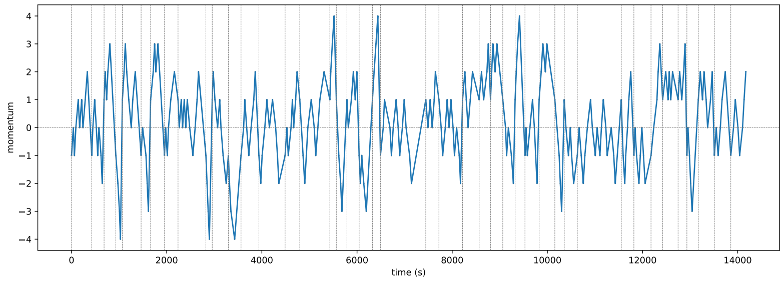


Figure 2: Momentum(score difference) of each game versus time, 1st match. Dotted lines sepaates each game.

We want a model that accounts for more than just score difference. We made 2 considerable additions to the definition of "momentum".

- Added in "server advantage". Numerous papers have stated that server advantage is an important psychological effect to increase the probability of scoring [3] [4], though it alone does not determine the competition outcome. **We model the effect by adding a "server scoring probability" to whomever is serving.**
- Added in "ace". Ace means a server wins a point by serving. This significantly boosts the server's morale.

A modified visualisation shows:

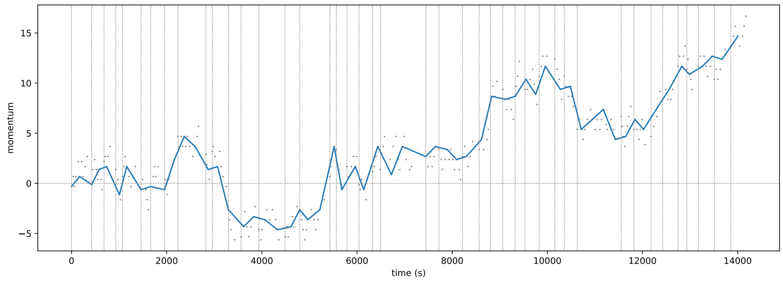


Figure 3: Modified global/cumulative momentum(score difference) graph versus time, 1st match. Dotted lines seperates each game.

