Algorithmic Sports Arbitrage Using Statistical Machine Learning

Matyas Huba

Academic Supervisor: Prof. Philip Treleaven
Technical Supervisor: John Goodacre

BSc Computer Science

2023/24

Department of Computer Science
University College London

Abstract

This thesis investigates the feasibility of employing algorithmic models and Machine Learning (ML) based arbitrage strategies in the sports betting markets. The original research presented in this thesis was conducted in collaboration with Quant Sports Trading and offers applied models for trading in real-time betting markets, specifically during football events. Systematic sports betting is a greenfield market with similar limit order book characteristics to financial markets in terms of volatility, liquidity or order flow imbalance. In sports betting, instead of buying or selling a financial instrument, market participants can back (bet in favour) or lay (bet against) sides in sporting events based on odds. This study investigates value betting and statistical arbitrage in football markets. Consequently, the microstructure of betting markets, which fundamentally differs from traditional exchanges in artificial order lag, is examined to inform the optimal execution of trading signals. These findings are incorporated with exchange data, historical match data and live match updates to establish a robust research platform for algorithmic betting strategies. Finally, the thesis proposes ML value betting models for football that exploit bookmakers' inefficient odds. To achieve the above results, the thesis comprises three empirical experiments that collectively lead to the applied models and their execution in betting markets.

Experiment 1: Research and Machine Learning Platform

The first experiment begins with the integration of in-play state of game data with millisecond exchange data from Betfair. This new, synthetic time-series data allows for truer modelling of inventory balance and probabilities of being executed in order to manage risk and avoid adverse selection. To benefit from these findings the experiment focuses on the development of a research, modelling and machine learning platform for quantitative researchers. The emphasis is on native data querying for feature engineering in machine learning models.

Experiment 2: Machine Learning for Pre-Game Football Predictions

This experiment utilises the Machine Learning platform from Exp. 1 to classify the outcome of upcoming football matches based on historical match data. By extending the Pi-rating system to the novel exponential time-decayed pairwise Pi-rating, the classification models reach similar accuracy to bookmakers.

Experiment 3: Statistical Arbitrage Strategies in Value Betting

To determine the viability of value betting on these preliminary results, the efficiency of starting odds is studied across multiple seasons and baseline models are established. The best performing models from Exp. 2 are then adopted by arbitrage strategies. Lastly, the Kelly Criterion is applied to the results of the classification models to maximise final wealth by deriving the optimal sizing of a sequence of bets.

By conducting these experiments, this thesis presents the following contributions to science:

Understanding of the Betting Market Microstructure

As an unsaturated market, sports betting microstructure has not yet been extensively explored. This thesis presents a comprehensive study of the nuances and anomalies of the betting market such as artificial order lag to help inform efficient order execution.

Novel Integration of In-play and Exchange Data

Granular in-play state of game data has previously been used solely for evaluating individual player and team performance. The studies in this thesis integrate in-play data with millisecond exchange data to generate synthetic features for algorithmic trading models.

Extension of the Pi-rating System for Value Betting

As part of Exp. 2 an extension of the Pi-rating system is developed. Namely, the exponential time-decayed pairwise Pi-rating which generates a comprehensive home and away rating for every team pair. Used within value betting models, the extended Pi-rating outperforms the original system to profit from inefficient bookmaker odds.

Impact Statement

Beyond contributions to science as laid out above, the thesis also presents findings and develops software tools that have a direct effect on risk management and profitability in sports betting. The collaboration with Quant Sports Trading Ltd (QST) has ensured the project stayed grounded in the applications of strategies and not just their theoretical performance.

In addition to the software infrastructure, viable value betting and in-play market-making strategies have been derived for football using classification algorithms and statistical arbitrage, respectively. Most notably, combining high-frequency exchange data with in-play data as introduced in Experiment 1 can help punters and betting businesses such as QST better manage inventory imbalances and; therefore, minimise risk. Overall, the impact of this thesis extends beyond scientific contributions to the profit-oriented sports betting industry.

Acknowledgements

I would like to extend my gratitude to Prof. Philip Treleaven, my academic supervisor, for his guidance and feedback throughout the year. Thank you for answering my calls at the most impossible of times of the day and turning your attention to my thesis. You were not only my academic supervisor, but also the first professor at UCL who I could turn to personally for general advice or help on career progression and postgraduate study. For this, I sincerely thank you.

I would also like to thank John Goodacre at QST for his technical insight and supervision. My gratitude extends to the entire team at QST, in particular, to Ben Schlagman. We spent countless days working together and I could not have asked for a better colleague.

Lastly, I owe my family a great debt of thanks. This short acknowledgement will not do you justice, but it has been because of the support and stability you provided that I could follow my own path in London and immerse myself in my studies. This and, as a matter of fact, any other of my accomplishments is as much your doing as it is mine. Köszi, 3M!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The introductory chapter presents a holistic view of this thesis, covering the key topics of classification and regression models in machine learning, and trading on electronic markets in the context of sports betting. The chapter outlines the motivation for conducting research on algorithmic sports trading and the objectives, experiments and contributions the thesis aims to achieve. It concludes with the thesis structure.

1.1 Research Motivations

First and foremost, this thesis is motivated by the unsaturated nature of the betting market compared to equities or options markets, providing unique opportunities for both original research and commercial endeavours. In addition, the most successful liquidity providers on traditional electronic exchanges operate on microsecond-level execution. While speed is still a driving factor in the profitability of sports trading algorithms, it is orders of magnitude lower frequency compared to financial markets. Hence, the barrier to entry is signifacntly lower.

The similarities between sports betting and financial markets, for example, liquidity provision, limit order book and volatility characteristics offer further incentive to conduct this research. Despite the prevalent academic literature in financial markets, the application of Machine Learning (ML) in live trading scenarios for sports trading remains largely untapped. The thesis aims to bridge this gap by implementing ML-based statistical arbitrage models that can manage risk, balance inventory and lock in profit in live trading. Such practical application of models contributes to the broader understanding of the real-world effectiveness and adaptability of algorithmic sports trading models.

Lastly, the collaboration with Quant Sports Trading Ltd provides the potential for significant contributions to both academia and industry and ensures that the research is grounded in real-world applications.

1.2 Research Objectives

The primary research objectives are to develop a robust research and Machine Learning platform and to implement and evaluate pre-game value betting models designed for football markets. The research platform should implement abstraction layers for handling large amounts of data from different sources with ease. From a researcher's point of view, this speeds up ideation by allowing for efficient, yet expressive data querying and filtering when constructing models.

Betting Exchange and State of Game Data Integration

In conjunction with the research platform, the aim is to gather and integrate football in-play state-of-game data with millisecond-level exchange data and historical match data. The resulting dataset serves as the

foundation for model training and aids models in developing a comprehensive understanding of the market microstructure as well as match specific events, such as momentum swings, drift and sudden changes in odds. Understanding certain limit order book changes from state-of-game data during training enables strategies to extrapolate movement in the back and lay prices when deployed in live trading.

Market Efficiency and Backtesting Evaluation

To explore the betting landscape for football and determine bookmakers' efficiency in setting accurate starting prices, classifiers are trained to predict the outcome (Home win, Away win, Draw) of matches. At this stage, data for the model comes from the historical database and the focus should be on engineering descriptive statistical features. The final models should be backtested to evaluate their accuracy and pinned against bookmaker starting odds to assess profitability. The backtesting framework should be able to replay seasons from the database and run multiple strategies in parallel. As a result, various conditions can be simulated while also taking into account how the different strategies influence each other.

Profitable Value Betting Model

With a practical understanding of the different betting markets football matches offer as well as having developed the data and research infrastructure, the final objective is to implement an arbitrage model for value betting prior to kick-off in football matches. The emphasis should be on ML models and their adaptability to the dynamic betting environment and ability to manage risk and inventory effectively. Further, baseline models are established to provide a benchmark for comparison against ML models and to determine the ML models' relative Sharp Ratio, Sortino Ratio and Final PnL.

1.3 Research Experiments

The main body and content of the thesis consist of three experiments, each investigating a different area of algorithmic sports trading. Yet they are closely aligned to build up to the development of various trading models. The code developed in Experiment 1 is publicly available on GitHub here, while the code for Experiments 2 and 3 can be found here.

Research and Machine Learning Platform

In the first experiment, in-play state-of-game data capturing significant events, player and ball positions is integrated with Betfair's millisecond-level exchange data. To enable efficient data retrieval for research purposes a cloud-native NoSQL time-series database architecture is implemented using MongoDB deployed to Amazon Web Services. The database forms the basis of the machine learning platform that researchers at QST can use to analyse high-frequency data and construct preliminary features for their models.

Machine Learning for Pre-Game Football Predictions

The second experiment focuses on accurately modelling football betting markets as well as exploring the foundation of pre-game value betting models. To model the expected performance of teams the Pi-rating system is extended to the exponential time-decayed pairwise Pi-rating which rates teams as pairs and accounts for the recency of historical matches. The derived ratings are enriched with traditional features such as average shots on target per match to serve as the input for linear classifiers. Lastly, the models' predictive ability is evaluated on four different sets of historical data.

Statistical Arbitrage Strategies in Value Betting

The final experiment builds on previous insights and tools to develop and evaluate ML statistical arbitrage strategies for pre-game football betting. It focuses on deriving strategies from the previously assessed classification models and applying the Kelly Criterion. Additionally, baseline models are defined and novel features are constructed from Exp. 1 data, enriching the strategies and contributing to a comprehensive evaluation of their effectiveness in dynamically adjusting the offered back volumes to lock in profits.

1.4 Scientific Contribution

Novel Football Prediction Technique

This thesis contributes to the field of sports betting by conducting various studies that dive into the nuances and anomalies of the market, bookmaker efficiency, profitability of value betting and in-play high-frequency trading for football markets. These findings from a wide range of subtopics not only add breadth to the current state of research on sports betting but also deep-dive into and contribute to existing research avenues, for example, efficient order execution given artificial order lag or market-making in betting. Additionally, the extension of the Pi-rating system presents more accurate metrics for predicting teams' performance in a specific match-up. This proves useful for bookmakers to adjust their starting prices and, naturally, to other punters who aim to capitalise on market inefficiencies.

Sports Betting Microstructure

Traditionally, in-play state-of-game data capturing key events during matches has served as input for evaluating player and team performance. This thesis takes a novel approach by integrating state-of-game data with exchange data, introducing an augmented dataset for model building. The new set of features that can be engineered from the sensible filtering of data now informs models on sudden market swings, loss of liquidity and inefficient pricing on the exchange. Overall, this integration expands the utility of state-of-game data beyond its conventional application, offering new perspectives and opportunities for understanding and predicting market behaviour in sports betting.

Value Betting with Statistical Arbitrage

While Machine Learning models have long been used to trade financial instruments, this thesis applies it to a new field by adapting ML specifically for sports betting. Unlike traditional financial markets, sports betting involves different types and intensities of market events and price changes. In fact, due to the shortening of odds which is a response to only one side winning, even a simple mean-reversion strategy would fail in betting. Statistical arbitrage models developed as part of this thesis account for such differences in price discovery, informing industry practitioners on value betting and future research. More generally, the application of ML in this context opens up opportunities to improve decision-making and profitability in the algorithmic sports betting domain.

1.5 Thesis Structure

Chapter 2 - Background and Literature Review

This chapter goes into the details of trading on electronic exchanges, sports betting and Reinforcement Learning for trading purposes to provide readers with a comprehensive foundation for the rest of the thesis. It will not only serve as background but will also cover relevant academic literature that is either utilised in the experiments or necessary for their understanding.

Chapter 3 - Research and Machine Learning Platform

Chapter 3 presents Experiment 1, the development of a research and machine learning platform, and the integration of different data sources. The chapter establishes the need for such a platform and to explain how the developed solution is fit for purpose by evaluating the researcher workflow, data querying latency and usability of the machine learning pipeline.

Chapter 4 - Machine Learning for Pre-Game Football Predictions

In this chapter, the focus shifts to understanding the intricacies of football betting, in other words, modelling team performance and developing a range of Machine Learning models to accurately predict the outcome of matches. The Pi-rating system is extensively studied and extensions are developed along with descriptive aggregate team statistics from historical data.

Chapter 5 - Statistical Arbitrage Strategies in Value Betting

Chapter 5 builds on the culmination of results from the previous chapters to implement Machine Learning statistical arbitrage models in the football betting market. It restates relevant literature and presents the research methodology, simulation and evaluation techniques used in deriving ML-based strategies with the Kelly Criterion. Lastly, ML models are compared to baseline models to establish their viability in live trading.

Chapter 6 - Conclusion

The final chapter discusses the main results from each chapter and draws conclusions with respect to the applicability of models and platforms to live trading scenarios. These conclusions will help to reiterate the original contributions this thesis offers to the science and industry of algorithmic sports betting. Lastly, the chapter suggests research opportunities to take the contents and research of this thesis further.

Chapter 2

Background and Literature Review

The chapter provides readers with a background in electronic exchanges, sports trading and Machine Learning for football predictions and arbitrage. Simultaneously, each section highlights the core research and literature that the thesis will build on. The aim is to set readers up for the consequent chapters and experiments.

2.1 Electronic Exchanges and Trading

An electronic exchange refers to a platform where participants interact anonymously via a computer network to execute trades (i.e., buy or sell equities, commodities, etc.). Since the emergence of the internet in the 1990s these platforms and the technological advancements they inspired have transformed financial markets, trading strategies and overall market efficiency. Nowadays, electronic exchanges facilitate algorithmic trading whereby orders can be submitted at millisecond-level by algorithms that statistically model market behaviour and place orders based on predefined risk aptitude [Cartea et al.,]. More recently, significant interest has been developed towards Machine Learning (ML) models and their ability to model the trading environment, learn dynamically and self-improve based on simulated trading scenarios and testing on historical data [Bachouch et al., 2018]. As a result of the clear financial incentive trading offers, there is an ongoing arms race in data collection, low-latency trading infrastructure and proximity to trading venues. The profitability of state-of-the-art algorithms in capital markets has prompted the same level of competition and research in other order book-oriented markets such as cryptocurrency or sports betting. Below is a literature overview of research in the equities and options space, followed by an introduction to market making that together will form the necessary background for techniques applied in this thesis for algorithmic sports trading.

2.1.1 Algorithmic Trading

Algorithmic trading (AT) is the act of executing trading strategies at a speed and complexity that is beyond human ability. Algorithms are deployed to automate or optimise one or more stages in the trading pipeline shown in Figure 2.1 from [Nuti et al., 2011] using complex computational and mathematical modelling. For the purposes of this thesis, the focus is on Data, Alpha model and Execution Model, however, it is necessary to understand how they fit into the larger AT pipeline.

Data

Data refers to data collection from various data sources that in one way or another could inform prices, manage risk and overall help create a more accurate model of the trading environment. Traditional data sources include exchange or order book data which describes the state of an exchange at the millisecond level. This can be either historical data recorded by algorithms during past trading sessions and used largely for backtesting trading strategies, or real-time order book data that is fed into AT models as input for live trading. Order books, the heart of any exchange, act as the public interface for keeping track of the buy and sell offers submitted by market participants for the security the exchange lists [Treleaven et al., 2013].

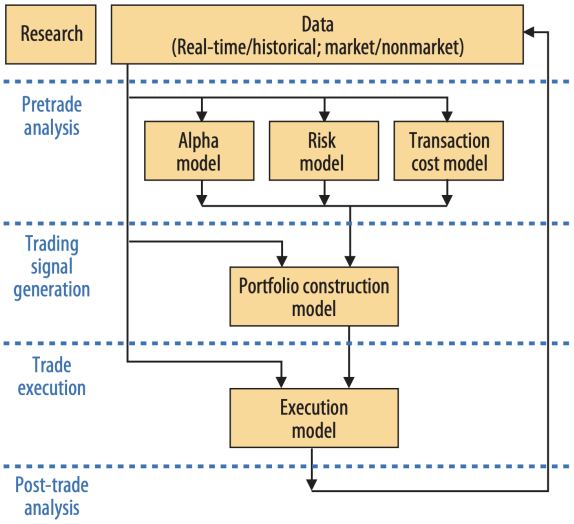

Figure 2.1: Algorithmic Trading System Components

Using the three data points that describe each order, namely price, volume and time of arrival, orders are aggregated and sorted to create the price ladder in Table 2.1, the default visual representation of the order book.

Less trivial and traditional data sources, for instance, social media activity, whose incorporation can lead to more accurate forecasting of macroeconomics or the price of particular securities are referred to as alternative data [Asur and Huberman, 2010]. Online search data scraped by algorithms can be used to track the consumer price index (CPI) or unemployment rates by analysing auto and home sales or even the relative box-office revenue of newly released movies as shown in [Goel et al., 2010]. Metrics such as CPI are used widely for asset pricing, and the ability to track them closely before central authorities release official metrics gives trading models a significant edge. Similar research has been done around the predictive power of satellite data in [Henderson et al., 2012] to measure economic growth. Using data on night light intensities the authors augmented official income growth measures over the Subsaharan region. Their work was taken forward by the authors of [Mukherjee et al., 2021] to directly predict crude oil prices from satellite imagery capturing oil supply in tanks.

Bid Volume	Price	Ask Volume
	1246.00	11,460
	1245.00	14,700
	1244.00	12,000
	1243.00	18,710
	1242.00	12,670
	1241.00	
5,080	1240.00	
13,100	1239.00	
12,000	1238.00	
8,770	1237.00	
6,290	1236.00	

Table 2.1: Limit Order Book Price Ladder

Alpha Model

As part of Pre-trade analysis the Alpha model takes cleaned data from the aforementioned sources to identify market inefficiencies that might be exploited in live trading [Treleaven et al., 2013]. Quantitative analysis, one of the three techniques detailed by Treleaven et al for Alpha modelling in AT, is the application of statistical and mathematical modelling. Generally, this includes differential calculus in models such as Black-Scholes [Black and Scholes, 1973] and stochastic calculus in optimisation problems with stochasticity (i.e., randomness) [Apollinaire and Amanda, 2022]. Further, ML models, traditionally regression and more recently deep neural networks, are trained to extrapolate movement in price, liquidity or volatility from historical order book data.

Execution Model

To act on the trading signals generated by the Alpha model, the Execution model is defined. Regardless of the strength of a signal, if efficient execution algorithms are not in place, the generated revenue is eaten away by transaction cost and/or slippage [Almgren and Chriss, 2001]. Slippage refers to the discrepancy between the expected price of a trade and the actual price at which it is executed [Stoll, 1995]. Stoll explains that this can occur due to the market's unpredictable slide but also as a result of the market impact caused by one's own trading algorithm. A common example of market impact is the case of executing a large buy order, where the order absorbs available buy-side liquidity, causing the price to rise as the order is filled. In conclusion, the Execution model aims to preserve the profitability of trading strategies by optimally submitting orders to the exchange.

2.1.2 Statistical Arbitrage

Statistical arbitrage, a widely studied phenomenon in financial markets, revolves around exploiting statistical relationships between assets to generate profits. This approach has gained significant attention due to its potential to capitalise on short-term market inefficiencies and anomalies as also explored by [Avellaneda and Lee, 2009]. Various studies have explored the theoretical foundations and practical applications of statistical arbitrage, evaluating its effectiveness and risk-adjusted return. One of the important works is by Gatev et al, who investigated pairs trading strategies based on cointegration and mean reversion [Gatev et al., 2006]. Their study demonstrated the profitability of exploiting temporary deviations from the long-term equilibrium between asset prices, highlighting the role of statistical techniques in identifying trading opportunities. Building on this foundation, numerous studies have explored different statistical arbitrage strategies, including momentum trading, mean-reversion strategies, and machine learning-based approaches.

