Vrije Universiteit Amsterdam

Bachelor Thesis

Data-Driven Prediction of ATP Tennis Match Outcomes Using Machine Learning Techniques

---

Author: Jakub Olaf Dryja (2732653)

1st supervisor:	Wan Fokkink
2nd reader:	Mauricio Verano Merino

A thesis submitted in fulfillment of the requirements for the VU Bachelor of Science degree in Computer Science

July 15, 2025

Abstract

This thesis investigates the use of machine learning models to predict the results of men's singles tennis matches on the ATP Tour, with a particular focus on Grand Slam competitions from 2010 to 2024. Four predictive models: Decision Tree, Logistic Regression, Random Forest, and XGBoost are employed in this study using historical match data and innovative feature engineering techniques, including dynamic player metrics, surface specific metrics, and an adaptive Elo rating system. Evaluation of these models utilizes accuracy, ROC AUC, log loss, and Brier score metrics to assess both binary classification accuracy and probability calibration. In addition to theoretical evaluation, practical implications are examined through simulated betting strategies designed to identify profitable and risk efficient approaches. Strategies such as Shorter Price, Threshold, and Agreement are developed and analyzed, highlighting conditions under which machine learning predictions effectively outperform traditional bookmaker odds. The findings indicate that advanced machine learning approaches, particularly the Random Forest model, can consistently achieve superior predictive performance and demonstrate potential for profitable betting strategies. The Random Forest model demonstrated the best performance of properly calibrated predictive models in sports betting markets by achieving a return on investment of 10.65% over 2047 bets under Threshold strategy. Overall, this research contributes valuable insights into predictive analytics in tennis, model calibration practices, and betting market efficiencies.

1. Introduction

Due to improvements in predictive modelling and the growing availability of comprehensive historical data, the use of machine learning techniques to forecast professional sports outcomes has become increasingly popular in recent years. The most likely explanation for such a surge is the potential use of these models to make profit. However, one should not overlook their scientific value, as they allow us to observe how machine learning models perform in real-life scenarios. Tennis, a highly dynamic and competitive sport, offers a particularly advantageous setting for these kinds of predictive analyses as a consequence of its clear regulations, organised tournament system, and extensive statistical records. Every rally produces precise numerical traces, while each match ends with a binary outcome, enabling direct comparison between forecasts and reality. This thesis makes use of these favourable circumstances to investigate how well data-driven machine learning models predict the results of men's singles matches on the ATP Tour, with a focus on Grand Slam competitions between 2010 and 2024.

To address the character of match prediction, the thesis frames three guiding questions: How well can we predict? Where do bookmaker prices diverge from model-based probabilities? and Under what conditions does that divergence translate into actionable value? This problem triad forces the modelling effort to balance pure statistical accuracy with the practical constraints of risk management and market frictions, constraints that any real-world decision system must ultimately respect.

Predicting tennis match outcomes can be approached through binary classification methods, determining a simple win/loss outcome, or through probability estimation, providing nuanced forecasts valuable in the context of sports betting. This study emphasizes the latter, comparing model generated probabilities directly with implied probabilities derived from bookmaker odds, thus exploring potential market inefficiencies. To achieve this, the research employs four distinct machine learning models, Decision Tree, Logistic Regression, Random Forest, and XGBoost, chosen for their diverse complexity, interpretability, and previously demonstrated effectiveness in predictive analytics. In order to get the most out of these models and reach the highest possible accuracy, this study follows several steps to achieve that. First of all, an innovative feature engineering framework, a crucial part of this work, that moves beyond static player attributes. By rolling exponentially weighted averages over time, interactions of various features, and maintaining advanced Elo ladders that adapt to surface-specific form, the thesis captures both who a player is and how they are trending at the moment of match-day. Subsequently, the models are rigorously trained using a time aware cross-validation strategy, preserving chronological order and preventing data leakage, ensuring robust and realistic predictive capabilities. Evaluation encompasses a variety of metrics, including accuracy, ROC AUC, log loss, and Brier score, highlighting both classification performance and probabilistic calibration. Ultimately, this thesis examines practical applications through betting strategy simulations to assess models in real-life scenarios. Shorter Price Strategy, Threshold Strategy, and Agreement Strategy are developed from scratch and thoroughly examined under various configurations to find lucrative setups. Each approach is evaluated on metrics such as return on investment (ROI), net profit and hit rate to identify the circumstances in which predictions powered by machine learning can beat conventional bookmaker odds.

This study enhances both the theoretical understanding and practical application of predictive models within tennis betting markets. By drawing on the research of previous similar works, adopting their most effective approaches, and blending them with novel ideas of my own, it delivers a comprehensive, complete, and clear framework to predict tennis match outcomes. Through the integration of advanced feature engineering, rigorous methodology, and real-world betting simulations, it evaluates predictive performance while also exploring the broader implications for market efficiency and the optimization of betting strategies.

2. Problem Domain

2.1 ATP Tennis as a Modeling Arena

The Association of Tennis Professionals (ATP) men's tour is the world's leading circuit of tournaments, consisting of the Grand Slams, the Masters 1000 events, and the ATP 250/500 tournaments - all feeding to a weekly player ranking. Whereas more extensive elaboration features in the Background (Section 4.1), worth mentioning here is the fact that the character of professional tennis is in constant evolution. The demographics of players change as the veterans retire and new talents break in, innovations in technology, such as the transition from wooden to composite rackets, and different court surfaces and ball specifications at tournaments. Cumulatively, these create nonstationarities in the analysis in the longer dimension but simultaneously make tennis the perfect testbed for data-driven forecasting. The sport's richly recorded match statistics, clearly defined outcome (match/winner), and public betting markets create a fertile environment in which predictive models can be rigorously developed and benchmarked.

2.2 Data Source and Scope

The empirical foundation for this study rests on two complementary datasets. The first is Jeff Sackmann's open source repository of ATP results, rankings, and player statistics, released under a Creative Commons BY-NC-SA 4.0 license. The database provides the match‑level outcomes, weekly ranking points, players' demography, and key performance indicators for nearly every recorded ATP event of the Open Era. The second dataset, drawn from tennis-data.co.uk, provides pre-match bookmaker odds from major providers (Betfair Exchange, Ladbrokes, Pinnacle Sports, Stan James, Bet365), generating the external reference point by comparing to the model‑extracted probabilities.

Some of the data that would be relevant in tennis modelling but is not available from Sackmann's dataset includes per-set statistics and detailed point-by-point match progression. While such data can be scraped from various websites for certain matches, this approach risks introducing inconsistencies. It is also worth noting that for many tournaments, far more detailed information than Sackmann's dataset exists from the Hawk-Eye ball-tracking technology. This uses multiple cameras to track the tennis ball's movement and to determine whether shots are in or out, primarily to assist with line-calling. It is, however, not open source as the Hawk-Eye is commercial equipment owned by Hawk-Eye Innovations, part of the Sony company.

Although Sackmann's archive extends back to 1968, data completeness and the structural stability of the game before 1990 are insufficient for robust modeling: several tournaments are unrecorded and detailed statistics are likely to be incomplete. Furthermore, the late 1980s marked substantive shifts in equipment and sports science that alter the underlying data-generating process. To avoid conflating these structural breaks with genuine predictive signals, the study period begins in 1998, the year verified as Roger Federer's professional debut and the earliest among the trio including Rafael Nadal (debut in 2001) and Novak Djokovic (debut in 2003). The 1998 initiation makes nearly complete coverage of the dataset available and brings the dataset in line with the timeframe of these superstars, providing 81831 match

records. From a machine learning perspective this amount might seem limited, but the dataset actually includes every single ATP match during the selected period. In previous literature, there is no trend toward using wider or narrower scope - most studies fall within similar range. For all experiments, a refined subset of Grand Slam matches from 2010 to 2024 is used, ensuring maximal data completeness and concentrating on players at or in peak form. The exclusive focus on Grand Slam tournaments allows for analysis of the most representative and highest-level matches in professional tennis, which are in fact also expected to be more predictable. The narrower temporal scope, limited to a more recent 15-year period, is also justified by the nature of feature engineering, which depends on the availability of sufficient and reliable prior match history to produce meaningful inputs.

2.3 Problem Formalization

This thesis addresses three core questions:

1. **Predictive accuracy**: How precisely can a machine learning model predict the result of an ATP singles match based solely on pre-match data?
2. **Market inefficiency**: To what extent do inconsistencies between the odds published by bookmakers and the probabilities predicted by the model suggest systematic inefficiencies that could be profitably exploited?
3. **Optimal wagering**: Which betting strategy, when applied to the model's probability estimates, maximizes return on investment (ROI) and overall turnover?

Model performance will be evaluated using classification and probability-calibration metrics: accuracy, ROC AUC, log loss, and the Brier score, while the effectiveness of betting strategies will be measured in terms of ROI, net profit and hit rate.

3. Related Work

Numerous approaches, such as traditional statistical modeling, machine learning, and betting market analysis, have been used to tackle the problem of tennis match outcome prediction. From basic rank-based models to complex learning algorithms, researchers have improved feature extraction, modeling, and evaluation metrics over time. This chapter provides context for the contributions of this thesis by reviewing these developments and highlighting significant gaps in the existing literature.

3.1 Literature Review

The majority of early efforts to model tennis match outcomes were statistical in nature. In order to model match outcomes based on the difference in ATP ranking points, Clarke and Dyte [1] employed logistic regression. Despite being easy to use and computationally straightforward, these models ignore a variety of contextual elements, including the playing surface and match context. Clarke and Barnett [2] improved statistical methods five years later in 2005 by adding statistics on serve performance to Markov Chain models. Their method provided in-depth probabilistic predictions of match dynamics and enabled point-level match simulations. Nevertheless, these models lacked the adaptability to include a wider range of contextual features and were predicated on the idea that points were independent of one another. In 2016, Kovalchik [3] carried out a comparative analysis of eleven predictive models, classifying them as regression-based, point-based, and paired comparison models. The best-performing regression models, particularly for players at the top of the rankings, used player rankings and Elo ratings. Predictive performance, however, declined dramatically for players with lower rankings, suggesting that player-specific factors and recent performance trends were not adequately taken into account. As machine learning gained popularity, researchers started using more features and algorithms. Sipko [4] presented a machine learning framework that used logistic regression and neural network models with 22 features, such as fatigue, injury status, and preferred surfaces. Sipko's method outperformed conventional models in terms of accuracy and betting return against the market. Surface-specific statistics were also used by Cornman et al. [5], who achieved a 69.6% accuracy rate and showed that betting strategies could produce profits. Van Rooij [6] compared SVM, Random Forest and XGBoost classifiers, analyzing the impact of both match statistics and player characteristics, demonstrating the necessity for models to concentrate on performance metrics rather than static attributes by showing that player attributes like age and height had less predictive value than in-match statistics. Yue et al. [7] used the Glicko rating system, which originated in chess, to predict tennis matches. Although their Bayesian method outperformed traditional Elo models in terms of prediction accuracy, it still had trouble generalising across tournament levels and surfaces. Ultimately, a comprehensive and systematic review of machine learning applications in sports betting, covering multiple sports, conducted by Galekwa et al. [19] in 2024 was analyzed to establish the current state of the art. A total of 219 studies were included in this review, depicting their results and findings. Concerning tennis, five top performing models were highlighted with Wilkens' [16] Neural Network (70% accuracy), Cornman et al.'s [5] Random Forest (73.5% accuracy), Solanki et al.'s [20] Neural Network (82% accuracy), Gao et al.'s [21] Random Forest (83.18% accuracy) and Ghosh et al.'s [22] Decision Tree (99.14% accuracy). While four out of five results seem reasonable, such a high score achieved by the Decision Tree model has raised concerns about its reliability for two reasons. Firstly, due to the simplicity and limitations of the algorithm's structure, and secondly, by reason of the

inherently unpredictable nature of the sport of tennis, where significant number of close matches in which both players have very similar chances of winning or unexpected events such as injuries should correspond to an accuracy way below 99%. A closer inspection of Ghosh et al.'s [22] study did not reveal a specific cause of this unrealistic result, although data leakage is suspected, as the dataset contains features derived directly from match outcome such as final set scores and thus violating the principle of causality by using the result to predict itself. Therefore, the results of Ghosh et al. [22] were excluded from the comparison between findings in this study and those of other research papers.

