Problem Chosen	2024	Team Control Number
C	MCM/ICM Summary Sheet	2412704

Momentum Quantifier (MoQ): Proving the "Big Mo" is Real

Abstract

The concept of the "Big Mo" ("Big Momentum") in competitive sports, particularly tennis, has long intrigued both enthusiasts and professionals. Often discussed in terms of a player's dominance or sudden shifts in the game's rhythm, momentum is frequently cited in spectacular matches, such as the 2023 Wimbledon Gentlemen's final between Carlos Alcaraz and Novak Djokovic. Despite its perceptible presence, momentum has remained largely qualitative, rooted in subjective experience and lacking a systematic, mathematical framework for analysis. This paper presents the **Momentum Quantifier (MoQ)**, a mathematical model designed to bridge this gap, offering a quantitative lens through which to view and understand momentum in tennis.

At the outset, it's crucial to acknowledge the complexity of momentum, which encompasses not only the physical and tactical aspects of the game but also psychological states and external conditions. Our approach to modeling momentum involves a comprehensive analysis of these factors, aiming to quantify how a player is performing relative to their opponent at any given point and to track the changes in this performance differential. To accomplish this, we employed **feature engineering**, which involves extracting and constructing variables influencing the match's flow, with a **sliding window** determining a unique set of temporal feature variables. The selection of a **Progressive Logistics Model** with forgetting factors drives the model to focus on recent play, ensuring the it remains dynamic and reflective of the match's current state. These aforementioned methods form a **CourtSense sub-model**, which predicts point outcomes using real-time match data, adjusted dynamically through the forgetting factor. The model eventually yielded an accuracy of **92.82%**.

To describe momentum concretely, we propose a **performance metric** that encapsulates the game's flow, correlating with observable factors and events. This metric serves as a cornerstone for the MoQ model, which, through high correlation with the metric, validates the existence and quantifiable nature of momentum. Central to MoQ is the innovative use of an adapted **Long Short-Term Memory (LSTM)** network, leveraging meticulously selected features—**the 4S (Scoring, Service, Stamina, Slip) factors**—to quantify momentum. These factors are derived from extensive analysis and are instrumental in capturing the nuanced dynamics of tennis matches. The model's effectiveness is evidenced by a promising **correlation analysis** result of **0.68** between the performance metric and the model's outputs, highlighting its capability to capture and predict momentum shifts. Furthermore, we define a "shift in momentum" in quantitative terms, identifying key indicators that may induce these pivotal changes in the game.

The MoQ model's robustness is further validated through application to an external dataset—the 2019 Australian Open Final between Novak Djokovic and Rafael Nadal. Despite differences in available features, the model demonstrates **generalizablity** and maintains performance, showing its potential for broader application in tennis and possibly other competitive sports.

In summary, the MoQ model's development, from its theoretical underpinnings to empirical validation, offers a comprehensive framework for analyzing momentum, providing valuable insights for players, coaches, and analysts alike. Furthermore, its adaptability and predictive accuracy, even in the face of varied datasets, highlight the model's potential for enhancing strategic planning and performance analysis in tennis.

Keywords: Feature Engineering, Progressive Logistic Regression, Time-Series Analysis with LSTM, Correlation Analysis

Contents

1 Problem Background

If you are a fan in competitive sports games like tennis, you might have heard a lot of the word "Momentum". When a player has momentum, they seem to control the dominance of the match, wielding the match's rhythm and flow. Novak Djokovic, a record-breaking tennis player, described this in his saying: "... if you have the mental ability to stay strong, stay patient and confident and just have belief in the right moments, then you get a win, you know."

However, momentum is not stationary; you probably have witnessed momentum being lost or stolen from a player during a match, entirely changing the outcome. Momentum can switch to the other player within the blink of an eye and stun unsuspecting players and spectators. This is especially spectacular during the 2023 Wimbledon Gentlemen's final between Carlos Alcaraz and Novak Djokovic (Figure 1). In this match, Djokovic completely dominated the first set with a score of 6 – 1, displaying enormous momentum. However, this quickly shifted to Alcaraz in the third set, where he won 6 - 1, before Djokovic regained control in the fourth set with a 6 - 3 win. Finally, in the late stages of the fifth set, the momentum swung once again, and Alcaraz secured victory with a score of 6 - 4.

But where **exactly** did that momentum come from? Despite its recognized impact, momentum has remained largely a qualitative factor and rooted in the subjective experience of spectators and self-perception of athletes. The question, then, is whether this intangible force can be analyzed and systematically understood through mathematical methods. If this proves feasible, it has the potential to greatly enhance the strategies used by competitive tennis players and coaches and, by extension, athletes and coaches in various sports.

Figure 1: Carlos Alcaraz (Left) and Novak Djokovic (Right) at Wimbledon [2]

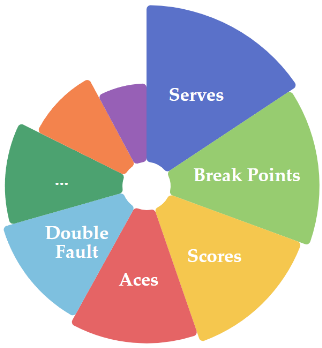


Figure 2: Data Aspects Provided by *Wimbledon_featured_matches.csv*

2 Problem Analysis: What is Momentum?

Momentum is often seen as a phenomenon that a player is playing exceedingly well at a time in the match. And because it its a *phenomenon*, it should be an outcome of a wide range of factors, including psychological states, like confidence and mental toughness, to external conditions like the current state of the match. To analyze and build a robust model that measures the momentum of players in competitive tennis, it is crucial to uncover and model these factors extensively. Specifically, our model should be capable of completing the following tasks:

(1) Evaluate the quantitative degree how one player is performing better than the other player at any given point in the match, and illustrate the changes of such performance difference in visual representation.

(2) Based on extensive factors, including the flow of match, the model should quantify the concept of "momentum" in tennis matches, and identify potential indicators that could predict shifts in momentum. To address the coach's skepticism, the model needs to be compared to a null model that assumes outcomes are random (e.g., based on point-by-point win probabilities without momentum).

(3) Finally, the developed "momentum" model should be tested on other matches from the different matches, including women's matches, to exhibit its predictive accuracy and generalizability, and identify any limitations or additional factors that may need to be incorporated.

2.1 Capturing the Flow of Match

To compare the performance ratings between the two players, we can use the data provided in *Wimbledon_featured_matches.csv* that describe the explicit match information, such as whether a player was serving or at a break point, etc. (shown in Figure 2); but we also need to consider other implicit information that has significant impact on the state of the match, for example, whether a player has a high scoring streak. Other unimportant information, potentially the speed of the serve, needs to be neglected to ensure model's simplicity and robustness. This inherently requires **feature engineering**: to extract and construct feature variables that capture both explicit and implicit aspects that control the flow of the match.

Since our model aims to dynamically assess the match's state as it evolves, it requires a granular focus on the state of the game, taking into account both minor and major developments. For example, the player may have a lead in games won, but when the other player just scored a break point, that player could be performing much better at that specific time. Therefore, to scope the data to recently played points, the **sliding window** technique can be employed; to handle older scores and progressively update the model as the match goes on, the introduction of a **forgetting factor** would be applicable, as illustrated in Figure 3. The remaining work would be finding an appropriate model to map these parameters to the performance or the winning probability of a player.

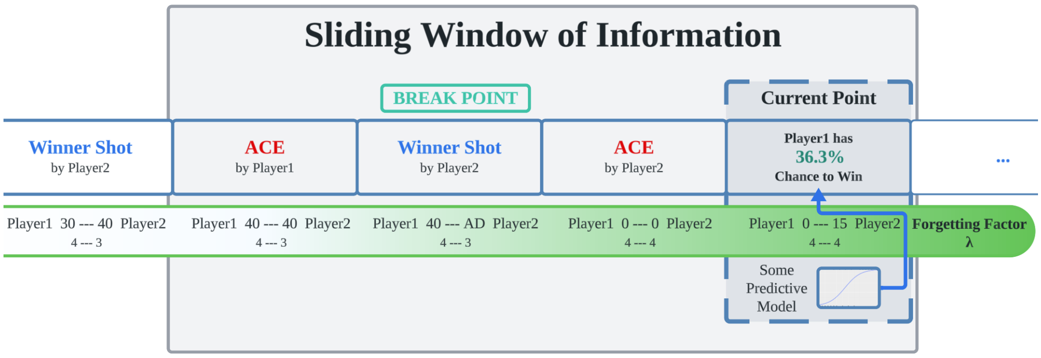


Figure 3: Information Is Constrained by Sliding Window and Forgetting Factor. (Note that the data in this graph is for understanding the process.)

2.2 How to Describe Momentum and What Affects It

To build an effective model that can analyze the player's momentum, we must first understand what "momentum" actually means. When we talk about momentum in tennis or any competitive sports, we are painting a picture of a player hitting their stride at just the right moment. The crowd can sense it, their cheers growing louder and more fervent. The players can feel it, their steps lighter, their spirits higher.

However, no matter how you would argue, "momentum" stays an elusive and subjective concept and largely a psychological state [1]; therefore, it is extremely hard to use an completely accurate numerical measurement for it, especially when we do not have available psychological data for these matches. With no measurement method, it is impossible to build a mathematical framework that could analyze the factors affecting the momentum, let alone predicting shifts in momentum as the match is being played. Worse yet, the momentum effect in tennis is highly controversial itself and is influenced by a complex interplay of factors, some measurable and some intangible.

Nonetheless, we argue that there is still a possible way to measure and model momentum. For example, in the final match of Wimbledon 2023, from when Djokovic won the first set with a dominating 6 to 1, to when Alcaraz took control in the third set with a same score but reversed for him, we can confidently say that there had been a shift in momentum. Therefore, while there is no established measurement of momentum, we can define a standard that momentum manifest itself in. In other words, we first designate a performance *metric* that reflects momentum, such as the changes in scoring patterns, and then analyze how changes in this metric correlate with other observable factors and events during a match. This approach allows us to use quantitative data to infer the presence and impact of momentum, even if we cannot measure it directly. This process described in detail in the Figure 4:

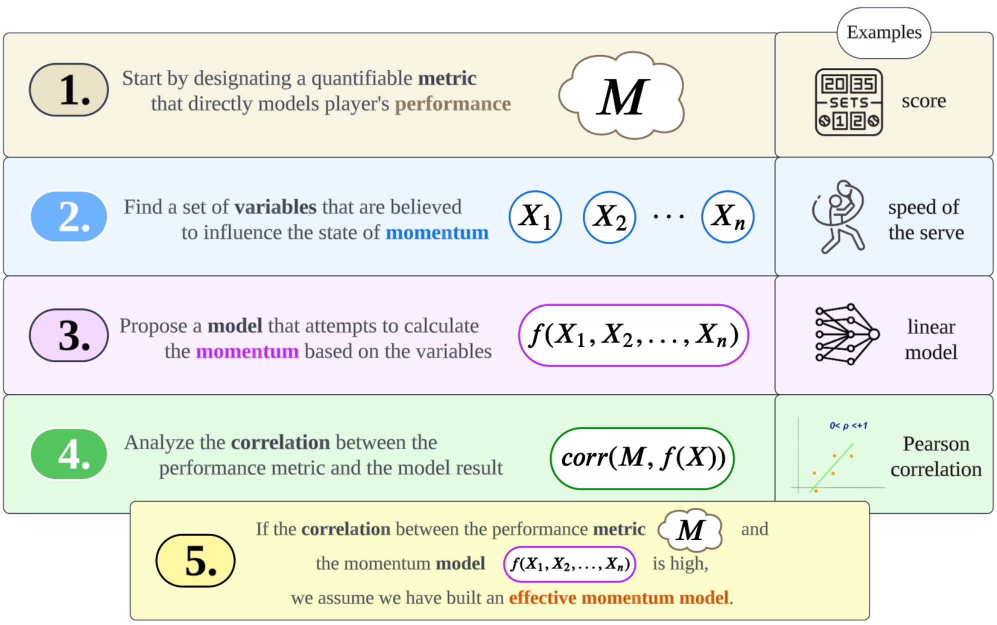


Figure 4: Process of Modeling Momentum

As described in the figure, if the correlation between the built momentum model and the performance metric are high, we can say that the momentum exists, affecting the ebb and flow during the match, and the model that attempts to calculate the momentum has a good credibility. The remaining work to do is to find the suitable parameters for the performance metric and look for factors to be incorporated into the momentum model, perhaps also leveraging the work of our first model that predicts the outcome of points. We note that any model of momentum, especially models that rely on physical match data, has a degree of uncertainty. We seek to capture the probability of shifts of games with our momentum model rather than deterministic outcomes.

If this method proves feasible in quantifying momentum, then other problems, such as predicting momentum shifts and utilizing the model to predict other matches, can be a straightforward process. If not, we will also identify the problems with our model and set a foundation for improving the model.

2.3 Our Work

It is our belief that a practical and inclusive mathematical framework to model the concept momentum would greatly help the tennis community in understanding the concept of momentum to improve performance, strategy, and training methodologies. In this paper, we seek to address this challenge using a set of match data from Wimbledon 2023 Men's Matches. Through extensive investigation in the field of tennis and thorough exploration, our main model - the **Momentum Quantifier (MoQ)** is built, which integrates a comprehensive set of useful features and use a variety of statistical methods to achieve this goal. The building of this model can be seen in Figure 5, or described as follows:

(1) To capture the flow of match in real-time, we constructed the CourtSense model as a sub-model of MoQ. The model starts by applying a Sliding Window machanism to a set of extracted features; these features are passed into the core Progressive Logisic Model to predict the winning chance of the point; due to the real-time progression of the competition, a Forgetting Factor is added to dynamically update this model to reflect the most recent match status.

(2) To quantify the concept of momentum, we designated a performance metric that can better reflect the flow of momentum, and use this metric to guide the Momentum Quantifier (MoQ)'s learning. At the heart of MoQ is an adapted LSTM model that uses a meticulously selected momentum factors - the 4S factors.