Research by Chan et al. introduced the concept of pairs trading, wherein long and short positions are taken in two correlated assets to exploit deviations from their historical relationship [Chan et al., 1999]. Subsequent studies by Vidyamurthy [Vidyamurthy, 2004] and Hogan et al. [Hogan et al., 2004] expanded on this idea, emphasizing the importance of robust statistical models and risk management techniques in implementing pairs trading strategies effectively. Furthermore, advances in machine learning and quantitative finance have led to the development of sophisticated statistical arbitrage models. [Chong et al., 2017] proposed a deep learning-based approach for pairs trading, demonstrating superior performance compared to traditional methods. Similarly, machine learning techniques such as support vector machines, random forests, and neural networks have been applied to identify profitable trading opportunities based on statistical patterns in financial data.

2.2 Sports Betting

Sports betting allows participants, known as punters to place bets on the outcome of predefined markets before and during sports events. The amount that is placed on a bet is called the wager. In the context of betting, markets are associated with a specific event (e.g., football match) and include predictions about whether both teams will score, full-time score and most notably match odds. Odds encode two key pieces of information - the payout on a bet and the probability that the event associated with the bet will happen.

As an example, assume Home and Away are playing a football match, and the score is 1:0 in the 50th minute. The current match odds are shown in Table 2.2 along with their conversion to total payout and implied probabilities with the formulas $TP = odds \times wager$ and $IP = \frac{1}{odds}$, respectively. Odds throughout this thesis refer to Decimal Odds. For the sake of completeness, note that odds can also be represented in Fractional (e.g., 5/4) and Moneyline (e.g., +150) formats. While their notations are different, they can be converted directly into one another and contain the same underlying information.

Runner	Odds	Total Payout on £100	Implied Probability
Home	1.98	£198	52.5%
Away	5.6	£560	18.85%
The Draw	3.15	£315	33.75%

Table 2.2: Example Odds, Payout and Probabilities

While sports betting is a form of gambling, the purpose of this thesis is to quantify uncertainty when it arises and ultimately handle betting as a rigorous scientific discipline. The preceding literature that the following chapters build on has explored various quantitative techniques to model betting markets and construct statistically significant predictive models. Albert and Koning compiled data-driven studies across Olympic events, football and tennis including the statistical analysis of the FIFA rankings or modelling baseball runs to determine if momentum is a significant factor in the sport [Albert and Koning, 2007]. In 2010, Croxon and Reade conducted the first study into price (i.e., odds) efficiency in 'in-play' markets and found that the betting exchange leads the process of information aggregation and consequently price discovery [Croxson and Reade, 2010]. Since the 2010 Croxon and Reade study much of the research into systematic sports betting has revolved around betting exchanges, their data, efficiency and microstructure.

2.2.1 Betting Exchanges

With the emergence of electronic exchanges for betting (e.g., Betfair), market participants can trade against each other. This allows exchanges to avoid taking on directional risk and instead let participants partake in price discovery based on in-play supply and demand. To generate their bottom line, betting exchanges take a commission on winning bets and transaction fees if certain frequency limits are reached. While fees reduce the profitability of punters, the nature of the exchange presents the opportunity to constantly profit from the betting markets if one can find an edge over other participants. Provided there is sufficient liquidity on the exchange, smart punters can grow their bankroll indefinitely which is not the case when dealing with bookmakers as they are quick to ban punters who steadily lose them money.

Background: Bookmakers

Traditionally, bookmakers set the odds and took on risk to enable gambling which inherently pinned them against profitable participants. When punters place bets on a runner the bookmaker becomes exposed to losses if the outcome favours the bettors. To offset their risk and balance their inventory. bookmakers dynamically adjust their odds to offer a less and less attractive payout and risk ratio to punters. Moreover, to ensure that they remain profitable in the long run and uncertainty is minimised, bookmakers operate under the concept of overround. Overround is the profit margin or *vigorish* embedded in the odds, ensuring that the total implied probability of all possible outcomes exceeds 100%. It can be calculated by summing the implied probabilities of all the odds to back. For example, in Table 2.2 the implied probabilities sum to 105.1% indicating that if a punter bet the same amount on all runners at the given odds, he would lose 5.1% of his initial wager. This 5.1% is the overround and what the bookmaker would pocket. If the sum of implied probabilities is below 100%, say 99.1%, it presents an immediate arbitrage opportunity where a punter who backed all of the runners is guaranteed a 0.9% profit.

Microstructure

Along with the profit structure, the greatest difference compared to bookmakers is the back and lay mechanism in betting exchanges. A back bet refers to betting in favour of the runner, while a lay bet is a bet against the runner, a concept identical to buying or selling a financial instrument on a traditional exchange. This gives participants the tool to actively trade outcomes during an event, allowing for dynamic risk management and profit-locking strategies based on live price shifts. As shown in Equation 2.1, the exposure resulting from a bet can be represented as a tuple where the calculations are taken on the back side:

$$E_b = [v_b \times (p_b - 1), -v_b] \text{ and } E_l = [-v_l \times (p_l - 1), v_l] \tag{2.1}$$

where E_b and E_l are the exposure resulting from a back bet and a lay bet. p_b, p_l, v_b, and v_l are the back odds, lay odds, back bet size and lay bet size, respectively. With a lay bet, the liability is the amount that would be paid to the punter who bought a back on the same event. How much punters can back or lay at different prices is determined by the liquidity of the market. When a bet is submitted to the order book and there is sufficient liquidity at the given price, it is said the bet has been matched. In the context of an exchange, this translates to someone else taking on the opposing direction of the trade. The spread between the highest available back price and the lowest available lay price forms what is known as the back/lay spread, akin to the bid/ask spread in financial markets, representing the cost of executing a trade and the potential profit or loss.

2.2.2 Football Betting Strategies

Predicting the full-time result (i.e., Home, Away or Draw) or modelling expected goals and the various statistics associated with football matches has been of research interest for decades. Conceptually, models can be broken down into two categories, pre-game and in-play. Pre-game models are used for value betting strategies and are concerned with placing bets before kick-off, usually relying on aggregate data points from historical matches, for instance, line-up, a team's average number of shots on target, number of cards or distance covered per match [Shahtahmassebi and Moyeed, 2016]. On the other hand, in-play models have access to some form of a data stream. This might include live updates on significant events (e.g., goals, substitutions, etc.) or more complex streams such as microsecond order book data from betting exchanges. In-play models have the advantage of dynamically modelling the markets and optimising price discovery as new information becomes available [Divos, 2020]. As a result, these models often place thousands of bets per match and can lock in a profit before full-time. Value betting strategies generally do not have this ability, but as highlighted in the literature, contemporary models employ a combination of pre-game and in-play models to better manage risk and identify when prices are out of touch.

Value Betting Strategies

Dixon and Coles set the stage for football betting research by introducing a parametric model using Poisson regression for predicting match outcomes [Dixon and Coles, 1997]. Their work focuses on exploiting potential inefficiencies in the betting market, demonstrating positive returns when employed as the basis for a betting strategy. However, the model's static nature, assuming constant parameters over time, prompted a shift toward dynamic modelling. Karlis and Ntzoufras contributed the dynamic double Poisson model by proposing a bivariate Poisson distribution to capture the correlation between the number of goals scored by teams in matches [Karlis and Ntzoufras, 2003]. This extension improves model fit and prediction, particularly in capturing the number of draws, the outcome that is often underrepresented in models. Building on these foundational ideas, more recent research explored Bayesian networks, feature engineering for ML and the integration of external information such as bookmakers' odds. Constantinou et al. presented the pi-football model that employs a Bayesian network to construct and update team ratings [Constantinou et al., 2012]. Unlike traditional models relying solely on historical data, this model incorporates subjective variables that proved useful for prediction but is not captured in pre-game data. Importantly, the model demonstrated profitability against bookmakers' odds. Egidi et al. were the first to look at improving match outcomes predictions by incorporating bookmakers' odds. Their hierarchical Bayesian Poisson model derived scoring rates for each team from parameter estimation based on historical data and bookmaker odds [Egidi et al., 2018].

In-Play Betting Market

Much of the in-play betting literature has close ties with algorithmic trading in financial markets. The structured and high-frequency nature of the limit order book on betting and financial exchanges lends itself to time-series analysis and ML algorithms in both instances. Dixon et al. trained a Deep Neural Network (DNN) with five hidden layers and 5-minute interval prices to trade foreign exchange (FX) and commodity instruments [Dixon et al., 2017]. Over 9,000 features were engineered enabling the model to memorise historical correlations between instruments. Øvregård looked at Neural Networks (NN) in the in-play tennis market on the Betfair exchange [Øvregård, 2008]. His experimentation included a Multilayer Perceptron (MLP) architecture to identify profitable trading opportunities instead of direct price predictions. Øvregård approached in-play betting as a regression problem instead of a classification one where the output would be discrete (i.e., odds either go up or down). Therefore, the MLP was able to capture the magnitude of price swings and its degree of confidence. Divos was one of the first to give a complete view of the in-play football betting market in [Divos, 2020] and showed many concepts from financial mathematics are applicable in the betting space. Notably, Divos devised the Constant Intensity Model, a risk-neutral framework for pricing and hedging in-play bets, that is analogous to the Black-Scholes model. Divos also compared K-Nearest Neighbour, Linear and NN models in predicting full-time scores from half-time in-play statistics. He found that in terms of log-likelihood, all three models performed similarly, but due to the NN's large number of parameters it was deemed to be the least preferable. While research is focused on ML, more recently Deep Learning, to predict certain outcomes, bet sizing is equally important in live trading as high predictive accuracy. Kelly derived the optimal bet sizing that allows gamblers (in the general sense) to maximize the expected logarithmic return on their wealth also known as the Kelly criterion [Kelly, 1956]. Half and Fractional Kelly criterion, variations of the original Kelly, offer more applicable, but also conservative approaches to determining the optimal size of a series of bets [Maclean et al., 1992]. The strategies from Maclean et al. minimise drawdowns (i.e., temporal losses) in highly volatile environments and are widely used in practice in trading, sports betting and gambling such as blackjack.

Chapter 3

Research and Machine Learning Platform

This chapter introduces and develops the first experiment of the thesis. In particular, the chapter discusses how pre-game, in-play and historical exchange data can be integrated, and details the AWS infrastructure used to develop the research and machine learning platform. It concludes with a discussion of results in several performance metrics such as data querying latency and ease of use for researchers.

3.1 Data

As shown in Figure 2.1, the Data subsystem is at the top of any algorithmic trading pipeline. In sports betting, there are three main sources of data:

- Pre-game
- In-play
- Exchange

either standalone or integrated can be used for backtesting, modelling, trading signal generation or strategy execution. In all use cases, simple architectures utilise only one data source. For example, a plug-and-play machine learning classifier from `scikit-learn` takes a table of historical matches and statistics to predict the full-time result label (Home, Away, Draw). By making a connection between the data sources, the goal is to set up a data strategy that effortlessly handles non-trivial queries and enables researchers to go beyond simple architectures.

Pre-game Data

It is a database of historical events where each row is an event and columns are compiled from basic event data (e.g., shots on target) and post-processed statistics. It is referred to as pre-game because the post-processed data becomes available for querying before the event that is being modelled takes place.

In-play Data

In-play data is the state of game data that comes through point-in-time during an event. In football, this might include each pass, its length and the start and end position of the ball. In-play data is usually lower latency than broadcasting and is used for individual player evaluation, by sportsbooks to adjust their odds based on match dynamics and by media to provide live score tables.

In-play data can be live or historical. Naturally, they have the same exact shape, the difference is how they are consumed. Live in-play data usually comes through a PUSH feed from the provider either directly

into a database or to an HTTP or FTP endpoint. More detail is given on the data feed implementation in Subsection 3.1.3. Historical in-play data is stored in a database as part of the research platform and event files can be queried and retrieved when necessary.

Exchange Data

Studies in this thesis and most of the literature are conducted on Betfair, the world's largest and most liquid betting exchange. Therefore, exchange data consists of Betfair's microsecond order book updates. Updates include the best prices punters can back and lay, liquidity at each available tick or the last traded price, and are used to construct the live price ladder. Note that the shape of the exchange data does not depend on the type of event (i.e., sport) it is associated with. Table 3.1 indicates the fields that are necessary to construct the order book and their description.

Code	Name	Description
pt	Published Time	Time in millisecond since epoch
tv	Traded Volume	The total amount matched across the market
ltp	Last Traded Price	The last price the runner traded at
trd	Traded	Price-Volume tuple delta of price changes
atb	Available to Back	Price-Volume tuple delta of price changes
atl	Available to Lay	Price-Volume tuple delta of price changes

Table 3.1: Fields in the Betfair exchange data that represent the state of the order book. Tuples represent the price and volume that is being offered at that price (i.e., (Price, Volume))

Similarly to In-play data, Exchange data can be live or historical, again, in the same shape. Live exchange from Betfair can be consumed via their REST API or a JSON-RPC connection that triggers a subroutine on the client's side whenever an update becomes available. Historical exchange data comes through the Betfair Exchange Historical Data service with the PRO subscription and is stored in the research platform's database. The PRO subscription provides the same level of granularity as the live updates which is essential for backtesting and deployed strategies that were trained on the historical exchange data.

3.1.1 Historical and In-play Data Collection

While the later experiments in this thesis focus particularly on football betting, it was important to make the research platform more generic for future use. Hence, football, tennis and horse racing data has been collected and embedded into the platform. By their nature, each sport prompts a completely different data structure demonstrating that the research platform is viable in real-life and can easily be extended. Table 3.2 lists all of the data providers used in this thesis and integrated with the research platform.

	Pre-game	In-play	Exchange
Football	FootyStats	StatsBomb	
Tennis	Opta	RunningBall	Betfair
Horse racing	RacingPost	Total Performance Data	

Table 3.2: Pre-game, In-play and Exchange data providers for football, tennis and horse racing

Football

Pre-game: FootyStats FootyStats is one of the most comprehensive data providers for football. They cover over 1500+ leagues worldwide, including the biggest competitions such as the English Premier League (EPL), World Cup, Euro Cup or La Liga. FootyStats provides individual player and aggregate team, league, and most importantly, match statistics with more than 219 fields for historical football matches since 2010.

They also keep track of the odds for the most popular betting markets, in total 68 of them, allowing for modelling that combines pure historical statistics with bookmaker odds, as also seen in [Egidi et al., 2018]. Since the 2016 season, FootyStats also calculates pre-game Expected Goals for each team. Expected Goals, commonly abbreviated as xG, builds on the low-scoring nature of football to approximate how many goals a team should score on average given historical statistics. More background is provided on xGs in Chapter 4.

FootyStats was the preferred data provider not only because of its comprehensive data but also because of the technical specifications and database architecture it utilises. The FootyStats Football Data API ensures easy access to its highly available and distributed MySQL database in JSON format. As the underlying database stores structured data, the API responses can easily be converted to tabular format which is the ideal shape for most ML purposes. In the end, 5320 EPL matches were downloaded, constituting all EPL seasons between 2010 and 2023, using the League Matches endpoint. The endpoint is accessible with the FootyStats Premium subscription for £19.99/month. Figure 3.1 show an example of how League Match data can be used for team analysis and visualisations by showing Arsenal's goal scoring and concerning intensity by 10-minute buckets in the 2018 EPL season.

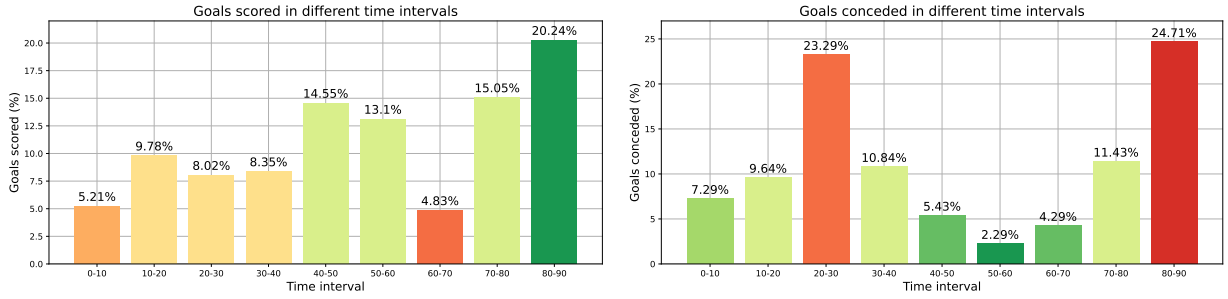

Figure 3.1: Arsenal goal scoring and conceding intensity by 10-minute intervals in the 2018 EPL season

In-play: StatsBomb StatsBomb is recognised as the most advanced in-play football data platform. It is used by coaches and scouts to evaluate players with data-driven statistical methods and by the media and betting industry to provide live updates and dynamic models during matches. StatsBomb utilises its proprietary camera calibration and pitch detection computer vision algorithms to collect data on each player's position, the ball's 3D position, pass start/end location and even more granular data points such as pass footedness (left or right) or shot velocity. With the impressive 2,400 data points StatsBomb creates for in-play analytics, they devised a live xG metric. Each in-play update comes with a live xG associated with the current position. Their calculation not only takes into account the ball and goalkeeper positions but also the proportion of the goal blocked by defenders and defenders that might potentially block the shot but are not yet in the attacker's way.

StatsBomb has developed libraries for Python and R for directly streaming data into clients' applications. Primarily using the `statsbombpy` package's `matches(competation_id, season_id)` and `events(match_id)` data was acquired for Champions League seasons between 2010 and 2018, and the 2015 EPL season, totalling 2630 matches. The JSON files returned from `events(match_id)` contain the sequence of updates on positions, pressure, live xG, etc. for each match. Given the quality of StatsBomb data, it is unaffordable for purely research purposes. Hence, all of the data that was acquired came from StatsBomb's open-sourced database.

3.1.2 Exchange Data Analysis

Betfair's exchange data is at the heart of in-play betting in any sports market. In its simplest form, exchange data is used to represent the price ladder and give punters an accurate view of the market's, and therefore the underlying event's, state. For example, the full-time score and time of goals can be reconstructed from the jumps in the best available back and lay prices for any runner. In a more complex setting, exchange data

allows backtesting, post-trade evaluation of a given strategy and trading signal generation with statistical analysis or machine learning. To carry out these tasks sensibly and efficiently, it is essential to have an in-depth understanding of the shape of the exchange data and how volatility, liquidity and other order book features play into price discovery.

Market Name	# of Runners	Example Runner
Player To Score a Hat-trick?	6	Harry Kane
Match Shots On Target	10	4 Or More Shots On Target
Corners Over/Under 8.5	2	Under 8.5 Corners
Aston Villa +2	3	Tottenham -2
Player First Goalscorer	43	Ryan Sessegnon
First Half Goals 1.5	2	Over 1.5 Goals
Match Odds	3	Aston Villa

Table 3.3: Non-exhaustive list of markets on Betfair during the Aston Villa v. Tottenham match. The number of runners for each market and an example runner are provided for context.

Betfair exchange data was downloaded from the aforementioned PRO Exchange Historical Data service. To develop strategies that are applicable to live trading scenarios, it was important to obtain exchange data for the same time periods that overlap with the pre-game and in-play data. In the end, data, which includes every event that Betfair records, from May 2015 to April 2016 was downloaded for football, tennis and horse racing. In practice, for football, this covers 71,513 events (i.e., matches) and 1,851,338 markets in total associated with the events. Generally, each football match has 26 markets, some of them conceptualised in Table 3.3, that punters can trade on.