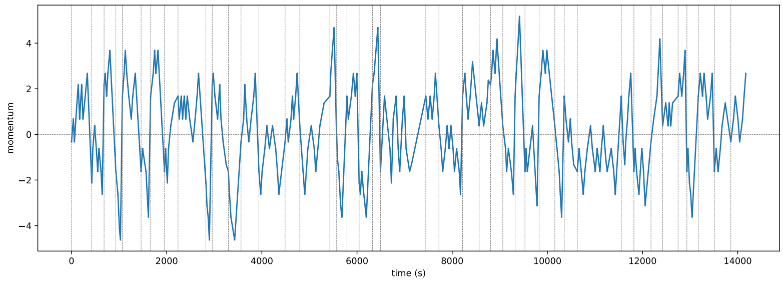
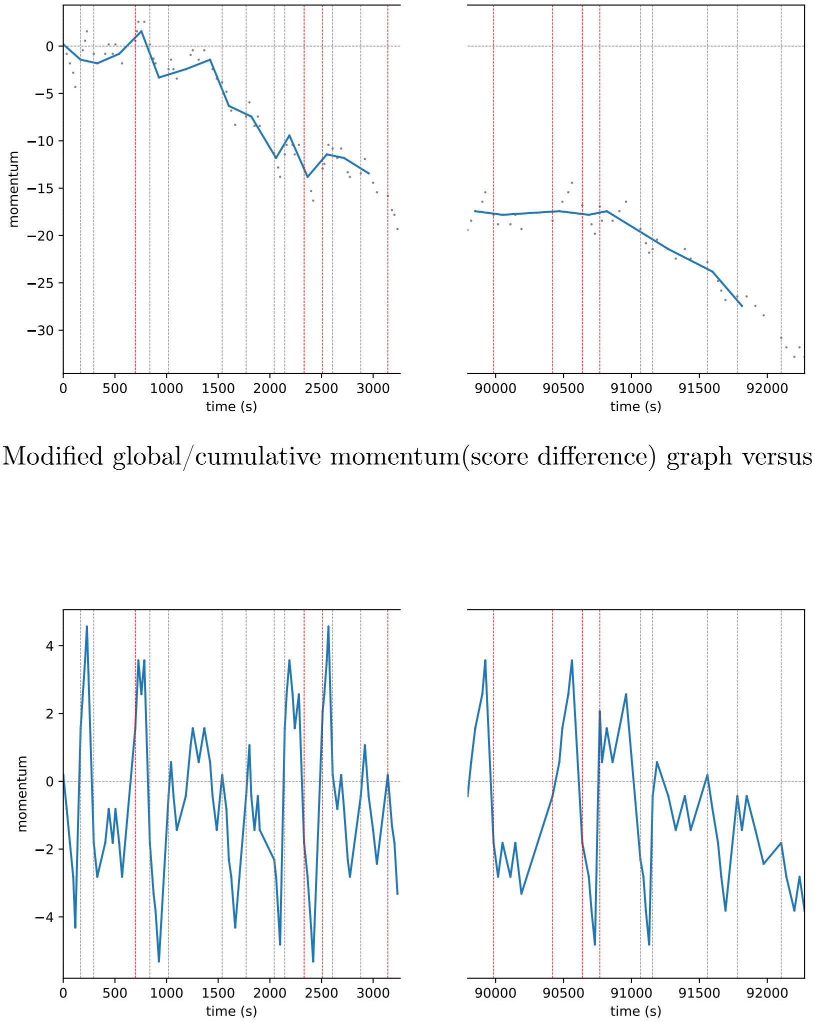


Figure 4: Modified momentum(score difference) of each game versus time, 1st match. Dotted lines sepaates each game.

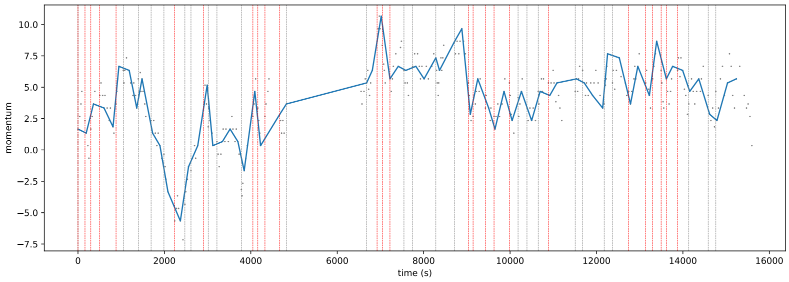
We also plotted momentum graphs of the remaining matches. Original data are grey dots , and blue trendline is drawn based on original data. Red lines are calculated based on original data.



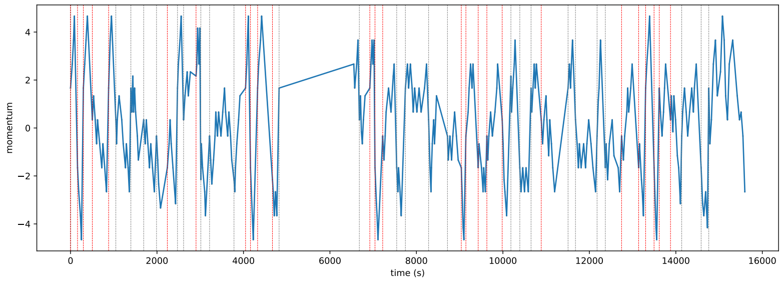
(a) Modified global/cumulative momentum(score difference) graph versus time.

(b) Modified momentum(score difference) of each game versus time.

Figure 5: 2nd match. Dotted lines seperates each game. Dotted red lines indicate swings



(a) Modified global/cumulative momentum(score difference) graph versus time.



(b) Modified momentum(score difference) of each game versus time.