3.2 Research Gaps and Contributions of This Thesis

The field has made progress, but there are still a number of gaps that need to be filled. Despite increasing prediction accuracy, many machine learning models still take a simplistic approach to feature engineering. Little is known about how features interact, such as surface-specific statistics and recent form. Furthermore, the practical applicability of many models is limited because they are not tested in actual betting situations or compared to bookmaker odds.

This thesis advances the field by addressing these gaps through several key contributions:

- Designing a feature engineering framework that captures player dynamic form, interaction of features and surface-specific statistics using dynamic historical data aggregation, thus addressing the feature interaction gap.
- Extending predictive modelling by simulating betting strategies and evaluating market inefficiencies by directly contrasting model outputs with bookmaker odds to assess market inefficiencies.
- Comparing and combining a range of machine learning models under consistent experimental conditions.
- Using a broad, recent dataset to guarantee that the findings accurately represent the dynamics of the ATP tour today.

4. Background

4.1 The Game of Tennis

Tennis is a well-known sport throughout the world that features both singles and doubles competitions on different professional circuits. The ATP Tour, which regulates professional tennis for men, and the WTA Tour, which regulates women's events, are the two main circuits that make up the sport. Although the two tours have largely similar structures, there are some differences in the player pools, competitive formats, and match dynamics. Although singles is the main focus of this thesis, professional tennis encompasses both singles and doubles formats. A tiered tournament system serves as the foundation for tennis' competitive structure. The Grand Slam tournaments, which include the US Open, Wimbledon, French Open (Roland Garros), and Australian Open, are the most prestigious events. Each one is held once a year. These competitions draw the top players, give out the most ranking points, and are played in longer formats (best-of-five), which puts players' endurance and skill to the test. The ATP Masters 1000, ATP 500, and ATP 250 competitions, which differ in the number of points given out and player participation, fall beneath the Grand Slams. These competitions create a worldwide circuit that crosses different continents and terrains.

The variety of tennis playing surfaces is a key feature. Professional tournaments are held on three main surface types:

- Hard courts (Australian Open and US Open) - the most popular surface, which offers medium-paced play with a balanced bounce and is typically constructed of acrylic layers over concrete or asphalt.
- Clay courts (French Open) - surface made of crushed brick or stone, that create a slower game, making spin and rally endurance more crucial. An essential component of the game on clay is sliding.
- Grass courts (Wimbledon) - the original surface of the sport, produce the fastest game with a low bounce, favoring serve and volley tactics.

The mechanical properties of these surfaces have been shown to influence match dynamics. As discussed by Miller [9], the interaction between modern tennis balls and surfaces affects friction and shock transmission, which in turn influence both gameplay and player biomechanics.

Another element shaping modern tennis is the ATP ranking system. Over a rolling 52-week period, players earn ranking points based on how well they perform in tournaments. The ranking not only reflects recent form but also determines tournament seedings and entry qualifications. Unlike some sports that rely on head-to-head results or subjective ratings, the ATP ranking is a transparent, point-based system that updates weekly, offering an objective measure of player performance.

4.2 Betting

In its simplest form, sports betting allows an individual to wager on the outcome of a match. When it comes to tennis, the simplest bet is to predict which of the two players will win the match. However, the betting markets have changed to give people a lot of different ways to bet. Bookmakers now offer odds on a wide range of in-game statistics and situations, like the number of sets played, the total number of games won, player-specific aces, and point or game handicaps. Despite this variety, the present study considers only the most fundamental betting option: the match winner market, where the bettor wagers on one player to win the match.

The inherent advantage that bookmakers have is known as the bookmaker's margin. This margin guarantees a profit over time by ensuring that the total of the implied probabilities derived from the offered odds exceeds 100%. According to Cortis [10], the margin and the distribution of bets across outcomes have a direct impact on a bookmaker's expected profit. Let O_1 and O_2 represent the decimal odds for each player in a two-player tennis match. The following formula can be used to determine the implied probabilities p_1 and p_2:

$$p_1 = \frac{1}{o_1}, \quad p_2 = \frac{1}{o_2}$$

The total of $p_1 + p_2$ typically exceeds 1, with the surplus considered as the bookmaker's margin. Take the 2024 French Open final between Carlos Alcaraz and Alexander Zverev, for example. The decimal odds were: 1.35 for Alcaraz and 3.35 for Zverev. The sum of implied probabilities ($\frac{1}{1.35} + \frac{1}{3.35}$) gives approximately 1.039, indicating a bookmaker margin of around 3.9%, which means that they have a roughly 4% edge in this market. If the sum were less than or equal to 1, there would be chances for arbitrage, which would let bettors make risk-free profits on more than one outcome. Bookmakers try to avoid this situation at all costs.

This study uses basic normalisation to get implied probabilities that add up to 1 by dividing the inverse odds by the book sum. The normalized implied probabilities p_1 and p_2 are calculated as follows:

$$p_1 = \frac{o_2}{o_1 + o_2}, \quad p_2 = \frac{o_1}{o_1 + o_2}$$

Štrumbelj [11] showed that basic normalisation doesn't always give the most accurate probability forecasts. Instead, he suggested Shin's model as a more advanced option that takes into account market inefficiencies and possible insider trading. However, this study does not aim to give exact estimates of true probabilities. Instead, the normalised probabilities are used as a starting point for comparing the predictive models made in this thesis. So, to keep things simple and consistent, basic normalisation is used.

Return on Investment (ROI), which calculates the profitability of a sequence of bets, is a popular metric used to evaluate the effectiveness of a betting strategy. ROI is calculated as:

$$ROI = \frac{\text{Total Profit}}{\text{Total Amount Wagered}}$$

It is crucial to consider ROI over a large number of bets to make sure that it is not the result of random variance. A good betting strategy should give a steady and consistent return on investment (ROI) over time, showing that it can predict the future instead of just short-term changes. ROI is the most important measure, but there are other important metrics that evaluate betting strategies. Net profit, since it indicates how much absolute money is won or lost, is a straightforward measure of success. Additionally, the hit rate, the proportion of winning bets, provides insight into the strategy's accuracy, though it must be interpreted in the context of the odds being played. For example, strategies focusing on high-odds underdogs may have a low hit rate but remain profitable if the wins compensate for the losses.

4.3 Machine Learning Techniques

An outline of the machine learning algorithms employed in this study is described in this section. Four classifiers, Decision Tree, Logistic Regression, Random Forest, and XGBoost, were chosen to reflect a range of model complexity and interpretability. Each of these algorithms offers unique benefits in terms of calibration and predictive power and has been successfully used in previous sports analytics studies.

4.3.1 Decision Tree

One of the simplest yet most understandable algorithms for classification tasks is a decision tree. Based on the values of the input features, the model divides the data into smaller subsets. The algorithm chooses the feature that most effectively separates the data into groups where one class is obviously dominant at each stage. These splits are arranged in a tree-like structure, with each leaf node assigning a predicted outcome and each internal node representing a decision based on a feature value, such as determining whether a player's win rate surpasses a particular threshold. The simplicity and transparency of decision trees make them especially appealing. They can handle both numerical and categorical features and don't require a lot of data preparation. Its key limitation is that if unchecked, trees have a propensity to overfit, which means they do well on training data but poorly on new, unseen matches.

Decision trees provide a useful starting point when it comes to tennis prediction. The Decision Tree is used as a standard for comparison in this study. While not expected to produce the best performance, it offers insights into how much of the match outcome variance can be explained by simple, rule-based splits.

4.3.2 Logistic Regression

One of the most popular algorithms for binary classification problems is logistic regression. It is a classification model, not a regression model, as its name suggests. The fundamental concept of logistic regression is to use a linear combination of input features that are then passed through a logistic (sigmoid) function to convert the output into a range between 0 and 1 in order to model the likelihood that a given input belongs to a specific class.

The probability of the positive class of p_1 winning the match is given by the following equation:

$$p_1 = \frac{1}{1 + e^{-\left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)}}$$

where:

- p_1 is the predicted probability
- β_0 is the intercept
- β_i are the model coefficients
- x_i are the features values

One of the major advantages of Logistic Regression is its simplicity and interpretability. Assuming all other features remain constant, each coefficient shows how a particular feature affects the likelihood of the positive class. Logistic Regression as a computationally efficient model, serves as a good choice for making it well-suited for baseline comparisons. However, Logistic Regression is limited by its linear decision boundary. It cannot capture complex, nonlinear interactions between features unless such interactions are explicitly introduced during feature engineering. and therefore it is less effective in tasks where there are essentially nonlinear relationships between features and the target.

In this study, Logistic Regression is included as a benchmark model, serving as a point of comparison for more complex algorithms. It is expected to provide respectable baseline performance and to generate accurately calibrated probabilities. Its simplicity and dependability offer useful context for evaluating the incremental benefit of more complex approaches, even though it might not be as accurate as more complex models.

4.3.3 Random Forest

Random Forest is a type of ensemble learning method that is based on Decision Trees. Instead of relying on solely a single tree, it constructs a large collection of decision trees, each of which is trained on a different random set of the data. To get the final prediction, the predictions of all the individual trees are combined, usually by majority vote in classification tasks.

The Random Forest algorithm introduces two key types of randomness to improve generalization and reduce overfitting:

- Bootstrap sampling (bagging): Each tree is trained on a randomly sampled subset of the training data. This ensures that each tree sees a slightly different set of data.
- Random feature selection: When determining the best split at each node, the algorithm considers only a random subset of the available features, which promotes diversity among the trees.

This set of techniques makes a model that can handle overfitting better than a single Decision Tree and can find complicated, nonlinear relationships and feature interactions. Furthermore, Random Forests are relatively insensitive to hyperparameters, which makes them a good choice for many classification tasks. While individual trees are easy to understand, the ensemble as a whole functions as a “black-box” model, making it difficult to trace specific predictions back to feature splits. Additionally, Random Forest models can be computationally intensive when a large number of trees is used.

This study chose Random Forest as a robust, general-purpose classifier, expected to provide high accuracy and to handle the complicated interactions in tennis match data. As an ensemble model, it is expected to outperform both the Decision Tree and Logistic Regression models, offering a strong balance between accuracy and generalizability. Its ability to automatically model feature interactions makes it particularly well-suited for the diverse and interconnected features used in this work.

4.3.4 XGBoost

XGBoost (Extreme Gradient Boosting) is an advanced ensemble learning algorithm based on the principle of gradient boosting. In Random Forest, trees are trained separately and their outputs are combined by averaging or voting. In XGBoost, on the other hand, trees are built one after the other, with each new tree trying to fix the mistakes of the trees that came before it. This process creates a strong predictive model out of a group of weak learners.

The algorithm optimizes a chosen loss function by computing the gradient of the loss with respect to the model’s predictions and adjusting the next tree accordingly. Each tree learns how to predict the mistakes of the preceding trees. After adding a new tree, the new prediction for a given instance can be written as:

$$\hat{y}^{(t)} = \hat{y}^{(t-1)} + \eta \cdot f_t(x)$$

where:

$\hat{y}^{(t)}$ is the updated prediction after the t-th tree

η is the learning rate (step size)

$f_t(x)$ is the prediction from the new tree

XGBoost differs from other boosting methods since it uses regularisation techniques to control model complexity, column sampling, and highly optimized computational performance. These improvements make XGBoost not only very accurate, but also capable of working with complex datasets. However, it is computationally more demanding and requires careful hyperparameter tuning to avoid overfitting.

In this study, XGBoost was included as a state of the art boosting model, expected to achieve the highest predictive accuracy and probability calibration among the evaluated algorithms. Given the structured nature of tennis match data and the presence of complex interactions between performance metrics, XGBoost is well-suited for this task. Moreover, as shown in Galekwa et al.'s [19] work, this algorithm has not yet been thoroughly examined in the context of sports outcome prediction, which makes it even more compelling to test.