Price Discovery

The most liquid market is usually Market Odds, where punters can bet on who they think will win the event. In football, where a draw is allowed, this implies three runners, Team A, Team B and The Draw. It is often the case the Match Odds market exceeds 70,000 order book updates across the 105 minutes the match is considered in-play (90 minutes match time + 15 minutes half-time break) leaving punters with ca. 11 updates per second.

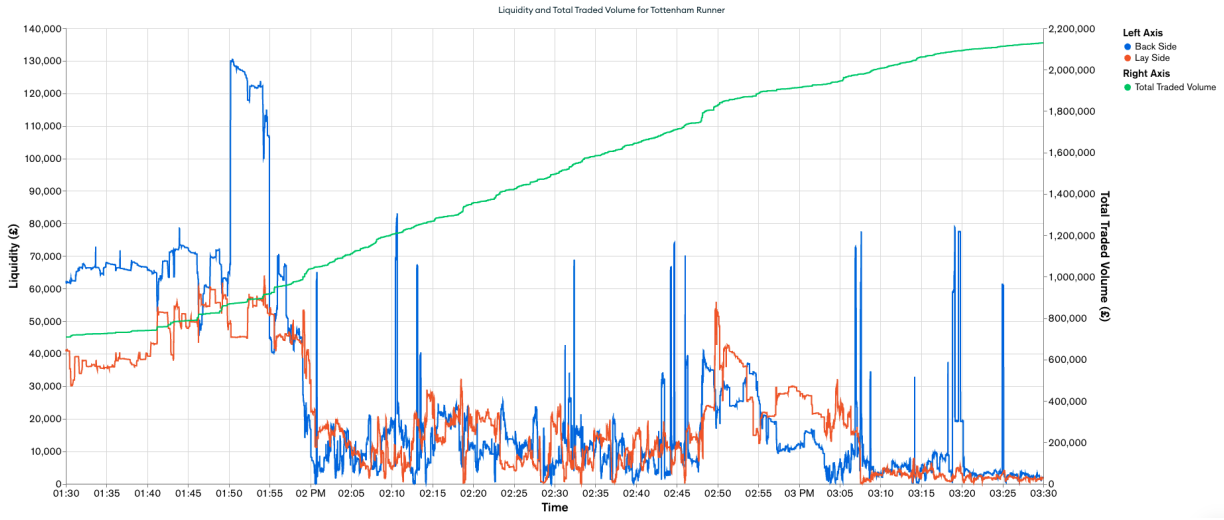

Figure 3.2: Back and lay liquidity progression and Total Traded Volume for Tottenham runner in Aston Villa v. Tottenham

Figure 3.2, created with MongoDB Charts, shows the progression of the liquidity of back bets and lay bets, respectively, in the Match Odds market for the Tottenham runner in the 01.01.2023 Aston Villa vs. Tottenham match. It is apparent that liquidity increases on both sides before the match turns in-play, however, on the back side liquidity is often by orders of magnitude larger. The maximum volume available to back is £130,000 at a given point in time, while on the lay side, it is £63,000 highlighting that punters still predominantly use the betting exchange for the traditional use case where one bets on who the winner will be. Such a large disparity is important to keep in mind for risk management. A strategy largely relying on back bets might be profitable but given the lower liquidity levels on the lay side, it may struggle with balancing its inventory quickly. Slow inventory balancing suggests higher risk levels as in the process of balancing the price can drift further and further away from optimal making it difficult for the strategy to exit positions and leading to losses.

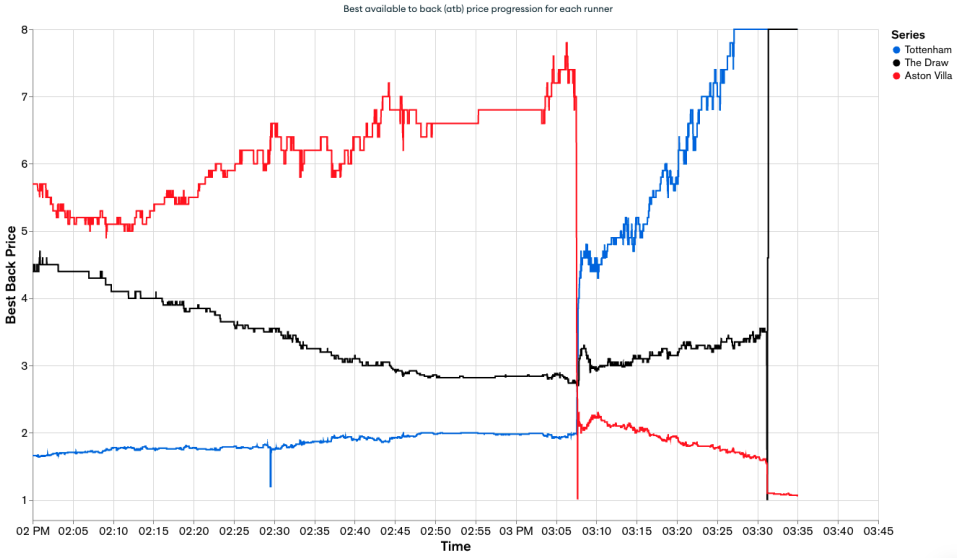

Figure 3.3: In-play Best Available to Back (atb) for each runner in Aston Villa v. Tottenham

Figure 3.3, also created with MongoDB Charts, displays the best available back price in the Match Odds market for each runner (Tottenham, Aston Villa, The Draw). Football games are often characterised by a constant drift in prices. Due to the fixed time limit, back prices tend to steadily shorten for the winning runner and lay prices shorten for the losing runners as time runs out. These assumption hold in Figure 3.3 which is apparent by the slow change in best back price with the exception of post-goal price discovery. It is also clear that goals occurred at 3:07PM and 3:31PM. Both goals were scored by Aston Villa, who started off initially as the underdog (i.e., Aston Villa had the highest available to back price at 5.8 at the start of the match).

3.1.3 Data Integration and Architecture

The need to efficiently store the large amounts of pre-game, in-play and exchange data, and make them readily available for querying, promoted the development of a highly available database architecture in the cloud. A combination of MongoDB and Amazon Web Services (AWS) Relational Database was chosen to define the final architecture shown in Figure 3.4 from MongoDB. The architecture relies on the AWS ecosystem as the current QST infrastructure, whose subsystems were part of the experiments in Chapter 4 and 5, also resides on AWS. However, Google Cloud Platform and Microsoft Azure have similar database services to what AWS offers, therefore, the architecture could easily be migrated to other cloud providers.

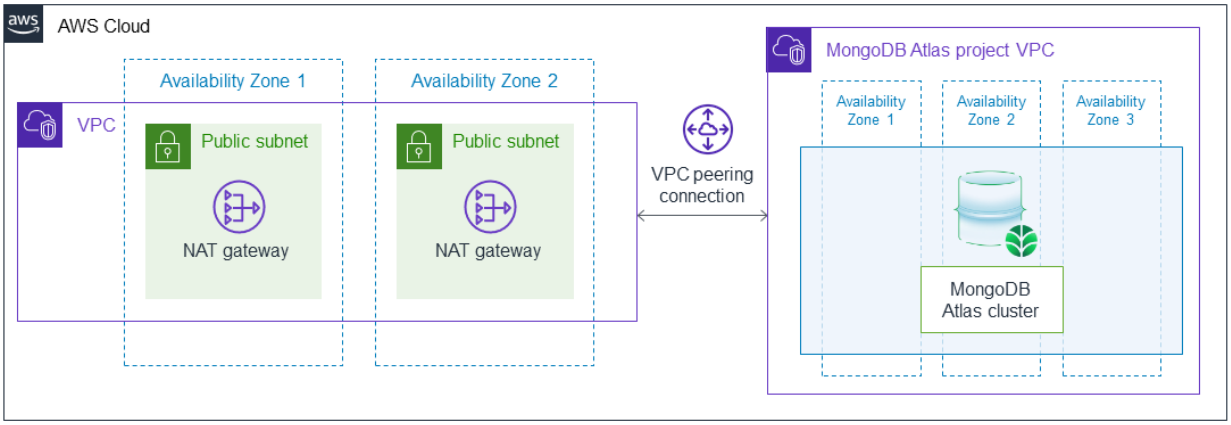

Figure 3.4: Database architecture diagram across MongoDB and MongoDB Atlas on AWS

Database Architecture Rationale

The combination of SQL and NoSQL databases was necessitated by the great difference in data shape and format across the three data sources. To provide some background, SQL stands for Structured Query Language and is ideal for storing data in table format. Columns or fields of a table must be pre-defined and cannot be altered once a table has been set up, hence the naming *structured*. SQL supports relationships between tables by defining primary and foreign keys on particular columns. To work with data in two separate tables the `join` operator iterates through pairs of records and concatenates data where the foreign key of one table equals the primary key of the other. Some join operations have a time complexity of $O(M \times N)$, where M and N are the number of records in the tables making SQL suboptimal where joins are frequently executed on millions of records. SQL data is stored in and queried from a Relational Database Management System (RDBMS). The equivalent AWS service is Amazon RDS which uses the MariaDB SQL engine. Many engines exist for SQL, popular ones include MySQL, PostgreSQL and Oracle, however, they bear no significant semantics difference.

No-SQL databases, on the other hand, offer a flexible data model that is not reliant on a predefined schema. This makes them ideal for handling unstructured or semi-structured data, such as JSON documents or key-value pairs. Unlike SQL, a No-SQL database does not enforce relationships between data entities through foreign keys. Instead, it allows for dynamic storage (i.e., files do not need to be in identical format), enabling developers to store and retrieve data in a more agile and scalable manner. Scalability comes from the property that No-SQL scales horizontally whereby new nodes added to the system allow for more efficient distribution of computational power. SQL traditionally scales vertically by increasing CPU or memory capacity on a single node, an overall less maintainable approach. One popular No-SQL database is MongoDB, which uses a document-oriented data model. In MongoDB, data is stored in collections, analogous to tables in SQL databases, and documents, which are JSON-like objects that contain data. Additionally, AWS's integration with MongoDB helps developers avoid getting locked inside the AWS infrastructure while having all the benefits that MongoDB and its Atlas database manager offer.

Pre-game Data Storage

Pre-game data comes from post-processed match statistics. While the individual stats are different for each sport, they are always a pre-defined and known set of measures. For example, FootyStats maintains 222 columns and Opta 116 columns in its database for each match/game. Given the pre-defined fields and, therefore, the schema, pre-game game data lends itself to storage in an SQL database. A MariaDB Amazon RDS instance was created to store pre-game data for football, tennis and horse racing with the appropriate SQL schemas. Here, data for any sport does not exceed 100,000 records ensuring join operations are executed efficiently. The main tables are `pre-game`, `pre-match` and `pre-race` with secondary tables for data such as league, tournament or competition info.

In-play Data Storage

In-play data is less structured than pre-game due to the plethora of different update types that might occur during an event. Looking at StatsBomb data, there is a total of x update types with different properties, some of them listed in Table 3.4 for reference. If such data is stored in SQL, hundreds of fields would have to be maintained in a table to account for all possible update types and properties. As an example, cards rarely occur but in all $\approx 3,000$ records that constitute a match there would be empty values for `foul_committed`, `card` and `fouled_player`. Following these considerations, a MongoDB was instantiated with one collection per sport. Documents are the JSON files compiled from in-play updates or, in the case of tennis, the XML file converted to JSON.

Update Type	Properties (non-exhaustive)
Pressure	player, possession team, duration, location
Pass	player, recipient, angle, height, body part
Block	player, blocking player, out
Shot	player, technique, one on one?, xg, nearby defenders
Foul Committed	player, fouled player, foul committed, card

Table 3.4: Example StatsBomb update types and associated JSON properties

Exchange Data Storage

Exchange data, similarly to in-play data, is less structured and does not have a predefined schema. Not necessarily all market updates contain the `atb` and `atl` fields, even though they carry the most important pieces of information for punters. A field is included if and only if there has been an update to the previously communicated value implying that data is never repeated. Additionally, one field, for example, `atb` may contain more than one Price-Volume tuple if the volume has changed on several tickers. These properties lead, again, to a MongoDB architecture, but in this scenario storage and retrieval are not trivial. Large market data JSONs often contain more than 70,000 updates per market resulting in files that are over 16MBs which is MongoDB's size limit for a single document. Complexity is further increased by the number of exchange data files which is more than 1.8 million.

AWS S3 with SQL To determine the most suitable NoSQL setup for the exchange data testing was carried out with Time Series Collections and GridFS inside MongoDB, and with S3 buckets on AWS. The naive solution was to upload Betfair's compressed exchange data files onto an S3 bucket and store file paths in an Amazon RDS table along with limited descriptive data to identify which files a researcher might wish to retrieve. The disadvantages of this approach are that *a.)* Amazon S3 and RDS are disjoint services and integrating them introduces an additional layer of complexity, and *b.)* to query exchange data, files would have to be downloaded to a local directory and decompressed every time they are used.

MongoDB GridFS GridFS, MongoDB's specification to access files over 16MBs without having to load its entirety into memory seemed like the appropriate choice. GridFS divides the file into chunks and stores them as separate documents. When querying exchange data with this approach GridFS automatically reconstructs the chunks. However, GridFS stores files in binary encoded format which makes them inapt for dynamic querying and retrieval.

MongoDB Time Series Collection In the end, Time Series Collections was the most appropriate solution given that the `pt` field could act as the timestamp for each update (i.e., document). Time series, in general, reduces file size by relying on the overall common shape of entries to run compression algorithms. Time Series Collections also promises effective compression, which was able to confidently reduce file sizes to under 16MBs. Importantly, Time Series Collections store each update with the `pt` field as a separate document, allowing for efficient querying, plotting with MongoDB Charts and skipping arbitrary time periods from the exchange data.

3.2 Results

3.2.1 Data Richness

The database derived in this chapter and experiment integrates pre-game, in-play, and exchange data. Each serves distinct purposes: pre-game data offers a historical view and key stats needed to build predictive models before games start. In-play data gives a live snapshot of the game, which is key for assessing player performance and updating betting odds. Betfair's exchange data provides a detailed look at betting market trends, essential for financial analysis in sports betting. By bringing these different data types together, the platform enables complex analysis, allowing for more sophisticated model building and research than basic models would allow. This rich data integration supports a thorough approach to creating and testing betting strategies, strengthening the model development and backtesting processes.

3.2.2 Research and Machine Learning Platform

To demonstrate a researcher's workflow consider a scenario where the aim is to access exchange data for conducting a comprehensive bulk backtest of a strategy involving pre-game odds, minute-by-minute odds, and goal counts. Picture this as a series of database queries yielding a list comprising exchange data for matches meeting the condition by the 52nd minute *1.)* three goals, *2.)* pre-game Over 3.5 Goals back price below 4.0, *3.)* current Over 3.5 Goals back above 1.5. It is known the timing of goals is incorporated into the pre-game database, conveniently provided by FootyStats. Attempting to calculate goal timings from exchange data for each query would prove inefficient. Hence, the sequence of queries proceeds as follows:

1. Initially, the pre-game football database is queried to identify matches from 2015 where three goals have been scored by the 52nd minute, resulting in a list of IDs.
2. The obtained list of IDs is then cross-referenced with Betfair events in MongoDB, leading to the retrieval of marketIndex.json files.
3. Subsequently, the marketIndex.json files are filtered in MongoDB to isolate Betfair market IDs associated with "Over/Under 3.5 Goals," generating a list of market IDs.
4. Next, the exchange data files corresponding to the identified market IDs are fetched from S3 and loaded into memory or a temporary MongoDB collection, yielding a collection of JSONs containing historic exchange data.
5. The exchange data is further filtered to identify instances where pre-game Over 3.5 Goals were backed below 4.0 and Over 3.5 Goals were backed above 1.5 at the 52nd minute, resulting in a refined list of JSONs with historic exchange data.
6. Finally, a bulk backtest is performed on the filtered files to analyze the effectiveness of the sports betting strategy.

3.3 Discussion

As data is pulled from several data sources it is tedious to join them for one-off querying and research. To provide easy access and look-up ability for researchers abstraction layers were developed for each data source in Python. Data is pre-processed from the database in Subsection 3.1.3 to achieve a research environment that intuitively integrates data and enables quick construction of statistics and data retrieval for backtesting. The platform also accounts for real-life trading scenarios by only supplying training data that will match the shape data that comes in during live trading.

The developed research and machine learning platform demonstrates practical viability in real-life scenarios, as evidenced by its adaptability to accommodate new sports seamlessly. Its purpose-driven design, outlined in the Research Objectives Section 1.2, ensures its efficacy for research endeavors. Moreover, the integration of pre-game and in-play exchange data, with a focus on machine learning applications, sets the

stage for the subsequent chapter's investigation. Specifically, the incorporation of EPL football data highlights the platform's versatility and readiness for in-depth analysis in upcoming research endeavors. This holistic approach not only streamlines data retrieval and processing but also lays a solid foundation for analysis and modelling, essential for informed decision-making in sports betting strategies.

Lastly, backtesting strategies on any sports has become a more streamlined and automated process with MongoDB's Time Series Collections. Thanks to the `pt` field, which represents the time of data updates, and storing each update as a distinct document, chaining updates and feeding them into QST's backtesting platform has become straightforward. Previously, the platform would retrieve an exchange data file from S3, process it on the client side and then load it packet by packet to the backtest. Now, by storing the already processed exchange packets on MongoDB clusters, the backtests can be instructed to directly access the Time Series Collections. This results in the backtesting platform being hosted completely on the server side and allows researchers to iterate over and validate strategies more effectively.

Chapter 4

Machine Learning for Pre-Game Football Predictions

Chapter 4 revolves around Experiment 2 to model the pre-game football betting market with focus on selected markets and its liquidity and back and lay spread. It goes onto develop a machine learning classification model for predicting the full-time result of matches. Lastly, the chapter helps construct a betting strategy based on the predictions and runs backtests to evaluate its performance.

4.1 Machine Learning for Football

Due to football's popularity across the world and the simplicity of electronic betting, liquidity is high in most football betting markets. In particular, the match odds market where punters can bet on three runners - Home, Away, Draw - is usually the most liquid, with English Premier League, Champions League and Euro/World Cup matches often reaching £10+ million in total wagered value. Subsequently, there is a great financial incentive to accurately forecast the result of matches which prompted research across academics and commercial entities as early as the 1990s.

4.1.1 Predicting Full-Time Result

One of the most prominent papers comes from Dixon and Coles in 1997 where the authors proposed a pure statistical technique employing the Poisson regression model to predict the number of goals scored by each team in a match [Dixon and Coles, 1997]. As the model showed high predictive accuracy, it quickly spread across bookmakers and punters. Therefore, the edge the Poisson (and similar statistical models) in the 1990s and early 2000s provided has soon diminished. More recently, machine learning (ML) techniques with higher algorithmic complexity have become the preferred way of forecasting results. For example, Canizares et al. demonstrated the use of a multi-agent system for predicting football match outcomes [Canizares et al., 2017]. Their study employed a Multilayer Perceptron learning algorithm and validated its approach with the 2015/2016 season data from the Spanish Premier Division reaching an accuracy of 61%.

This chapter delves into the construction and evaluation of several modern ML classification models, employing a rigorous comparative analysis to assess their performance across various accuracy metrics on an extensive collection of datasets. Central to the investigation is the feature engineering to uncover the driver behind the predictive power of the models. By systematically altering and enhancing the input features, the aim is to isolate and identify the characteristics that impact model performance the most. Furthermore, the chapter explores the implications of feature selection and transformation techniques in enhancing model interpretability. Through dimensionality reduction, variable transformation and label/one-hot encoding, raw data is refined into potent predictors, ultimately improving the accuracy and applicability of ML classification models in live betting scenarios.