Figure 6: 3rd match. Dotted lines sepaates each game. Dotted red lines indicate swings

3 Do Swings Happen Randomly?

The study of momentum in sports generated great interests among scholars, sports coaches and athletes. The debate is still ongoing–some people do not believen in momentum [2], just like our dear tennis coach.

It was mentioned earlier that the trends of momentum rising and falling reflect performance. We can define the turning point of the rising and falling trends as the inflection point. In each game, by linearly fitting the overall performance of player 1 in the game with the number of balls scored, we can determine the slope that reflects the rising and falling trends. When the performance changes (i.e., when the sign of the slope changes), an inflection point occurs. To filter out the inflection points where the trend changes slowly, only |slope| above a certain threshold are considered as inflection points. These points are marked with red vertical lines on the graph.

In this scenario, we aim to develop a mathematical model to identify turning points in a player's performance trend during a game. The turning points are defined as moments when the trend of performance changes from increasing to decreasing or vice versa, determined by the slope of a fitted line.

- Data Representation
  - Let t_i represent time in the game.
  - Let y_i represent the player's score at each t_i.
- Linear Regression Model
  - We use a simple linear regression model to fit the data points (t_i, y_i).
  - The linear model is given by $y = mt + b$, where m is the slope and b is the y-intercept. The slope m indicates the trend of the player's performance. A positive slope implies an increasing trend, while a negative slope indicates a decreasing trend.
  - To find the best-fit line, we minimize the sum of squared differences between the actual performance scores and the scores predicted by the linear model. The objective is to minimize the Mean Squared Error (MSE), given by:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - (mt_i + b))^2$$

- Identifying Turning Points
  - A turning point is identified when there is a change in the sign of the slope (from positive to negative or vice versa).
  - Moreover, to filter out minor fluctuations, a threshold is set for the slope. Only when the absolute value of the slope changes and is greater than a specified threshold (in this case, 0.15) is a turning point acknowledged.
  - Mathematically, if m_{prev} and $m_{current}$ are the slopes of two consecutive segments and $|m_{current}| > 0.15$, a turning point is identified when $\text{sign}(m_{prev}) \neq \text{sign}(m_{current})$.

Now we have identified all turning points that are actually swings, we need to test for their randomness. We have imposed certain conditions on turning points and the randomness test is as follows:

Let's denote:

1. N as the total number of games.
2. n_1 as the number of games with turning points.
3. n_2 as the number of games without turning points.

4. $N = n_1 + n_2$.

5. R as the total number of runs (a run is a sequence of consecutive games either all with turning points or all without turning points).

The expected number of runs ($E(R)$) and the variance of the number of runs ($Var(R)$) are given by:

1.
$$E(R) = \frac{2n_1n_2}{N} + 1$$

2.
$$Var(R) = \frac{2n_1n_2(2n_1n_2 - N)}{N^2(N-1)}$$

The test statistic (Z) is then calculated as:

$$Z = \frac{R - E(R)}{\sqrt{Var(R)}}$$

While P-value is given by:

$$\text{p-value} = 2P(Z > |z|)$$

Interpretation

- If the value of Z is significantly high or low, it indicates that the turning points are not randomly distributed, suggesting a pattern or trend in the player's performance changes.

- A high p-value (usually > 0.05) implies that the turning points are randomly distributed, indicating no specific pattern in performance changes.

insert table here			

Results We have obtained very low p-values, which suggests that swings are not random.

4 How To Predict Swings?

As soon as we have tested the non-randomness of swings, a new question immediately surfaces: can we predict the flow of play? This is very important to sports coaches as it can be used for post-match analysis. Fear not! We have developed a model using deep learning.

But the first step might be the most important step. In order to predict swings, or turning points in a match, we shall first consider the most influential parameters that can have impact on a player's performance, which can be deduced by the information from the preceding few points. Such parameters are called **features**.