(3) To verify the model's effectiveness at capturing momentum, we do a correlation analysis between the performance metric and the model's result, and obtained a promising result. We also defined what a "shift in momentum" means and found out which factors are has the most potential in causing such momentum shifts. Finally, the MoQ undergoes a set of optimization, exhibiting its generalizability, adaptibility, and future directions.

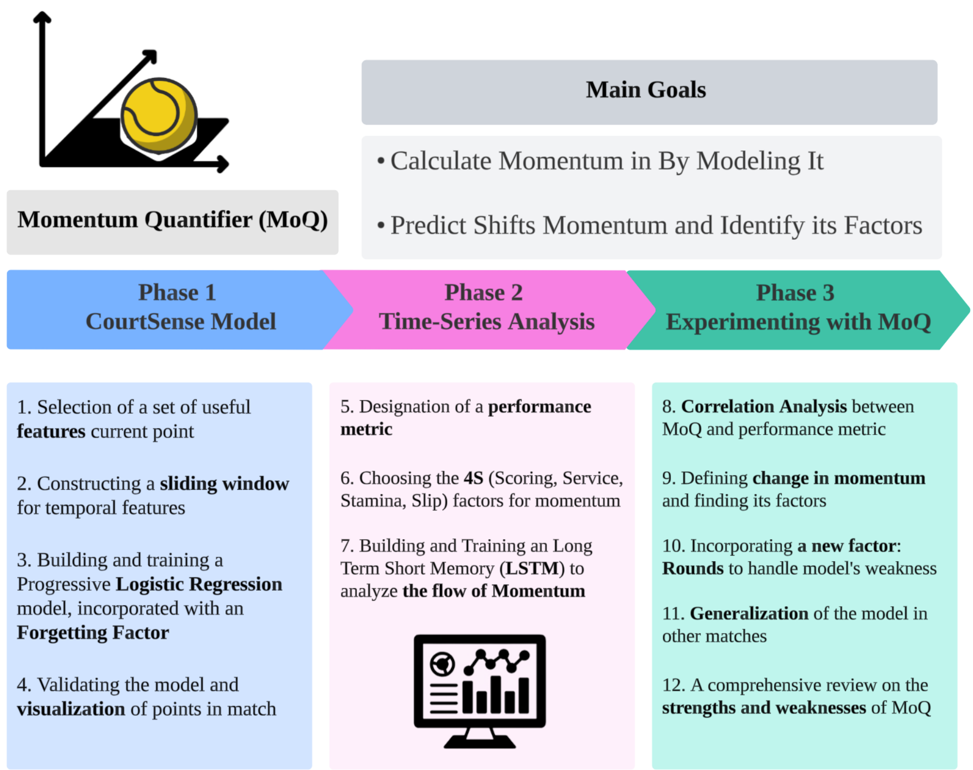


Figure 5: Journey Map of Our Work

3 Model Preliminaries

Considering that practical problems always contain many complex factors, first of all, we need to make reasonable assumptions to simplify the model, and each assumption is closely followed by the corresponding explanation:

3.1 Assumptions

Assumption 1: **Momentum is largely a subjective experience and there is no accurate way to measure it.**
Explanation: Momentum, especially in tennis, is a concept based subjectivity; for example, the psychological state of the players and the intangible atmosphere of the competition.

Assumption 2: **Momentum can be reflected in scoring patterns of the tennis player.**
Explanation: The concept of momentum in a tennis match is not merely abstract or psychological but has tangible manifestations in the scoring patterns observed during the game. This assumption is crucial for building a mathematical model for momentum.

Assumption 3: **Momentum is a universal experience in competitive environment.**
Explanation: We assume momentum is not an exclusive concept in tennis, and the analysis of momentum in other fields like basketball or table tennis can be investigated and hence, referenced.

Assumption 4: **The errors, if any, in the *Wimbledon_featured_matches.csv* are ignored.**
Explanation: After a inspection on the dataset, we initially found logical errors and missing values. Thus, we treat this dataset as fully credential.

*Additional assumptions made in individual sections to simplify analysis.

3.2 Notations

Some important mathematical notations used in this paper are listed in Table 1.

Table 1: Notations used in this paper

Symbol	Description
t	a specific timestep, corresponding to a point being played
$F^{(t)}$	the feature set selected for the CourtSense Model at timestep t
$x^{(t)}$	the input vector for logistic regression at timestep t
$M^{(t)}$	the performance metric at timestep t
$s^{(t)}$	the 4S factors as LSTM model input at timetep t
$\Delta M^{(t)}$	the difference between $M^{(t)}$ and $M^{(t-1)}$
M_{shift}	the threshold $\Delta M^{(t)}$ must reach to identify a momentum shift
$\sigma_{\Delta M}$	The standard deviation of ΔM

*Additional variables that not listed here will be discussed in detail in their corresponding section.

4 CourtSense: Understanding the State of Match

The CourtSense model aims to capture the current state of the match by predicting the outcome of the next unplayed point, using available match information. First, we need to apply feature engineering to the given data to extract useful features.

4.1 Feature Engineering

4.1.1 Data Extraction

The data provided by the problem (*Wimbledon_featured_matches.csv*) is a comprehensive dataset that contains the complete match information, including scores, serve details, point details which are all applicable to analyze the state of the match. After a thorough inspection of the data, we find that the data is clean and usable. However, these are all explicit match information and is limited to a specific point, for example, whether this player is serving, what is the ratio of scores, etc. We think that more temporal features like whether this player is leading in score are more direct in describing the match's status; hence we added them to the feature set. These additional features are included in Table 2.

Table 2: Added Match Information

Column Name	**Description**	**Example Value**
p1_streak	The winning / losing streak of player 1 up until that point	5, 0, -3
p2_streak	The winning / losing streak of player 2 up until that point	3, 0, -5
p1_leading	Whether p1 is leading in score	1 (leading), 0, -1
p2_leading	Whether p1 is leading in score	-1 (losing), 0, 1
p1_leading_game	Whether p1 is leading in game	1, 0, -1
p2_leading_game	Whether p1 is leading in game	-1, 0, 1
p1_leading_set	Whether p1 is leading in set	1, 0, -1
p2_leading_set	Whether p1 is leading in set	-1, 0, 1

These features, representing nuanced indicators of the current state of the match, are better to feed into the model than raw scores for simplicity and accuracy. For example, the winning streak is a direct reflection of the player's current performance and momentum.

Moreover, for the data columns like '*game_victor*' that represent the winner of the game (or set), we split them into 2 columns to be player-specific. For each point i in the match, we transform, for example, the *game_victor* column into two player-specific columns, $V_{p1}^{(i)}$ and $V_{p2}^{(i)}$, defined as:

$$V_{p1}^{(i)} = \begin{cases} 1 & \text{if } \textit{game\_victor} = 1 \text{ at point } i \\ -1 & \text{if } \textit{game\_victor} = 2 \text{ at point } i \\ 0 & \text{if if } \textit{game\_victor} = 0 \text{ at point } i \end{cases}$$

$$V_{p2}^{(i)} = \begin{cases} 1 & \text{if } \textit{game\_victor} = 2 \text{ at point } i \\ -1 & \text{if } \textit{game\_victor} = 1 \text{ at point } i \\ 0 & \text{if if } \textit{game\_victor} = 0 \text{ at point } i \end{cases}$$

this procedure ensures these data can be modeled as their respective performance.

Finally, we discarded other rather unimportant or redundant information for the match, such as the raw score information, the speed of the serve (which cannot be used for points already in play when predicting point outcomes), unforced errors (less indicative of player's performance), and information about net points. These choices are made after conducting a thorough research on related Sports Analytics in the field of tennis [3][4][5]. The remaining features alongside the constructed features can be used as raw input data for our model.

4.1.2 Sliding Window Mechanism

We recognize that the current state of the match is heavily influenced by the temporal features of, not only for the current point, but also the last few points. For example, if the player ran a long distance in the past few point rounds, they will carry more amount of fatigue, influencing the player's performance of the current point. Or if the player has just won a break point, they will gain much more confidence in their coming plays. Therefore, by employing a sliding window mechanism, the model can utilize the temporal features on the most recent and relevant data points, ensuring that the analysis remains sensitive to the current state of play and the immediate history before it. This is described in the formula below:

Let F_t and F_{nt} represent the sets of temporal and non-temporal features, respectively. For a given point in time t, the combined feature set $F^{(t)}$ is given by:

$$F^{(t)} = S_t^{(t-1)} \cup F_{nt}^{(t)}$$

where

$$S_t^{(t-1)} = \sum_{i=t-3}^{t-1} F_t^{(i)}$$

represents the sum of temporal features of last 3 points played, and F_{nt} represents the non-temporal features at the current point t. The temporal features include whether the player just had a break point, or double fault, etc., which could affect the performance of the current play; the non-temporal features include whether this player is serving at this point, and the winning streak, etc.

So far, we have acquired a granular and exhaustive set features that completely describe the match's status at a specific point and is ready to be fed into our Progressive Logistic Model - the core of the CourtSense model.

4.2 Progressive Logistic Model

To effectively capture the dynamic flow of a tennis match, we introduce the Progressive Logistic Model. This model is designed to predict the outcome probability of the current point, providing a real-time assessment of the match's status and the relative performances of the players. The Progressive Logistic Model utilizes a multivariate logistic regression framework, due to its suitability for binary outcome prediction (e.g., point won or lost) and its ability to output probabilities based on multiple features; furthermore, combined with the Forgetting Factor, the model is capable of progressively updating the itself according to the latest match information as the match progresses. The general process of this model is depicted in Figure 6.

4.2.1 Model Establishment

Logistic regression is a statistical model used for binary classification problems, developed by David Cox in 1958. It is a type of generalized linear model (GLM) that uses a logistic function to model a binary dependent variable. The goal of logistic regression is to find the best fitting

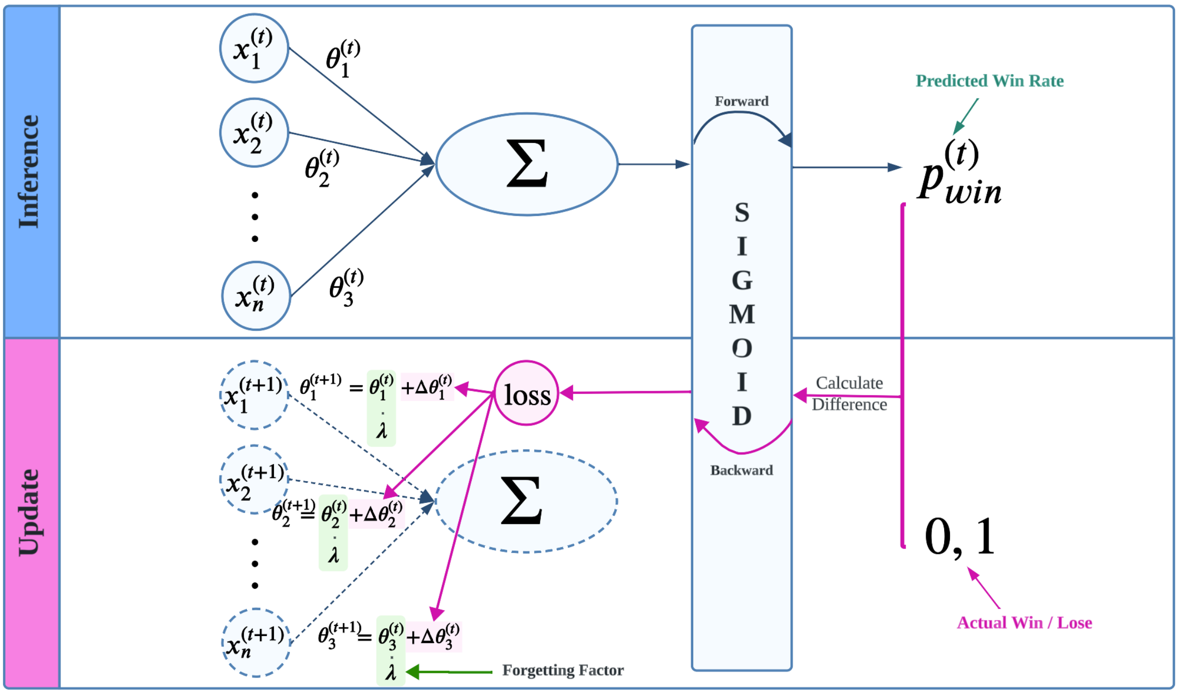


Figure 6: Process of the Progressive Logistic Regression Model

model to describe the relationship between the dichotomous characteristic of interest (dependent variable = 0 or 1) and a set of independent variables (predictors) [6]. While the assumption of variable independence is ideal, logistic regression can still perform effectively in real-world scenarios where complete independence is rare, such as our feature set that describe match's information.

To feed the aggregated features into a logistic regression model, each feature set is represented as a vector $\mathbf{x}^{(t)}$ at timestep t (which means the t-th point played in this match), and these vectors are used to calculate the probability of winning a point. $\mathbf{x}^{(t)}$ is directly derived from the feature set $F^{(t)}$, which contains both temporal features and non-temporal features.

The logistic regression model predicts the probability $P(y^{(t)} = 1|\mathbf{x}^{(t)}; \theta)$ where $y^{(t)} = 1$ indicates the event of interest (i.e., winning the point), $x^{(t)}$ is the feature vector at timestep t, and θ represents the model parameter (weights). The model calculates the probability using the formula below:

$$P(y^{(t)} = 1|\mathbf{x}^{(t)}; \theta) = \sigma(\theta^\top \mathbf{x}^{(t)})$$

where $\sigma(z)$ is the sigmoid function defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

and $z = \theta^\top \mathbf{x}^{(t)}$ represents the linear combination of the features $\mathbf{x}^{(t)}$ weighted by the model parameters θ, expressed as:

$$z = \theta_0 + \theta_1 x_1^{(t)} + \theta_2 x_2^{(t)} + \cdots + \theta_n x_n^{(t)}$$

where θ_0 is the intercept term, $\theta_1, \theta_2, \ldots, \theta_n$ are the weights associated with each feature $x_1^{(t)}, x_2^{(t)}, \ldots, x_n^{(t)}$ in the feature vector $\mathbf{x}^{(t)}$.