4.2 Data Transformation and Exploration

Data used in this and Chapter 5 comes exclusively from the Research Platform developed as part of Chapter 3. Thanks to the platform's intuitive programming interface, and low-latency database architecture, retrieving structured FootyStats data ready for transformation and processing was done easily and efficiently. It is important to note that both ML models and betting strategies utilise solely English Premier League (EPL) data. The choice of EPL determines a key aspect of modelling and feature engineering, namely, the number of outcomes of a match (i.e., Home win, Away win or Draw). This is different, for example, from the knockout stage of the Euro or World Cup where the possible outcome is either a Home win or an Away win. Having more than two classes as the potential output prevents the use of traditional binary classifiers which will prompt different encoding techniques below.

To capture various trends, shifts in team performance and rotation of players in the EPL, the past 10 seasons of the league were pulled into the modelling environment. To gain a better idea of the shape and form of data, general statistics, including the minimum, maximum, mean and standard deviation (std) of each field were plotted in Table 4.1.

	Home				**Away**			
	mean	**min**	**max**	**std**	**mean**	**min**	**max**	**std**
Full-time Goals	1.53	0	9.0	1.32	1.22	0	9.0	1.19
Half-time Goals	0.69	0	5.0	0.85	0.54	0	5.0	0.75
Shots	12.37	0	37.0	5.26	10.35	0	31.0	4.66
Shots on Target	5.43	0	19.0	2.67	4.63	0	17.0	2.42
Corners	5.78	0	19.0	3.09	4.68	0	19.0	2.71
Fouls	10.40	0	23	3.45	10.71	0	26	3.56
Yellow Cards	1.60	0	6	1.25	1.81	0	9	1.30
Red Cards	0.06	0	2	0.24	0.07	0	2	0.27
Posesssion %	51.20	12	83	11.90	48.65	8	83	11.88
Odds	2.93	1.02	23	2.13	4.57	1.07	42.75	3.71
Expected Goals (Xg)	1.09	0.52	3.59	0.84	0.94	0.29	2.84	0.72

Table 4.1: Home vs. Away Team Statistics

Table 4.1 reveals clear patterns regarding the performance metrics of home and away teams. Notably, the mean Full-time Goals scored by home teams at 1.53 are higher compared to 1.22 for away teams, indicating a significant home advantage in terms of scoring over the long run. This advantage is also evident in the Half-time Goals and Shots on Target, suggesting that teams tend to perform better offensively in front of their home crowd. A key metric heavily relied upon is Expected Goals (Xg) which further underpins the home advantage by recording a higher average Xg figure of 1.09 for home teams compared to 0.94 for away teams. While these metrics show a clear trend, it is interesting the average Possession % for home teams stands at 51.2%, only slightly higher than 48.65% for away teams. Additionally, the maximum values for both home and away team goals and shots do not differ significantly, indicating that while home teams generally perform better, away teams can have similar performance peaks.

Lastly, betting odds, as indicated by Odds for home wins and away wins, average 2.93 and 4.57 respectively, highlighting a market expectation favouring home teams. The wide range in these odds, with home odds reaching up to 23 and away odds up to 42.75, underscores the unpredictability and variance in match outcomes, which can be influenced by numerous factors that the features and models below in this chapter aim to understand.

The findings presented in Table 4.1 are corroborated by the work of Clarke and Norman, as documented in their 1995 study on the home ground advantage of individual clubs in English soccer [Clarke and Norman, 1995]. This study, like the above analysis, identifies a distinct advantage for teams playing in their home stadium, which is reflected in higher scoring, more shots on target, and higher expected goals metrics for home teams

compared to their counterparts playing away. Naturally, bookmakers understand this advantage too, hence the steadily lower odds (i.e., odds are less favourable from punters' point of view) for home team wins. Even the minimum of home wins odds (1.02) is below the minimum odds for away wins.

4.2.1 Data Pre-Processing

Effective machine learning necessitates rigorous data pre-processing, a process vital for converting raw data into a refined format amenable to ML algorithms. This phase encompasses data cleaning, normalisation, label encoding, and data splitting, aimed at enhancing data quality and ensuring algorithmic efficiency and correctness. Concurrently, data exploration is imperative for identifying inherent patterns, outliers, and correlations within the dataset, providing crucial insights for subsequent feature engineering and training stages.

Data Cleaning

A comprehensive set of pre-processing steps was implemented to clean, transform, and organise the raw football data into a format suitable for analysis and model implementation. This began with the instantiation of the `FootyStatsCleaner` class which shows a streamlined approach to data pre-processing, tailored for the specific needs of ML models focusing on football match statistics.

Firstly, the class filters the matches by ensuring only those with a status of 'complete' are retained, thereby eliminating incomplete or future matches which could introduce noise or inaccuracies into the dataset. This step is necessary for maintaining the integrity of the training set. Subsequently, it applies a date filter to include only matches up to a specified cutoff, aligning the dataset with the temporal scope relevant for the analysis or predictive modelling. The conversion and sorting by date step transitions match timestamps from UNIX to a human-readable datetime format, then sorts these entries chronologically. The ordering is essential for analyses that may consider trends or changes over time, facilitating time series analyses or the creation of features based on temporal patterns, for example, a continuously updating rating system.

Following, the cleaner class selects and renames the appropriate columns from the original dataset to fit the predefined schema expected by downstream ML algorithms. This column selection process is critical, as it discards irrelevant information from the FootyStats data and focuses the data exploration and model's training stages on features likely to influence the outcome of interest, such as match results or goal counts. Recall that FootyStats records more than 219 fields, many of which are for broadcasting or media purposes. Also, out of the 68 odds column, only three are relevant for predicting the Full-Time Result. The full list of retained columns can be seen in Table 4.2. Lastly, two fundamental data cleaning operations are carried out: dropping rows with missing values and duplicates. Removing missing values prevents algorithms from encountering errors or biases during training, while eliminating duplicates ensures that the model is not unduly influenced by repeated entries.

Table 4.2: List of column retained from the FootyStats data

Field	Description	Type	Example
Date	Date of the match	datetime	2020-01-01
HomeTeam	Home Team	String	"Liverpool"
AwayTeam	Away Team	String	"Arsenal"
FTHG	Full-time Home Goals	Integer	2
FTAG	Full-time Away Goals	Integer	1
FTR	Full-time Result	String	"H"
HTHG	Half-time Home Goals	Integer	1
HTAG	Half-time Away Goals	Integer	0
HTR	Half-time Result	String	H
HS	Home Shots	Integer	10

Continued on next page

Continued from previous page

Field	Description	Type	Example
AS	Away Shots	Integer	12
HST	Home Shots on Target	Integer	5
AST	Away Shots on Target	Integer	7
HC	Home Corners	Integer	5
AC	Away Corners	Integer	3
HF	Home Fouls	Integer	10
AF	Away Fouls	Integer	12
HY	Home Yellow Cards	Integer	1
AY	Away Yellow Cards	Integer	2
HR	Home Red Cards	Integer	0
AR	Away Red Cards	Integer	1
HP	Home Possession %	Integer	65
AP	Away Possession %	Integer	35
B365H	Home Win Odds from Bet365	Float	1.5
B365D	Draw Odds from Bet365	Float	3.0
B365A	Away Win Odds from Bet365	Float	2.0
PreHXG	Home xG	Float	1.5
PreAXG	Away xG	Float	1.0
Status	Status of the match	String	"Completed"

Individual Team Statistics

Once equipped with the right columns, match-by-match data was aggregated using the `IndividualTeamStats` class to construct a detailed statistical profile for each team within the dataset. This began with the initialisation of "team dictionaries" (e.g., `team_home_wins`) to calculate various metrics for each team, including wins, losses, draws, goals scored, goals conceded, shots on goal, shots on target, corner kicks, fouls committed, yellow and red cards received, and possession percentages. These metrics are accumulated over match-by-match, as well as segregated into home and away categories to highlight any home-ground advantage or away-game challenges.

Furthermore, the class tracks the number of seasons each team has participated in, providing insights into their experience and longevity in the league and potential funding advantages. It also monitors the performance trends by maintaining a record of goals and goal differences from the last n matches, offering a glimpse into recent form which can be pivotal for short-term predictions or trend analyses.

This systematic accumulation and organisation of team statistics serve as the foundation for feature engineering and constructing descriptive statistics around each team. In Section 4.3, the team dictionaries enable analysis of team performances, shotting accuracy or trends over the seasons.

Pairwise Team Statistics

The `PairwiseTeamStats` class extends individual team statistics by focusing on matchups between teams, using dictionaries to track pairwise statistics (e.g., `pairwise_home_away_goals`). These dictionaries are populated through list comprehension that pairs each team with every other team, explicitly excluding self-pairings, to maintain separate records for home and away scenarios. The approach ensures distinct statistical tracking for each team pairing, allowing for analysis of team performance in specific contexts. For example, comparing Arsenal's performance at home against Tottenham versus their performance visiting Tottenham. Such methodology allows for a comprehensive exploration of inter-team dynamics, including how historical rivalries, advantages, or tactical matchups influence game outcomes. Similar to individual team statistics, the output of this section of the data processing pipeline is used for feature engineering below.

4.2.2 Data Exploration

To enhance the understanding of the data and newly created statistics beyond Table 4.1 and uncover relationships between specific features, correlation matrices and distribution plots were created. The aim was to identify potential relationships and compare the impact of individual and pairwise statistics with the target variable FTR. High correlations between metrics may suggest redundancy, while high correlation with the target variable provides information on statistics to further focus on and refine. These relationships will inform the feature selection and engineering process and enhance the interpretability and performance of the model. The widely known Pearson correlation coefficient was used to measure the linear relationships between two normally distributed variables with the equation

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{4.1}$$

where x_i and y_i are individual data points for the given two features, $\overline{x}$ and $\overline{y}$ are their respective means, and n is the total number of data points. The correlation values obtained range from -1 to +1, with values greater than 0 indicating a positive linear correlation and values less than 0 indicating a negative one. Prior to running the analysis, the FTR label was encoded such that Home win = +1, Draw = 0 and Away win = -1, resulting in a more logical relationship with the -1 to +1 scale outputted by the Pearson coefficient.

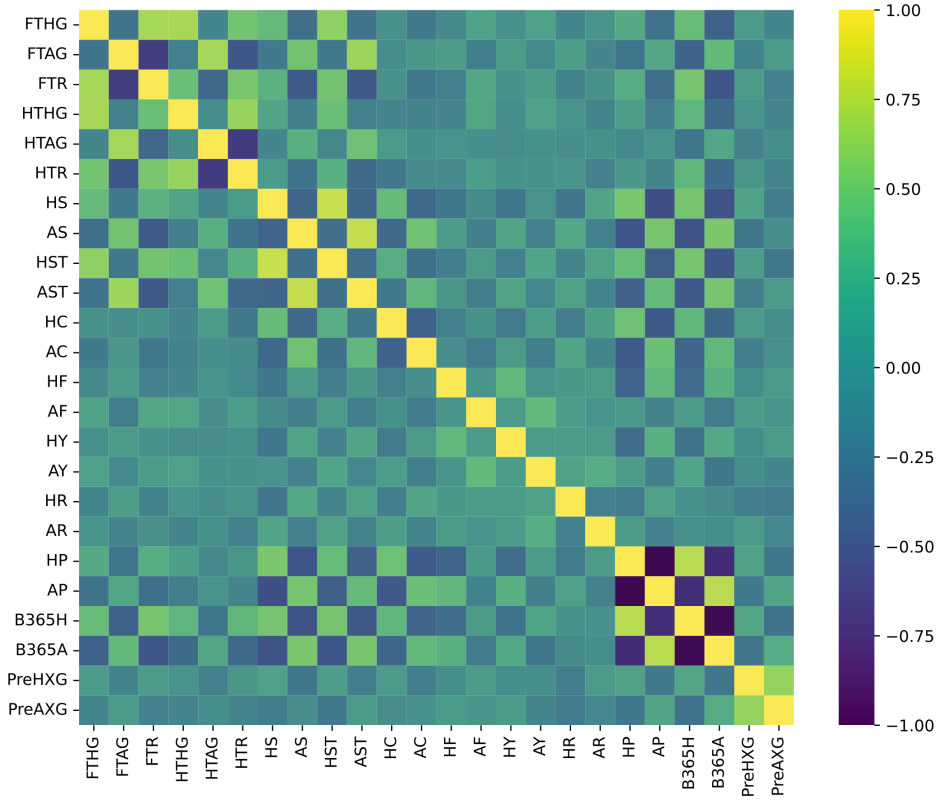

Figure 4.1: Correlation matrix of retained fields from FootyStats

Figure 4.1 shows the correlation coefficients between each feature pair. It is immediately apparent from the brighter regions that relationships across the number of goals (both full-time and half-time), number of shots (both on and off-target), possession percentages, the outcome of the match and odds are the strongest. For example, FTHG (Full-time Home Goals) has a strong positive correlation with HTHG (Half-time Home Goals) and a moderate positive correlation with HST (Home Team Shots on Target). FTAG (Full Time Away Goals) has a strong positive correlation with AST (Away Team Shots on Target) and a moderate

positive correlation with HST. The label of most interest, FTR (Full-time Result) naturally has the highest positive correlation with FTHG and the lowest negative correlation with ATHG. Similarly, HST and AST have a significant correlation with FTR. Interestingly, none of the corner (HC, AC), card (HY, AY, HR, AR) or foul (HF, AF) metrics correlate with FTR or shot metrics. Therefore, they will be excluded from future analysis.

Lastly, given the B365H and B365A (i.e., odds) columns' high correlation with FTR, and HST and AST, which in turn are also correlated with FTR, they will be of great importance in the feature engineering stage. This is because odds (and expected goals - PreHXG, PreAXG) are the only data points the model has access to before a match commences. Expected goals show no notable correlation with any relevant field, therefore, it is projected not to encode predictive accuracy in terms of FTR.

Data Splitting

To model a realistic betting environment, datasets were not randomly split into train and test sets. While random splitting is common in machine learning to mitigate data patterns and biases, for sporting event predictions where the model only needs to forecast future match outcomes, a sequential split approach was chosen. To be able to evaluate models' performance on periods of different lengths with varying training (and testing) data, models are trained on four sets of data; Past One Season (`P1`), Past Two Seasons (`P2`), Past Five Seasons (`P5`) and finally Past 10 Seasons (`P10`). Data splitting for each period can be seen in Table 4.3.

Dataset Name	Period	Split	Seasons	League	# of Matches
`P1`	**Past Season**	Training Set	2022-2023	EPL	190
		Validation Set	2022-2023	EPL	95
		Test Set	2022-2023	EPL	95
`P2`	**Past Two Season**	Training Set	2021-2022	EPL	380
		Validation Set	2022-2023	EPL	190
		Test Set	2022-2023	EPL	190
`P5`	**Past Five Season**	Training Set	2018-2021	EPL	950
		Validation Set	2021-2022	EPL	475
		Test Set	2022-2023	EPL	475
`P10`	**Past Ten Seasons**	Training Set	2013-2019	EPL	1900
		Validation Set	2019-2021	EPL	950
		Test Set	2021-2023	EPL	950

Table 4.3: Data Split for each time period

To carry out data splitting, the `DataProcessor` class was used. It offers flexibility in splitting the dataset into training, testing, and optional validation sets, catering to scenarios where validation is either required or not. By default, the dataset is divided based on a specified train-test ratio, with an 80-20 split being the common practice. When validation is enabled, the testing set is further divided into test and validation subsets which by default equally splits the test set. This additional split facilitates a more nuanced model evaluation, allowing for hyperparameter tuning and performance assessment on data unseen during the training phase. The addition of the `split_data_last_n` method offers an alternative split strategy, segregating the last n records as the test set and the remainder as the training set. This technique will be particularly useful in Chapter 5 where training is done on all data points except the last one.

For data assignment to training, testing, and validation sets, the class ensures that the features (X) and target variable (y) are appropriately segregated, with y representing the match outcomes (FTR). It also prevents data leakage by specifically selecting features for the X datasets in test and validation sets, limiting

the training data to only pre-match variables - team names, date, betting odds and expected goal. This ensures that the models are trained and evaluated on information available before a match begins.

4.3 Feature Engineering

Having cleaned, pre-processed and analysed the FootyStats data with the use of a correlation matrix, the next stage is feature selection to construct and identify the most important characteristics that will be used to train the model. Findings from the correlation matrix, which shows the relationship between the FootyStats fields and the target variable FTR, are taken as the base of the feature selection process. Following, more descriptive statistics and ratios are devised. These are complemented by several popular features from the literature and an independently developed rating system (based on the PI Ratings).

4.3.1 Pi-Rating System

The Pi-Rating system dynamically reflects both teams' performance trends and the importance of match outcomes, ensuring a robust and responsive rating mechanism. Utilising Pi-ratings and its extensions developed in this section a comprehensive set of features are generated including the home team's home rating (`HT_HomeRating`), the home team's away rating (`HT_AwayRating`), the away team's home rating (`AT_HomeRating`), and the away team's away rating (`AT_AwayRating`). The extensions include pairwise rating features with the `PW` prefix to the aforementioned four features (e.g., `PWHT_HomeRating`) and weighted pairwise with `WPW` prefix features. The additional `PW` and `WPW` features are aimed at offering a more nuanced view of team strengths and weaknesses in the context of a specific match-up, an idea similar to what was discussed in Section 4.2.1.

To compute the ratings, first, an initial value of 0 is assigned to each team's home and away rating, which represents an average team relative to the residual team [Constantinou et al., 2012]. One iteration, or match, of calculating pi-ratings involves seven steps as follows:

Calculate Home and Away Team's Expected Goal Difference Home Team α: $\hat{g}_{DH}$ is the goal difference of team α against the average opponent when playing at home. Here, $R_{\alpha H}$ represents the rating of Team α when playing at home.

$$\hat{g}_{DH} = \begin{cases} 10^{\left|\frac{R_{\alpha H}}{3}\right|} - 1, & \text{if } R_{\alpha H} \geq 0 \\ -\left(10^{\left|\frac{R_{\alpha H}}{3}\right|} - 1\right), & \text{otherwise} \end{cases} \tag{4.2}$$

Away Team β: $\hat{g}_{DA}$ is the goal difference of team β against the average opponent when playing away. Here, $R_{\beta A}$ represents the rating of Team β when playing away.

$$\hat{g}_{DA} = \begin{cases} 10^{\left|\frac{R_{\beta A}}{3}\right|} - 1, & \text{if } R_{\beta A} \geq 0 \\ -\left(10^{\left|\frac{R_{\beta A}}{3}\right|} - 1\right), & \text{otherwise} \end{cases} \tag{4.3}$$

Predict the Match Goal Difference Compute the predicted goal difference $\hat{g}_D$ by subtracting $\hat{g}_{DA}$ from $\hat{g}_{DH}$. A positive value indicates the home team is predicted to score more goals than the away team, while a negative value indicates the opposite.

$$\hat{g}_D = \hat{g}_{DH} - \hat{g}_{DA} \tag{4.4}$$

Record the Actual Goal Difference Log the actual goal difference (g_D) from the match, which is the difference between the home team's goals (g_H) and the away team's goals (g_A).

$$g_D = g_H - g_A \tag{4.5}$$

Determine Prediction Error e Calculate the error e as the absolute value of the difference between the predicted goal difference ($\hat{g}_D$) and the actual goal difference (g_D).

$$e = |\hat{g}_D - g_D| \tag{4.6}$$

Exponential Time Decay Apply a decay factor $\alpha^{t_{\text{diff}}}$ to account for the time elapsed since the match. The time difference t_{diff} is calculated as the difference between the date of the latest match d_{latest} and the date of the current match d_{match}, both expressed in days.

$$t_{\text{diff}} = d_{\text{latest}} - d_{\text{match}} \tag{4.7}$$

$$\text{decay} = \alpha^{t_{\text{diff}}} \tag{4.8}$$

Weighted Error Calculation First, calculate the weighted error $\psi(e)$ using a logarithmic function with $c = 3$ as the constant and incorporate the decay. Then, adjust $\psi(e)$ based on prediction accuracy for home ($\psi_H(e)$) and away ($\psi_A(e)$) teams.

$$\psi(e) = \text{decay} \cdot c \cdot \log_{10}(1 + e) \tag{4.9}$$

$$\psi_H(e) = \begin{cases} \psi(e), & \text{if } \hat{g_D} < g_D \\ -\psi(e), & \text{otherwise} \end{cases} \tag{4.10}$$

$$\psi_A(e) = \begin{cases} \psi(e), & \text{if } \hat{g_D} > g_D \\ -\psi(e), & \text{otherwise} \end{cases} \tag{4.11}$$

Revise Pi-Ratings Update team ratings using weighted errors. Hyperparameters λ (for direct adjustments) and γ (for catch-up adjustments) are used.