5. Implementation

The whole project was conducted in a Python environment, involving the pandas and NumPy libraries for data processing, scikit-learn for model development and evaluation and XGBoost for gradient boosted decision trees. The entire codebase is publicly available as open source on GitHub at:

https://github.com/qdryja/Data-Driven-Prediction-of-ATP-Tennis

5.1 Data Acquisition

This study utilizes open source data sources that offer comprehensive coverage of ATP men's professional tennis over the modern era. A more detailed discussion of the data scope and domain-specific considerations was presented in Section 2.2. The primary match data are sourced from Jeff Sackmann's ATP github repository, an open access archive of ATP match results, player demographics, rankings, and basic performance statistics. This repository provided 27 CSV files that covered every ATP match played between 1998 and 2024, each of which corresponded to a single season. The resulting slice contains 81831 match records. Each match yields two symmetric rows, one for the winner and one for the loser, with player specific columns prefixed accordingly (e.g., winner_id, loser_id, winner_rank, loser_rank, etc.). Table 5.1 depicts how the variables are grouped.

Table 5.1: Variable groups from the Sackmann's dataset

Group	Columns	
Tournament context	tourney_id tourney_name tourney_level tourney_date best_of round	surface draw_size match_num minutes score
Player demographics and ranking	_id _seed _entry _name _hand	_ht _ioc _age _rank _rank_points
Point-aggregate performance	_ace _df _svpt _1stIn _1stWon	_2ndWon _SvGms _bpSaved _bpFaced

In addition, data on bookmaker odds were gathered from the public archives of tennis-data.co.uk, which keeps track of pre-match odds for tennis tournaments. For this study, a subset of 59 CSV files containing odds data for Grand Slam tournaments from 2010 to 2024 was extracted. Every year, there are usually four tournaments: the Australian Open, the French Open, Wimbledon, and the US Open. The only exception is Wimbledon 2020, which was cancelled because of the COVID-19 pandemic.

5.2 Data Preprocessing

The raw extraction that begins in 1998 supplies 81 831 tour-level rows. An initial inspection showed that every row contained at least one missing cell, a pattern driven largely by administrative fields (seed, entry status) rather than technical performance variables.

5.2.1 Missing Values

Three overlapping analytical strata were established to assess the practical impact of deletion decisions. See Table 5.2.

Table 5.2: Strata definitions

Id	Description	Number of rows
S_1	Full corpus: all tour-level matches since 1998	81831
S_2	Grand-Slam participants (2010-2024): rows involving any of the 599 players who appeared in at least one Grand-Slam main draw in the modern live-scoring era	64900
S_3	Grand-Slam matches (2010-2024): matches actually played at Slams during the same window	7493

To keep the dataset's fluency and reliability, it is critical in this study to keep the number of dropped rows to a minimum. Excessive pruning could weaken the reliability of dynamic metrics, particularly the Elo system discussed in Section 5.3.5, by reducing the continuity of historical records. The row removal strategy used a system of priorities: S_3 rows were preserved at all costs, followed by S_2, then S_1. This hierarchy ensures that the most useful observations, Grand Slam matches with relevant players, are kept for modelling, enhancing models' ability for better predictions.

The first step involved removing non-predictive administrative columns (tourney_seed and entry) that are used for tournament bookkeeping but do not provide any information about a match. It reduced the number of incomplete rows significantly without losing any match data. Following, 412 rows with walkovers and unfinished matches were dropped as these do not contribute meaningful predictive signals. After these changes, the number of rows with missing values decreased to 10253 (S_1), 5971 (S_2), and 546 (S_3). The next phase required dropping rows on incomplete entries in essential performance metrics (like _ace, _df, and _svpt, etc.), which mostly happened in lower-tier tournaments with less skilled players.

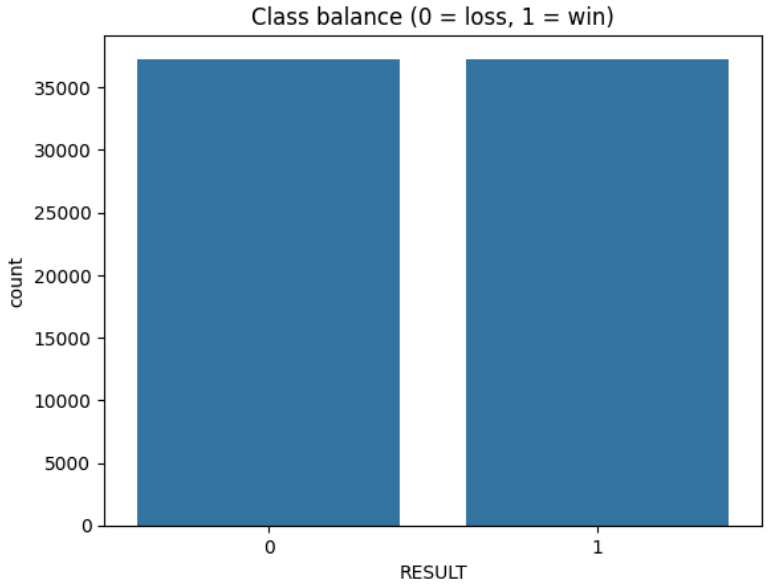
This step left out 6866 rows in S_1, 3407 in S_2, and only one row in S_3, demonstrating that the highest-priority matches remained nearly unaffected by these steps.

Remaining missing values were: player_age (2 blanks, recovered via manual lookup using ITF biographies), player_height (838 blanks, filled using the mean of all players), player_rank (570 blanks, filled with value 2000 that corresponds to the lowest ATP ranking), player_rank_points (573 blanks, set to zero), and minutes (1854 blanks, approximated by multiplying number of games by 6 minutes). Even though this might cause some noise in the estimates, it is still acceptable for aggregate-level modelling and is backed up by Bennett [12] who affirms that imputation, even simplistic, can reduce bias.

5.2.2 Symmetric Recording of Players

The next step involved preventing positional bias that might occur with the current dataset since the winner is always on the left hand side, which could cause the model to learn that the player on one side is more likely to win. To prevent this, a random mask was applied on every match to change the format to $player_1$/$player_2$ - with the winner randomly assigned to either position. Finally the target variable (result) is set to 1 if $player_1$ won, 0 otherwise. This keeps class balance and positional neutrality, as shown in Figure 5.1.

Figure 5.1: Class balance for the binary target variable



5.2.3 Post Cleaning Summary

Table 5.3 presents a numerical summary of the preprocessing process. The final dataset obtains 74553 rows (S_1), 61147 rows (S_2), and 7461 rows (S_3), achieving dropping rates of 8.4 %, 5.3 %, and 0.01 % respectively. These rates are well within the commonly cited threshold of 90 % completeness. Bennett [12] underscores that missingness above 10 % may bias statistical inference unless explicitly modeled.

Table 5.3: Post cleaning summary

Id	Initial number of rows	% of rows dropped
S_1	81 831	8.4%
S_2	64 900	5.3%
S_3	7 493	0.01%

5.3 Features Design

The raw ATP match logs, while extensive and detailed, are limited in their native form by a lack of analytical expressiveness. The original per-match statistics do not reflect the nuanced differences in performance between players nor do they contextualize historical performance. Therefore, a critical step in this thesis involved the design of an expansive feature set to summarize comparative dynamics between players. This section discusses the reasoning, structure, and methodology underlying the creation of these features.

5.3.1 Design Philosophy and Feature Groups

The design approach was based on constructing relational features. That is, statistics reflecting the difference in performance between Player 1 and Player 2. This was achieved by computing each metric separately for both players based solely on their prior matches and subtracting Player 2's value from Player 1's. The resulting value forms the final feature passed to the model. This design choice allows the model to learn patterns that are specific to the relative strengths and weaknesses in a given matchup. Features where values are near zero suggest a balanced contest, while increasingly positive or negative values indicate directional advantages. Features are classified in two complementary ways - thematically and in terms of methodological construction (see Appendix Tables A.1 and A.2).

5.3.2 Leakage Prevention and Feature Construction API

To prevent information leakage, a major concern in sport predictive modeling, strict steps were involved to ensure that training data only included information available prior to the prediction time. Leakage refers to scenarios where data unavailable at prediction time is accidentally used during model training, often leading to misleadingly high validation scores and poor generalizability. To mitigate this, an API was built to handle feature computation in a forward rolling manner. During training set generation, the API iterates through the match dataset chronologically. For each match, the *generate_features_for_match()* function is called to compute features using only statistics accumulated before the match. Once the features are generated, the match result with corresponding match statistics, is used to update the API's internal state. This guarantees that feature values for each match are strictly based on past data, avoiding any bias. For prediction tasks involving future matches that have not yet been played (e.g., forecasting matches from 2025), the same *generate_features_for_match()* function can be used, but without updating the API afterward, since no match results are yet available. This design

ensures consistent feature generation across both training and prediction scenarios and protects the model pipeline safe from being polluted by new data.

5.3.3 Normalization and Rate Based Transformation

Direct use of raw statistics would produce misleading insights due to differences in match formats and durations. For instance, Grand Slam matches are held in best-of-five format, whereas others follow best-of-three, introducing variability in point and game counts. To mitigate this, all primary metrics were normalized into rate-based forms. Aces and double faults were expressed per service game, break point statistics were converted into conversion or save percentages, and other performance data were scaled by the number of games or points played. These transformations create a coherent and scale invariant set of indicators. However, even these per-match rates are often insufficient as they lack temporal smoothing. A player's performance can fluctuate significantly from one match to another due to various factors like fatigue, injury, or opponent strength. To capture performance trends rather than snapshots, historical averaging becomes necessary, a problem addressed via EWMA (described in Section 5.3.4).

5.3.4 Exponentially Weighted Moving Averages (EWMA)

The Exponentially Weighted Moving Average (EWMA) is a statistical method that is often used to smooth out time series data. Unlike Simple Moving Averages (SMA), which assign equal weight to all observations, EWMA progressively gives greater weight to more recent observations, allowing it to capture dynamic changes and trends more effectively. Another advantage of EWMA over SMA is its ability to maintain continuity in the analysis. In SMA, discontinuities occur when older observations suddenly leave the calculation window, potentially producing artificial jumps and hiding real performance trends. By contrast, EWMA integrates all historical data with a fading effect, and thus preventing sudden changes when older data points are removed. This makes EWMA particularly suitable for performance tracking contexts where recent trends carry significant predictive value, such as in professional sports, stock price forecasting, and quality control processes.

Formally, the EWMA calculation at time t can be represented by the recursive formula:

$$EWMA_t = (1 - \alpha) \cdot EWMA_{t-1} + \alpha \cdot m_t$$

Where:

m_t	is the observed value at time t
$EWMA_{t-1}$	is the previously computed EWMA
α	is the smoothing factor ($0 < \alpha \leq 1$)

The choice of the smoothing factor α is significant as it determines the sensitivity of EWMA to recent changes. A larger α assigns greater weight to recent data, making EWMA more responsive to short-term fluctuations, whereas a smaller α produces a smoother slope, emphasizing long-term trends. In this study, both short-term and long-term performance dynamics for each feature were computed to model player form. Two α values of 0.18 and 0.05 were selected as optimal for these two horizons. The higher α of 0.18 corresponds to a half-life of about 3.5 matches, capturing recent performance. On the other hand, the

lower α of 0.05 represents a half-life of around 14 matches, better reflecting sustained performance trends. Figures 5.2 and 5.3 illustrate the difference in weight distribution across past observations for these two smoothing factors.