Once we have the correct set of model parameters, we can determine the winning probability of a player based on the feature vector of the match for that player.

4.2.2 Parameters Training & Updating

One of our Progressive Logistic Model's highlights is its ability to progressively adjust itself during the play of the match. But this does not mean it relies only on the real-time data; instead, we first train the model using the data given by the problem (pre-training phase), and when applied to a specific match, the progressive real-time data updates the parameters as the match unfolds (updating phase).

The training of the logistic regression model's parameters θ is achieved by minimizing the logistic loss function, which quantifies the difference between the predicted outcomes and the actual results of matches:

$$\mathcal{L}(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log(\sigma(\theta^\top \mathbf{x}^{(i)})) + (1-y^{(i)})\log(1-\sigma(\theta^\top \mathbf{x}^{(i)}))\right]$$

where m represents the number of observations in the training dataset, $y^{(i)}$ denotes the outcome of the i-th point (win or loss), and $\mathbf{x}^{(i)}$ is the feature vector associated with the i-th point in a match. Then, the parameter vector θ is optimized performed using Gradient Descent, described as:

$$\theta := \theta - \alpha\nabla_\theta\mathcal{L}(\theta)$$

where α is the learning rate that regulates the step size during each iteration, and $\nabla_\theta\mathcal{L}(\theta)$ represents the gradient of the loss function with respect to the parameter vector θ. Considering the constraints of the size of the dataset and the model's ability to update itself during a match, we meticulously choose different values for pre-training our model and updating our model during the match:

$$\alpha = \begin{cases} 0.01 & \text{at pre-training phase} \\ 0.05 & \text{at updating phase} \end{cases}$$

in this way, for each data point, both when pre-training and when updating, the parameter vector is iteratively updated in the direction that reduces the loss function.

At updating phase, the logistic regression model's parameters (θ) need to reflect the latest match data. This requires us to place more weight to the latest information of the points. Therefore, aside from using a higher learning rate value, a forgetting factor λ is incorporated into the parameter update rule:

$$\theta^{(t)} = \lambda\cdot\theta^{(t-1)} + \Delta\theta^{(t-1)}$$
$$\Delta\theta^{(t-1)} = -(1-\lambda)\alpha\nabla_{\theta^{(t-1)}}\mathcal{L}(\theta^{(t-1)})$$

this rule adjusts $\theta^{(t)}$ by blending the previous parameters $\theta^{(t-1)}$, scaled by λ, with the changes updated by new data ($-\alpha\nabla_{\theta^{(t-1)}}\mathcal{L}(\theta^{(t-1)})$), scaled by $(1-\lambda)$. In such a way, the forgetting factor allows the model to prioritize recent information and adapt to changes in gameplay strategies or player performance over time.

To this point, we have thoroughly established our CourtSense model, which first uses feature selection combined with a sliding window mechanism, and leverages a Progressive Logistic Model to follow the flow of the match and predict the result of a specific point.

The original dataset of *Wimbledon_featured_matches.csv* included the matches played at Wimbledon 2023 from the 3rd round to the final round (7th round), with a total of 7256 points. To separate the training and testing sets, we use the data of the 30 matches from the 3rd round to the semi-final round (a total of 6950 points), and use the epic match in the final round (334 points) between Carlos Alcaraz and Novak Djokovic to test our model's accuracy.

We train our CourtSense model using Scikit-learn, a machine learning framework for Python. After pre-training, the CourtSense Model is then tested on the testing set, where the model starts from predicting the winning probabilities of both players at the first point, and updates itself based the actual result of the first point, and then moves on to predict the second point, etc. The aggregated results of all point predictions and their actual outcomes are used for validating our model's performance.

4.3 Model Validation

In validating the performance of our CourtSense model, we focused on a suite of standard evaluation metrics, including Mean Squared Error (MSE), accuracy, confusion matrix, and AUC-ROC. MSE represents how far the model's predictions are from the actual values; accuracy measures the proportion of total predictions our model gets right; the confusion matrix shows exactly how many values were predicted correctly or falsely; AUC provides an insight into the model's ability to distinguish between the two possible outcomes (winning or losing a point), with higher values close to 1 indicating better performance. The detailed explanation can be found in [7]. The result of these metrics are shown in Table 3 and Figure 7.

Table 3: Evaluation Metrics Result for CourtSense Model

Metric	Value for Player 1	Value for Player 2
MSE	19.57%	19.02%
Accuracy	92.52%	93.11%
AUC-ROC	93.886%	93.445%

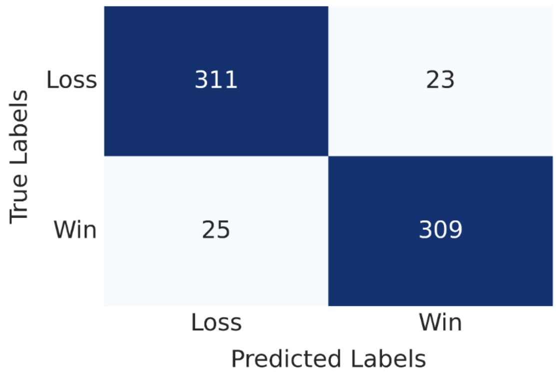


Figure 7: Confusion Matrix for CourtSense Model

To our surprise, the CourtSense model performs exceedingly well at predicting each point's outcome during the match. It is almost too good: the model achieved an average accuracy of 92.82%, while the AUC stood at an average of 93.67%, suggesting that the model can reliably forecast match dynamics, providing valuable insights into players' performance. This could be attributed to our meticulous selection of features and the various mechanisms our model used.

In addition to evaluating the overall performance of our CourtSense model, we also analyzed on its adaptability throughout the progression of a match. We analyzed the model's accuracy in predicting outcomes by computing a moving average across 20-point windows. This approach allowed us to observe how the model's predictive accuracy fluctuated over the course of a match. The resulting data is illustrated in Figure 8.

We can see the model starts at a relatively low accuracy, and moves up as the match unfolds. Generally, for each set of the game (as separated by the vertical lines in the graph), the model is able to progressively adapt its parameters to better fit the match's status, although there are some fluctuations, probably signaling a randomness for the outcome of the points.

4.4 Match Flow Visualization

To visualize the flow of the match, we select the 2nd set of the final match between Carlos Alcaraz and Novak Djokovic, which was an intense play with a tie-breaker of 7-6. Figure 9 presents

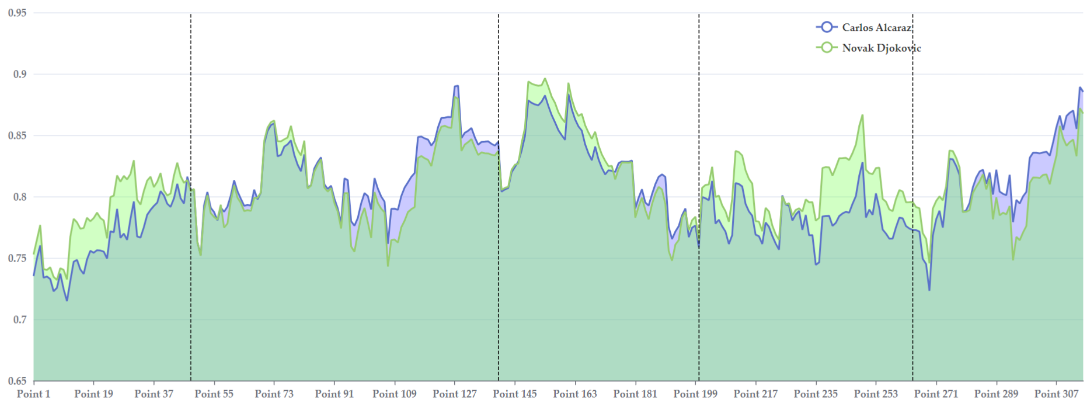


Figure 8: Average Predictive Power of CourtSense as Match Unfolds

a graphical depiction of the prediction difference between the two players, which indicates their performance differences, over the course of this set. The vertical axis is the difference in the predicted probability of winning each point; while the horizontal axis represents the sequence of points played. When the curve rises above the zero line, it suggests that Player 1 is more likely to win the point; when it dips below, Player 2 is favored. The dotted line at 0.08 reflects the average prediction difference across all points, which is a little above 0 (happens to reflect the tie-breaker win of Alcaraz).

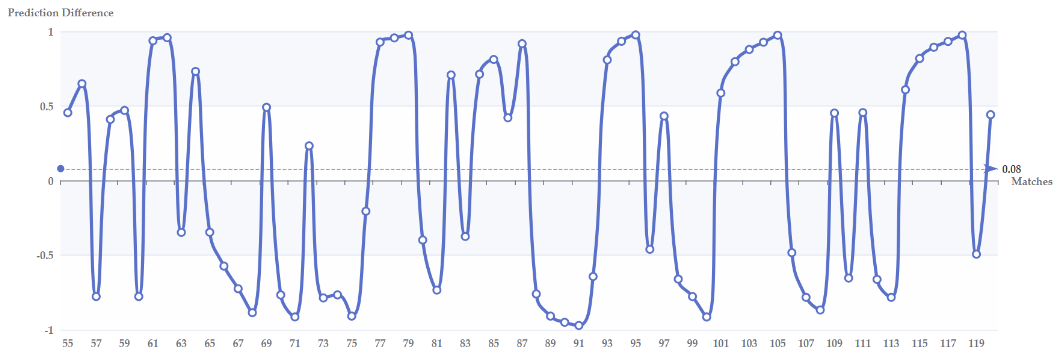


Figure 9: Match's Flow during the 2nd Set of the Final Match, Indicated by the Winning Chance Difference

The fluctuations in the graph traces the ebb and flow of the match, or perhaps, the real-time shifts in "momentum" between the players during the match. These shifts are marked by rapid ascents and descents in the curve. Notably, a player often maintains this momentum for a sequence of points before it changes to the other player, illustrating the competitive nature of a tennis match. But does the graph capture a true flow of "momentum", or is it merely an illusion of temporary advantage? We'll discuss this problem by advancing our model below.

5 Momentum Quantifier (MoQ): Decoding the Myth of Momentum

As discussed in the Problem Analysis section, we think that momentum is an elusive concept and cannot be measured directly; but it can be reflected by the changing patterns of score. By defining a performance metric like this, we can try to develop a model (Momentum Quantifier, MoQ) that calculates the amount of momentum based a set of anticipated factors, and if the MoQ's result has high correlation value with the metric, we are safe to say the model is effective for analyzing and quantifying momentum.

5.1 Designating a Performance Metric

This brings back to the question: does the chance of winning a specific point reflect the momentum? While the granular details of every point matter, we think that momentum is more about the *broader strokes* of the match — the ebb and flow felt over games and sets. So although predicting each point's outcome can be influential, we also need to consider **a suitable scope of** score that can reflect the volume of momentum. Under this scope, we calculate gained-score differences. For example, during the 5th set between Alcaraz and Djokovic, their difference between their gained points in a scope of 12 points, are depicted as below in Figure 10:

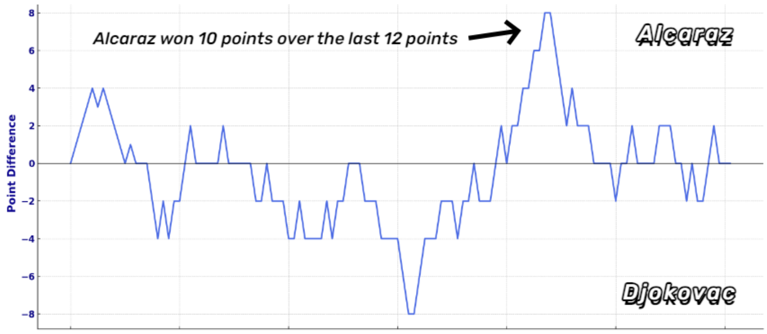


Figure 10: Sum of gained point differences within a scope of 12 points

We can clearly see that at the middle point in the match, Djokovic has a significant edge over Alcaraz at first, before Alcaraz taking back the momentum and chased the points back by winning 10 points in the last 12 points. The actual flow of momentum, as we watched the match from [8] and inferred from the live atmosphere, is quite similar to the flow of this graph. This means this the scoped gained-score difference is indeed a performance metric good at reflecting momentum.

We designated our performance metric as the following. Denote the set of points won by player A in a given scope as P_A, and P_B for player B, with the scope defined as the last N points played. The performance metric, M, for player A over player B in this scope can then be defined as:

$$M = \sum_{i=1}^{N} (P_A^i - P_B^i)$$

where P_{A_i} and P_{B_i} are binary indicators of whether player A or B won the ith point in the scope, respectively. Specifically, $P_{A_i} = 1$ if player A won the ith point, and 0 otherwise; similarly for P_{B_i}. After careful consideration and collection, we eventually chose a value of 12 for the scope N.

However, there is some nuance here. As discussed, our goal is to build a momentum model that incorporates some important factors at a *specific point* to measure its momentum. Apparently, the gained-score difference within the scope N is an *average* performance difference in that scope, not indicator of momentum of any specific point. For instance, if we look at the highest point in Figure 10, it means Alcaraz took the 10 points of the 12 points **before** the play of this point. This difference only signals he has played well in the last 12 points in general, not a performance indicator of the current point; and as we can see, this difference dropped sharply after this point, signaling the momentum is shifting towards his opponent at this point.