Team $\boldsymbol{\alpha}$ (Home Rating): Adjust with home error.

$$\hat{R_{\alpha H}} = R_{\alpha H} + \psi_H(e) \cdot \lambda \tag{4.12}$$

Team $\boldsymbol{\alpha}$ (Away Rating): Adjust based on home rating change.

$$\hat{R_{\alpha A}} = R_{\alpha A} + (\hat{R_{\alpha H}} - R_{\alpha A}) \cdot \gamma \tag{4.13}$$

Team $\boldsymbol{\beta}$ (Away Rating): Adjust with away error.

$$\hat{R_{\beta A}} = R_{\beta A} + \psi_A(e) \cdot \lambda \tag{4.14}$$

Team $\boldsymbol{\beta}$ (Home Rating): Adjust based on away rating change.

$$\hat{R_{\beta H}} = R_{\beta H} + (\hat{R_{\beta A}} - R_{\beta A}) \cdot \gamma \tag{4.15}$$

The `PiRatingsCalculator` class implements these seven calculations in an iterative, match-by-match fashion by looping over each match in the dataset. To better visualise the Pi ratings, Figure 4.2 shows the time-series progression of both the home and away ratings of several teams.

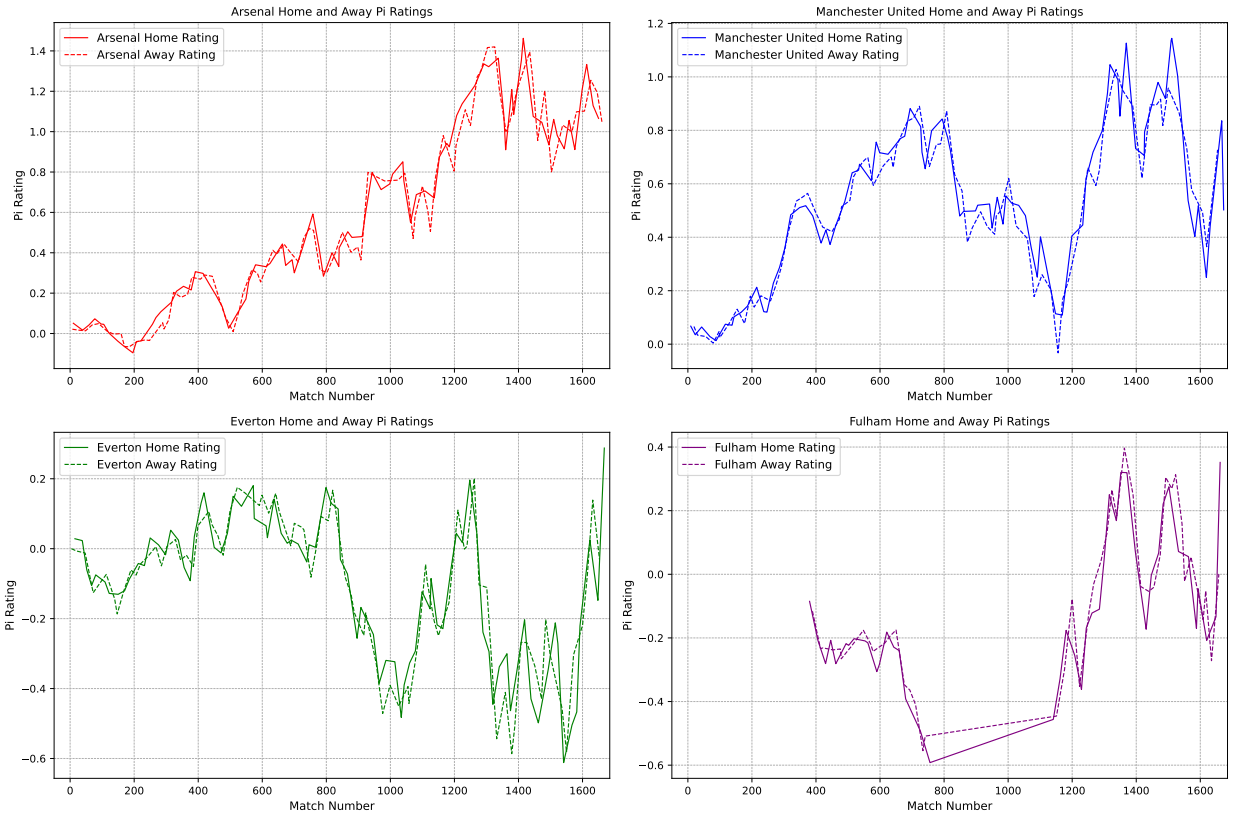

Figure 4.2: Time series graph of Home and Away ratings for Arsenal, Manchester United, Everton and Fulham

The data used for this plot came from the past five seasons, hence the approximately 1700 data points on the x-axis. The plots show a close relationship between the home and away ratings which was an expected behaviour as the Pi rating system adjusts for the home advantage to normalise ratings. Occasionally, the away rating creeps above the home rating, indicating the team performed better than expected in away matches. Crucially, this does not imply the team performed better in away than in home matches. It is also apparent Fulham (bottom right plot) only joined the league later and can be assumed that between approximately match 750 and 1150 they were out of the league.

4.3.2 Pairwise Pi-Rating Extension

Pairwise Pi-Rating

The Pi Pairwise rating is an advanced approach developed as part of this thesis that extends the original system defined by Constantinous [Constantinou et al., 2012]. It evaluates how teams compare against one another in both home and away contexts by calculating unique ratings for every possible match-up. This takes into account the specific dynamics of each head-to-head encounter. The Pairwise Pi model offers a more nuanced and targeted understanding of team dynamics. It does this by considering the specific strengths and weaknesses in direct matchups, thereby capturing unique dynamics and strategies encounters. The increased space complexity that is explained below is a minor compromise for the more information value.

Weighted Pairwise Pi-Rating

A potential issue with Pairwise Pi is overfitting by interpreting noise or anomalies as significant trends. This can especially cause problems if the available data for certain match-ups is insufficient, as the model might overfit to limited or random fluctuations. As a mitigation technique, Weighted Pairwise Pi-Rating

combines the Pairwise match-up specific ratings with the broader home and away team ratings from the original system. Such an integration reduces the risk of overfitting by balancing detailed head-to-head data with general team performance. The model adjusts the weight of Pairwise vs. original ratings to enhance reliability, ensuring a robust, well-rounded analysis of team performance.

Figure 4.3 shows the time-series progression of Weighted Pairwise Pi-Ratings for Manchester City vs. Liverpool matchup. The negative correlation between one team's home and the other team's away ratings is notable as well as general the upward/downward trend throughout the dataset (i.e., past five seasons) for both teams.

With the extended Pi ratings a clearer separation is apparent as compared to the ratings in Figure 4.2 indicating that indeed the Weighted Pairwise ratings better account for team-pair dynamics. There is also a greater variance between home and away ratings further enriching the Pi-systems descriptive performance.

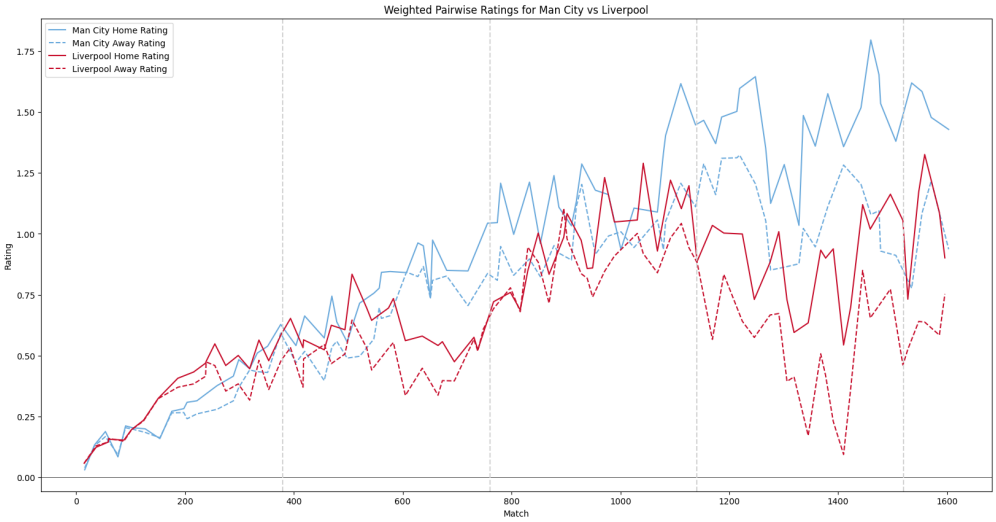

Figure 4.3: Time series graph of Weighted Pairwise Pi-Ratings for Manchester City vs. Liverpool

Implementation

While the calculations in Pairwise and Weighted Pairwise are similar to that of the original rating, thus maintaining similar time complexity, the space complexity is higher. This is due to the need to store unique ratings for every possible pairing among n teams, resulting in $n(n-1)$ unique ratings. This increase in data storage is a reflection of the model's detailed approach.

To extend the original system, the `PiRatingsManager` class was developed. Most notably, the class initialises and keeps track of three important instance variables:

- `self.pi_ratings: pd.DataFrame = self.init_ratings()`
- `self.pi_pairwise: pd.DataFrame = self.init_pairwise_ratings()`
- `self.pi_weighted: pd. DataFrame = self.init_pairwise_ratings()`

To track every match-up pair, `self.init_pairwise_ratings()` initalises $n(n-1)$ dictionaries. In the case of Pairwise Ratings, these are only updated when the given team pair is playing. On the other hand, Weighted Pairwise makes updates to `self.pi_weighted` after each iteration (i.e., match-by-match) based on the predefined weight β. When the given team pair is playing, $1-\beta$ is used to incorporate the pairwise ratings too.

4.3.3 Win Streak

A simple, yet information-heavy feature regarding a team's recent form is the win streak. Intuitively, this keeps track of the number of games (home or away) won consecutively by a team. When a team loses a match, their streak resets at 0. Figure 4.4 the progression of the win streak metric for Arsenal, Manchester United, Everton and Fulham.

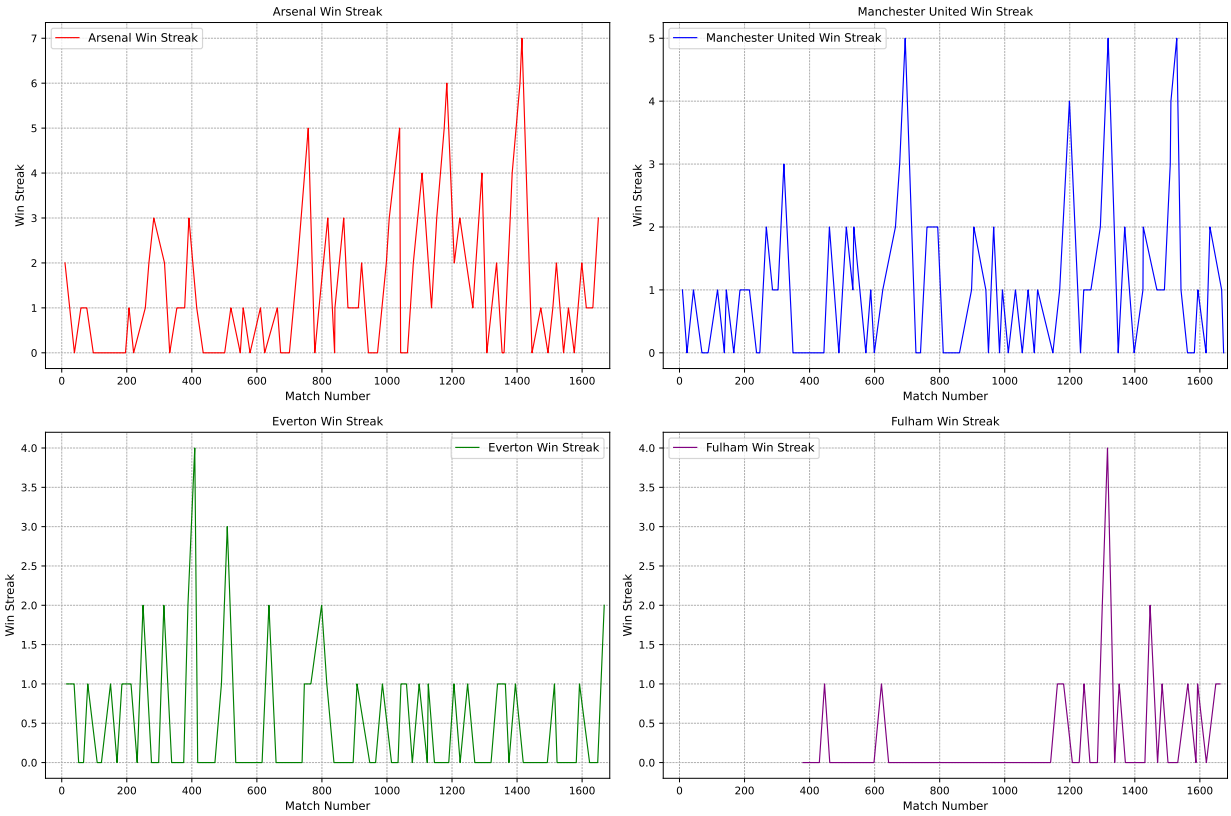

Figure 4.4: Time series graph of the Win Streak feature for Arsenal, Manchester United, Everton and Fulham

4.3.4 Last n-Match Form

Similarly to Win Streak, the Last n-Match Form represents a team's recent performance. However, it is more than Win Streak as it records a wider set of data points than just wins. Last n-Match retains and accumulates on a rolling basis a team's goal difference $G_{diff} = G_{scored} - G_{conceeded}$, total goals scored and average shot accuracy.

4.4 Training and Methodology

Predicting the full-time result (FTR) of matches where a result is one of Home win, Away win or Draw is a multiclass classification problem. Ultimately, the aim is to develop a strategy that can profit from the bookmaker's or exchange's mispriced odds. Odds are implied probabilities for the outcomes making it essential that the ML models not only predict the outcome label but also their confidence in the label. Confidence then directly translates info implied probabilities and therefore becomes comparable with odds. However, for this chapter, the focus is purely on the models' ability to forecast results (although keeping in mind the output should be a set of probabilities), the betting element and its corresponding strategies are introduced in Chapter 5.

4.4.1 Model Selection

In tackling the multiclass classification challenge, the model selection process is guided by several factors including predictive accuracy, capability to handle class imbalance, computational efficiency, and ease of interpretation. The optimal models must not only accurately forecast the full-time result but also effectively estimate the confidence levels of each prediction, translating these into probabilities. Therefore, algorithms must be able to handle the variability of football data, including the hidden correlations between features and the skewness towards certain outcomes (e.g., home advantage). In the end, four classifiers were chosen that match the above criteria.

XGBoost Classifier - XGB

XGBoost, an ensemble-based classification technique, leverages multiple simpler models to refine its predictive accuracy. Initially, it begins with a singular decision tree predicting the mean of the target variable for all samples. Then, it iteratively constructs additional trees, enhancing the model's predictive ability. These trees collectively contribute to the final label prediction through their aggregated outputs.

CatBoost Classifier - CAT

The CatBoost Classifier, a more advanced ensemble method, boosts predictive accuracy by integrating various simpler models. It initiates with a foundational decision tree, similar to XGBoost and sequentially constructs additional trees based on the gradient of the loss function, halting once a specific criterion is met. Deeper models can discern more complex patterns but pose the risk of overfitting. However, this can be mitigated through cross-validation or grid search.

Support Vector Machine - SVM

The Support Vector Machine (SVM) is a supervised learning algorithm. It operates by identifying the optimal hyperplane which separates different class labels with the maximum margin. This is achieved by transforming the input data space into a higher-dimensional space where a linear separation is possible if the data is otherwise not linearly separable in the original space. The key components, support vectors, are the data points closest to the hyperplane and are critical in defining the margin. SVM models are highly adaptable, allowing for customisation through the kernel trick, which adjusts how data is transformed, catering to both linear and non-linear datasets.

Random Forest Classifier - RFC

The Random Forest Classifier (RFC), an ensemble machine learning algorithm, uses numerous uncorrelated decision trees created from data and feature subsets. Every tree makes a class prediction for an input, and these predictions are combined for the final model output. This method diminishes overfitting and improves accuracy. The RFC was set up with balanced class weights in response to the underrepresentation of draw classes in the dataset. Adjusting class weights according to their frequency aims to reduce data bias and enhance the representation of all outcomes.

4.4.2 X Set Construction

Following feature engineering in Section 4.3 the FootyStats data is now transformed into team ratings and features with high (either positive or negative) correlation with the FTR result label. However, for use in the selected classifiers having these metrics calculated for a single point in time (e.g., end of the season) is not sufficient. Instead, features must be appended to every row, or match, in the dataset to ensure training closely resembles the testing environment.

To achieve this, the `XTableConstructor` class was defined which, most importantly, takes in a dictionary of arguments `**kwargs` that specifies which set of features to append to the training set `X_train` and, later to the test set `X_test`. The full set of features is displayed in the JSON in Figure 4.5.

```
feature_param = {
    "GOAL_STATS": True,
    "SHOOTING_STATS": True,
    "POSSESSION_STATS": True,
    "ODDS": True,
    "XG": True,
    "HOME_AWAY_RESULTS": True,
    "CONCEDED_STATS": True,
    "LAST_N_MATCHES": True,
    "WIN_STREAK": True,
    "PAIRWISE_STATS": True,
    "PI_RATINGS": True,
    "PI_PAIRWISE": True,
    "PI_WEIGHTED": True
}
```

Figure 4.5: JSON Configuration for Feature Selection

Important to note that one set of features (e.g., `PI_RATINGS`) involves several feature columns (e.g., `HomeRating`, `AwayRating`). Allowing for quick inclusion and/or exclusion of features in the JSON format portrayed in Figure 4.5 will assist greatly in feature selection, model comparison and evaluation with different feature sets and hyperparameter tuning. Once the `feature_params` JSON has been defined, `XTableConstructor` iterates through each match in the training set, applying the feature inclusion logic specified in the JSON. For each match, it computes the relevant statistics and ratings for both home and away teams from all the available data up to the given match (i.e.,) point in time.