Figure 5.2: EWMA distribution (α = 0.18)

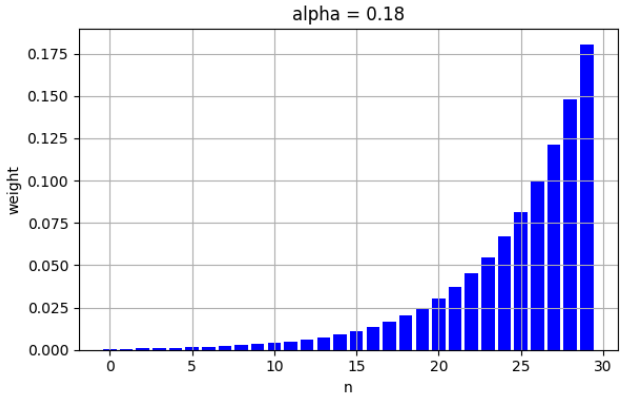
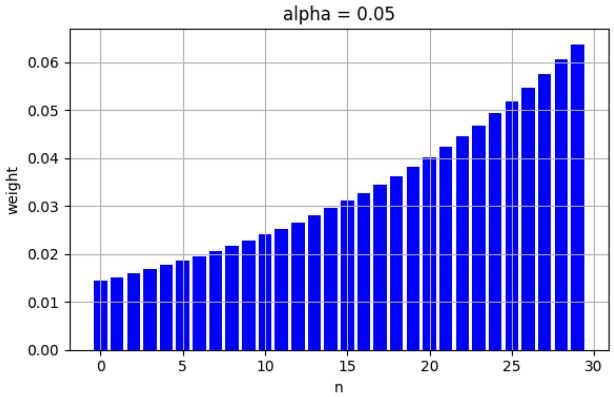


Figure 5.3 EWMA distribution (α=0.05)



There have been a number of studies reflecting EWMA efficiency in performance prediction contexts. For example, a cricket analytics study by De Silva et al. [13] applied a one-sided EWMA to keep track of how well each player was batting over time. This method successfully detected drops in player performance, like streaks of low scores, much earlier than static averages would have. These drops were considered a good indication when they were "out of form". This study shows that EWMA can quickly adjust to changes in performance, capturing dynamic trends that static averages often fail to reflect.

All features derived using the EWMA methodology are summarized in Table A.2 under the feature group labeled “EWMA”.

5.3.5 Composed Metrics

Direct match-level statistics like win percentages or raw counts of aces and double faults are not sufficient to reflect a player's skill or competitive form. These elementary metrics often fail to consider deeper contextual dynamics, such as opponent strength, surface type, and temporal variation in form. A set of higher-order, composite features was engineered to fill in this analytical gap. These features are not observed directly in the raw dataset but are instead derived through structured transformation, aggregation, or synthesis of multiple base-level metrics. They are used to capture hidden traits like overall player strength, ability to adapt to different surfaces, and balance between offensive and defensive skills. These composed features make the model better at generalising across different matchups and times by turning complicated relationships and changing traits into measurable metrics.

Elo Rating System

The Elo rating system, first developed by Arpad Elo for the game of chess, became a common method to dynamically measure players' skill levels based on competitive outcomes. The system relies on the idea that every player has a hidden skill level that can be measured with a number. The difference between the ratings of two players is seen as the main factor that will determine the outcome of the match: a bigger gap in ratings means a higher probability that the player with higher ranking will win. Crucially, the Elo

system is designed to update these skill ratings iteratively. After each match, the winner's and loser's ratings are adjusted based on the match outcome and the pre-match expectations. If the stronger player wins, their rating increases slightly, which shows that their expectations were met. If the weaker player upsets the favorite, though, the adjustment is larger, reflecting the surprise and revising both players' ratings more sharply. Elo is purely match-based and recalibrates after every single match, unlike ATP rankings, which add up points from tournament finishes over a rolling 52-week period. Its simplicity, interpretability, and adaptability have led to its application beyond chess, in various sports. Moreover, several studies have highlighted Elo's strong performance in tennis prediction tasks. For example, Kovalchik [3] found that Elo-based models outperformed ATP rankings in forecasting Grand Slam match outcomes. These advantages motivated its adoption for this thesis, where interpretability and predictive reliability were key objectives.

The formula for the expected outcome of a match between Player 1 and Player 2 is:

$$E_1 = \frac{1}{1 + 10^{\frac{R_2 - R_1}{400}}}$$

And the updated rating for Player 1 after the match is given by:

$$R_1' = R_1 + K(S_1 - E_1)$$

In this formula, K is the update factor, which is a constant that determines how sensitive the rating changes. Higher values increase responsiveness but risk instability. In chess, K values typically range from 10 (for elite, stable players) to 40 (for new players). In tennis research, fixed K values are commonly used in the range of 20 to 50. However, using a constant K across all players fails to account for differences in experience and rating stability. To address this, a dynamic K is used in this study, adjusting the magnitude of rating updates based on the players' experience. This approach, inspired by FiveThirtyEight's Elo modeling methodology [14], reduces volatility for experienced players and increases responsiveness for new players. The dynamic K is calculated using the following formula:

$$dynamic\_k = \frac{K}{(n + offset)^{shape}}$$

where:

K	is a constant multiplier
n	is the number of matches played by the player
$offset$	is a baseline adjustment to reduce volatility for new players
$shape$	is an exponent controlling how quickly K decreases as match count increases

A limitation of traditional Elo systems is the assumption that player skill remains static in the absence of play. In reality, time away from the tour due to injury, rest, or suspension introduces uncertainty. To address this, ratings are decayed toward a neutral baseline (1500) after 90 days of inactivity. The decay is calculated as:

$$R' = 1500 + (R - 1500) \cdot e^{-\lambda \cdot days\_idle}$$

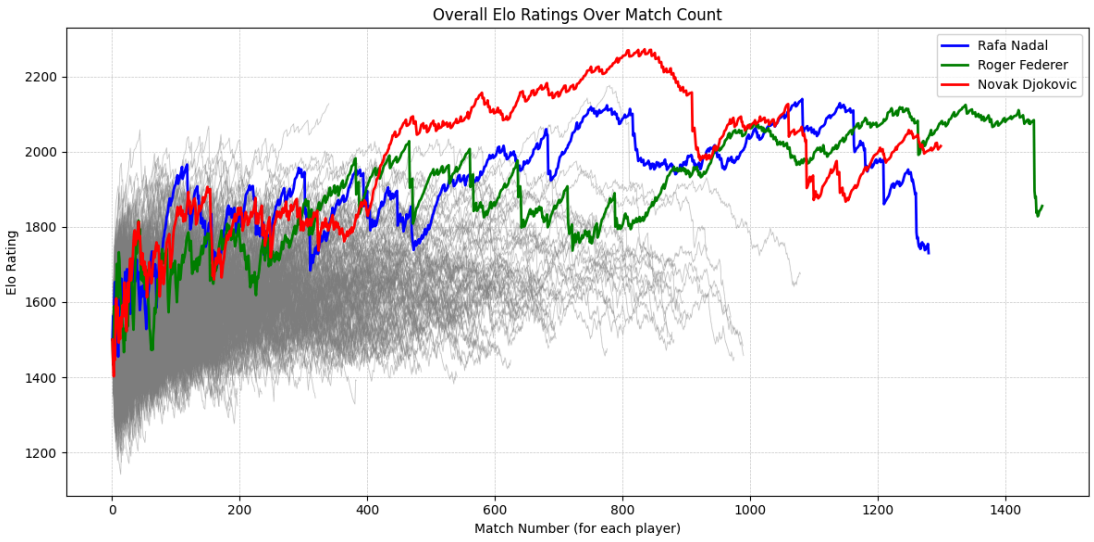
where:

R	is the player's most recent rating before inactivity
λ	controls how quickly the rating decays toward 1500 during inactivity
$days\_idle$	is the number of days since the player's last match

This kind of decay indicates that players coming back from injury, a break, or retirement tend to underperform. This approach also introduces a comeback boost with K-factor being temporarily increased after a player's absence (90 days or more). This compensates for the fact that there is a higher uncertainty about how the returning player will play, allowing Elo ratings to adapt more quickly to sharp recoveries or continued decline. Without this, decayed ratings might be over-trusted and respond too slowly to unexpected changes in performance.

Figure 5.4 illustrates the overall Elo of all players over time (matches played), with Djokovic, Nadal, and Federer being highlighted. The Big Three's dominance is clearly visible, as their Elo ratings consistently exceed the rest of players. While the majority of players stabilize in the range of 1400 to 1800, Djokovic, Nadal, and Federer steadily achieve over 1800 Elo points, reflecting their sustained lead. This chart also illustrates the distinct career patterns of these players, with Federer's longer career along the horizontal axis, Nadal's periods of fluctuation, and Djokovic's later but highly consistent rise.

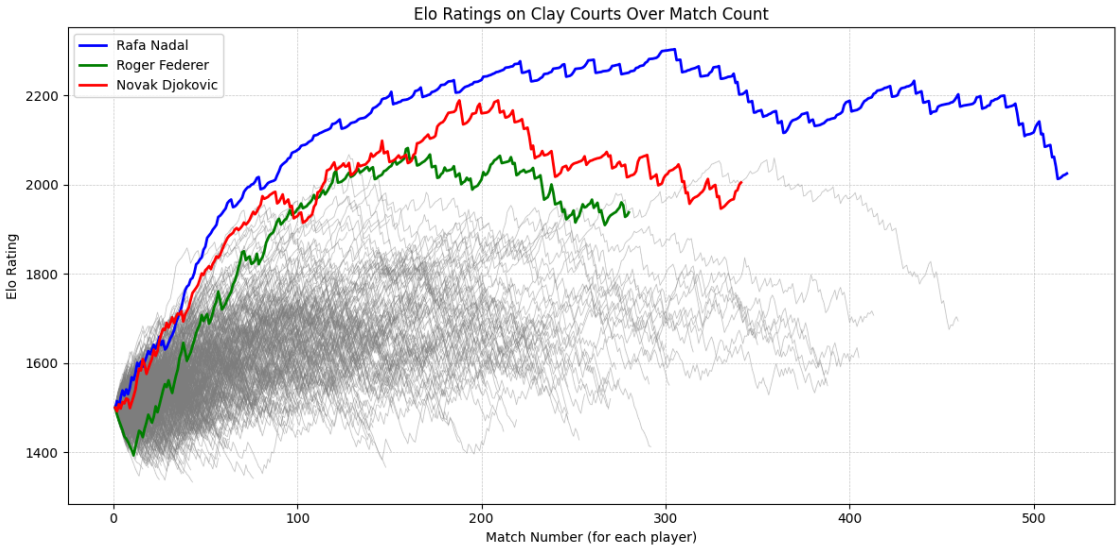
Figure 5.4: Overall Elo ratings over match count



In addition to tracking a single global Elo rating for each player, this study keeps track of separate Elo ladders for each court surface. This approach takes into account the fact that tennis performance strongly depends on the surface. This is very clear in Nadal's case, where his unmatched dominance on clay does not translate to performance on hard or grass courts. it's not as clear on grass or hard courts. This is illustrated in Figure 5.5, which isolates Elo ratings for matches played on clay. Unlike in Figure 5.4, where multiple players reach comparable peak ratings, Nadal stands significantly apart from all other competitors on clay. This reflects his top performance at Roland Garros, the only Grand Slam played on clay, where he has claimed the title more times than any other player in history and became widely recognized as the "King of Clay". It validates the use of surface specific Elo ratings and highlights the model's ability to capture context-dependent player dominance.

Surface specific Elo system in this study is slightly simplified in contrast to the overall Elo. It uses fixed K and ignore the decay rule. This is because each surface's competitive season is periodic. For instance, the season on grass courts usually lasts six to seven weeks, making it impractical to decay Elo. Elo changes only happen based on match results on the right surface, which keeps ratings stable between surface seasons.

Figure 5.5: Clay court Elo ratings over match count



These improvements make the Elo system better suited to the unique characteristics of professional tennis, such as changing forms, different surfaces, and players being absent. As a result, Elo serves as a stand-in for player strength, along with the other features in this predictive framework.