Reviewing our problem, we eventually aim to predict shifts in momentum as the match progresses. Therefore, when training our model, it would be rational to use the N points *after* the point of concern rather than *before* that point; in this way, the model is able to *predict* the average momentum after that point. This choice is made after we found that, the model trained using scoped gained-score difference before that point achieved little correlation with this metric. Finally, the performance metric of point t in the match is defined as:

$$M^{(t)} = \sum_{i=t}^{t+N} (P_A^i - P_B^i)$$

5.2 Determining the Factors of Momentum

Before defining the structure of the MoQ model, we first need to find the possible factors of momentum. After an extensive research in this field [9] [10], we eventually defined the **4S** factors (Scoring, Serve, Stamina, and Slip) to account for factors that may affect the shifts in momentum:

S1. Scoring Factor: point_win
Explanation: We use the result of our CourtSense model, which is capable predicting the outcome of the immediate point, as scoring factor. This will model how the chances of winning the current point affect the flow of momentum.

S2. Service Factor: server, break_pt_won, ace
Explanation: The statistics of serves heavily impacts the shift of momentum in tennis, especially when acing and winning a break point.

S3. Stamina Factor: distance_run, speed_mph, rally_count, serve_depth, serve_width, return_depth
Explanation: We believe these stamina factors that reflect the quality of the play are also influential in momentum.

S4. Slip Factor: double_fault, unf_err
Explanation: Small mistakes like unforced errors and double fault of a player, although not significant in actual performance, can potentially cause a psychological fluctuation, thereby affecting momentum.

5.3 MoQ's Core: Time-Series Analysis with LSTM

In this part, we describe the main model structure used in our MoQ model. Now that we have the factors that may affect the volume of momentum, we need to build a model that quantifies momentum and predict momentum shifts. Note the 4S factors are point-specific data, and points are essentially a time-series within a match:

$$S = \{s^{(1)}, s^{(2)}, ..., s^{(n)}\}$$

where each element in series S is a value such that $s^{(t)} = f(x_1^{(t)}, x_2^{(t)}, ..., x_m^{(t)})$, with $x_1^{(t)}, x_2^{(t)}, ..., x_m^{(t)}$ being our 4S factors, the function f defined by a specific model. Also, note the model's goal of forecast shifts in momentum. In this scenario, we think of using **multivariate time series** models in the field of machine learning [11]. Finally, we choose the Long-Term Short Memory Recurrent Neural Network (LSTM), considering this model's capability of to handle multivariate series data [12].

5.3.1 LSTM Structure

The LSTM unit comprises three primary **gates**: the input gate, the forget gate, and the output gate, alongside two **states**: the cell state and the hidden state. To explain what these mean, imagine LSTM as a kind of smart agent that watches a sequence of events (momentum changes in a tennis match in our case) unfold over time. This agent has a special notebook (**the cell state**) where it can jot down important information to remember and also erase things that are no longer relevant. Alongside, it has a certain level of focus or attention (**the hidden state**) that it adjusts based on the latest events and what it has decided to keep in its notebook. These decisions are made by three specialized mechanisms, often referred to as gates. **The Forget Gate** decides what from its notebook is no longer useful and can be erased. It looks at the new information and the current state of focus and decides what memories to let go of. **The Input Gate** helps the agent decide which of the *new* information is important enough to note down in its notebook. After updating its notebook and deciding what to focus on, **the Output Gate** uses this gate to determine what it should pay attention to now. This is based on its updated notebook and the new information, guiding its current state of focus or attention. This process is illustrated in Figure 11.

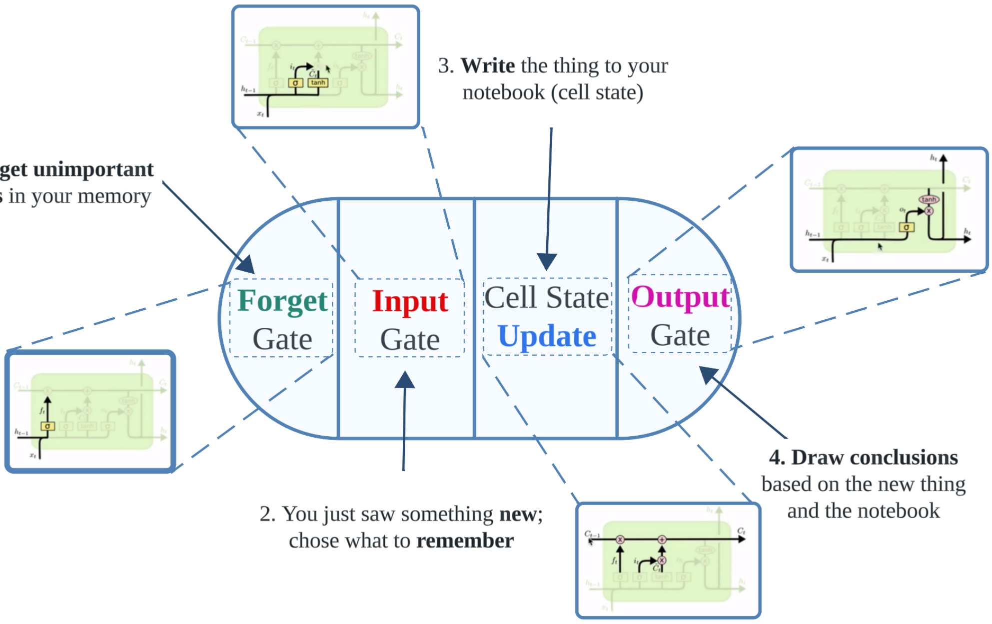


Figure 11: Simplified Description of the Process within a LSTM cell [13]

In our MoQ model, we successfully integrated the 4S factors (Scoring, Service, Stamina, Slip) with the LSTM's structure. Each point in a match is represented as a multi-dimensional vector, where each dimension corresponds to one of the 4S factors. These vectors form a time series, serving as the input to the LSTM model:

$$S = \{s^{(1)}, s^{(2)}, ..., s^{(n)}\}$$

where $s^{(t)} = LSTM(x_1^{(t)}, x_2^{(t)}, ..., x_m^{(t)})$ encapsulates the 4S factors at point t, with $x_1^{(t)}, x_2^{(t)}, ..., x_m^{(t)}$ representing the individual metrics within those factors, and $LSTM$ symbolizing the LSTM's complex function that maps these inputs to an output. This complex function can be mathematically described as the following sequence (2), with the **bold** symbols representing trainable parameters:

1. Forget Gate: $f_t = \sigma(\mathbf{W_f} \cdot [h_{t-1}, x^{(t)}] + \mathbf{b_f})$
2. Input Gate: $i_t = \sigma(\mathbf{W_i} \cdot [h_{t-1}, x^{(t)}] + \mathbf{b_i})$
3. Cell State Update: $\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x^{(t)}] + b_C)$
4. Final Cell State: $C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$

5. Output Gate: $o_t = \sigma(\mathbf{W_o} \cdot [h_{t-1}, x^{(t)}] + \mathbf{b_o})$

6. Hidden State: $h_t = o_t * \tanh(C_t)$

where σ denotes the sigmoid function (described at (2)), $*$ represents element-wise multiplication, the square brackets in $[h_{t-1}, x^{(t)}]$ means concatenation (taking the two vectors and combining them into a single, longer vector), $\mathbf{W}$ and $\mathbf{b}$ are the weight and bias vectors used in each respective gate, h_{t-1} is the previous hidden state, $x^{(t)}$ is the input vector at time t, C_t is the current cell state, and the hyperbolic tangent function ($\tanh$) is given by:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

The LSTM structure above sets the foundation for our predictive momentum model. Now given a set a time-series 4S factors, we will attempt to train this model to meet the performance metric as described above.

5.3.2 Model Training

The training of the MoQ model requires structuring the data from the 30 matches (excluding the final match) into a format suitable for time-series analysis. First, each data point consists of the 4S factors for one player at a specific point in the match. The model used only the 4S factors of one of the players in the match, because we do not want the same performance metric to be repeatedly trained on different factors for different players. Then, each match is treated as a separate sequence for the LSTM to process. Within each match, the model iterates through the points (time steps) in sequential order, learning the patterns and dynamics of momentum as the match unfold.

The training process adjusts the LSTM's parameters - the weights ($\mathbf{W}$) and biases ($\mathbf{b}$) of different gates, to minimize the distance between the model's predictions and the actual outcomes defined by our performance metric $M^{(t)}$. This is accomplished through forward propagation of inputs (the 4S factors at a data point) through the model to generate momentum predictions, and then calculating the difference between these predictions and the actual momentum measured by the performance metric, and then backpropagating this error to adjust the model parameters.

Forward Propagation After processing the input at (1) to form a time-series, for each iteration, At the start of a match, the LSTM's hidden state h_0 and cell state C_0 are initialized to small random values. The model then processes the match point-by-point. For each point t, the 4S factors at that time step $s^{(t)}$ are fed into the LSTM as input. The LSTM updates its cell state based on this input and its previous states, producing an output that corresponds to the predicted momentum at that point, as described on (3).

Back Propagation The model's predictions produced at the forward propagation phase are compared to the actual momentum metrics $M^{(t)}$ to calculate the loss, using a specific loss function. This loss is then backpropagated through the model to adjust the parameters, improving the model's predictions for the next iteration (point) within the match. We used the default optimizer and loss function of the TensorFlow framework. The detailed description for their implementation of LSTM can be found in [14].

So far we have thoroughly described the model structure and training process of MoQ. But is the model any good? Can it truly model the momentum? We will see them in the Experiment section below.

6 Experiment

In this section, we provide a validation of the MoQ model, testing its accuracy and provide a correlation analysis to show the model's capability to quantify momentum; moreover, we use the result of the model to predict flow of momentum, and see what can lead to the change in flow of play.

6.1 Model's Result on Quantifying Momentum

The testing of MoQ involves predicting all 322 points (excluding the first 12 points) for the Wimbledon 2023 Men's Final, on Carlos Alcaraz (Player 1). We compared the LSTM's prediction results of momentum, to the actual performance metric that we designated. The result is illustrated in Figure 12.

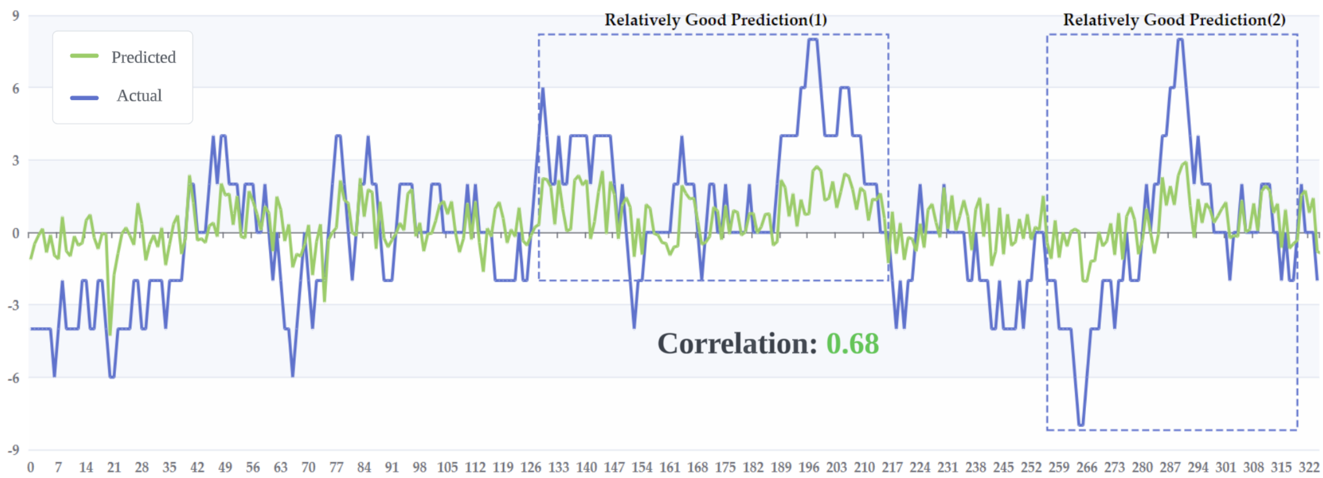


Figure 12: MoQ Prediction Result on the Final Match (excluding the first 12 points)

From the graph, we can judge that the model has a good predictive power of capturing when there's a significant shift of momentum, for example, the "Relatively Good Prediction" areas marked by the dashed lines. The model is mostly right in capturing the swings in momentum, although the momentum's volume between the prediction and actual is still large, as depicted in the figure. This could either due to the limitation of time-series analysis for being bad at predicting extreme values [15], or the unpredictability nature of momentum.

As mentioned in problem analysis, to measure whether this model is effective at predicting momentum, we have to conduct a correlation analysis between the performance metric and the predicted momentum value. The Pearson correlation coefficient, denoted as r, providing a measure of the linear correlation between these two sets of data.

The Pearson correlation coefficient is defined as:

$$r = \frac{\sum_{i=1}^{n}(s^{(t)} - \bar{s})(M^{(t)} - \bar{M})}{\sqrt{\sum_{i=1}^{n}(s^{(t)} - \bar{P})^2 \sum_{i=1}^{n}(M^{(t)} - \bar{M})^2}}$$

where $s^{(t)}$ i is predicted momentum values at point t, $M^{(t)}$ represents the actual momentum metrics at point t, $\bar{s}$ and $\bar{M}$ are the means of the predicted values and actual metrics, respectively, and n is the total number of points analyzed (322). After calculation, we found that the correlation between the performance metric and the result of the momentum model is exceptionally high: $\mathbf{0.68}$. This is out of our expectation because this means our model aligns well with the observable changes in the performance metric, meaning it has been successful at modeling momentum to some level, only using the limited data provided by the original problem! Most importantly, it indicates momentum can be a *real thing* that affects the ebb and flow of the match!