Lastly, `XTableEncoder` is used to encode and normalise the features, ensuring that the data is in a suitable format for the classification algorithms. Initially, categorical variables with string values, specifically the home team (HT) and away team (AT) columns, are transformed into a numerical format via one-hot encoding. One-hot encoding expands these columns into multiple binary columns, each representing a unique team, thereby eliminating model interpretation issues associated with categorical data. Following encoding, the class applies Min-Max scaling to bring all variable measurements to a common scale, between 0 and 1, which improves the algorithm's convergence speed and stability during the training phase.

y Set Construction

The target label, FTR is in the format of "H" for home win, "A" for away win and "D" for draw. To be able to input FTR into the multiclass classifiers, label encoding was implemented to map the string values to integers. The encoding follows the pattern: `{"H": 0, "A": 1, "D": 2}`.

4.5 Testing and Evaluation

When testing, the input table (i.e., `X_test`) initially has the columns: Home Team, Away Team and, optionally, pre-match odds from Bet365. Data from `X_train` is used to retrieve and append the most up-to-date statistics and ratings to each match in the test set following a similar logic to `XTableEncoder`.

To obtain preliminary results, the models (`XGB, CAT, SVM, RFC`) were tested on the full set of features for each dataset defined in Table 4.3. Additionally, a benchmark SVM model (`BENCH`) was set up that only has access to the Team Performance Statistics.

4.5.1 Evaluation

After training, K-fold cross-validation was conducted to evaluate the models based on the following metrics: Accuracy, F1 Score, Precision, and Recall. Each metric holds distinctive significance in assessing the

effectiveness and reliability of the models in the predictive context.

$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.16}$$

Accuracy, considered the fundamental metric, provided an overall measure of the correctness of the model's predictions by assessing the ratio of correctly predicted games to the total number of them. Additionally, the F1 Score, which is the harmonic mean of precision and recall (see Equation 4.16), offers a balanced assessment of the model's performance. It signifies the model's ability to identify positive cases while minimizing false positives and negatives.

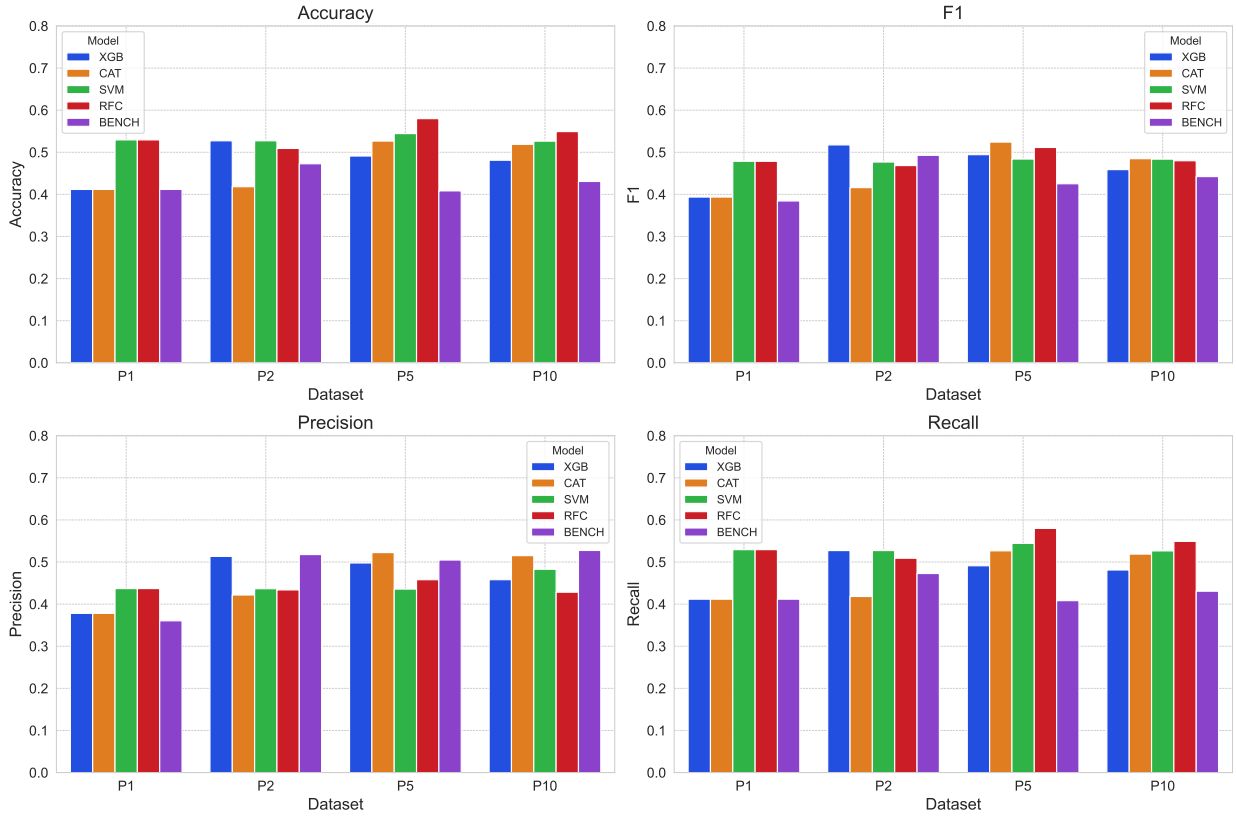

Figure 4.6: Preliminary Accuracy, F1, Precision and Recall results from models XGB, CAT, SVM, RFC, BENCH

Table 4.6 presents the preliminary accuracy, F1, precision and recall metrics from these models on every dataset. From the accuracy results, it is clear that `BENCH` is overall inferior to models with a more comprehensive set of features. On the other hand `RFC` and `SVM` are steadily at the top of the accuracy metric, while `CAT` and `XGB` show lower numbers with greater fluctuation across datasets. In terms of datasets, `P5` was identified as the most optimal with `RFC` reaching an accuracy of 58.52%, which was the highest percentage measured in this test. In practical terms, this means that based on 950 training samples from the 2018 to 2021 seasons, `RFC` was able to accurately predict the outcome of 278 matches out of 475 in the test set (i.e., from the 2022 and 2023 seasons). This indicates that general trends and power relations hold for several years in the English Premier League, hence ratings and statistics maintain their predictive power into the future.

4.5.2 Feature Selection using SHAP Importance

A thorough feature selection method is carried out next to improve the accuracy of the preliminary models and eliminate potentially redundant, or even harmful features. To do so, the Shapely Additive exPlanations (SHAP) was used which scores each feature in terms of its direct impact on the final predictions. Through an iterative process, features with lower SHAP values are gradually removed from the training set, resulting in a remodelled data set without low-impact features. This improves model interpretability and improves predictive power and efficiency as less data is explored during training.

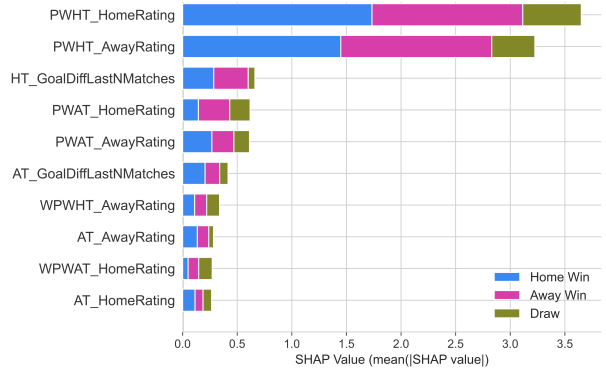

Figure 4.7: Top 10 Most Significant Features from SHAP Analysis

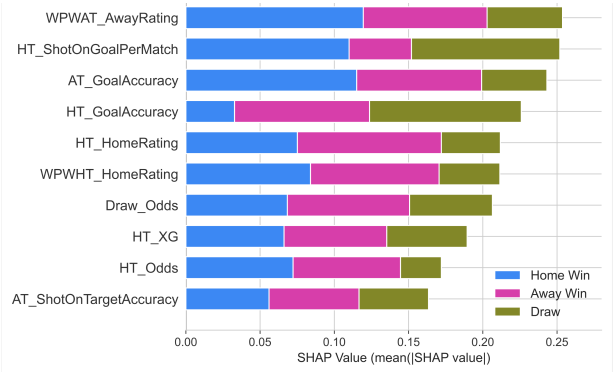

Figure 4.8: 11th to 20th Most Significant Features from SHAP Analysis

This methodology was applied to the `CAT` model, which demonstrated a high preliminary accuracy (52.32%) on the `P5` dataset and is compatible with SHAP. The SHAP values were computed on (`X_train`) using the SHAP Tree Explainer, providing insights into the impacts of each feature on specific label (i.e., Home win, Away Win, Draw) predictions. Figure 4.7 presents the Top 10 features based on SHAP and their significance for each label. The plot highlights the power and robustness of the newly developed Pairwise Pi systems as they contribute to the model's output by orders of magnitude more than the original Pi-rating or descriptive statistics. The major drop in importance from the Pairwise Home Team Ratings (`PWHT`) to the Pairwise Away Team Ratings (PWAT) suggests redundancy between rating pairs. This is explained by high home team ratings directly implying lower away team ratings (as the rating values are normalised). The redundancy is similar to that of possession metrics. If one team's possession metric is x, the other team's metric can be inferred from $100 - x$. Following, a list of feature sets was for removal based on SHAP values.

It is only in the Top 11-20 features, seen in Figure 4.8, where the Team Statistics feature set appears. Namely, Shots on Goal Per Match, Goal Accuracy and Shots On Target Accuracy. Interestingly, the Bet365 odds only placed as the 17th (`Draw_Odds` and 19th (`HT_Odds`) most significant features with SHAP values of 0.21 and 0.16, respectively. Comparing these to the SHAP values of 3.21 and 3.6 for the Pairwise Pi Ratings indicates the superior predictive accuracy of the engineered ratings over bookmakers' odds.

The `feature_params` JSON resulting from the final, and restricted set of features the models will use is presented in Figure 4.9. By default, feature sets are set to `False`, therefore, the JSON needs only to contain the included features. In the end, 7 sets of features were removed, which added up to 28 feature columns. Such a large reduction in model complexity presents the potential for improvements in predictive accuracy, effectiveness as well as explainability.

```
feature_param = {
    "GOAL_STATS": True,
    "LAST_N_MATCHES": True,
    "PI_RATINGS": True,
    "PI_PAIRWISE": True,
    "PI_WEIGHTED": True
}
```

Figure 4.9: Final JSON Configuration for Feature Selection after SHAP Analysis

4.6 Results

The final training and testing of the models was done with the selected features in Figure 4.9 and on all five datasets to provide a more comprehensive account on feature strength across multiple seasons. Overall, the implementation of SHAP-based feature selection increased the model's final accuracy, elevating it from 58.52% to 60.05%. This method has highlighted the relative importance of the extended Pi Rating system (e.g., Pairwise Pi) in predicting match outcomes to bookmaker odds or team performance metrics and the importance of robust feature selection.

4.6.1 Final Model Accuracy Metrics

The results, including accuracy, F1 score, precision, recall and crossv-alidation for each model on all four data splits, are presented in Table 4.4. The Random Forest Classifier attained the highest accuracy of 60.05% on the `P5` dataset, with the mean accuracy across all models standing at 53.38%.

The final results from the machine learning models provide a comprehensive view of The F1 score is a balanced measure that combines precision and recall by computing the harmonic mean of the two—precision is the ratio of true positive predictions to all positive predictions made by the model, while recall, also known as sensitivity, measures the ratio of true positive predictions to all actual positives. Cross-validation is a technique to assess the generalisability of the models, providing an average score across multiple partitions of the data to ensure the model's robustness against overfitting.

Analysing Table 4.4, it is apparent that alongside RFC and XGB displays a consistent performance across datasets P1 to P10, with slight variations in F1, precision, and recall scores. The benchmark model BENCH demonstrates inferior cross-validation scores, particularly on dataset P1, suggesting its low reliability in unseen data scenarios. The XGB and SVM classifiers show less variability across datasets, with the XGB model performing better in cross-validation on P1 but with a decrease in F1, precision, and recall as the dataset size increases to P10. Conversely, the CAT model maintains a relatively stable F1 score but with a noticeable decline in precision and recall from P1 to P10.

Table 4.4: Performance Metrics from Final Models XGB, CAT, SVM, RFC, BENCH

Model	Dataset	Accuracy	F1 Score	Precision	Recall	Cross-Validation
`XGB`	`P1`	53.98%	0.52	0.53	0.53	0.69
	`P2`	51.02%	0.5	0.53	0.49	0.65
	`P5`	54.09%	0.48	0.49	0.47	0.54
	`P10`	49.15%	0.42	0.45	0.46	0.51
`CAT`	`P1`	41.94%	0.43	0.46	0.41	0.66
	`P2`	49.49%	0.5	0.52	0.49	0.65
	`P5`	49.10%	0.48	0.48	0.49	0.56
	`P10`	48.31%	0.45	0.46	0.48	0.51

Continued on next page

Continued from previous page

Model	Dataset	Accuracy	F1 Score	Precision	Recall	Cross-Validation
SVM	P1	54.83%	0.48	0.44	0.53	0.61
	P2	54.48%	0.48	0.44	0.53	0.62
	P5	55.29%	0.48	0.43	0.54	0.56
	P10	53.09%	0.48	0.48	0.52	0.55
RFC	P1	53.15%	0.48	0.44	0.53	0.64
	P2	55.22%	0.5	0.5	0.51	0.66
	P5	60.05%	0.47	0.43	0.51	0.58
	P10	55.83%	0.51	0.55	0.56	0.57
BENCH	P1	48.45%	0.48	0.44	0.53	0.53
	P2	47.51%	0.47	0.47	0.47	0.46
	P5	49.34%	0.55	0.57	0.54	0.47
	P10	45.05%	0.46	0.49	0.45	0.46

The confusion matrix in Figure 4.10 shows the three out classes: Home Win, Away Win, and Draw. The diagonal cells (41 for Home Win, 34 for Away Win, and 19 for Draw) represent the number of correct predictions that the RFC model made for each class, which are true positives. The off-diagonal cells show the misclassifications: for instance, 15 occurrences where the model predicted an Away Win but the actual outcome was a Home Win, or 12 occurrences where a Draw was predicted but it actually was a Home Win. The darker the cell colour, the higher the number of observations for that particular combination of predicted and actual values, as indicated by the colour scale on the right side of the matrix.

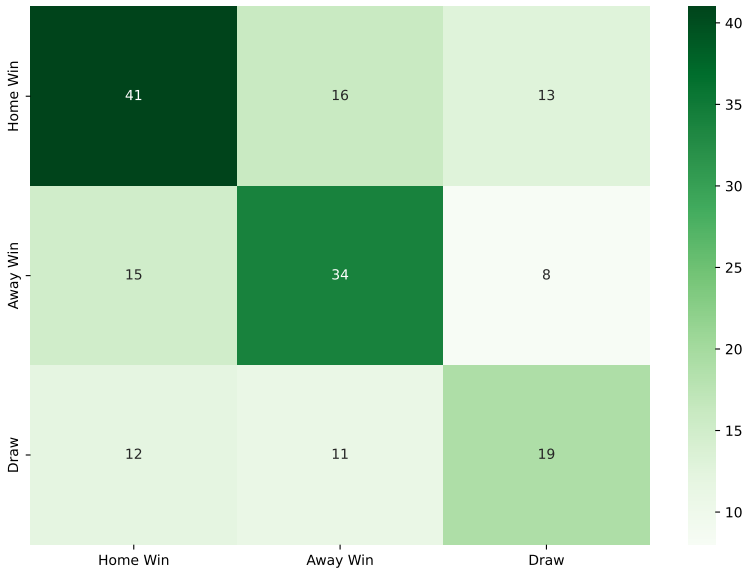

Figure 4.10: Confusion Matrix of y_pred vs. y_test Label for RFC on the P5 dataset

The matrix reveals that the model effectively predicts Home Wins (41 instances) and Away Wins (34 instances), suggesting it accounts for home advantage and away game dynamics well. However, it only correctly predicts 19 Draws, indicating the underrepresented weighting of draws. Misclassifications between

Home and Away Wins suggest the model does not fully capture the distinctions between these outcomes, while the low draw prediction accuracy points to a lack of data and model limitations in recognising the nuances of tied matches. Overall, the matrix highlights successes by showing the highest values in the true positives diagonal and underscores opportunities for refining the model's accuracy.

4.7 Discussion

In Chapter 4, the process of data transformation and exploration established a data-driven foundation for model input preparation and feature engineering. Pre-processing techniques were applied to ensure data integrity and compatibility with machine learning workflows. Most notably, the correlation matrix served as a critical step for identifying key relationships between variables and informing subsequent feature selection. The chapter's greatest strength lies in the exponential time-decay adjusted Pairwise and Weighted Pairwise Pi-rating scheme, which provides a new, robust, and accurate rating of teams. This originality approach allowed for a more nuanced understanding of team dynamics and performance, contributing significantly to the accuracy of match result predictions. The system can dynamically adjust for head-to-head team performance in each match-up pairing, offering a temporal perspective on the unique dynamics between specific team pairings. Following, using the SHAP feature importance analysis, the predictive power of features was quantified, thus streamlining the models and boosting their overall performance.

Ultimately, the chapter focused on pragmatic steps in ML model development: identifying relevant correlations, mitigating overfitting through cross-validation and feature selection, and assessing models using established metrics like accuracy, F1 score, precision, and recall. The Random Forest Classifier stood out for its performance on the P5 dataset, reaching 56.40% accuracy as per Table 4.4. This is on par with recent and relevant research around Full-time Result predictions and highlights the RFC model's potential for application in real-world scenarios.

The models show that beating the bookmakers in a live betting scenario is possible. In order to further investigate this claim, the FTR prediction task and models were defined above in such a way that enables easy construction of betting strategies. For example, models' output can directly be converted to probabilities to measure confidence and size bets accordingly. Additionally, the Research and Machine Learning Platform, developed as part of Chapter 3, was utilised to its full capabilities by dynamically retrieving FootyStats data based on the specified set of feature parameters. The next chapter will use the models and ML pipeline developed in this chapter to pin down whether bookmakers' odds can be exploited.

Chapter 5

Statistical Arbitrage Strategies in Value Betting

This chapter focuses on the implementation of value betting strategies in football which resulted from the research conducted in previous chapters. First, the efficiency of bookmakers and betting exchanges is evaluated to identify potential market entry points for the strategies. Building on the predictive models from Experiment 2 in Chapter 4, viable strategies are devised and backtested to evaluate their performance.

5.1 Football Betting Markets

In football betting markets, Expected Goals (xG) serve as a crucial metric for evaluating the likelihood of scoring opportunities during a match and serve as a fundamental metric in assessing team performance and predicting match outcomes. By quantifying the quality of scoring opportunities in a match, xG provides valuable insights into the offensive prowess and defensive resilience of teams. Over/Under bets, also known as totals bets, are another significant aspect of football betting. This type of wager allows bettors to predict whether the total number of goals scored in a match will be over or under a predetermined threshold set by the sportsbook, such as 2.5 goals. The concept of Over/Under adds depth to betting strategies, enabling bettors to focus on the overall performance of both teams rather than just the match outcome. Additionally, factors like price drift due to time constraints, such as the 90-minute duration of matches, influence the betting odds and market dynamics. Understanding these elements is key for identifying potential market entry points and devising effective betting strategies in football betting markets.