4.1 Feature Engineering

In our data analysis, the critical parameters that have great impact on dependent variable, or anticipated value shall be regarded as features. Since we have too much data, we shall introduce the method of machine learning. We must feed ML algorithms data in the form of features, to aid in efficient predictions. In our case, predict the performance of each player and if a "swing of momentum" is going to occur.

However, the validity and feasibility of machine learning remarkably depends on the selection and processing of features, due to the different importance of parameters. We shall process the data in a way that generates features that has the greatest correlation on the output. Such step is known as feature engineering, which is the most time-spending but the most critical step in machine learning.

- Dependent Variable: Whether a player scores a point (binary 1 or 0), essentially predicting the probability of a player scoring in the next rally.
- Independent Variables: Data from all past rallies in the match.
- Serve Related Features
  - Server Identity: Identifying who is serving, as the server generally has a higher scoring probability.
  - Ace Balls: Frequency and recent occurrence of ace serves, indicating a higher probability of scoring through aces.
  - First or Second Serve: First serves generally have a higher win rate.
  - Serve Direction (Width): Wider serves may lead to higher scoring probabilities.
  - Serve Depth: Deeper serves may increase the likelihood of scoring.

  - Serve Speed: Faster serves could correlate with higher scoring probabilities.
  - Double Faults: Recent double faults could negatively impact the server's scoring probability.

- Rally Related Features
  - Winning Shots: Recent winning shots can boost a player's scoring probability in the next rally.
  - Type of Winning Shot: Differentiating between forehand and backhand winning shots.
  - Net Approaches: Frequency and success rate of net play, indicating a higher scoring probability.
  - Unforced Errors: Recent unforced errors could decrease a player's scoring probability.
  - Rally Length: Longer rallies indicating greater physical exertion and potential impact on scoring probability.
  - Running Distance: Total distance run by each player, indicating physical exertion.
  - Return Depth: Depth of returns, with deeper returns potentially reducing scoring probability.

- Critical Points
  - Break Points: Frequency and outcomes of break points, indicating scoring probabilities under high-pressure situations.
  - Missed Break Points: Impact of missed opportunities on subsequent scoring probability.
  - Won Break Points: Winning break points could boost a player's scoring probability.

- Global Data Features (Reflecting Overall Match Performance)
  - Total points, games, sets played.
  - Match duration.
  - Total aces, first serve success rate, average serve angle, depth, speed.
  - Double faults, winning shots, net approaches, and successful net points.
  - Unforced errors, average return depth, total rally count.
  - Break point statistics (attained, missed, won).

- Local Data Features (Reflecting Recent Performance)

- Duration of recent play, number of serves, aces, first serve success.
- Recent serve statistics (angle, depth, speed).
- Recent double faults, winning shots, net play frequency, success at the net.
- Recent unforced errors, average return depth, rally count.
- Recent break point statistics.
- Total recent running distance.

4.2 Regression Model

After preparing data, we decided to use a 2-layer linear regression model to train the data. The neural network is a simple feedforward network with two linear layers and a ReLU activation function in between.

- Layers:
  - First Linear Layer: This layer takes the input vector $\mathbf{x} \in \mathbb{R}^n$ (where n is the number of features) and transforms it to a hidden vector $\mathbf{h} \in \mathbb{R}^m$ (where m is the size of the hidden layer, 256 in our case). The transformation is defined as:

    $$\mathbf{h} = \mathbf{W}_1\mathbf{x} + \mathbf{b}_1$$

    where $\mathbf{W}_1 \in \mathbb{R}^{m \times n}$ is the weight matrix and $\mathbf{b}_1 \in \mathbb{R}^m$ is the bias vector of the first layer.
  - ReLU Activation: The ReLU (Rectified Linear Unit) function is applied element-wise to the hidden vector $\mathbf{h}$. It is defined as:

    $$\mathrm{ReLU}(h_i) = \max(0, h_i)$$

    for each element h_i of $\mathbf{h}$.
  - Second Linear Layer: This layer maps the hidden vector $\mathbf{h}$ to the output $y \in \mathbb{R}$. The transformation is:

    $$y = \mathbf{W}_2\mathbf{h} + b_2$$

    where $\mathbf{W}_2 \in \mathbb{R}^{1 \times m}$ is the weight matrix and $b_2 \in \mathbb{R}$ is the bias of the second layer.
- Training:
  - Loss Function: The model uses Mean Squared Error (MSE) as the loss function, which for a set of predictions $\hat{y}_i$ and true values y_i is given by:

    $$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

    where N is the number of samples.