Serve Advantage

Sipko [4] introduced the concept of serve advantage as a more informative alternative to relying solely on raw serve and return percentages. Traditional metrics usually look at serve and return performances separately, without taking into account how these parts of the game interact with each other and with the opponent. Serve advantage, on the other hand, is designed to measure not only how well a player serves, but also how that performance compares to the defensive abilities of the opponent. This perspective gives

a better view of a player's likely performance. In practice, this is computed by evaluating each player's success rate on serve against their opponent's return effectiveness, resulting in a net serve advantage value. For Player 1, this is expressed as the difference between their own serve points won and Player 2's return points won.

$$SERVE\_ADV_1 = SRV\_PTS\_WON_1 - RET\_PTS\_WON_2$$

This feature captures the important relationship between offensive and defensive dynamics in tennis and lets the model analyze player matchups in a way that raw averages cannot. Moreover, it reflects how players are likely to challenge each other's strengths directly, and for this reason making it a strategically meaningful indicator. Feature importance has shown that this metric has stronger predictive power than its component statistics alone, especially in matches that are tight. This higher-order metric includes performance traits that are otherwise overlooked.

Completeness

The completeness metric serves as another advanced metric that was developed. In professional tennis, players such as Roger Federer impersonate completeness. It makes him stay competitive on all surfaces against a wide range of opponents. This metric is based on the idea described by Sipko [4], who emphasized the importance of balanced capabilities in both offensive and defensive play styles. The completeness metric is designed to quantify this balance by multiplying a player's serve points won and return points won. This formulation rewards all-round players, those who not only dominate on their serve but are also capable returners.

$$COMPLETENESS = SRV\_PTS\_WON \cdot RET\_PTS\_WON$$

Momentum

Another strategic feature developed during the engineering process is the momentum metric, aimed at capturing form variance trends. Momentum in sports analytics typically means if a player is currently improving or declining in form relative to their longer-term baseline. It is especially useful in scenarios where recent form diverges significantly from historical averages due to injury, tactical adjustments, or confidence gained from recent wins.

Initially, momentum was computed for each performance metric by calculating the difference between the short-term and the long-term EWMA. This provided granular insights into a player's directional trend for each skill metric. However, including multiple individual momentum features inflated the dimensionality of the dataset and risked introducing redundant or noisy signals that could decrease the model's generalizability. To mitigate this, a more compact composite momentum metric was designed. This single metric attempts to sum up the collective direction and volatility of recent performances across a subset of key statistics. Instead of tracking individual trends for every feature, the composite momentum aggregates these shifts into a unified score, offering a concise yet meaningful view of short-term player dynamics.

$$MOMENTUM = \left(\frac{\sum_{i=1}^{N} \text{sign}(m_i)}{N} \right) \cdot \sqrt{\frac{\sum_{i=1}^{N} m_i^2}{N}}$$

where:

m_i	is the difference between short and long EWMA for performance metric i
$sign(m_i)$	indicates whether the recent trend for metric i is positive or negative
N	is the number of features included in the momentum calculation

5.4 Model Training

5.4.1 Time Aware Train-Test Segmentation

Given the temporal nature of the dataset, a standard random train–test split would introduce information leakage from future matches into past data. Such leakage occurs when the model gains access to information from events that, in a real-world scenario, would not yet have taken place at prediction time, thereby leading to inflated performance estimates and poor generalization. Instead, a time-aware approach was implemented using the *TimeSeriesSplit* method from scikit-learn. This method divides the data chronologically into training and validation folds that imitates the passage of time. In each split, the model is trained exclusively on past observations and validated on a subsequent, unseen block of data. This approach respects the sequential nature of the problem, ensuring that all observations in the training fold come before those in the test fold, thus replicating the conditions under which a predictive system would be deployed. Such a scheme was shown by Bergmeir et al. [15] to reduce estimate variance and deliver more reliable error metrics for time series forecasting models.

The training process was organized into an outer 5-fold time series cross-validation loop, where the model was trained on an expanding window of historical matches and validated on the next sequential block. This mirrors the real-world forecasting scenario, where predictions are made for future matches using only past data. An inner 3-fold time series split was used during hyperparameter optimization to tune model parameters on the training set without contaminating the validation fold. This nested structure improves the robustness of performance estimates and mitigates overfitting risks.

5.4.2 Hyperparameters Tuning

Hyperparameter tuning was performed using randomized grid search, balancing thoroughness and computational feasibility. For each candidate algorithm, a predefined search space was constructed based on prior literature and practical considerations. The search process evaluated 40 randomly sampled parameter combinations, finding a balance between coverage of the search space and computational efficiency.

The best hyperparameter configurations for each algorithm, as selected through the inner cross-validation process, are summarized in Table B.1 in the appendix.

5.4.3 Evaluation Metrics

Model performance was assessed using a set of metrics suited for binary classification and probabilistic forecasting. The primary evaluation criteria included:

- Accuracy: The proportion of matches where the predicted class correctly matches the actual outcome. While commonly used, accuracy alone does not reflect the quality of probability estimates or the model's calibration.
- Logarithmic loss (log loss): A measure of the accuracy of predicted probabilities, penalizing confident but incorrect predictions more heavily. Lower log loss values indicate better probabilistic calibration and discrimination.
- Brier score: The mean squared difference between the predicted win probabilities and the actual match outcomes. This metric captures both the accuracy and the calibration of the probabilistic forecasts, with lower values indicating better performance.
- Receiver Operating Characteristic Area Under the Curve (ROC AUC): A measure of the model's ability to distinguish between winning and losing players. Higher AUC values indicate that the model ranks winners above losers more consistently, regardless of the decision threshold.

These evaluation metrics align with best practices in sports outcome modeling, where both classification accuracy and probability calibration are significant. Yuan et al. [17] and Kovalchik [3] highlight that log loss is particularly well-suited for betting contexts, as it punishes overconfidence in incorrect predictions, an important property when probabilistic forecasts are compared to bookmaker odds. Wilkens [16] further demonstrates that combining accuracy, ROC AUC, log loss, and Brier score provides a comprehensive assessment of model performance, capturing both the model's ability to distinguish winners from losers and the reliability of its predicted probabilities. This multi-metric approach ensures that the models evaluated in this thesis are assessed not only on predictive correctness but also on the quality of the underlying probability estimates.

Besides numerical evaluation, several diagnostic plots were generated to provide visual insights into model behaviors. Calibration curves assessed whether predicted probabilities aligned with observed outcomes, revealing any tendency toward overconfidence or underconfidence. ROC curves visualized the trade-off between true positive and false positive rates, offering a graphical summary of classification performance across various probability thresholds.

While visualization of ROC curves does not provide particular differences between folds, calibration curves, however, reveal much more insightful trends regarding the reliability of predicted probabilities. Calibration plots for each model and fold are shown in Figures 5.6 and 5.7.

Figure 5.6: Calibration curves in cross-validation fold 1

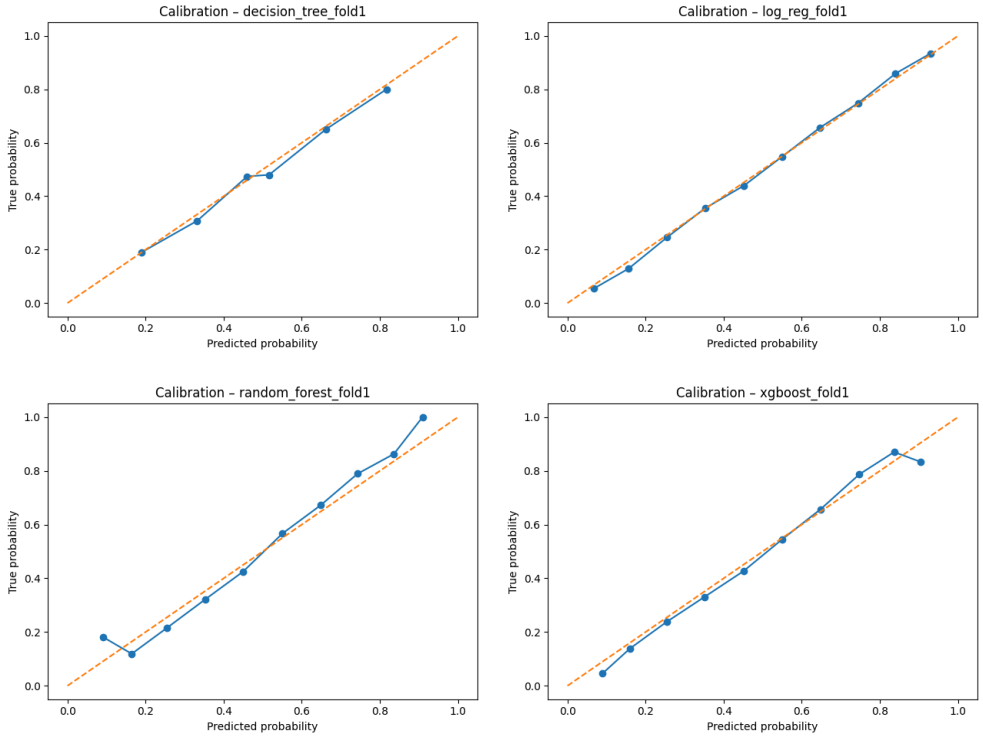
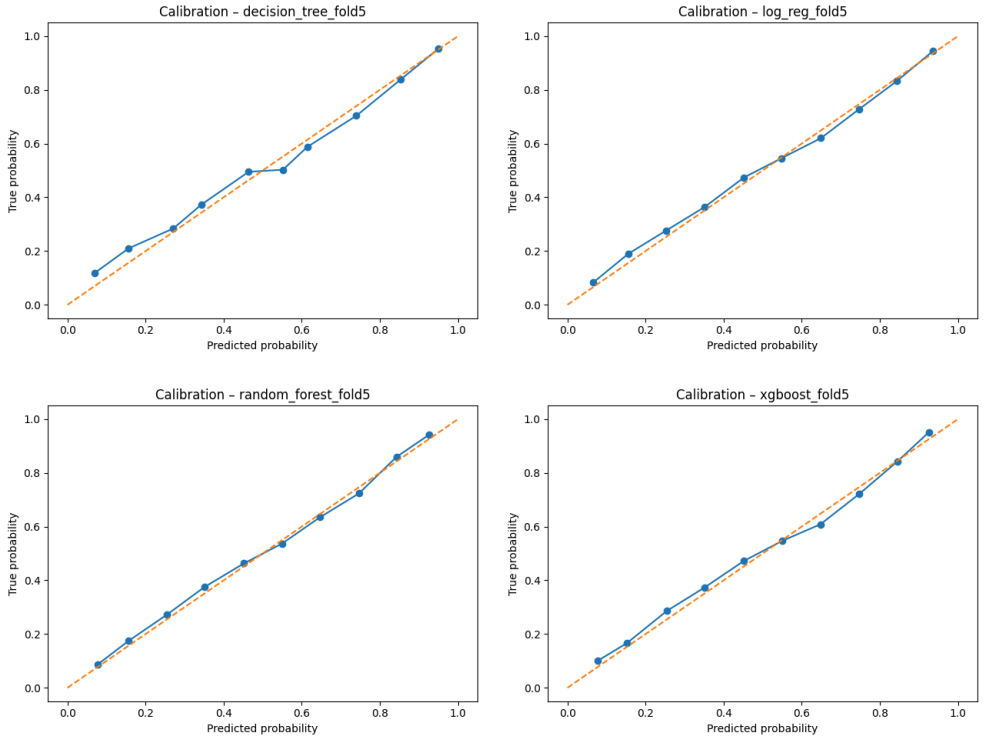


Figure 5.7: Calibration curves in cross-validation fold 5



The calibration between folds 1 (Figure 5.6) and 5 (Figure 5.7) improves noticeably as the training process progresses. In fold 1, models display greater deviations from the ideal diagonal line, reflecting less reliable probabilities. In contrast, by fold 5, calibration curves align much more closely with the ideal line, suggesting improved reliability of predicted probabilities. There are two primary causes for this progression. First, models' capacity to generalise is improved as the training dataset grows in subsequent folds, exposing them to a larger and more representative sample of past matches. Second, certain features, like the Elo rating, require a sufficient number of matches to stabilize and accurately reflect player skill. Since each new player in the dataset begins with an initial Elo of 1500, which does not yet reflect their actual playing ability, Elo ratings in the early stages of the dataset may be incomplete and unrepresentative.