5.2 Market Efficiency

Understanding the efficiency of pre-game match odds is essential for both bettors and researchers aiming to capitalize on potential opportunities within the market. Assessing market efficiency involves evaluating the accuracy of bookmakers' odds and the extent to which they reflect the true probabilities of various outcomes. This evaluation not only informs bettors about the quality of starting prices but also provides insights into the effectiveness required by predictive models to outperform the market. Researchers often employ various metrics to determine market efficiency, such as the discrepancy between implied probabilities derived from odds and observed outcomes. Additionally, the examination of betting exchange data alongside traditional bookmakers' odds can offer further insights into market dynamics and potential inefficiencies. Data used for assessing market efficiency includes a range of bookmakers' starting prices and match results over multiple seasons, allowing for a comprehensive analysis of betting patterns and price movements.

5.2.1 Metrics and Statistics

Unlike financial markets, where outcomes are often unknown, the digital odds in sports betting offer implied probabilities that can be compared against observed outcomes. The section explores various metrics aimed at evaluating the strength of probabilistic predictions before conducting any strategy creation and backtesting.

Receiver Operating Characteristic and Area Under Curve

The Receiver Operating Characteristic (ROC) curve is a graphical representation of the true positive rate against the false positive rate. It shows the trade-off between sensitivity and specificity.

$$\text{ROC} = \frac{\text{True Positive Rate}}{\text{False Positive Rate}} \tag{5.1}$$

The Area Under Curve (AUC) gives the area under the ROC curve and measures how well the model can distinguish between two classes. An AUC of 1 denotes a perfect model, while an AUC of 0.5 implies no discrimination capability. In the context of this analysis, a high AUC would indicate that the bookmaker odds are good predictors of the actual match outcomes.

$$\text{AUC} = \int_0^1 \text{ROC} \tag{5.2}$$

5.2.2 Bookmaker Evaluation

The data presented in Table 5.1 provides a concise overview of the match odds from the entire training set that was utilised. Interestingly, there is a significant difference in betting odds between a win for either team and a draw which can explained by the historical statistical significance of wins compared to draws. Furthermore, the disparity in odds between home and away wins indicates a clear understanding of the home advantage by the bookmaker.

	Minimum	Median	Maximum	Mean
Home Win	1.01	2.19	26	2.69
The Draw	2.48	3.76	27	4.11
Away Win	1.07	3.50	51	4.83

Table 5.1: Summary of Bet365 Odds Statistics

In the AUC plots below, a high value suggests a strong capability to distinguish between different outcomes. In the context of bookmaker efficiency, a high AUC for the ROC curves indicates that the model's predictions align closely with the actual outcomes, making it highly predictive. This metric is crucial for evaluating the effectiveness of models in assessing bookmaker odds and their ability to provide accurate forecasts in the betting market.

In Figure 5.1 the AUC values for the Home and Away labels are both 0.73, and for the Draw label, it is 0.60. These high AUC values for Home and Away outcomes suggest that the Bet365 odds are effective in discerning wins from non-wins. However, the Draw outcome, with a lower AUC of 0.60, indicates a less effective differentiation. The implications for the betting market are significant. A bookmaker with efficient odds that align closely with actual outcomes will offer less favourable odds for punters to gain an edge. Figure 5.1 shows that betting the Draw outcome offers the most potential for punters to exploit inefficient odds. However, as seen in Chapter 4, the underrepresented nature of Draws makes it difficult for any model to reach high accuracy.

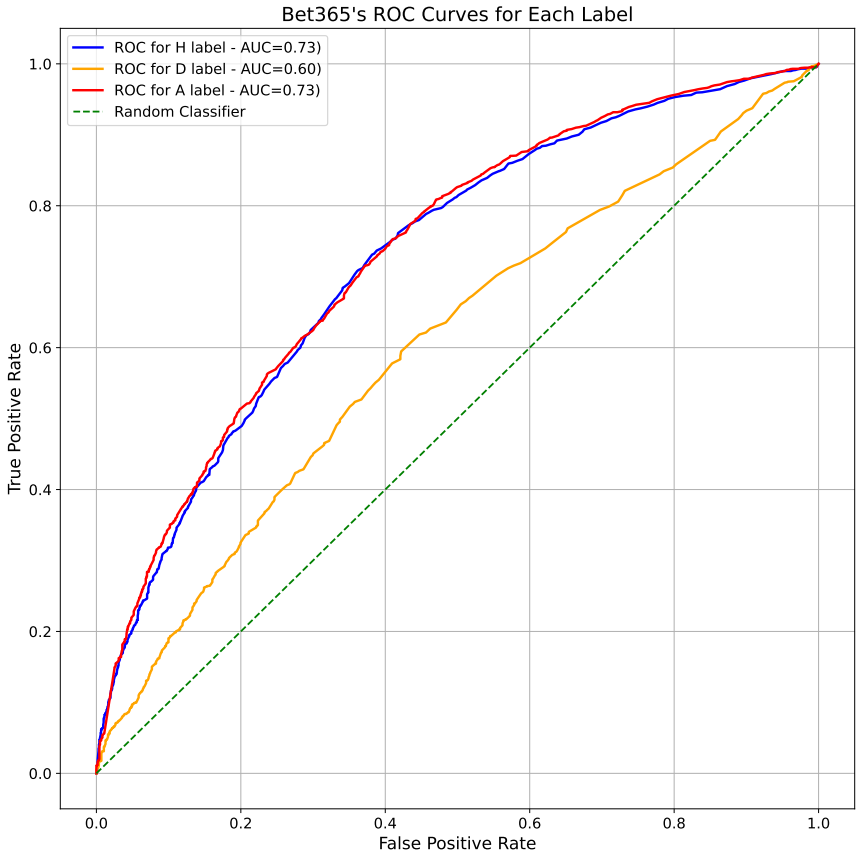

Figure 5.1: ROC curves for Target Label based on Bet365 Odds, $AUC_H = 0.73$, $AUC_A = 0.73$, $AUC_D = 0.60$

The calibration plots in Figures 5.2, 5.3 and 5.4 show how well the implied probabilities match the real outcomes for Home, Away and Draw labels, respectively. If the lines are near each other, the betting markets (i.e., odds) are fair and unbiased. When the actual outcome rate is regularly higher than the implied probability, the market is underestimating that outcome. If it's lower, the market is overestimating it. Figure 5.2 and 5.3 emphasise the findings from the ROC curves by showing a low level of bias for Home and Away predictions. On the other hand, the underrepresented Draw labels resulted in skewed probabilities, hence the corrupted plot in Figure 5.4.

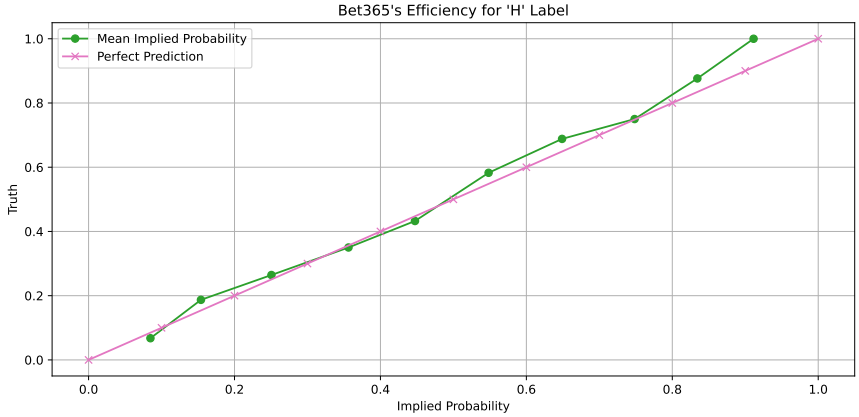

Figure 5.2: Bet365 bias estimation for Home outcomes

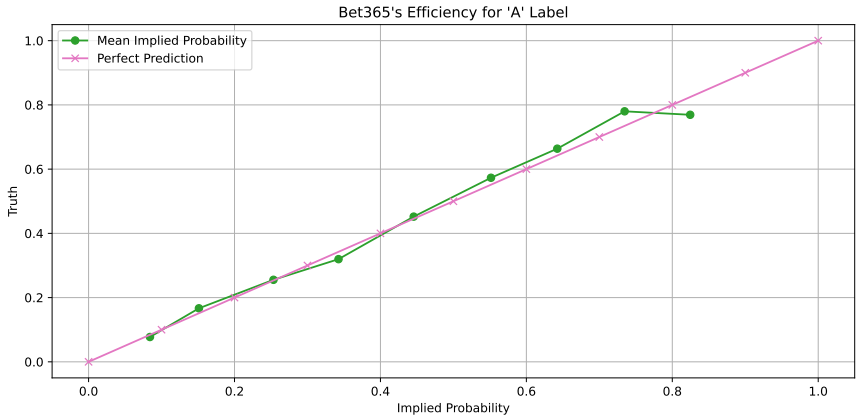

Figure 5.3: Bet365 bias estimation for Away outcomes

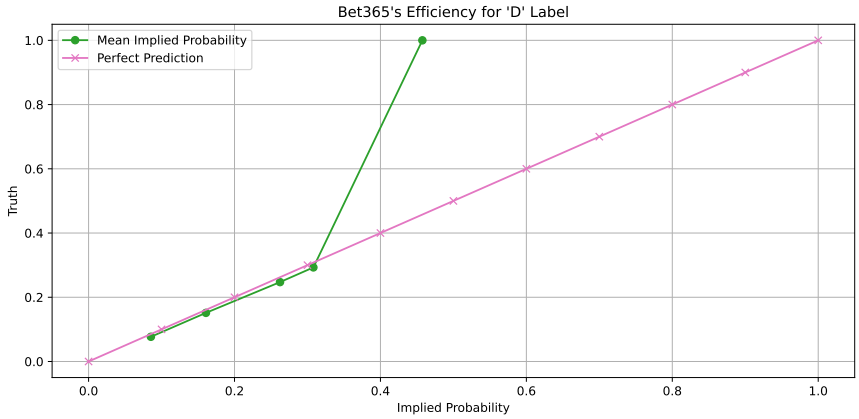

Figure 5.4: Bet365 bias estimation for Draw outcomes

5.3 Betting Strategies

Naive betting strategies include simple approaches such as blindly backing the favourite or always backing the Home team, without considering factors like team form, historical performance or head-to-head ratings. More sophisticated strategies involve the models and features developed as part of Chapter 4. Once a model can predict outcomes, it should be embedded into a betting strategy that executes the bets. Some common strategies include the Kelly Criterion, which calculates the optimal bet size based on the perceived edge and bankroll size.

5.3.1 Kelly Criterion

The Kelly Criterion [Kelly, 1956] is a mathematical formula that helps investors and gamblers calculate what percentage of their capital they should allocate to each trade or bet. The goal is to maximize the long-term growth rate of the capital. Based on the outcome with probability p, where p is the estimated value from the model), and odds b, which represents the payout defined by the bookmaker, the fraction f^* of the bankroll to bet is given by the formula:

$$f^* = \frac{bp - q}{b} \tag{5.3}$$

where f^* is the fraction of the current bankroll to wager, b is the odds received on the bet, p is the probability of winning, and q is the probability of losing.

5.3.2 Fractional-Kelly Criterion

The Fractional Kelly Criterion is a modification of the Kelly Criterion that allows for a fraction of the optimal bet size to be wagered. The formula is given by:

$$f = f^* \times k \tag{5.4}$$

where f is the fraction of the current bankroll to wager, f^* is the optimal fraction of the current bankroll to wager, and k is the fraction of the optimal bet size to wager. A widely used value for k is 0.5, which is also known as the Half-Kelly Criterion. By reducing the wager, punters reduce the risk of ruin and the volatility of the bankroll, while still maintaining a positive expected value. However, profits will be more moderate too.

5.4 Backtesting Simulation

A backtesting environment is developed next to utilise the match outcome predictions from the ML models in Experiment 2 and to exploit the obtained predictions using statistical arbitrage strategies. Specifically, the Support Vector Machine (`SVC`), Random Forest (`RFO`) and XGBoost (`XGB`) models were used to predict the final-time result with the narrowed-down features after the SHAP Feature Importance Analysis.

5.4.1 Training and Test Data

The main difference compared to data used in Chapter 4 is its split. The models below were trained on FootyStats EPL match data available up to the respective point in time. To achieve this, the dataset was segmented into consecutive study periods, each offset by a single round. Within each study period, a test set was designated to represent the specific round under consideration, while a training set encompassed all preceding rounds. This approach resulted in the expansion of the training set from 49 rounds (comprising one season and the initial matches of the subsequent season) to 199 rounds (encompassing five seasons minus one round). Ensuring the exclusion of any data pertaining to future events was critical throughout the prediction process.

The training data (`X_train`) starts from the 2018/19 season and ends, initially, at the halfway of the 2018/19 season. Consequently, the initial test set is the first round after the halfway point of the 2018/19 season where one round consists of ten fixtures. Then, as the rounds progress, both `X_train` and `X_test` are shifted round by round, therefore, `X_test` only ever contains a single round's fixtures. Table 5.2 shows this initial setup and the data in each dataset.

Dataset	Start Date	End Date	# of Rounds	# of Matches
`Train`	10.08.2018	05.01.2018	19	190
`Test`	06.01.2018	13.01.2018	1	10

Table 5.2: Initial Training and Test Dataset Setup to Backtest Strategies

After the initial training, retraining and testing were repeated for the following 95 rounds, equating to two and a half seasons (1 season = 39 = 390 matches). As a result, the strategies were tested on a total of 950 matches ($2.5 * 390$). Note that such a backtesting system implies that, in order to accurately and successfully run the backtest, each model is retrained after data becomes available from a fully concluded round.

5.4.2 Baseline Strategies

Benchmark strategies were established to provide a baseline for the more comprehensive ML strategies. These models do not take any complementary data beyond the Bet365 odds. The two benchmark strategies are defined as:

- `FAV` backs the favourite runner (i.e., runner with the lowest odds) with 1 unit per match. If the odds are the same, no bet is placed. Notably, the lowest betting odd consistently corresponds to either Home win or Away win, therefore `FAV` would never bet on Draw.
- `HOME`: strictly backs the Home team with 1 unit per match. No other information is available to the strategy, therefore, its only purpose is to evaluate the strength of the home advantage and how well bookmakers account for it.
- `SVM`: backs the team with the highest probability of winning with 1 unit per match. Probabilities are based on the `SVM` model's target label predictions.

5.4.3 Machine Learning Strategies with Kelly Criterion

The best-performing models from Chapter 4 were taken to construct value betting strategies that return small but consistent profits. In each study period, the objective was to forecast the match results `y_test` for the designated round. As mentioned `X_test` was derived from the optimal feature set from Chapter 4. This approach ensures the avoidance of any lookahead bias (thanks to the implementation of `XTableConstructor`) while also factoring in the relevant team performance statistics and Pi Ratings.

Instead of a fixed bet size (1 unit per match), the Kelly Criterion is applied to adjust sizing and maximise the final wealth of the agents. The bet size is now given by Equation 5.4. The agents will operate based on the Half-Kelly Criterion, meaning $k = 0.5$ in Equation 5.4. In the original Kelly Criterion $k = 1$, however, following research and the practical anecdotes from QST, it was decided that Half-Kelly is more appropriate and applicable.

Crucially, the ML strategies only ever place bets if the confidence or `p_threshold`, given by the softmax function of the regression models, for the given target label, is over 0.4. This can be expressed as $y_{\text{pred}} >= 0.4$. Also, note that $p = y_{\text{pred}}$ in Equation 5.4 for the Half-Kelly calculations. Taking the optimised models `SVM`, `RFO` and `XGB` the aim was to capture the mispriced odds from bookmakers with the following statistical arbitrage strategies:

- `HK-RFO`: backs the team with the highest probability of winning with f unit per match where f is determined based on Half-Kelly in Equation 5.4, but only if $y_{\text{pred}} >= 0.4$. Probabilities are based on the `RFO` model's target label predictions.
- `HK-SVM`: backs the team with the highest probability of winning with f unit per match where f is determined based on Half-Kelly in Equation 5.4, but only if $y_{\text{pred}} >= 0.35$. Probabilities are based on the `SVM` model's target label predictions.
- `HK-XGB`: backs the team with the highest probability of winning with f unit per match where f is determined based on Half-Kelly in Equation 5.4, but only if $y_{\text{pred}} >= 0.7$. Probabilities are based on the `XGB` model's target label predictions.

The differences in `p_threshold` are the result of the underlying nature of the classification models. XGB predicted probability associated with `y_pred` is generally twice as high as the probabilities from the RFO or SVM models.

5.5 Results

5.5.1 Machine Learning vs. Baseline Model Comparison

Finally, a comparison is provided between the baseline strategies and Half-Kelly ML (i.e., `HK`) strategies. In addition to the statistics from Subsection 5.2.1 and trivial metrics such as profit, final bankroll and accurate label prediction rate, more complex, risk-adjusted measures are evaluated:

Sharpe Ratio

The Sharpe Ratio is a measure of risk-adjusted return and is calculated as:

$$\text{Sharpe Ratio} = \frac{R_p - R_f}{\sigma_p} \tag{5.5}$$

where R_p is the expected bankroll return, R_f is the risk-free rate, and σ_p is the bankroll standard deviation.

Sortino Ratio

The Sortino Ratio is a measure of risk-free return, and is calculated as:

$$\text{Sortino Ratio} = \frac{R_p - R_f}{\sigma_d} \tag{5.6}$$

where R_p is the expected bankroll return, R_f is the risk-free rate, and σ_d is the downside deviation.

5.5.2 Statistical Analysis

The overall findings from backtesting are summarised in Table 5.3. The rows are categorised by different metrics, while the columns represent various betting strategies. Accuracy, defined as the percentage of correctly predicted outcomes predicted label, `y_pred` compared to the true label, `y_test` is provided. Importantly, the profit and loss (PnL), final bankroll and risk-adjusted metrics allow for a more comprehensive analysis. Lastly, the number of bets made by each strategy and their breakdown (i.e., Home, Away, Draw) are presented to identify potential biases, especially associated with the confidence levels (`p_threshold`) defined under Subsection 5.4.3.

	HK-SVM	HK-RFC	HK-XGB	SVM	HOME	FAV
PnL %	52.39%	-63.03%	-42.45%	-65.08%	-3.53%	-61.93%
Bankroll	152.39	36.97	57.55	34.92	96.47	38.07
Accuracy	56.80%	53.21%	55.39%	53.07%	45.48%	55.37%
Sharpe ratio	0.82	-0.08	-0.03	-0.06	0.02	-0.04
Sortino ratio	1.14	-0.05	-0.02	-0.04	0.07	-0.03
Average payoff	0.07	-0.07	-0.05	-0.07	0	-0.07
# of bets	702	950	808	950	950	950
Home bets	393	692	515	635	950	628
Away bets	278	215	279	268	0	322
Draw bets	31	43	14	47	0	0

Table 5.3: Bet and risk-return characteristics for each strategy averaged across the 950-match test set

As illustrated in Table 5.3, the `HK-SVM` model demonstrates both the highest prediction accuracy (56.8%) and PnL (52%) with a moderate Sharpe ratio of 0.82. Following is the `HOME` baseline strategy that incurred a

minor loss of -3.53% while betting exclusively on the Home team for the two and a half seasons in the dataset. Conversely, the `HK-RFC` and `HK-XGB` strategies, as well as the simpler approach `SVM` recorded large losses of approximately half the initial bankroll. Also, note the 55.37% accuracy of the `FAV` strategy which represents the bookmaker's predictive accuracy. This shows that a high predictive accuracy does not immediately reflect in increased PnL levels. Lastly, results from Chapter 4 are reiterated in Table 5.3, namely, the number of Home bets dominates the number of Away, and even more so, the number of Draw bets. It is notable that the best-performing model, `HK-SVM` had the highest ratio of both Away, 39% $\approx$ 278/702, and Draw, 4.5% $\approx$ 31/702, bets compared to the total number of bets.