  - Optimization: The Adam optimizer is used to minimize the loss function. Adam is an adaptive learning rate optimization algorithm that's considered effective for deep learning models.

- Prediction: The final model takes an input vector $\mathbf{x}$, processes it through the layers as described, and outputs a prediction y.

After 1000 epoches of training, our final result is Loss = 0.2189, while MSE = 0.2287.

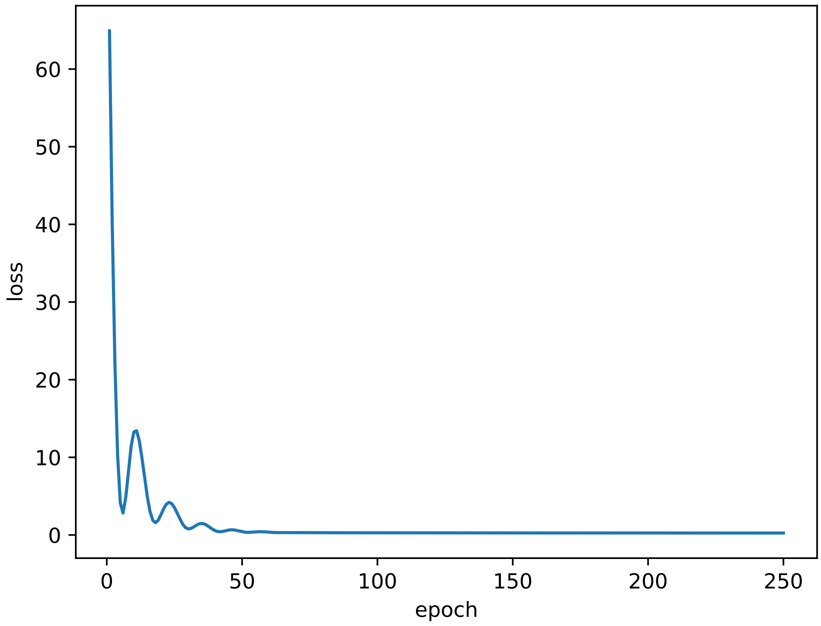
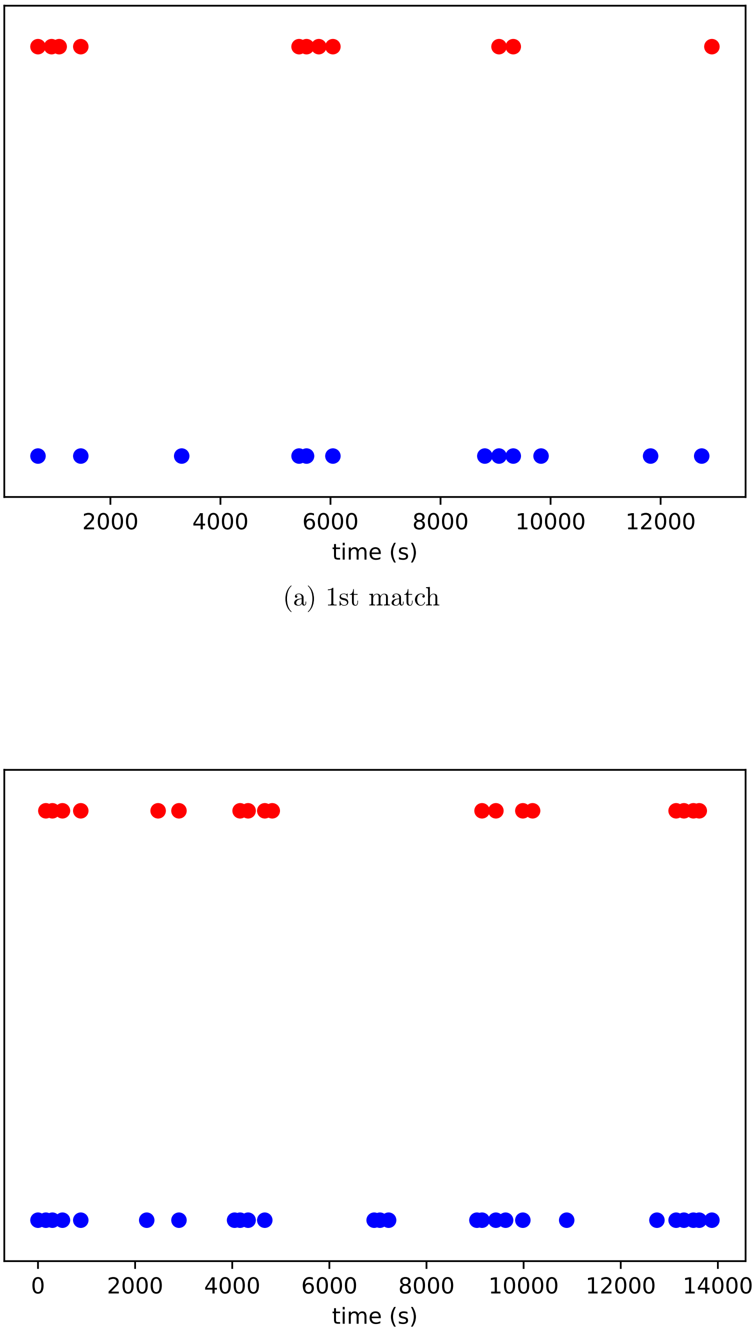