In contrast, logistic regression maintained a similar calibration across all folds, which is in line with its probabilistic nature and reduced reliance on intricate features. These findings support the selection of a time-aware validation approach and show that model calibration naturally improves as more contextual player data accumulates over time.

6. Experiments and Results

This chapter presents a comprehensive empirical assessment of the four candidate probabilistic classifiers developed earlier and links their predictive quality to real-world profitability. Each model's performance is evaluated on the complete match dataset and, in particular, on the chronologically held-out Grand Slam window from 2010 to 2024. The resulting probability forecasts are fed into three fixed-stake betting strategies, from which measures of return, risk and turnover are derived.

6.1 Results

Table 6.1 shows a clear yet modest ordering of model quality. The gradient boosted tree delivers the strongest results on every metric, with gains that, although small in absolute terms, remain consistent across accuracy, ROC AUC, log loss, and the Brier score. The random forest performs nearly the same as logistic regression. In the calibration-sensitive measures, where a lower log loss and Brier score validate sharper, better-calibrated probability estimates, their slight inferiority to the boosted variant is most apparent. The limitations of simpler, non-ensemble algorithms in this predictive context were highlighted by Decision Tree's poorer performance.

Table 6.1: Overall performance metrics over entire dataset

Model	Accuracy	Roc_Auc	Log_Loss	Brier
Decision Tree	0.661087	0.723360	0.611788	0.211699
Logistic Regression	0.675590	0.744027	0.594782	0.204757
Random Forest	0.674753	0.740171	0.598531	0.206338
XGBoost	0.676716	0.744799	0.594227	0.204575

6.1.1 Model Accuracy in Grand Slam Tournaments

Model accuracy for Grand Slam matches played between 2010 and 2024 is shown in Table 6.2. Compared to the broader dataset, all four classifiers obtain notably higher scores, which is reasonable because of the number of key tournament structural characteristics. The frequency of upsets is decreased by top-ranked players' increased participation and more consistent form. Match statistics are captured with greater precision, lowering measurement noise. The longer five-set format for men reduces the impact of short-term variance and increases the impact of underlying skill differences on results. Collectively, these conditions create a cleaner predictive environment in which even moderately calibrated models can translate player attributes into more accurate forecasts.

Table 6.2: Overall performance metrics on Grand Slam matches (2010–2024)

Model	Accuracy	Roc_Auc	Log_Loss	Brier
Decision Tree	0.7347	0.8138	0.5283	0.1768
Logistic Regression	0.7468	0.8271	0.5133	0.1712
Random Forest	0.7642	0.8525	0.4876	0.1600
XGBoost	0.7518	0.8352	0.5054	0.1676
Bookmakers	0.7653	0.8452	0.4888	0.1608

An inspection of Table 6.2 indicates that random forest now offers the best overall calibration, its log loss and Brier score are the lowest and its accuracy (0.7641) sits only a fraction below the bookmaker benchmark (0.7653). This outcome was somewhat unexpected, as gradient boosting was initially considered the likely top performer. The fact that random forest performs better than XGBoost might indicate that additional calibration is necessary for the latter to fully optimise its probabilistic outputs. This outcome is not wholly unexpected, though, as Wilkens [16] discovered that random forest performed marginally better than gradient-boosted trees in tennis match prediction, especially when it came to probability calibration. Although XGBoost's performance does not match random forest across all metrics, it still remains competitive, trailing the forest by about one percentage point in accuracy. Logistic regression and the single decision tree continue to lag behind both ensemble methods. The ordering confirms that ensemble methods preserve their advantage even in this cleaner, lower-variance setting and the near-parity between the random forest and market prices suggests that the model captures most of the information already reflected in the odds.

Considering the ranking of the best performing models presented by Galekwa et al. [19], the models in this study placed third, which is a more than satisfactory result. However, this cannot be considered an unprecedented success, as different scientific studies take into account varying data scopes - whether in terms of time range or types of matches. Moreover, accuracy and related discrimination metrics alone are insufficient to gauge profitability from a betting perspective. Betting success relies on identifying value bets, situations where the model assigned probability significantly differs from bookmaker odds, providing profitable opportunities over the long term. Therefore, the further analysis presented in Section 6.2 evaluates these models in the context of betting strategy performance.

6.1.2 Comparison with Bookmakers' Odds

There are two complementary approaches to tennis match prediction: either as a regression problem that estimates the continuous probability of a Player 1 victory, or as a binary classification task that predicts the match winner. Probability forecasting yields a richer output that can be directly compared with bookmaker odds, whereas classification concentrates on assigning a discrete outcome. In this study, model predictions are evaluated primarily as probability forecasts, reflecting their real-world relevance in betting contexts. Bookmakers' implied probabilities, calculated using the formula described in Section 4.2, serve as a baseline against which model probabilities are compared.

6.2 Betting Strategies

The strategies explored in this section focus strictly on utilizing the predictions of machine learning models rather than traditional betting tactics commonly used with bookmaker odds. Each bet involves a fixed stake of 10 units to ensure consistent comparisons. The evaluation prioritizes both ROI and net profit, emphasizing strategies with robust performance over extensive samples rather than isolated cases of high accuracy. Three distinct betting strategies are investigated: Shorter Price Strategy, Threshold Strategy, and Agreement Strategy.

6.2.1 Shorter Price Strategy

This straightforward strategy entails betting on the player that each model rates as more likely to win in every match, without adjusting for confidence levels or other factors. Although this method offers a thorough and transparent evaluation of each model's overall forecasting capability, it is not the most efficient or realistic strategy to optimise return on investment. The main results produced by using this tactic for every Grand Slam game played between 2010 and 2024 are shown in Table 6.3.

Table 6.3: Shorter Price Strategy results (Grand Slam Matches 2010-2024)

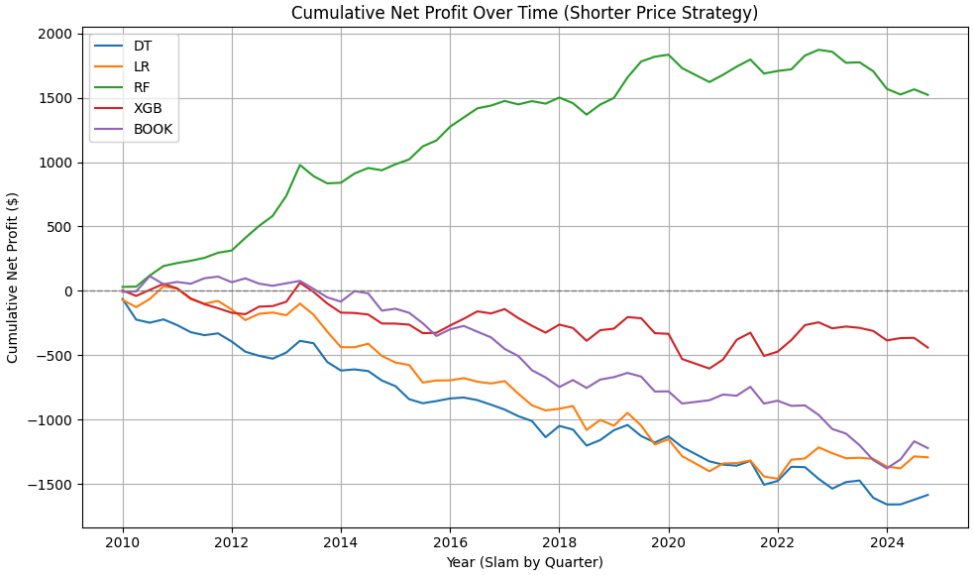
Model	Net Profit	ROI
Decision Tree	-1583.34	-2.12%
Logistic Regression	-1276.94	-1.71%
Random Forest	1616.26	2.17%
XGBoost	-655.94	-0.88%
Bookmakers	-1219.84	-1.63%

The Random Forest model delivers the standout performance in this straightforward strategy with a positive ROI in excess of 2 %, the only configuration to generate an absolute profit over the 59 grand slam tournaments horizon. Moreover, Random Forest achieves its superior financial return despite recording slightly lower point-estimate accuracy than the bookmakers, reflecting its stronger discrimination and calibration, as evidenced by a higher ROC AUC, lower log loss and Brier scores. The XGBoost model, although yielding a modest negative ROI (≈ –0.6 %), still comfortably surpasses the bookmakers' baseline (≈ –1.6 %). In an environment widely regarded as informationally efficient, any systematic edge over the market odds is already noteworthy, as the ability to generate consistent profit indicates the presence of exploitable inefficiencies, as shown by Ramesh et al. [18].

Figure 6.1 charts cumulative net profit under the Shorter Price strategy from 2010 through 2024, with each Grand Slam marked by a quarter tick on the x-axis. The Random Forest's equity curve climbs steadily overall, ending with the highest cumulative gain, but it is not monotonic, several minor drawdowns occur with specific tournaments, indicating periods where market inefficiencies were less pronounced. In contrast, XGBoost presents the most unstable trajectory, with sharper peaks and declines

that reflect its occasionally overconfident probability estimates. Logistic regression and the decision tree both follow downward trends after initial flat or modest gains, underscoring their limited ability to sustain profitability. A detailed, year by year breakdown appears in Appendix C.1.

Figure 6.1: Cumulative net profit over time (Shorter Price Strategy)



6.2.2 Threshold Strategy

This strategy refines the betting process by focusing only on players where the model indicates a clear value advantage over the bookmaker odds. The basic concept is to wager on a player whose estimated probability of winning is greater than the implied probability based on bookmaker odds. However, without additional conditions, this approach could lead to betting on players with very low absolute chances of winning, cases where both the model and the bookmakers assign small probabilities, but the model's estimate is only slightly higher. In order to prevent it, a threshold is applied to the player's odds, guaranteeing that wagers are only made on players whose odds are within a reasonable range, effectively filtering out extreme long shots with outsized risk. Formally, for a given player (Player 1), a bet is placed if both of the following conditions are met:

1. The odds for Player 1 (O_1) are lower than or equal to a preset threshold value, controlling for excessive variance:

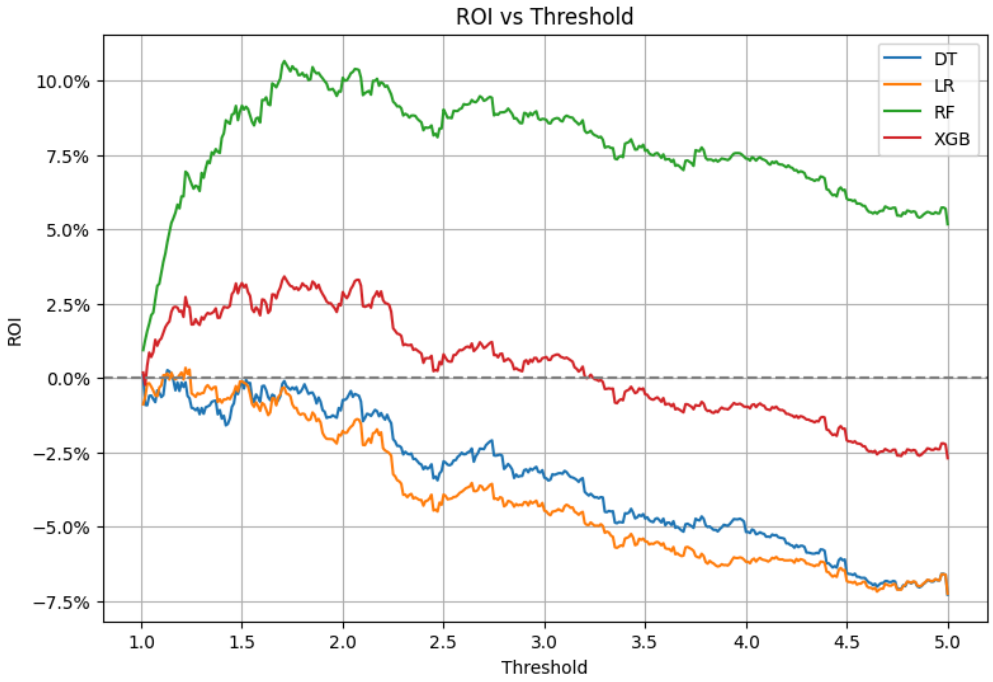
$$0_1 \leq Threshold$$

2. The model's predicted probability of Player 1 winning (P_m) exceeds the implied probability calculated from the bookmaker odds (P_b):

$$P_m > P_b$$

By combining these two conditions, the strategy concentrates wagers on players where the model identifies a meaningful edge and where the risk profile, as reflected by the odds, remains within acceptable bounds. Different threshold values were tested to explore the trade-off between the number of qualifying bets and the return on investment, as illustrated in Figure 6.2.