6.2 Are Shifts in Momentum Random?

Now that we have a model that quantifies momentum, is there any indicator in the match that can predict a shift of momentum? To answer this question, we have to first understand what is "a shift of momentum". Since our model is trained on using the average momentum of the 12 points after the predicting point, if at this point the value is significantly different than the last point's value, we can say that this point could potentially cause a shift of momentum and it will be reflected in the subsequent 12 points. Specifically, we can define a shift of momentum at point t if the absolute difference in the momentum values between point t and the previous point $t-1$ exceeds a certain threshold M_{shift}. This can be expressed as:

$$\Delta M^{(t)} = |M^{(t)} - M^{(t-1)}| > M_{\text{shift}}$$

where $M^{(t)}$ is the calculated momentum value using MoQ at point t, $M^{(t-1)}$ is the momentum value at point $t-1$, M_{shift} is a predefined threshold that quantifies what constitutes a "significant" change in momentum. The choice of M_{shift} is important, which must be large enough to filter out minor fluctuations that are normal. Figure 13 showed a distribution of ΔM:

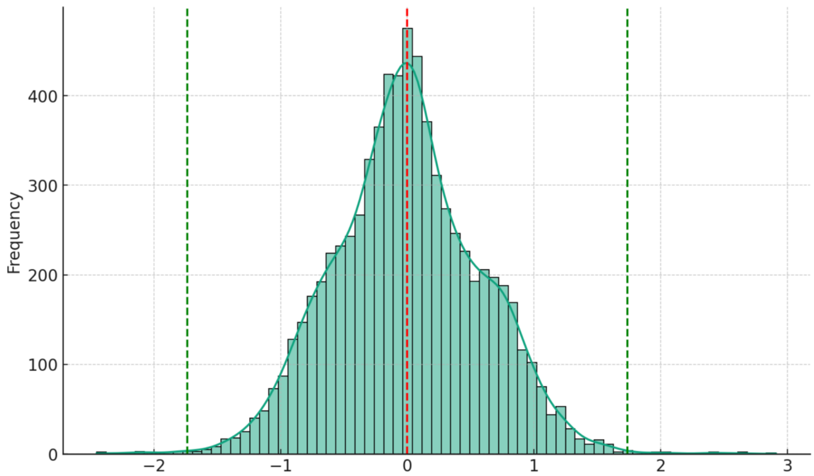


Figure 13: Distribution of value ΔM

Initially it looked like a normal distribution, so we used a normal distribution test like Shapiro-Wilk's test (the detail of this testing can be found in [16]). The calculation result, p-value $=$ 0.00031, clearly rejected the null hypothesis of normality. With this result, we can already conclude that the shifts in momentum are not random. However, we still have to define the threshold ΔM to count for the data points that lead to a change in momentum. In statistics, the common way to set up such threshold is using the formula below:

$$M_{\text{shift}} = k \times \sigma_{\Delta M}$$

where M_{shift} is the threshold for defining a significant shift in momentum, k is the multiplier that determines how stringent the threshold is, and $\sigma_{\Delta M}$ is the standard deviation of the ΔM values, representing the typical deviation from the average change in momentum. When assuming that distribution is normal, $k = 2$ is often selected to represent approximately the 95th percentile of a distribution. In our model where ΔM does not follow a normal distribution but resembles the shape, we can still use $k = 2$ to count the most significant shifts in momentum.

By applying this method to the whole dataset, we calculated $M_{\text{shift}} = 1.25698$, and accounted for a total of 694 points. This means when using our MoQ model to predict momentum, if the change of momentum $\Delta M^{(t)}$ at a timestep t is larger than 1.25698, a swing in the match might be happening.

Finally, using this 694 points, we try to filter the most important factors from the 4S factor that can cause a shift in momentum. This is done by using the correlation analysis, as defined in (), between the value of momentum change ΔM and the value of each of these factors in the 4S factors set. The result is described in the Figure 14.

We can interpret this result as the following:

- For the scoring factor, there is a positive correlation of 0.42, indicating the prediction results of our CourtSense model is moderately related to shifts in momentum. In other words, winning the current point is can be associated with a shift in momentum, but not every point winning leads to a momentum shift.
- The service factor has a moderate positive correlation of 0.49 with shifts in momentum. This suggests that factors related to service are moderately associated with momentum shifts. Specifically, aces having a correlation of 0.20 with momentum shifts, meaning scoring an ace has a little chance to cause a momentum shift.
- The stamina factor in total can have a good correlation with the shifts in momentum, but for individual factors, only the dist_run (the total distance covered when playing that point) and rally count shows a weak negative correlation with the momentum shifts. This could be due to the fatigue during the match; hence we encourage the tennis coaches to focus on some amount of stamina training.
- The slip factor is pretty important in influencing the match's flow. We can see unforced errors have a moderate negative correlation of -0.42 with momentum shifts. Although not quite indicative of the player's performance, making unforced errors may be able to cause a psychological effect. Therefore, make sure to avoid unforced errors during the play of tennis.

Nonetheless, we cannot find a very important factor in the 4S factors correlations that has a high correlation with shifts in momentum. We believe this is due to the nature and complexity of momentum. Being able to find some moderate correlations, we believe, is good enough to provide insights to the help determine the changes in momentum.

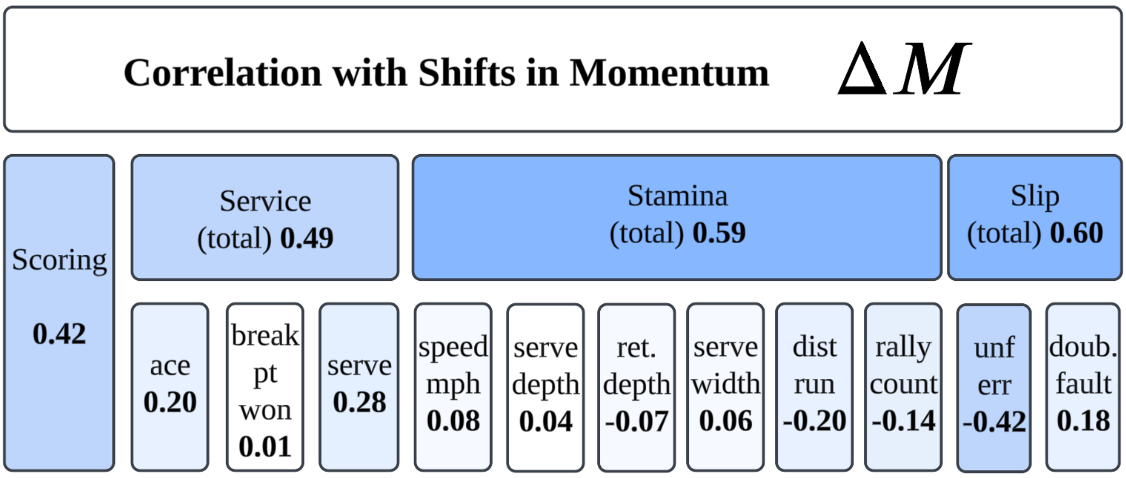


Figure 14: Correlations between Change in Momentum with different 4S Factors. The total value is obtained by simple addition of the absolute values of individual factors in hope to capture the combined effect on momentum.

6.3 An endeavor to further improve MoQ

As mentioned, the model excels at capturing the flow of momentum, but the specific value between the prediction and actual is still not very good. Analyzing from Figure 12, we can tell this is especially true for extreme points. If the extreme points exhibit unique or complex patterns that deviate significantly from other data, the performance of our model is limited. Table 4 shows the points of most extreme differences between predicted and actual momentum value across the whole dataset.

Table 4: Deviation of Predicted Momentum at Extreme Values

Actual	**Predicted**
12	1.574508666
12	1.381220818
-10	-0.544439316
-10	-1.485133172
-10	-0.986423492

We hope to incorporate more factors into the model to diminish this effect. We investigated a broader scale of studies and papers [18][19] on the factors that affect the performance of athletes in competitions, not just tennis, and obtained the following conclusions.

First, it was found that there is no correlation between competition ranking and the actual performance, but a significant correlation between competitive strength (a specific metric for athlete's ability) and actual performance. Second, in multiple rounds of competition, it was found that different rounds have a significant impact on the performance of athletes. For example, there will be significant differences in the number of aces and double errors of athletes in the 1/16, 1/8, 1/4, and final rounds, for example, this table below showed the average number of aces in a match for different rounds:

Table 5: Different Rounds can Impact Player Performance, Exampled by the Number of Aces

ace_count_average	**Round**
10.0625	1/16
14	1/8
13	1/4
9	Semi-Final

To further improve the accuracy of the model, we added the factor of the Round in the game into the 4S factors as an independent factor, transformed them into normalized values, and retrained the model using the same process. The result shows that adding the Round factor of the number of rounds played in the competition has improved the prediction of some extreme points, enhancing the accuracy of the model, depicted in Figure 15; however, as we noticed, this scrambled some other predictions as well.

In addition, we also focused on psychological factors while researching on data of athletes during competitive competitions. Relevant studies suggest that the performance of athletes is greatly influenced by their concentration, stress, potential energy level, psychological resilience, and emotional control on the field. At the same time, the growth and competition experiences of each athlete should be collected, and detailed analysis should be conducted based on the athlete's own technical characteristics. These aforementioned factors could serve as our model's future directions.

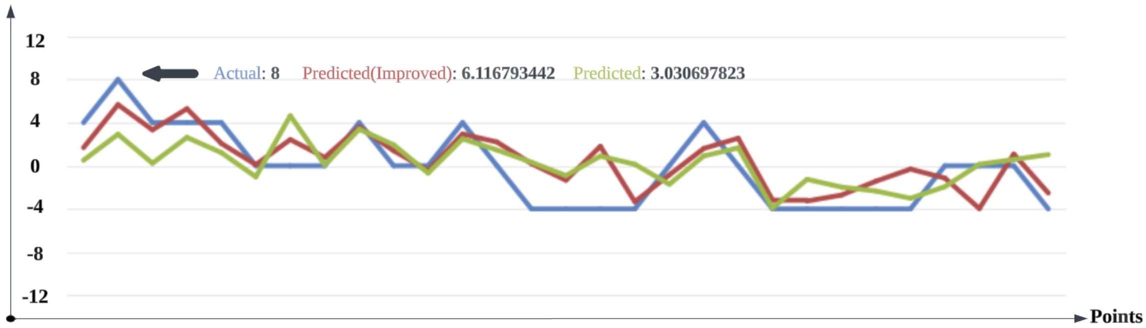


Figure 15: Comparison between Actual and MoQ model prediction result without & without incorporating the Round factor, at the First Match of the 16/1 Round.

6.4 On other matches: Generalizability Analysis

It is not easy to find other match data that has a same set of features. The closest data we can obtain is the Tennis ATP dataset from Kaggle[17], which records every serve and rally between two tennis players, Novak Djokovic and Rafael Nadal, during the 2019 Australian Open Final (ATP). After analyzing the dataset, the directly extractable indicators are "server" (which side is serving) and "rally_count". After categorizing the "reason" column in the dataset, we systematically extracted indicators such as "p1_ace," "p1_break_pt_won," and "p1_unf_err" for model training. Additionally, we incorporate the newly discovered factor of match rounds. In total, seven indicators are extracted.

Upon re-training the MoQ model with these seven indicators, including the novel match rounds factor, the model shows an ability to adapt its predictions to the unique dynamics of this match. The result can be seen in Figure 16. As we can see from the picture, the model achieves similar performance, even with the absence of some original factors. Notably, the model is now a little more aggressive at extreme points. We conclude that the model exhibit a certain level of Generalizability and Adaptability.

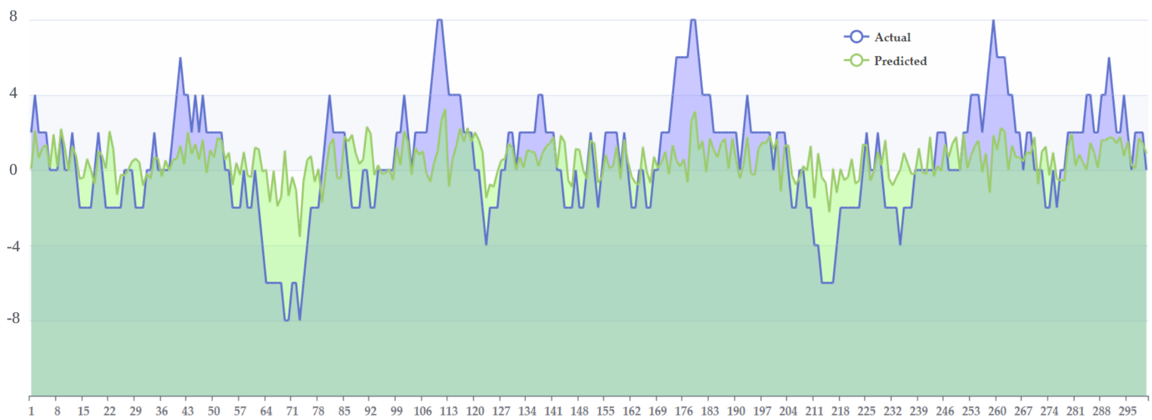


Figure 16: Prediction Result of Adapted MoQ on ATP 2019 Men's Final

7 Strength and Weakness

Our Momentum Quantifier (MoQ) model excels at some aspects in the context of predicting momentum flow in tennis matches:

- **Meticulous Feature Selection**: The temporal features and non-temporal features selected during the feature engineering of CourtSense greatly complemented the model's ability; the MoQ model benefits from a carefully curated set of 4S factors including the scoring factor derived from the CourtSense model. This meticulous selection process ensures our model captures and comprehensively utilize a wide array of factors influencing momentum.
- **Works as Match Flows**: By incorporating a sliding window approach and a forgetting factor in the CourtSense, along with the use of Long Short-Term Memory (LSTM) networks in MoQ, the model adeptly focuses on the most relevant data points. This temporal sensitivity allows the MoQ model to adapt to the dynamic nature of momentum, capturing shifts as they occur in real-time.
- **Credibility**: The underlying CourtSense model has demonstrated high predictive accuracy, with over 90% accuracy in point outcome predictions; the MoQ model exhibits a moderate correlation score with the designated performance metric, underscoring its capability to meaningfully quantify momentum shifts in line with observed match dynamics. Credibility of the model: our CourtSense model achieved an accuracy of over 90% in predicting the point outcome and the correlation between the performance metric and the MoQ model had a moderate correlation score.