Figure 7: Loss versus epoch graph. X-axis is trimmed because loss converges to 0.22 from 250 epoches onwards.

Based on this model, we plotted a graph of actual swings and predicted swings . The accuracy from our model stands around 63.4%



(a) 1st match

(b) 3rd match

Figure 8: Predicted swings(red) versus actual swings(blue)

4.3 Classification Model

We have tested another deep learning model against our data. The text below outlines how our model works:

- Layers:
  - Input Layer: Number of nodes equals the number of input features. Let's denote this number as n.
  - Hidden Layer: Consists of 256 nodes. This layer uses a ReLU (Rectified Linear Unit) activation function.
  - Output Layer: 2 nodes, corresponding to the classification categories. This layer uses a Softmax activation function for probability distribution.
- From Input to Hidden Layer:
  - Let $\mathbf{x} \in \mathbb{R}^n$ be the input vector of size n.
  - The weights connecting the input layer to the hidden layer can be represented as a matrix $\mathbf{W_1} \in \mathbb{R}^{256 \times n}$.
  - The bias for the hidden layer is a vector $\mathbf{b_1} \in \mathbb{R}^{256 \times 1}$.
  - The output of the hidden layer before activation, $\mathbf{H}$, is given by $\mathbf{H} = \mathbf{W_1 x} + \mathbf{b_1}$, where $\mathbf{H} \in \mathbb{R}^{256 \times 1}$.
  - After applying ReLU, the output of the hidden layer becomes $\mathbf{H'} = \max(0, \mathbf{H})$, with $\mathbf{H'} \in \mathbb{R}^{256 \times 1}$.
- From Hidden to Output Layer:
  - The weights from the hidden layer to the output layer can be represented as a matrix $\mathbf{W_2} \in \mathbb{R}^{2 \times 256}$.
  - The bias for the output layer is a vector $\mathbf{b_2} \in \mathbb{R}^{2 \times 1}$.
  - The final output before applying softmax, Y, is given by $Y = \mathbf{W_2 H'} + \mathbf{b_2}$.
  - The Softmax function is applied to Y to get the probability distribution over the two classes. If $Y = [y_1, y_2]$, the softmax function is defined as $Y = \mathbf{W_2 H'} + \mathbf{b_2}$, where $Y \in \mathbb{R}^{2 \times 1}$.
- Loss Function:
  - The model uses Cross-Entropy Loss, which is commonly used in classification tasks. For a single instance with true label c and predicted probabilities p_1, p_2, the cross-entropy loss is:

$$\text{Loss} = -\sum_{i=1}^{2} \mathbf{1}(c = i) \log(p_i)$$

where $\mathbf{1}(c = i)$ is the indicator function, equal to 1 when $c = i$ and 0 otherwise.