5.5.3 Financial Analysis

Figures 5.5 and 5.6 delve into the bet-by-bet performance of the Half-Kelly machine learning and baseline strategies, respectively, using the size of the bankroll as the key measure (which is directly correlated with the PnL). Similar to the Statistical Analysis, this section presents an overview of the strategies' effectiveness in generating returns over time. In certain cases, the strategies' inability to deliver a steady profitable performance is also highlighted by large draw-downs.

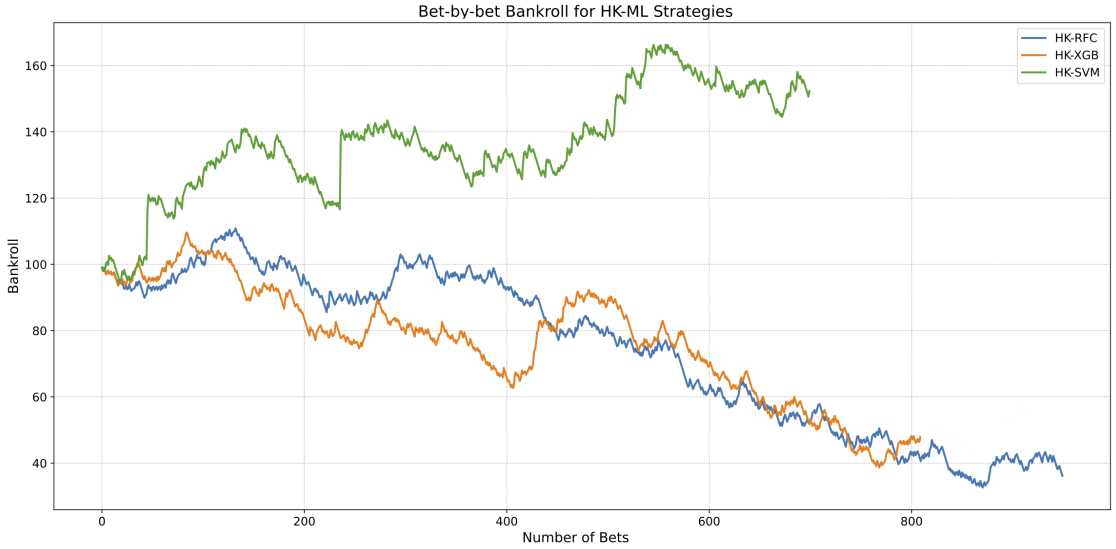

Figure 5.5: Bet-by-bet progression of the bankroll of the machine learning and Halk-Kelly strategies HK-RFC, HK-SVM and HK-XGB. The shorter line shows the strategy made fewer bets.

It is clear from Figure 5.5 that `HK-SVM` has placed the fewest bets (by the shorter line) but constantly and steadily outperformed `HK-XGB` and `HK-RFC` in terms of bankroll size. `HK-SVM` was better able to exploit mispriced odds, apparent by the occasional sharp jumps in bankroll size, while also limiting its losses. Even though `p_treshold=0.35` for `HK-SVM`, the strategy could refrain from placing risky bets more effectively than, for instance, `HK-XGB` whose threshold was set to `p_treshold=0.7`. This implies the `SVM` model generally predicts the target label with conservative confidence (e.g., 0.34) and only exceeds the 0.35 threshold in limited scenarios (hence the shorter line chart).

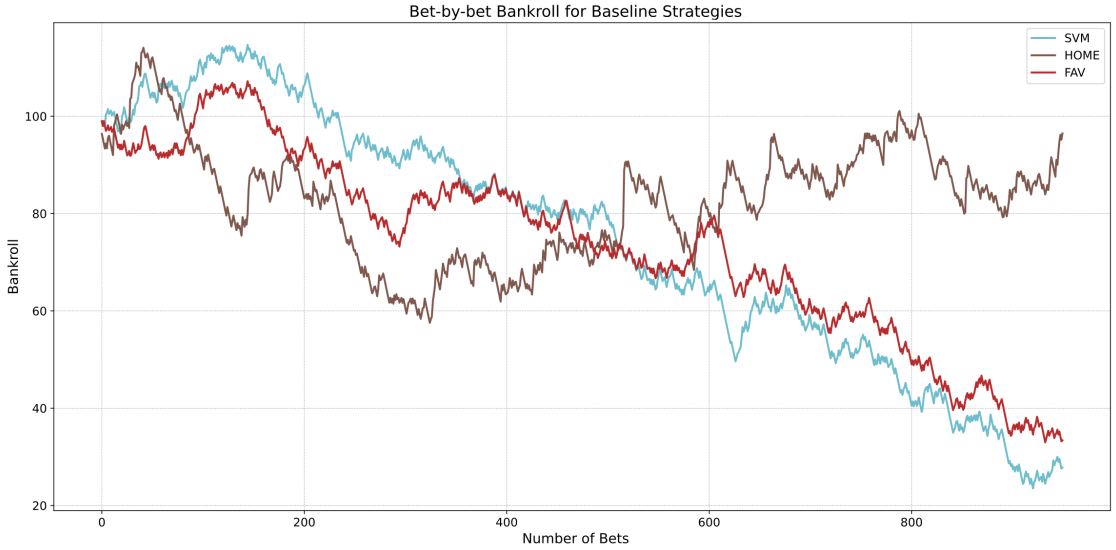

Figure 5.6: Bet-by-bet progression of the bankroll of the baseline strategies SVM, HOME, and FAV.

Figure 5.6 shows the progression of the bankroll size for the baseline strategies `SVM`, `HOME` and `FAV`. The `HOME` strategy, which simply bet on the home team, demonstrated a surprising efficacy across the 950 matches by keeping close to the initial bankroll size. While the testing set, consisting of two-and-a-half seasons, was determined to be sufficiently large to filter out randomness and temporal trends, the performance of `HOME` indicates the test set size should be increased. It is understood from historical data and similar research studies that solely backing the Home team should result in more significant losses over time. Lastly, note that both the `SVM` strategy, which does determine bet sizing based on the Kelly Criterion, and the FAV strategy, which would represent the bookmaker's performance by backing the favourite runner, both showed substantial losses with little to no upward trends.

5.6 Discussion

By evaluating how accurately the bookmaker's odds reflect the true probabilities of game outcomes, the chapter first aimed at identifying inefficiencies in the pre-game value betting markets. When employing metrics such as the discrepancy between implied probabilities and observed outcomes, the Receiver Operating Characteristic (ROC) curve and the corresponding Area Under Curve (AUC) measure became the most informative. They provided a graphical representation and a numerical measure, respectively, of the bookmaker's ability to distinguish between outcomes, with a perfect model scoring an AUC of 1. Odds associated with Home and Away wins are generally efficient, with high AUC values of 0.73, indicating a strong correlation with actual match results. However, the odds for Draws, represented by a lower AUC of 0.60, suggest a lower predictive power which resents inefficiencies.

Following, to test and deploy both machine learning and *naive* baseline strategies to the value betting market, a comprehensive backtesting simulation was set up. The focus was primarily on the accuracy and profitability of the `SVM`, `RFC` and `XGB` models derived in Chapter 4. To maximise the agents' bankrolls while also mitigating risk, the Half-Kelly Criterion and confidence thresholds, `p_threshold` were implemented and tailored to each model. The (re)training and testing on the gradually expanding datasets over two and a half successive seasons ensured a robust and bias-free assessment of these strategies. Lastly, to provide a benchmark and draw conclusions on relative profitability, the ML strategies were pinned against baseline strategies, which relied solely on the bookmaker's odds to place bets throughout the testing period.

Finally, the chapter concluded that the `HK-SVM` strategy outperforms others with a profit and loss (PnL) percentage of 52.39% and an accuracy rate of 56.80%, complemented by moderate risk-adjusted return

measures such as a Sharpe ratio of 0.82 and a Sortino ratio of 1.14. On the other hand, both `HK-RFC` and `HK-XGB` strategies, as well as the baseline `FAV`, recorded substantial losses and poor risk-adjusted performance. The HOME strategy, betting solely on the home team, maintained a more stable performance with a minor PnL loss of -3.53%. The discrepancy between accuracy and PnL across strategies, and `HK-SVM`'s performance which placed more bets on Away and Draw outcomes show that returns originate from the less significant target labels. In other words, predicting the same label as the bookmaker, even if the prediction is correct, provides no edge as odds have already been adjusted with overround.

Chapter 6

Conclusion

The last chapter summarises the findings, and industry and scientific contributions presented throughout the thesis, in particular throughout the live trading scenarios. Additionally, the chapter discusses future research opportunities and concludes by taking a step back to evaluate the thesis's impact on the field of algorithmic sports trading.

6.1 Summary

The thesis was driven by several key motivations, primarily stemming from the untapped nature of the betting market compared to traditional financial markets which presents unique opportunities for both academic research and commercial ventures. Namely, motivation arose from the similarities in liquidity provision, limit order book dynamics, and volatility characteristics. Despite the extensive literature on machine learning (ML) applications in financial markets, its utilisation in sports trading remains relatively low. This gap motivates the thesis to explore ML-based statistical arbitrage models tailored for live sports trading scenarios, aiming to bridge academia with practical industry applications.

To address these motivations, the research objectives are delineated into distinct areas. Firstly, the development of a robust research and ML platform is paramount, facilitating efficient data handling, model construction and research. The platform integrates football in-play state-of-game data with millisecond-level exchange data and historical match data to train comprehensive models capturing market microstructure dynamics. Subsequently, the research explores the football betting landscapes via a comprehensive literature review, assessing bookmakers' efficiency, and developing pre-game value betting models. This involves training multiclass classifiers to predict the full-time results, backtesting models against bookmaker odds, and establishing baseline models for comparison. Lastly, the research implements ML-based arbitrage models with the Half-Kelly Criterion for pre-game football betting, emphasising adaptability to dynamic environments, effective risk management, and comparison against baseline models to evaluate performance metrics.

The thesis comprises three main experiments aligned with these objectives. Experiment 1 focuses on integrating in-play and exchange data using a cloud-native NoSQL time-series database architecture. Experiment 2 delves into accurately modelling football betting markets and developing pre-game value betting models, while Experiment 3 builds on previous insights to develop and evaluate ML-based statistical arbitrage strategies for pre-game football betting.

6.2 Conclusion

In conclusion, the research objectives defined in Chapter 1 were reached by successfully carrying out the experiments in Chapter 3 - Research and Machine Learning Platform, Chapter 4 - Machine Learning for Pre-Game Football Predictions and Chapter 5 - Statistical Arbitrage Strategies in Value Betting. These experiments yielded important results and contributions that are applicable to both the industry and academia.

Experiment 1: Research and Machine Learning Platform

Firstly, the establishment of a comprehensive research and machine learning platform provides a robust framework for conducting sophisticated analyses and model development in the betting space. The platform, equipped with abstraction layers for handling diverse data sources, enables researchers to efficiently query and process large datasets, accelerating the pace of ideation and experimentation in sports betting research. While the focus of this research was on football value betting, the platform also hosts all of the pre-game, in-play and historical exchange data for different sports. As a result, the research platform became the standard tool for data experimentation and machine learning modelling for researchers at QST.

Experiment 2: Machine Learning for Pre-Game Football Predictions

One of the standout contributions of this research is the original extension of the Pi-rating system, which enhances the predictive accuracy of team performance metrics. By incorporating time-decayed pairwise ratings and integrating additional statistical features, the extended Pi Pairwise and Weighted Pairwise systems offer valuable insights for bookmakers and punters alike, enabling more informed decision-making and strategy formulation. As shown by the SHAP feature importance analysis, the Home and Away team features derived from the Pi Pairwise ratings outperformed both the original Pi system in predictive accuracy and the evaluated bookmaker's odds.

Experiment 3: Statistical Arbitrage Strategies in Value Betting

Last but not least, the exploration and implementation of statistical arbitrage value betting strategies demonstrate the feasibility of generating profits in football betting markets with a moderate risk profile. Leveraging machine learning techniques, the `HK-SVM` strategy effectively identifies mispriced odds and exploits market inefficiencies, highlighting the potential for algorithmic trading approaches in sports betting. Overall, the ML models derived in Experiment 2 were successfully backtested and integrated into arbitrage strategies, in particular through the use of the Half-Kelly Criterion for bet sizing. Due to its moderate risk profile and profitability, the `HK-SVM` strategy is part of the end-to-end algorithmic sports betting pipeline at QST.

6.3 Future Work

While the research has made significant strides in developing robust value betting strategies for football markets, several avenues for future exploration and improvement remain open. Given the predominant focus on the Pi-rating system, further investigation into external statistics could yield valuable insights and improvements to the pre-game classification models. For example, incorporating point-in-time player statistics and sentiment analysis might enhance the models' predictive capabilities by capturing additional contextual information, such as player transfers, injuries, or match stakes, which are often driving factors behind match outcomes.

Moreover, the research platform developed for integrating pre-game, in-play, and exchange data provides a rich foundation for future experimentation and innovation. One promising direction is to explore in-play market-making strategies using reinforcement learning techniques. By leveraging the platform's easy accessibility to real-time data streams, researchers can develop and test sophisticated algorithms for dynamically adjusting back volumes to optimize profitability in rapidly changing in-play markets. Additionally, there is potential for in-play drift betting strategies in football markets, by integrating pre-game and in-play data to identify and exploit market inefficiencies as odds predictably shorten/lengthen throughout a match.

Lastly, in terms of the arbitrage strategies in this research, the parameters of the strategies could be fine-tuned. In particular, the probability thresholds, `p_threshold` to place bets and the fraction of $k = 0.5$ in the Kelly Criterion calculation to determine bet sizing were chosen based on empirical evidence and manual search. Future work could investigate these parameters with more robust scientific approaches, for example, by running a large-scale Grid search or Bayesian hyperparameter optimisation with varying parameter pairs and configurations.

Bibliography

[Albert and Koning, 2007] Albert, J. and Koning, R., editors (2007). *Statistical Thinking in Sports*. Chapman and Hall/CRC, 1st edition.

[Almgren and Chriss, 2001] Almgren, R. and Chriss, N. (2001). Optimal execution of portfolio transactions. *Journal of Risk*, 3:5–40.

[Apollinaire and Amanda, 2022] Apollinaire, N. M. and Amanda, P. N. (2022). Stochastic optimal control theory applied in finance. *Mathematics and Computer Science*, 7(4):59–67.

[Asur and Huberman, 2010] Asur, S. and Huberman, B. A. (2010). Predicting the future with social media. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 492–499.

[Avellaneda and Lee, 2009] Avellaneda, M. and Lee, J.-H. (2009). Statistical arbitrage in the u.s. equities market. *Quantitative Finance*, 10(7):761–782.

[Bachouch et al., 2018] Bachouch, A., Huré, C., Langrené, N., and Pham, H. (2018). Deep neural networks algorithms for stochastic control problems on finite horizon: numerical applications.

[Black and Scholes, 1973] Black, F. and Scholes, M. S. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654.

[Canizares et al., 2017] Canizares, P. C., Merayo, M. G., Núñez, M., and Suárez-Paniagua, V. (2017). Case study on the prediction of football matches using multi-agent systems. pages 572–576.

[Cartea et al.,] Cartea, A., Donnelly, R., and Jaimungal, S. Algorithmic trading with model uncertainty.

[Chan et al., 1999] Chan, L. K., Lakonishok, J., and Sougiannis, T. (1999). The stock market valuation of research and development expenditures. NBER Working Paper No. w7223.

[Chong et al., 2017] Chong, E., Han, C., and Park, F. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. *Expert Systems with Applications*, 83:187–205.

[Clarke and Norman, 1995] Clarke, S. and Norman, J. (1995). Home ground advantage of individual clubs in english soccer. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 44(4):509–521.

[Constantinou et al., 2012] Constantinou, A., Fenton, N., and Neil, M. (2012). pi-football: A bayesian network model for forecasting association football match outcomes. knowledge-based systems, 36, 322-339. *Knowledge-Based Systems*, 36:332–339.

[Croxson and Reade, 2010] Croxson, K. and Reade, J. J. (2010). Exchange vs. dealers: A high-frequency analysis of in-play betting prices exchange vs. dealers: a high-frequency analysis of in-play betting prices *.

[Divos, 2020] Divos, P. (2020). Modelling of the in-play football betting market.

[Dixon et al., 2017] Dixon, M., Klabjan, D., and Bang, J. H. (2017). Classification-based financial markets prediction using deep neural networks. *Algorithmic Finance*, 6:67–77.

[Dixon and Coles, 1997] Dixon, M. J. and Coles, S. G. (1997). Modelling association football scores and inefficiencies in the football betting market. *Journal of the Royal Statistical Society. Series C: Applied Statistics*, 46:265–280.

[Egidi et al., 2018] Egidi, L., Pauli, F., and Torelli, N. (2018). Combining historical data and bookmakers' odds in modelling football scores. *Statistical Modelling*, 18:436–459.

[Gatev et al., 2006] Gatev, E., Goetzmann, W. N., and Rouwenhorst, K. G. (2006). Pairs trading: Performance of a relative value arbitrage rule. Yale ICF Working Paper No. 08-03.

[Goel et al., 2010] Goel, S., Hofman, J. M., Lahaie, S., Pennock, D. M., and Watts, D. J. (2010). Predicting consumer behavior with web search. *Proceedings of the National Academy of Sciences*, 107(41):17486–17490.

[Henderson et al., 2012] Henderson, J. V., Storeygard, A., and Weil, D. N. (2012). Measuring economic growth from outer space. *American Economic Review*, 102(2):994–1028.

[Hogan et al., 2004] Hogan, S., Jarrow, R., Teo, M., and Tse, Y. K. (2004). Testing market efficiency using statistical arbitrage with applications to momentum and value strategies. *Journal of Financial Economics*, 73(3):525–565.

[Karlis and Ntzoufras, 2003] Karlis, D. and Ntzoufras, I. (2003). Analysis of sports data by using bivariate poisson models.

[Kelly, 1956] Kelly, J. L. (1956). A new interpretation of information rate. *Bell System Technical Journal*, 35(4):917–926.

[Maclean et al., 1992] Maclean, L., Ziemba, W., and Blazenko, G. (1992). Growth versus security in dynamic investment analysis. *Management Science*, 38(11):1562–1585.

[Mukherjee et al., 2021] Mukherjee, A., Panayotov, G., and Shon, J. (2021). Eye in the sky: Private satellites and government macro data. *Journal of Financial Economics*, 141(1):234–254.

[Nuti et al., 2011] Nuti, G., Mirghaemi, M., Treleaven, P. C., and Yingsaeree, C. (2011). Algorithmic trading. *Computer*, 44:61–69.

[Shahtahmassebi and Moyeed, 2016] Shahtahmassebi, G. and Moyeed, R. (2016). An application of the generalized poisson difference distribution to the bayesian modelling of football scores. *Statistica Neerlandica*, 70:260–273.

[Stoll, 1995] Stoll, H. (1995). Market microstructure theory. *The Review of Financial Studies*, 8(4):1235–1238.

[Treleaven et al., 2013] Treleaven, P., Galas, M., and Lalchand, V. (2013). Algorithmic trading review. *Communications of the ACM*, 56:76–85.

[Vidyamurthy, 2004] Vidyamurthy, G. (2004). Pairs trading : Quantitative methods and analysis / g. vidyamurthy.

[Øvregård, 2008] Øvregård, N. (2008). Trading "in-play" betting exchange markets with artificial neural networks.