Despite its strengths, we deeply acknowledge that the MoQ model faces certain limitations that highlight areas for potential improvement:

- **Constrained by Data**: One of the model's primary constraints lies in the low dimensionality of the available data, which only consists of explicit match statistics. This limitation means we have to overlook the significant psychological factors and other nuanced elements that can also influence momentum, such as crowd support and and mental resilience.
- **Limited Understanding of Momentum**: The model's understanding of momentum is inherently tied to the specific performance metric chosen. This design choice, while necessary for an operational model, inherently narrow the scope of momentum's manifestation, excluding other possible expressions and dimensions of this complex phenomenon.

8 Conclusion

In this paper, we introduced the Dynamic Momentum Quantifier (MoQ) model, a novel approach to quantifying and understanding momentum in tennis matches. Through meticulous feature selection, incorporating advanced modeling techniques such as LSTM networks, building up a specific CourtSense model to that can predict the outcome of a point during match, and using the innovative use of a sliding window and forgetting factors, the MoQ model is a sophisticated tool for capturing the elusive concept of momentum in tennis.

References

[1] Andrew Hamilton, Psychological Aides Sector, Sports Performance Bulletin. The Role of Momentum in Sports Performance.

[2] Skysports | Wimbledon: Carlos Alcaraz dismantles Daniil Medvedev to set up final against Novak Djokovic.

[3] Medium | Predicting ATP Tennis Match Outcomes Using Serving Statistics.

[4] Avinash Dixit, Susan Skeath. The Most Important Situations in Tennis – and in R&D Competition.

[5] Pual K. Newton, Joseph B. Keller. Probability of Winning at Tennis I. Theory and Data.

[6] JS Cramer, University of Amsterdam, 2022. The Origins of Logistic Regression.

[7] Analytics Vidhya | Metrics to Evaluate your Classification Model.

[8] A FINAL FOR THE AGES | Carlos Alcaraz vs Novak Djokovic Full Match | Wimbledon 2023

[9] Helmut Dietl, Cornel Nesseler, University of Zurich, 2017. Momentum in tennis: Controlling the match

[10] Min Kyu Sim, Dong Gu Choi, Pohang University of Science and Technology, 2019. The Winning Probability of a Game and the Importance of Points in Tennis Matches

[11] John Gamboa, University of Kaiserslautern, 2017. Deep Learning for Time-Series Analysis. arXiv:1701.01887v1

[12] Reza Ghanbari, Keivan Borna, IEEE, 2021. Multivariate Time-Series Prediction Using LSTM Neural Networks.

[13] Analytics Vidhya | Shipra Saxena, What is LSTM? Introduction to Long Short-Term Memory

[14] Github | keras/keras/layers/rnn/lstm.py at v2.15.0 · keras-team/keras

[15] Han Lin Shang, Macquarie University; Ruofan Xu, Monash University, 2020. Functional time series forecasting of extreme values. arXiv:2012.10763v1

[16] Wikipedia | Shapiro–Wilk test

[17] Kaggle | Tennis ATP Tour Australian Open Final 2019

[18] Zhu Mingquan, School of Physical Education, China University of Geosciences (Wuhan). Stability of World Excellent Men's Speed Climbing Performance

[19] Exploring the Characteristics and Trends of Competitive Performance in Multi round and Multi round Racing Programs (Athletics Thesis)

Memorandum

To: Coaching Staff
From: Team 2412704
Date: Feb 5th, 2024
Subject: Insights on Momentum in Tennis Matches and Strategies for Player Preparation

Our recent analysis using the Momentum Quantification (MoQ) model has provided valuable insights into the concept of momentum in tennis and its impact on the flow of play. This memo summarizes our findings and offers advice on preparing players to respond effectively to events that influence momentum during a match.

The MoQ model's testing during the Wimbledon 2023 Men's Final revealed its adeptness in capturing significant momentum shifts, correlating well with observable performance changes. Notably, momentum shifts are not random but can be anticipated through certain indicators, allowing for strategic interventions.

In our analysis of momentum shifts, unforced errors were notably detrimental, indicating that minimizing mistakes is crucial for maintaining positive momentum. Scoring emerged as a less important factor, with a moderate to low correlation between winning some point and momentum shifts. This means the immediate scoring of a point is not a clear indicator; momentum is more pronounced in the long run. Physical stamina and endurance also played roles, suggesting that players' physical preparation can impact their ability to maintain or shift momentum. Service factors like aces has probability to cause a swing as well, so training on the quality of serves would prove beneficial, too. Below is a more detailed approach:

Emphasize the importance of scoring and converting opportunities. Every point won can contribute to a favorable momentum shift, making it essential to capitalize on scoring chances. **Incorporate stamina and endurance training.** Preparing players for the physical demands of a match can help mitigate the impact of fatigue on momentum. **Focus on minimizing unforced errors.** Reducing mistakes is critical for maintaining momentum and can prevent giving the opponent psychological and strategic advantages. **Prepare players mentally for momentum shifts.** Mental resilience and adaptability can help players navigate the highs and lows of a match, enabling them to respond effectively to changing dynamics.

We have also developed a sub-model named CourtSense, which complements our main MoQ model. CourtSense is specifically designed to provide point-specific insights into player behaviors and strategies during key moments of the match, particularly those that influence momentum shifts. This model has reached a very high accuracy, therefore we suggest considering using our mathematical modeling approach to enhance the understanding of your players.

In conclusion, the combination of MoQ and CourtSense models offers a comprehensive toolkit for analyzing and influencing momentum in tennis. Although the model still has weakness for not incorporating the psychological side of the player, we believe we have laid the groundwork for future analysis into player momentum.

Report On the Use of AI

1. **OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)**

   Query1:

   Why is the LSTM model insensitive to extreme point processing?

   Output:

   Long ShortTerm Memory (LSTM) is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem, which often occurs in traditional RNNs. LSTMs are capable of learning and remembering longterm dependencies in sequential data, making them wellsuited for tasks such as natural language processing, time series prediction, and speech recognition.
   When you mention "extreme point processing," it's not entirely clear what specific aspect you're referring to. If you mean extreme values or outliers in the input data, LSTMs, like many neural network models, can be somewhat resilient to them. Here are a few reasons why LSTMs may exhibit some insensitivity to extreme points:
   1. Sigmoid and tanh activation functions: LSTMs typically use activation functions such as sigmoid and hyperbolic tangent (tanh), which are bounded between 0 and 1 or 1 and 1, respectively. These functions help prevent the activation values from exploding or vanishing during training. As a result, extreme input values may be scaled to fit within these bounds, reducing their impact on the model.
   2. Memory cell gating: LSTMs have a memory cell and a set of gates (input, forget, and output gates) that control the flow of information into and out of the cell. The gates act as filters, allowing the network to decide which information to remember, forget, or output. This gating mechanism helps the LSTM selectively retain relevant information while ignoring less important or noisy signals, which could include extreme values.
   3. Robust learning: LSTMs are designed to learn and remember patterns in sequential data over long distances. This inherent ability to capture temporal dependencies might contribute to their robustness in handling extreme points as they learn to distinguish between relevant patterns and outliers.
   However, it's essential to note that while LSTMs may exhibit some resilience to extreme points, they are not immune to issues caused by outliers. The degree of insensitivity can vary based on the specific characteristics of the data and the complexity of the model. Additionally, preprocessing steps, such as scaling or outlier handling, are often applied to improve the model's performance and robustness.

2. **OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)**

   Query2: Can you give me some factors that influence the performance of tennis players?

   Output:

The performance of tennis players is influenced by a combination of physical, mental, technical, and strategic factors. Here are some key factors that can impact a tennis player's performance:

1. Physical Fitness:

Endurance: Tennis matches can be physically demanding, requiring players to have good cardiovascular endurance to sustain highintensity efforts over long periods.

Speed and Agility: Quick lateral movements, rapid sprints, and agility are crucial for effective court coverage and quick responses to opponents' shots.

Strength: Both upper and lower body strength contribute to powerful serves, groundstrokes, and effective net play.

2. Technical Skills:

Stroke Technique: Proficient forehands, backhands, volleys, and serves are fundamental for success in tennis. A player's ability to execute these strokes with consistency and accuracy is critical.

Footwork: Efficient footwork helps players move quickly around the court, enabling them to reach balls in optimal positions and maintain balance during shots.

3. Mental Toughness:

Focus and Concentration: Tennis requires intense concentration, and maintaining focus during long matches is essential. Mental lapses can lead to errors and impact overall performance.

Resilience: The ability to bounce back from setbacks, stay composed under pressure, and handle the emotional ups and downs of a match is crucial for success.

4. Strategy and Tactics:

Shot Selection: Knowing when to play aggressive shots, defensive shots, or approach the net requires strategic thinking.

Court Awareness: Understanding the opponent's strengths and weaknesses, as well as adapting to changing match conditions, is key to gaining a competitive edge.

5. Experience and Match Play:

Tournament Experience: Players often benefit from participating in various tournaments, gaining exposure to different playing styles, and learning to handle the pressures of competitive environments.

Match Management: Effectively managing energy levels, pacing oneself throughout a match, and making strategic decisions during critical points contribute to overall performance.

6. Equipment and Technology:

Racquet and String Setup: Players often have preferences for specific racquets and string tensions that suit their playing style.

Use of Technology: Video analysis, data tracking, and other technological tools can help players and coaches assess performance, identify areas for improvement, and make datadriven decisions.

7. Nutrition and Recovery:

Proper Nutrition: A wellbalanced diet is essential for maintaining energy levels, enhancing recovery, and supporting overall health.

Rest and Recovery: Adequate rest, sleep, and recovery strategies are crucial for preventing injuries and ensuring peak performance during matches.

8. Coaching and Support Team:

Quality Coaching: Having a knowledgeable and supportive coaching team can provide players with guidance on technical, tactical, and mental aspects of the game.

Physiotherapy and Medical Support: Access to physiotherapists and medical professionals helps players manage injuries and maintain their physical wellbeing.

Each player is unique, and the interplay of these factors varies from person to person. Successful tennis players often possess a wellrounded combination of these attributes and continually work to improve and adapt throughout their careers.

3. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query3: If given a column of natural number data, how can you determine at which point the trend of change has changed and provide the most common threshold setting method.

Output:

The common threshold setting method is to determine the threshold based on the standard deviation of the data, that is, to set the threshold as a multiple of the standard deviation of the data. This method can dynamically determine the threshold based on the distribution and range of changes of the data. The following are the specific steps:

Firstly, calculate the standard deviation of the data column. In Excel, the STDEV function can be used to calculate the standard deviation. For example, if the data column is stored in column A, you can enter=STDEV (A1: A100) in another cell to calculate the standard deviation of the data in the range of A1 to A100.

Determine the multiplier of the threshold. Common threshold multiples include 1 standard deviation, 1.96 standard deviation (corresponding to 95% confidence level), 2 standard deviation, etc.

Calculate the threshold. In another cell, use the formula of multiplying the standard deviation by the standard deviation to calculate the threshold. For example, if the threshold multiple is set to 2 times the standard deviation, the calculation formula is=2 STDEV (A1: A100).

Using the calculated threshold as the threshold for determining the turning point, when the difference between adjacent data is greater than this threshold, it is considered that a turning point has occurred.

Using multiples of standard deviation to set thresholds is a common and universal method that can be adjusted according to different situations and distributions of data, making it more flexible and accurate.

4. **OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)**

Query4: show me the structure of LSTM

Output:

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture designed to address the vanishing gradient problem in traditional RNNs, allowing for the learning of long-term dependencies in sequential data. The structure of an LSTM unit involves several components, including memory cells and various gates. Here is a simplified representation of the LSTM architecture:

1. Memory Cell (C):
   - The memory cell is a key component that stores information over time.
   - It runs parallel to the rest of the LSTM structure and interacts with the gates.
2. Input Gate (i):
   - Determines which information from the input should be stored in the memory cell.
   - Computed using the sigmoid activation function.
3. Forget Gate (f):
   - Decides which information from the memory cell should be discarded or forgotten.
   - Computed using the sigmoid activation function.
4. Cell State Update:
   - Calculates the new values to be added to the memory cell.
   - Involves a tanh activation function that produces candidate values for the cell state.
5. Cell State (C_t):
   - The updated memory cell state after considering input, forget, and cell state update operations.
6. Output Gate (o):
   - Determines what information from the memory cell should be output to the next time step.
   - Computed using the sigmoid activation function.

7. Hidden State (h_t):

- The output of the LSTM unit, which is a filtered version of the memory cell state.
- Calculated by applying the output gate to the cell state using the hyperbolic tangent (tanh) activation function.

In mathematical terms, the operations in an LSTM unit can be summarized as follows:

- Input Gate: (i_t = sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}))
- Forget Gate: (f_t = sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}))
- Cell State Update: (widetilde{C}_t = tanh(W_{ic}x_t + b_{ic} + W_{hc}h_{t-1} + b_{hc}))
- New Cell State: (C_t = f_t odot C_{t-1} + i_t odot widetilde{C}_t)
- Output Gate: (o_t = sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}))
- New Hidden State: (h_t = o_t odot tanh(C_t))

In these equations:

- (sigma) represents the sigmoid activation function.
- (odot) denotes element-wise multiplication.
- (W) and (b) are weight matrices and bias vectors, respectively, for the various gates.

The LSTM architecture allows for the learning of long-term dependencies by selectively updating and retaining information in the memory cell, making it well-suited for sequential data processing tasks.

5. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query5: I want to obtain detailed data on each game of Wimbledon or other tennis matches in the past few years, and which websites should I go to to obtain it.

Output:

If you want to obtain detailed data on each game of Wimbledon or other tennis matches in the past few years, the following are some websites you can try:

1.ATP Official Website (www.attour.com) - This is the official website of the Men's Professional Tennis Association, providing data, scores, and statistical information for various competitions.

2.WTA Official Website (www.wtatennis.com) - This is the official website of the Women's Professional Tennis Association, providing data, scores, and statistical information on women's events.

3.The official website of the International Tennis Federation (ITF) (www.itftennis.com) - The ITF website provides data and information on various international tennis competitions, including Wimbledon.