- Optimization:
  - Adam Optimizer: This is used for adjusting the weights $\mathbf{W_1}, \mathbf{W_2}$ and biases $\mathbf{b_1}, \mathbf{b_2}$ to minimize the loss function. Adam is an adaptive learning rate optimization algorithm that combines the advantages of two other extensions of stochastic gradient descent, namely AdaGrad and RMSProp.
- Model Evaluation:
  - After training, the model's performance is evaluated by its accuracy on the test dataset. For predicted labels $\hat{y}$ and true labels y, accuracy is defined as:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} = \frac{\sum_{i=1}^{N} \mathbf{1}(\hat{y}_i = y_i)}{N}$$

where N is the total number of predictions.

After 300 epoches of training, we achieved an accuracy of 63.96%. The loss-epoch diagram is shown below:

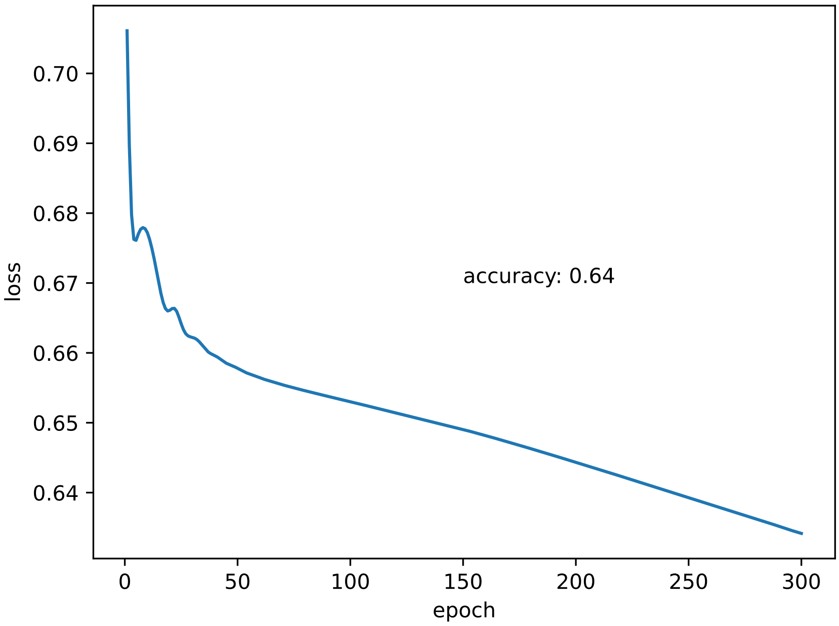


Figure 9: Loss versus epoch graph for classification model. Accuracy is around 0.64

References

[1] Tao Chen, Qingliang Fan, Kai Liu, and Lingshan Le. Identifying Key Factors in Momentum in Basketball Games. *Journal of Applied Statistics*, 48(16):3116–3129, 7 2020.

[2] David Hale. Is momentum real? an in-depth investigation of sports' most overused term - espn. *ESPN.com*, December 2021.

[3] Jan R. Magnus Klaassen and Franc J. G. M. On the advantage of serving first in a tennis set: Four years at wimbledon. *Journal of the Royal Statistical Society. Series D*, 48(2):247–256, 1999.

[4] I. M. MacPhee, Jonathan Rougier Pollard, and G. H. Server advantage in tennis matches. *Journal of Applied Probability*, 41(4):1182–1186, 2004.