Figure 6.2:



As the threshold increases from tighter limits (around 1.3) to higher values (up to 1.8), all models initially experience an improvement in ROI. Among the models, Random Forest stands out with the most consistent and substantial gains. Its ROI peaks at over 10% when the odds threshold falls between approximately 1.70 and 1.85, a range that captures many well-priced opportunities without exposing the model to excessive variance. With higher odds, the Random Forest's ROI gradually decreases, as bets on long shots begin to reduce the model's advantage. XGBoost also achieves a positive ROI within a certain threshold range but at a significantly lower level compared to Random Forest. Moreover, as the threshold increases further, its ROI eventually turns negative, indicating that its predictions become less reliable when betting on higher-odds matches. Logistic regression and the decision tree keep flatter, consistently lower ROI curves across the threshold range, highlighting their weaknesses in finding profitable bets. Overall, the figure demonstrates that setting the threshold at an intermediate level, neither too low nor too high, is essential for achieving the best balance between profit and risk.

Table 6.4 depicts the threshold ranges that achieve the highest ROI values under the threshold strategy, representing the most effective configurations across all tested models. In contrast, Table 6.5 reorders those same results by total net profit and presents a different perspective. Random Forest continues to perform strongly when the threshold is raised to odds around 4.0. Although its ROI at this higher threshold decreases to approximately 7%, the larger number of qualifying bets, combined with higher stakes per bet, leads to much bigger cumulative profits compared to competing models. This contrast

highlights the trade-off between maximizing return on individual bets and maximizing overall financial return through scale. Lower thresholds improve ROI per wager but limit the number of betting opportunities, whereas higher thresholds allow for increased volume. For well-calibrated models like Random Forest, these additional bets translate moderate ROI into significantly higher total profits.

Table 6.4: Threshold Strategy results sorted by ROI

Threshold	Model	Bets	Net	ROI	Hit rate
1.71	RF	2045	2113.1	10.33%	0.8592
1.70	RF	2034	2086.3	10.26%	0.8599
1.72	RF	2075	2122.7	10.23%	0.8554
1.75	RF	2148	2195.6	10.22%	0.8478
1.85	RF	2293	2326.1	10.14%	0.8321

Table 6.5: Threshold Strategy results sorted by Net profit

Threshold	Model	Bets	Net	ROI	Hit rate
3.97	RF	4882	3578.5	7.33%	0.6016
3.96	RF	4879	3568.8	7.31%	0.6018
3.95	RF	4879	3568.8	7.31%	0.6018
3.11	RF	4143	3564.4	8.60%	0.6582
3.98	RF	4884	3558.5	7.29%	0.6014

6.2.3 Agreement Strategy

The final explored strategy builds on the idea of model agreement to increase the confidence of betting decisions. By having several models independently agree on a bet rather than depending on just one model's prediction, this method lowers the possibility of false positives brought on by calibration quirks or model errors. The strategy tries to find bets with more predictive support by combining the strengths of different models, which theoretically leads to more reliable selections and mitigates the risk of overfitting to one model's biases. Initially, the Decision Tree model was excluded due to its weaker results in both classification and probability calibration, as well as the presence of better tree-based models in the ensemble.

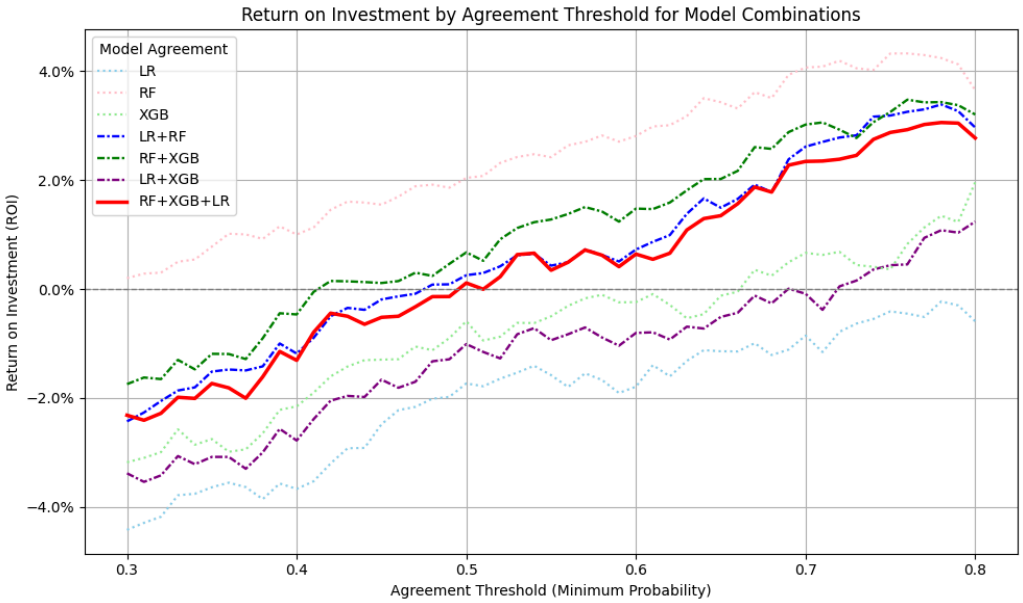
The formal condition for placing a bet is expressed as:

$$\forall j \in \{1, \ldots, n\} : P_j < Threshold$$

This condition states that for a bet to be placed on a given player, the predicted win probability for this player P_j from every model j in the selected model set must exceed a predefined threshold. In other words, all models must independently indicate a sufficiently high likelihood of victory for the bet to qualify. This strict agreement criterion reduces the number of bets but increases their expected quality.

Figure 6.3 demonstrates the outcomes from various model combinations applied under the agreement strategy. The results show that while some combinations yielded positive ROI, single models generally outperformed these combined configurations.

Figure 6.3: ROI for various configurations under the Agreement Strategy



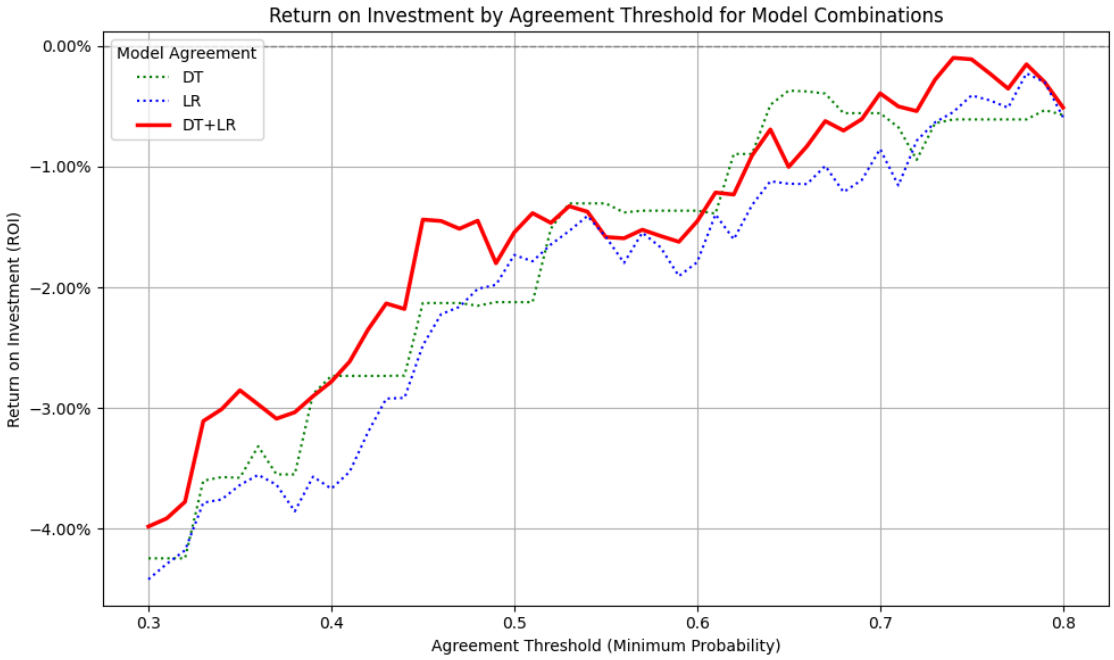
Looking at these trends, it becomes clear that adding more models into the agreement condition gradually reduces ROI. This is examined in greater detail in Table 6.6, which presents the outcomes for every combination at a fixed agreement threshold of 0.76. As more models are added, the ROI decreases from 4.32% for Random Forest alone to 2.29% for the full combination. This highlights the trade-off between increasing agreement and reducing the number of qualifying bets. Although the accuracy increases when multiple models agree (from 0.9397 for Random Forest alone to 0.9423 when all models agree), the accuracy gain is marginal and insufficient to make up for the profitability loss. As a result, despite a slight gain in predictive accuracy, this strategy does not represent an effective approach for maximizing financial returns.

Table 6.6

Threshold	Models	Bets	Net	ROI	Hit rate
0.76	RF	2652	1146.76	4.32%	0.9397
0.76	RF+XGB	2490	865.26	3.47%	0.9406
0.76	RF+LR	2387	776.46	3.25%	0.9422
0.76	RF+XGB+LR	2323	679.46	2.29%	0.9423

These results motivated further analysis. Since combining all three models discussed above did not yield promising results in terms of ROI or net profit, and given that their standalone performance differences were already well known, the next step focused on pairs of models with more similar results. Logistic Regression and Decision Tree, although weaker overall, demonstrated relatively comparable performance patterns. This led to an additional analysis exploring how these two models would perform when combined, under the assumption that models with similar performance levels might interact differently than a mix of strong and weak performers. Figure 6.4 illustrates the ROI for Logistic Regression, Decision Tree, and their combination across varying agreement thresholds.

Figure 6.4: ROI for Logistic Regression + Decision Tree under the Agreement Strategy



Although combining Decision Tree and Logistic Regression does not achieve a positive ROI, Figure 6.4 reveals that at certain threshold levels, their combined performance is better than when the models are used individually. This suggests that even weaker models can complement each other, reducing isolated errors and improving overall outcomes. While the improvement is not enough to generate profit in this

scenario, the results highlight a useful insight. If two models with similar and ideally stronger performance were combined, for example, another model matching Random Forest or XGBoost, the combined results could potentially be improved. This opens a promising direction for future work, where developing or integrating additional models of comparable strength could further improve both the accuracy and profitability of the agreement strategy.