4.FlashScore (www.flashscore.com) - This is a global sports score website that provides real-time scores and detailed statistical data for various tennis matches.

5.Tennis.com (www.tennis.com) - This is a website dedicated to tennis, providing news reports, scores, and statistics on various events.

I hope the above information can help you obtain the tennis match data you need!

6. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query6: use mathematical language to describe this process: designate a metric such as game score, and build a model of momentum, and when this metric correlates high with the model, we can infer the presense and the impact of momentum

Output:

To formalize the process of modeling momentum in a mathematical framework, we can start by designating a quantifiable metric, \(M \), which could represent aspects such as game score, point differentials, or other performance indicators in sports. This metric is chosen as a proxy to represent momentum shifts within a game.

Next, we define a set of variables \(\{X_1, X_2, ..., X_n\} \) that are believed to influence or reflect the state of momentum. These variables could include, for example, the rate of unforced errors, first serve success rate, number of winners, break points converted, etc. Each of these variables represents a factor that could either contribute to gaining momentum or indicate that a shift in momentum has occurred.

We propose a predictive model \(f \) that attempts to estimate the momentum metric \(M \) based on these variables:

\[M = f(X_1, X_2, ..., X_n) \]

The function \(f \) could be a linear model, a complex nonlinear model, or even a neural network, depending on the complexity of the relationships between the variables and the momentum metric. The choice of model would depend on the nature of the data and the specific dynamics of the sport being analyzed.

To evaluate the model's effectiveness in capturing the concept of momentum, we analyze the correlation between the predicted momentum metric \(\hat{M} \) and the observed outcomes or performance metrics in the game. This can be done using statistical measures such as the Pearson correlation coefficient, denoted by \(r \), where a higher \(|r| \) value indicates a stronger relationship:

\[r = \frac{\sum (M_i - \bar{M})(\hat{M}_i - \bar{\hat{M}})}{\sqrt{\sum (M_i - \bar{M})^2 \sum (\hat{M}_i - \bar{\hat{M}})^2}} \]

Where:
- \(M_i \) is the observed momentum metric for the \(i \)-th observation.
- \(\hat{M}_i \) is the predicted momentum metric for the \(i \)-th observation.
- \(\bar{M} \) is the mean of the observed momentum metrics.
- \(\bar{\hat{M}} \) is the mean of the predicted momentum metrics.

A high correlation coefficient between \(M \) and \(\hat{M} \) would suggest that the model is effective at capturing the dynamics associated with momentum. This would allow us to infer the presence and impact of momentum in a game based on the observed values of \(\{X_1, X_2, ..., X_n\} \) and their relationship to game outcomes. It's important to note that this model would likely need to be dynamic, adapting to different contexts and individual player characteristics, and it would require continuous validation and recalibration based on new data to maintain its predictive power.

7. **OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)**

Query7: lstm loss 2.2 is it good?

Output:

When it comes to evaluating the "goodness" of a loss value, such as an LSTM (Long Short-Term Memory) model's loss of 2.2, it largely depends on the context and specifics of the task at hand. LSTMs are a type of recurrent neural network (RNN) particularly adept at handling sequences, such as time series data, natural language processing, and other sequence-related tasks. The loss value is a quantitative measure of how well the model's predictions match the actual target values, with lower values generally indicating better performance. However, the interpretation of the loss value can vary based on several factors:

Nature of the Task: Different tasks have different scales and distributions of the target variable, which can greatly influence what constitutes a "good" loss. For example, in a task predicting stock prices, a loss of 2.2 might be quite good, whereas in a task categorizing text into a few categories, it might be relatively poor.

Loss Function Used: The type of loss function can also impact the interpretation of the value. Common loss functions for LSTMs in different tasks include Mean Squared Error (MSE) for regression tasks, Cross-Entropy Loss for classification tasks, and others like Mean Absolute Error (MAE). Each of these scales differently, and thus their numerical loss values are not directly comparable.

Baseline Comparisons: It's essential to compare the LSTM's loss to a relevant baseline. For instance, what loss does a simple model achieve on the same task? If a simple model or the industry standard model achieves a loss significantly lower than 2.2, then the LSTM's performance might not be considered good. Conversely, if the LSTM outperforms simple models or is close to the state-of-the-art models, a loss of 2.2 could be seen as an achievement.

Training Progress: The loss value should also be viewed in the context of the model's training progress. Early in training, a loss of 2.2 might be excellent, indicating rapid learning. If the loss stagnates at 2.2 after extensive training and tuning, it might suggest that the model has reached its limit given the current architecture, data, and hyperparameters.

Validation and Generalization: A "good" loss on the training set is less meaningful if it doesn't translate to similar performance on unseen data. If your LSTM achieves a training loss of 2.2 but performs significantly worse on the validation set, this could indicate overfitting. The ultimate goal is to minimize the loss on new, unseen data, ensuring the model generalizes well.

Relative Improvement: If previous iterations or models achieved higher loss values on the same task, then a reduction to 2.2 could represent a significant improvement and, in that sense, could be considered "good."

Given these considerations, it's crucial to evaluate the LSTM's loss of 2.2 within the broader context of your specific task, the data at hand, and the performance of alternative models or approaches. It might also be helpful to look into more detailed metrics specific to your task, such as accuracy, precision, recall, F1 score for classification tasks, or more domain-specific evaluations for other types of tasks.

8. **OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)**

Query8: how to do a non-linear correlation analysis between 2 series of data

Output:

Non-linear correlation analysis between two series of data involves exploring the relationship between them without assuming this relationship to be linear. Unlike Pearson's correlation coefficient, which measures the degree of linear relationship between two variables, non-linear methods can capture more complex patterns. Here's how you can approach it:

1. Visual Inspection
Start by visually inspecting the data through scatter plots. This can give you an initial sense of the relationship. Look for patterns that are not straight lines, such as curves, clusters, or other complex shapes.

2. Transformation
Sometimes, a non-linear relationship can be transformed into a linear one through mathematical functions such as logarithms, square roots, or exponentials. Apply transformations to your data and then use Pearson's correlation on the transformed data to assess linearity.

3. Rank Correlation
Methods like Spearman's rank correlation and Kendall's tau are non-parametric and can be more effective at identifying non-linear relationships. They work by ranking the data points and then analyzing the correlation between these ranks, which can capture monotonic relationships.

Spearman's Rank Correlation: It assesses how well the relationship between two variables can be described using a monotonic function. It's based on the ranked values for each variable rather than the raw data.

Kendall's Tau: Similar to Spearman's, Kendall's tau measures the strength of the monotonic relationship between two variables. It compares the number of concordant and discordant pairs of data points.

4. Mutual Information
Mutual Information (MI) is a measure from information theory that quantifies the amount of information obtained about one random variable through another. It is particularly useful for capturing non-linear relationships because it doesn't assume any specific type of relationship between variables.

5. Non-linear Regression Models
Consider using non-linear regression models if you have a hypothesis about the form of the relationship. Models like polynomial regression, logistic regression, or other non-linear models can fit complex relationships between variables. The coefficients and fit of these models can provide insights into the nature of the relationship.

6. Machine Learning Techniques

Advanced machine learning techniques can be particularly effective at uncovering complex relationships in data:

Decision Trees and Random Forests: These can model complex, non-linear relationships through hierarchical, piecewise partitions of the data space.

Neural Networks: Especially useful for capturing complex, non-linear interactions between variables, though they require larger datasets and more computational resources.

7. Distance Correlation

Distance correlation is a measure of association between two random variables that can detect both linear and non-linear associations. Unlike Pearson's correlation, distance correlation is zero only when variables are independent.

Implementation Steps

For most of the statistical methods mentioned (like Spearman's, Kendall's tau, and MI), you can find implementations in statistical software packages and libraries like Python's scipy.stats and sklearn.feature_selection. For machine learning models, libraries like scikit-learn, tensorflow, or pytorch can be utilized.

Example in Python

Here's a brief example of how to calculate Spearman's rank correlation in Python using scipy.stats:

```
python
Copy code
import numpy as np
import scipy.stats as stats

# Sample data series
x = np.random.rand(100)
y = np.random.rand(100)

# Spearman's rank correlation
rho, pval = stats.spearmanr(x, y)

print(f"Spearman's rank correlation: {rho:.3f}, P-value: {pval:.3f}")
```

This code generates two random series of data and computes Spearman's rank correlation along with the p-value, which tests the hypothesis of no correlation.

Remember, choosing the right method depends on your data and the specific characteristics of the relationship you're investigating. It's often useful to combine several methods to get a comprehensive understanding of the data relationship.

9. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query9: write an 1000 words abstract for this paper "Momentum Quantifier (MoQ): Proving the

"Big Mo" is Real in Tennis", using the content below:

\section{Problem Background}

If you are a fan in competitive sports games like tennis, you might have heard a lot of the word “Momentum”. When a player has momentum, they seem to control the dominance of the match, wielding the match's rhythm and flow. Novak Djokovic, a record-breaking tennis player, described this in his saying: “… if you have the mental ability to stay strong, stay patient and confident and just have belief in the right moments, then you get a win, you know.”

However, momentum is not stationary; you probably have witnessed momentum being lost or stolen from a player during a match, entirely changing the outcome. Momentum can switch to the other player within the blink of an eye and stun unsuspecting players and spectators. This is especially spectacular during the 2023 Wimbledon Gentlemen’s final between Carlos Alcaraz and Novak Djokovic (Figure 1). In this match, Djokovic completely dominated the first set with a score of 6 – 1, displaying enormous momentum. However, this quickly shifted to Alcaraz in the third set, where he won 6 - 1, before Djokovic regained control in the fourth set with a 6 - 3 win. Finally, in the late stages of the fifth set, the momentum swung once again, and Alcaraz secured victory with a score of 6 - 4.

But where \textbf{exactly} did that momentum come from? Despite its recognized impact, momentum has remained largely a qualitative factor and rooted in the subjective experience of spectators and self-perception of athletes. The question, then, is whether this intangible force can be analyzed and systematically understood through mathematical methods. If this proves feasible, it has the potential to greatly enhance the strategies used by competitive tennis players and coaches and, by extension, athletes and coaches in various sports.

```
\begin{figure}[htbp]
    \centering
    \begin{minipage}{0.5\textwidth}
    \centering
\includegraphics[width=.95\textwidth]{carlos-alcaraz.jpg}
\caption{Carlos Alcaraz (Left) and Novak Djokovic (Right) at Wimbledon \cite{1}}
\label{fig:Fire Situation}
    \end{minipage}\hfill
    \begin{minipage}{0.45\textwidth}
    \centering
\includegraphics[width=.6\textwidth]{data-given.png}
\caption{Data Aspects Provided by \textit{Wimbledon\_featured\_matches.csv}}
\label{fig:Fire Situation}
    \end{minipage}
\end{figure}

\section{Problem Analysis: What is Momentum?}
```

Momentum is often seen as a phenomenon that a player is playing exceedingly well at a time in the match. And because it its a \textit{phenomenon}, it should be an outcome of a wide range of factors, including psychological states, like confidence and mental toughness, to external conditions like the current state of the match. To analyze and build a robust model that measures the momentum of players in competitive tennis, it is crucial to uncover and model these factors extensively. Specifically, our model should be capable of completing the following tasks:

\begin{enumerate}[\bfseries (1)]

\setlength{\parsep}{0ex}

\setlength{\topsep}{0ex}

\setlength{\itemsep}{0ex}

\item Evaluate the quantitative degree how one player is performing better than the other player at any given point in the match, and illustrate the changes of such performance difference in visual representation.

\item Based on extensive factors, including the flow of match, the model should quantify the concept of "momentum" in tennis matches, and identify potential indicators that could predict shifts in momentum. To address the coach’s skepticism, the model needs to be compared to a null model that assumes outcomes are random (e.g., based on point-by-point win probabilities without momentum).

\item Finally, the developed "momentum" model should be tested on other matches from the different matches, including women's matches, to exhibit its predictive accuracy and generalizability, and identify any limitations or additional factors that may need to be incorporated.

\end{enumerate}

\subsection{Capturing the Flow of Match}

To compare the performance ratings between the two players, we can use the data provided in \textit{Wimbledon_featured_matches.csv} that describe the explicit match information, such as whether a player was serving or at a break point, etc. (shown in Figure 2); but we also need to consider other implicit information that has significant impact on the state of the match, for example, whether a player has a high scoring streak. Other unimportant information, potentially the speed of the serve, needs to be neglected to ensure model's simplicity and robustness. This inherently requires \textbf{feature engineering}: to extract and construct feature variables that capture both explicit and implicit aspects that control the flow of the match.

Since our model aims to dynamically assess the match's state as it evolves, it requires a granular focus on the state of the game, taking into account both minor and major developments. For example, the player may have a lead in games won, but when the other player just scored a break point, that player could be performing much better at that specific time. Therefore, to scope the data to recently played points, the \textbf{sliding window} technique can be employed; to handle older scores and progressively update the model as the match goes on, the introduction of a \textbf{forgetting factor} would be applicable, as illustrated in Figure 3. The remaining work would be finding an appropriate model to map these parameters to the performance or the winning probability of a player.

\begin{figure}[htbp]

\centering

\includegraphics[width=.76\textwidth]{sliding-window.png}

\caption{\centering Information Is Constrained by Sliding Window and Forgetting Factor. (Note that the data in this graph is for understanding the process.)}

\end{figure}

\subsection{How to Describe Momentum and What Affects It}

To build an effective model that can analyze the player's momentum, we must first understand what "momentum" actually means. When we talk about momentum in tennis or any competitive sports, we are painting a picture of a player hitting their stride at just the right moment. The crowd can sense it, their cheers growing louder and more fervent. The players can feel it, their steps lighter, their spirits higher.

\textit{However}, no matter how you would argue, "momentum" stays an elusive and subjective concept and largely a psychological state \cite{-1}; therefore, it is extremely hard to use an completely accurate numerical measurement for it, especially when we do not have available psychological data for these matches. With no measurement method, it is impossible to build a mathematical framework that could analyze the factors affecting the momentum, let alone predicting shifts in momentum as the match is being played. Worse yet, the momentum effect in tennis is highly controversial itself and is influenced by a complex interplay of factors, some measurable and some intangible.