7. Conclusions and Future Work

7.1 Conclusions

This thesis aimed to evaluate the potential for machine learning approaches to predict outcomes in ATP tennis matches and to uncover possible inefficiencies in betting markets. The experiments' outcomes validate the potential for predictive models, if adequately conceived and tested, to attain levels of achievement similar to, and in some cases superior to, those reflected by bookmaker probabilities in the context of Grand Slam events. Of the considered models, the Random Forest classifier consistently revealed the best general performance, striking the best balance between predictive competence, calibration, and profitability. In the context of Grand Slam tournaments, its accuracy reached over 76% - the same level as bookmakers' with only negligible deviation of less than one percentage point, placing it third in models' ranking presented by Galekwa et al. [19]. Additionally, it produced the best probabilistic predictions, as its superior log loss and Brier score metrics reflect. The Random Forest model also excelled over the other classifiers in simulated betting experiments, yielding positive return on investment (ROI) in several approaches. This suggests that, despite the general efficiency of tennis betting markets, exploitable discrepancies still exist, particularly when using advanced probabilistic models. The XGBoost model also excelled, especially in rank-based metrics like ROC-AUC, but its profitability in simulated betting experiments was less stable due to the overconfidence the model occasionally displayed in probability estimates. Logistic Regression and the Decision Tree classifier, being more basic and more interpretable, fell behind the group of ensemble predictors in predictive achievement and financial return, both times stressing the value of modeling the interactions between features and non-linear relationships in this application. The results further confirm that Grand Slam tournaments offer a cleaner predictive environment, with higher model accuracy compared to the full dataset. Importantly, the study demonstrated that market odds and model predictions are closely aligned in aggregate, yet meaningful differences exist on a match-by-match basis. Betting strategies that selectively exploit these differences, such as the threshold-based approach, produced significantly better ROI than naive, straightforward betting schemes, particularly when filtering for moderate odds ranges where model confidence was highest.

In principle, the paper justifies the feasibility of using the data-driven models to not only to predict match outcomes but also to construct profitable betting strategies under specific market conditions. However, it is important to keep in mind the nature of the sport, which inherently cannot be predicted with 100% accuracy. Despite positive results, the models are not guaranteed to succeed in forecasting future matches. This study assessed the effectiveness of these models solely for academic purposes, and therefore their use for potential profit-making is not recommended - particularly considering the bookmakers margins, which place them in a favorable position

7.2 Future Work

Several areas for future research stem from the limitations and findings of this study. The possibilities would enhance the predictive models, expand their scope and make them even more useful in real-world applications.

Extension to Women's Tennis

The current study is only interested in the men's ATP tour. An extension to the Women's Tennis Association (WTA) circuit is a natural step forward. The WTA differs from the ATP in terms of player characteristics, match dynamics and competitive structure dimensions. Comparative analysis can reveal whether match outcomes in women's tennis are qualitatively more or less stable and whether similar features and algorithms are comparably effective in both settings.

Temporal Dynamics of Odds

This study evaluated bookmaker odds as they were before the start of a match. However, in practice, odds are published and adjusted over a period of days or even weeks prior to a match. Bookmakers frequently modify these odds in response to player-specific factors, as well as to balance their financial exposure when betting activity is skewed. Analyzing how odds evolve over time could uncover early inefficiencies or shifts in market sentiment. Incorporating this temporal dimension might lead to the identification of value bets at earlier stages, improving the practical utility of the models.

Feature Expansion with Enhanced Datasets

Although the datasets employed in this study are extensive, they lack certain technical and physiological performance indicators. Metrics such as serve speed, unforced errors, and detailed fatigue measures are not systematically recorded in the primary data sources. Including such features could enable the models to capture more nuanced aspects of player performance. Specifically, precise match timestamps, as opposed to roughly estimated tournament-level dates, would make fatigue modelling viable. Additionally, access to set-by-set or point-by-point statistics would enable the construction of more detailed features, for instance reflecting in-match momentum, potentially improving predictive accuracy further.

Application to Non-Grand Slam Tournaments

While betting experiments in this study concentrated solely on Grand Slam matches, where data completeness and player performance stability are relatively high, further odds analysis encompassing ATP 250, 500, and Masters 1000 tournaments would offer a more thorough evaluation. These tournaments usually involve players with larger disparities in skills and more frequent upset occurrences, and hence the complexity in predicting outcomes might be more challenging. Nevertheless, the larger complexity might also lead to more severe market inefficiencies, providing more room for effective betting approaches to yield profits.

8. Bibliography

[1] S. R. Clarke and D. Dyte, “Using official ratings to simulate major tennis tournaments” Int. Trans. Oper. Res., vol. 7, no. 6, pp. 585–594, 2000.

[2] T. Barnett and S. R. Clarke, “Combining player statistics to predict outcomes of tennis matches” IMA J. Manag. Math., vol. 16, no. 2, pp. 113–120, 2005.

[3] S. A. Kovalchik, “Searching for the GOAT of tennis win prediction” J. Quant. Anal. Sports, vol. 12, no. 3, pp. 127–138, 2016.

[4] M. Sipko, “Machine learning for the prediction of professional tennis matches” MEng. final year project, Dept. of Computing, Imperial College London, London, UK, June 2015.

[5] A. Cornman, G. Spellman, and D. Wright, “Machine learning for professional tennis match prediction and betting” Stanford Univ., Stanford, CA, Tech. Rep., 2017.

[6] C. van Rooij, “Machine learning in tennis: predicting the outcome of a tennis match based on match statistics and player characteristics” M.Sc. thesis, Dept. of Cognitive Science & Artificial Intelligence, School of Humanities and Digital Sciences, Tilburg University, Tilburg, The Netherlands, May 2021.

[7] J. C. Yue, E. P. Chou, M.-H. Hsieh, and L.-C. Hsiao, “A study of forecasting tennis matches via the Glicko model” PLoS ONE, vol. 17, no. 4, art. no. e0266838, Apr. 2022.

[8] Entain, “The rising popularity of live tennis betting” Entain News & Insights, Jan. 29, 2025. [Online]. Available:
https://www.entaingroup.com/news-insights/insights/2025/the-rising-popularity-of-live-tennis-betting/

[9] S. Miller, “Modern tennis rackets, balls, and surfaces” Br. J. Sports Med., vol. 40, no. 4, pp. 401–405, 2006.

[10] D. Cortis, “Expected values and variances in bookmaker payouts: a theoretical approach towards setting limits on odds” J. Prediction Markets, vol. 9, no. 1, pp. 1–14, 2015.

[11] E. Štrumbelj, “On determining probability forecasts from betting odds” Int. J. Forecast., vol. 30, pp. 934–943, 2014.

[12] D. A. Bennett, “How can I deal with missing data in my study?” Aust. N. Z. J. Public Health, vol. 25, no. 5, pp. 464–469, Oct. 2001.

[13] D. De Silva, R. M. Silva, and C. L. Jayasinghe, “A study of batting out-of-form in One-Day International cricket” Estud. Econ. Aplic., vol. 40, no. 1, Jan. 2022, doi:10.25115/eea.v40i1.7041.

[14] B. Morris, C. Bialik, and J. Boice, “How we’re forecasting the 2016 U.S. Open” FiveThirtyEight (blog), Aug. 28, 2016. [Online]. Available:
https://fivethirtyeight.com/features/how-were-forecasting-the-2016-us-open/

[15] C. Bergmeir, M. Costantini, and J. M. Benítez, “On the usefulness of cross-validation for directional forecast evaluation” Comput. Stat. Data Anal., vol. 76, pp. 132–143, Aug. 2014.

[16] S. Wilkens, “Sports prediction and betting models in the machine learning age: The case of tennis” J. Sports Anal., vol. 7, pp. 99–117, 2021, doi:10.3233/JSA-200463.

[17] L.-H. Yuan, A. Liu, A. Yeh, A. Kaufman, A. Reece, P. Bull, A. Franks, S. Wang, D. Illushin, and L. Bornn, “A mixture-of-modelers approach to forecasting NCAA tournament outcomes” J. Quant. Anal. Sports, vol. 11, no. 1, pp. 13–27, 2015.

[18] S. Ramesh, J. Dobelman, A. Kaufman, and L. Bornn, “Beating the House: Identifying Inefficiencies in Sports Betting Markets”, arXiv:1910.08858, Oct. 2019.

[19] R. M. Galekwa, J. M. Tshimula, E. G. Tajeuna, and K. Kyandoghere, “A systematic review of machine learning in sports betting: Techniques, challenges, and future directions”, arXiv:2410.21484, Oct. 2024.

[20] S. Solanki, V. Jakir, A. Jatav, and D. Sharma, “Prediction of tennis match using machine learning” Int. J. Progress. Res. Eng. Manag. Sci., vol. 2, pp. 5–7, 2022.

[21] Z. Gao and A. Kowalczyk, “Random forest model identifies serve strength as a key predictor of tennis match outcome” J. Sports Anal., vol. 7, no. 4, pp. 255–262, 2021.

[22] S. Ghosh, S. Sadhu, S. Biswas, D. Sarkar, and P. P. Sarkar, “A comparison between different classifiers for tennis match result prediction” Malaysian J. Comput. Sci., vol. 32, no. 2, pp. 97–111, 2019.

9. Appendix

Table A.1: Feature groups (thematically)

Group	Features	
Demographic	AGE HEIGHT	
Ranking & Overall Performance Metrics	ATP_RANK ATP_PTS ELO ELO_SURFACE ELO_S CMPLT_S MOMENTUM	 ELO_L CMPLT_L
Win Rates	WINRATE_S TB_WINRATE HAND_WINRATE H2H	WINRATE_L
Serve Quality	ACE_S DF_S 1ST_IN_S 1ST_WON_S 2ND_WON_S SRV_PTS_WON_S SRV_GMS_WON_S SRV_ADV_S	ACE_L DF_L 1ST_IN_L 1ST_WON_L 2ND_WON_L SRV_PTS_WON_L SRV_GMS_WON_L SRV_ADV_L
Return & Pressure	RET_PTS_WON_S RET_GMS_WON_S BP_CONV_S BP_SAVED_S	RET_PTS_WON_L RET_GMS_WON_L BP_CONV_L BP_SAVED_L
Target	RESULT	

Table A.2 Feature groups (in terms of methodological construction)

Group	Features	
Regular	AGE HEIGHT ATP_RANK ATP_PTS TB_WINRATE HAND_WINRATE H2H	
EWMA	ELO_S WINRATE_S ACE_S DF_S 1ST_IN_S 1ST_WON_S 2ND_WON_S SRV_PTS_WON_S SRV_GMS_WON_S RET_PTS_WON_S RET_GMS_WON_S BP_CONVERSION_S BP_SAVED_S	ELO_L WINRATE_L ACE_L DF_L 1ST_IN_L 1ST_WON_L 2ND_WON_L SRV_PTS_WON_L SRV_GMS_WON_L RET_PTS_WON_L RET_GMS_WON_L BP_CONVERSION_L BP_SAVED_L
Composed	CMPLT_S SRV_ADV_S ELO ELO_SURFACE MOMENTUM	CMPLT_L SRV_ADV_L
Target	RESULT	

Table B.1: Model hyperparameters settings

Model	Hyperparameters	Values
Decision Tree	min_samples_leaf max_depth	30 5
Logistic Regression	C penalty	0.215 L1
Random Forest	n_estimators min_samples_leaf max_features max_depth	600 4 sqrt 10
XGBoost	subsample n_estimators max_depth learning_rate lambda gamma colsample_bytree	0.8 500 5 0.01 1 0.2 0.8

Table C.1: Year by year profit under Shorter Price Strategy

Year	DT	LR	RF	XGB	BOOK
2010	-221.00	52.40	164.60	72.40	52.10
2011	-108.40	-126.80	37.70	-158.30	59.30
2012	-196.50	-118.60	293.50	13.10	-71.40
2013	-25.70	-194.30	298.30	-0.10	-90.10
2014	-143.70	-216.20	102.70	-171.20	-102.80
2015	-159.60	-191.00	277.80	-40.90	-196.90
2016	-28.10	23.40	281.80	99.50	-10.70
2017	-252.60	-209.30	47.80	-199.60	-311.20
2018	-22.10	-73.40	7.30	-10.80	-16.50
2019	-18.85	-152.15	384.85	5.15	-92.75
2020	-146.51	-208.71	-216.11	-289.71	-67.51
2021	-181.40	-40.80	-5.10	16.50	-26.10
2022	45.72	226.22	170.42	239.12	-87.88
2023	-146.90	-89.10	-96.20	-110.70	-350.30
2024	22.30	41.40	-133.10	-120.40	92.90