Nonetheless, we argue that there is still a possible way to measure and model momentum. For example, in the final match of Wimbledon 2023, from when Djokovic won the first set with a dominating 6 to 1, to when Alcaraz took control in the third set with a same score but reversed for him, we can confidently say that there had been a shift in momentum. Therefore, while there is no established measurement of momentum, we can define a standard that momentum manifest itself in. In other words, we first designate a performance \textit{metric} that reflects momentum, such as the changes in scoring patterns, and then analyze how changes in this metric correlate with other observable factors and events during a match. This approach allows us to use quantitative data to infer the presence and impact of momentum, even if we cannot measure it directly. This process described in detail in the Figure 4:

```
\begin{figure}[htbp]
    \centering
    \includegraphics[width=0.8\textwidth]{momentum-model.png}
    \caption{\centering Process of Modeling Momentum}
\end{figure}
```

As described in the figure, if the correlation between the built momentum model and the performance metric are high, we can say that the momentum exists, affecting the ebb and flow during the match, and the model that attempts to calculate the momentum has a good credibility. The remaining work to do is to find the suitable parameters for the performance metric and look for factors to be incorporated into the momentum model, perhaps also leveraging the work of our first model that predicts the outcome of points. We note that any model of momentum, especially models that rely on physical match data, has a degree of uncertainty. We seek to capture the probability of shifts of games with our momentum model rather than deterministic outcomes.

If this method proves feasible in quantifying momentum, then other problems, such as predicting momentum shifts and utilizing the model to predict other matches, can be a straightforward process. If not, we will also identify the problems with our model and set a foundation for improving the model.

\subsection{Our Work}

It is our belief that a practical and inclusive mathematical framework to model the concept momentum would greatly help the tennis community in understanding the concept of momentum to improve performance, strategy, and training methodologies. In this paper, we seek to address this challenge using a set of match data from Wimbledon 2023 Men's Matches. Through extensive investigation in the field of tennis and thorough exploration, our main model - the \textbf{Momentum Quantifier (MoQ)} is built, which integrates a comprehensive set of useful features and use a variety of statistical methods to achieve this goal. The building of this model can be seen in Figure 5, or described as follows:

\begin{enumerate}[\bfseries (1)]

\setlength{\parsep}{0ex}

\setlength{\topsep}{0ex}

\setlength{\itemsep}{0ex}

\item To capture the flow of match in real-time, we constructed the CourtSense model as a sub-model of MoQ. The model starts by applying a Sliding Window machanism to a set of extracted features; these features are passed into the core Progressive Logisic Model to predict the winning chance of the point; due to the real-time progression of the competition, a Forgetting Factor is added to dynamically update this model to reflect the most recent match status.

\item To quantify the concept of momentum, we designated a performance metric that can better reflect the flow of momentum, and use this metric to guide the Momentum Quantifier (MoQ)'s learning. At the heart of MoQ is an adapted LSTM model that uses a meticulously selected momentum factors - the 4S factors.

\item To verify the model's effectiveness at capturing momentum, we do a correlation analysis between the performance metric and the model's result, and obtained a promising result. We also defined what a "shift in momentum" means and found out which factors are has the most potential in causing such momentum shifts. Finally, the MoQ undergoes a set of optimization, exhibiting its generalizability, adaptibility, and future directions.

\end{enumerate}

Output:

The concept of "momentum" in competitive sports, particularly tennis, has long intrigued both enthusiasts and professionals. Often discussed in terms of a player's dominance or sudden shifts in the game's rhythm, momentum is frequently cited in spectacular matches, such as the 2023 Wimbledon Gentlemen’s final between Carlos Alcaraz and Novak Djokovic. Despite its palpable presence, momentum has remained largely qualitative, rooted in subjective experience and lacking a systematic, mathematical framework for analysis. This paper presents the Momentum Quantifier (MoQ), a pioneering model designed to bridge this gap, offering a quantitative lens through which to view and understand momentum in tennis.

At the outset, it's crucial to acknowledge the complexity of momentum, which encompasses not only the physical and tactical aspects of the game but also psychological states and external conditions. Our approach to modeling momentum involves a comprehensive analysis of these factors, aiming to quantify how a player is performing relative to their opponent at any given point and to track the changes in this performance differential. Key to this endeavor is feature engineering, which involves extracting and constructing variables that capture both the explicit and implicit elements influencing the match's flow. Techniques such as sliding windows and forgetting factors are employed to focus on recent play, ensuring the model remains dynamic and reflective of the match's current state.

To describe momentum concretely, we propose a performance metric that encapsulates the game's flow, correlating with observable factors and events. This metric serves as a cornerstone for the MoQ model, which, through high correlation with the metric, validates the existence and quantifiable nature of momentum. The model's foundation is the CourtSense sub-model, which predicts point outcomes using real-time match data, adjusted dynamically through a forgetting factor to mirror the latest developments.

Central to MoQ is the innovative use of an adapted Long Short-Term Memory (LSTM) network, leveraging meticulously selected features—the 4S factors—to quantify momentum. These factors are derived from extensive analysis and are instrumental in capturing the nuanced dynamics of tennis matches. The model's effectiveness is evidenced by a promising correlation analysis between the performance metric and the model's outputs, highlighting its capability to capture and predict momentum shifts. Furthermore, we define a "shift in momentum" in quantitative terms, identifying key indicators that may herald these pivotal changes in the game.

The MoQ model's robustness is further validated through application to an external dataset—the 2019 Australian Open Final between Novak Djokovic and Rafael Nadal. Despite differences in available features, the model demonstrates adaptability and maintains performance, underscoring its potential for broader application in tennis and possibly other competitive sports.

This research not only substantiates the quantifiable nature of momentum in tennis but also sets a precedent for future studies. The MoQ model's development, from its theoretical underpinnings to empirical validation, offers a comprehensive framework for analyzing momentum, providing valuable insights for players, coaches, and analysts alike. Its adaptability and predictive accuracy, even in the face of varied datasets, highlight the model's potential for enhancing strategic planning and performance analysis in tennis. Furthermore, the exploration of momentum shifts and their predictors opens new avenues for understanding the game's psychological and tactical dimensions. As the first of its kind, the MoQ model represents a significant advancement in sports analytics, with implications extending beyond tennis to the broader realm of competitive sports.

10. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query10: 5th set.xlsx

Spreadsheet

add 2 new columns about if the player gain a point at that point

Output:

To add two new columns that indicate whether a player gained a point at a particular point during a match, we would typically consider the scoring patterns and the current score for each player. The columns could be named something like "Player1_Gained_Point" and "Player2_Gained_Point," with boolean values (TRUE or FALSE) or numerical values (1 for a gained point, 0 for no point gained).

Given the context, we will assume the data in "5th set.xlsx" includes the scores or outcomes of individual points in a match. I'll start by loading the spreadsheet to understand its structure and contents. Then, I can guide you through or directly implement the addition of these columns based on the data available. Let's begin by examining the file.

The data from the "5th set.xlsx" file appears to include columns that may represent the following:

A match identifier (e.g., "2023-wimbledon-1701").
Player 1's name (e.g., "Carlos Alcaraz").
Player 2's name (e.g., "Novak Djokovic").
A numerical value associated with Player 1, which could represent points, games, or another scoring metric (e.g., 108, 109, ...).
A similar numerical value for Player 2 (e.g., 104).
Based on this structure, to add two new columns indicating whether each player gained a point at a particular instance, we need to calculate the differences in their respective scores between each row. An increase in a player's score from one row to the next would indicate that the player gained a point at that instance.

Let's proceed with adding these columns. We'll denote a gained point with a 1 and no change with a 0 for each player.

I've added two new columns to the dataset:

Player1_Gained_Point: Indicates whether Player 1 (Carlos Alcaraz, in this dataset) gained a point at a particular instance. A value of 1 means a point was gained, and a value of 0 means no point was gained.
Player2_Gained_Point: Indicates whether Player 2 (Novak Djokovic, in this dataset) gained a point at a particular instance, with the same logic as above.
For the first row, both new columns were set to 0 since there's no previous point to compare and determine if a point was gained.

This approach assumes that an increase in the numerical value from one row to the next for a player indicates a point gain. If your data represents a different aspect of scoring or if there are any other specific rules or conditions you'd like to apply, please let me know so we can adjust the logic accordingly.

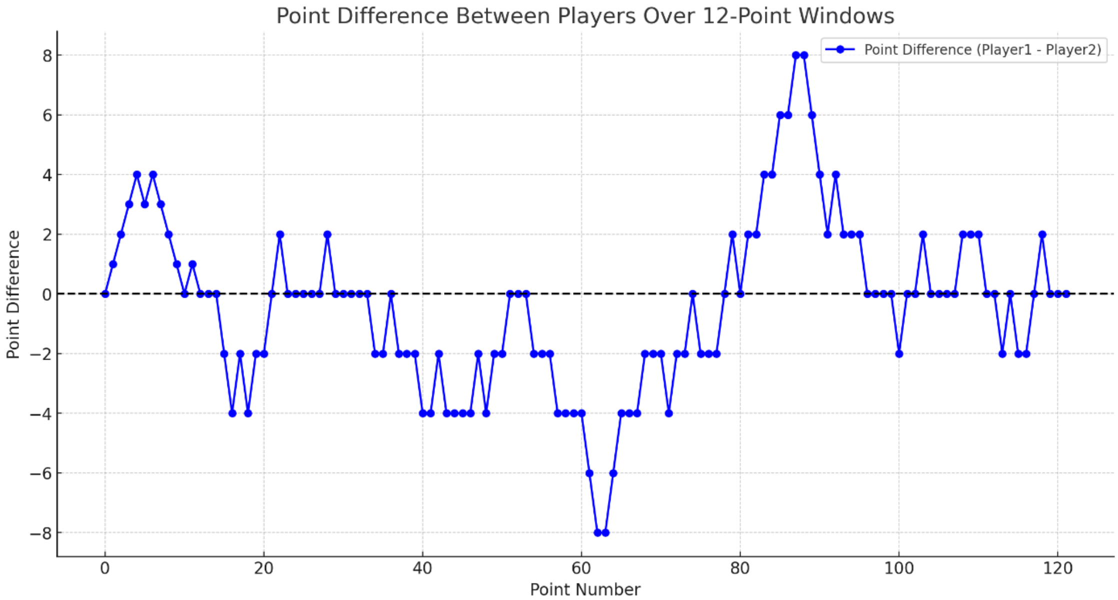
11. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query11: sum up the gained points in a 12-point window. calculate the gained point differences between the players. as for the first 11 points just use available data. and plot a graph.

Output:

To sum up the gained points within a 12-point window and calculate the differences between the players, we will:

Use a rolling window of 12 points to sum up the gained points for each player.
Calculate the difference in the summed points between the two players within each window.
For the first 11 points, we'll sum up the available data without applying the window.
Plot a graph showing the differences over the course of the match.
Let's start with the calculations and then move on to plotting the graph.



The graph illustrates the point difference between the two players over 12-point windows throughout the match. Each point on the graph represents the difference in the sum of gained points within a rolling window of 12 points, with Player 1's sum subtracted by Player 2's sum. A positive value indicates that Player 1 has gained more points than Player 2 in the respective window, while a negative value suggests the opposite.

The horizontal line at 0 serves as a reference to easily identify which player is leading in terms of points gained within these windows. The fluctuations in the graph reflect the changes in momentum between the players as the match progresses.

If there's anything more specific you'd like to analyze or any adjustments to the graph, feel free to let me know!

12. OpenAI *ChatGPT* (Jan 10, 2024 version, ChatGPT-4)

Query12: Uploaded image

interpret this graph

Output:

This image appears to be a graphical representation of the correlation between various match indicators and shifts in momentum, denoted by

Δ

�

Δ M. Let's break down the information presented:

Scoring: There is a strong positive correlation of 0.72, indicating that scoring is significantly related to shifts in momentum. In other words, changes in scoring are likely to be associated with a shift in momentum.

Service (total): This has a moderate positive correlation of 0.49 with shifts in momentum. This suggests that factors related to service are moderately associated with momentum shifts. Within this category:

ace: Aces have a correlation of 0.20 with momentum shifts.
break pt won: Winning break points has a very low correlation of 0.01 with momentum shifts.
serve: This likely refers to successful serves in general, with a correlation of 0.28.
speed mph: The speed of the serve (in mph) has a low correlation of 0.08.
serve depth: The depth of the serve has a very low correlation of 0.04.
Stamina (total): There is a moderate positive correlation of 0.59, suggesting that stamina-related factors have a moderate relationship with momentum shifts. Under this category:

ret. depth: Return depth has a small negative correlation of -0.07 with momentum shifts.
serve width: The width of the serve has a very low correlation of 0.06.
dist run: The distance run by a player has a moderate negative correlation of -0.20.
rally count: The number of shots in a rally has a low negative correlation of -0.14.
Slip (total): This category has a strong positive correlation of 0.60, indicating that slipping or errors are strongly related to momentum shifts. Within this category:

unf err: Unforced errors have a strong negative correlation of -0.42 with momentum shifts.
doub. fault: Double faults have a low positive correlation of 0.18 with momentum shifts.
Interpretation:
Positive Correlation: A positive value indicates that as the indicator increases, shifts in momentum are more likely to occur in the same direction (e.g., more scoring or aces are associated with gaining momentum).
Negative Correlation: A negative value indicates that as the indicator increases, shifts in momentum are likely to occur in the opposite direction (e.g., more unforced errors are associated with losing momentum).
The larger the absolute value of the correlation coefficient (ranging from -1 to 1), the stronger the relationship. Values closer to 0 indicate a weaker relationship.

In summary, scoring, stamina-related factors, and slipping/errors are significantly correlated with shifts in momentum. Scoring has the strongest positive correlation, while unforced errors have the strongest negative correlation, suggesting these are key factors in predicting momentum shifts in a match.