Modelling of the In-Play Football Betting Market

Peter Divos

Supervisors: Prof. Tomaso Aste
Dr. Sebastian del Bano Rollin

A dissertation submitted in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy
of University College London
Department of Computer Science.

August 2, 2020

Declaration

I hereby declare that the contents of this dissertation are my original work. Any information derived from other sources has been explicitly referenced and acknowledged in the text. I also declare that this dissertation has not been submitted in whole or in part for consideration for any other degree or qualification at this, or any other University.

Peter Divos

Abstract

This thesis is about modelling the in-play football betting market. Our aim is to apply and extend financial mathematical concepts and models to value and risk-manage in-play football bets. We also apply machine learning methods to predict the outcome of the game using in-play indicators.

In-play football betting provides a unique opportunity to observe the interplay between a clearly defined fundamental process, that is the game itself and a market on top of this process, the in-play betting market. This is in contrast with classical finance where the relationship between the fundamentals and the market is often indirect or unclear due to lack of direct connection, lack of information and infrequency or delay of information. What makes football betting unique is that the physical fundamentals are well observable because of the existence of rich high frequency data sets, the games have a limited time horizon of usually 90 minutes which avoids the buildup of long term expectations and finally the payoff of the traded products is directly linked to the fundamentals.

In the first part of the thesis we show that a number of results in financial mathematics that have been developed for financial derivatives can be applied to value and risk manage in-play football bets. In the second part we develop models to predict the outcomes of football games using in-play data.

First, we show that the concepts of risk-neutral measure, arbitrage freeness and completeness can also be applied to in-play football betting. This is achieved by assuming a model where the scores of the two teams follow standard Poisson processes with constant intensities. We note that this model is analogous to the Black-Scholes model in many ways.

Second, we observe that an implied intensity smile does exist in football betting and we propose the so-called Local Intensity model. This is motivated by the local volatility model from finance which was the answer to the problem of the implied volatility smile. We show that the counterparts of the Dupire formulae [31] can also be derived in this setting.

Third, we propose a Microscopic Model to describe not only the number of goals scored by the two teams, but also two additional variables: the position of the ball and the team holding the ball. We start from a general model where the model parameters are multi-variate functions of all the state variables. Then we characterise the general parameter surfaces using in-play game data and arrive to a simplified model of 13 scalar parameters only. We then show that a semi-analytic method can be used to solve the model. We use the model to predict scoring intensities for various time intervals in the future and find that the initial ball position and team holding the ball is relevant for time intervals of under 30 seconds.

Fourth, we consider in-play indicators observed at the end of the first half to predict the number of goals scored during the second half, we refer to this model as the First Half Indicators Model. We use various feature selection methods to identify relevant indicators and use different machine learning models to predict goal intensities for the second half. In our setting a linear model with Elastic Net regularisation had the best performance.

Fifth, we compare the predictive powers of the Microscopic Model and the First Half Indicators Model and we find that the Microscopic Model outperforms the First Half Indicators Model for delays of under 30 seconds because this is the time frame where the initial team having the ball and the initial position of the ball is relevant.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my primary supervisor Professor Tomaso Aste and to my secondary supervisor Dr. Sebastian del Bano Rollin for their inspiration, patience, support and motivation. I would also like to thank Zsolt Bihary, Luigi Colombo and Benoit Jottreau for the fruitful conversations about football and modelling.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Football betting provides a unique opportunity to observe the interaction between a fundamental process, that is a football game and a market on top of this process, the betting market.

1.0.1 Overview of the In-Play betting market

In-play bets are traded during the live game, their price fluctuate rapidly as the teams score goals. This is similar to financial markets where the price of an option changes according to the fluctuations of the underlying instrument.

In traditional football betting, also known as pre-game or fixed odds betting, bets are placed before the beginning of the game. In-play football betting enables bettors to place bets on the outcome of a game during the live game. The main difference is that during in-play betting, as the game progresses and as the teams score goals, the chances of certain outcomes change dynamically and so do the odds of the bets. With the proliferation of mobile devices, in-play betting became increasingly popular in recent years. For instance, [33] recently reported that for one particular bookmaker (Unibet) in-play betting revenues exceeded pre-game betting revenues by 2013Q2

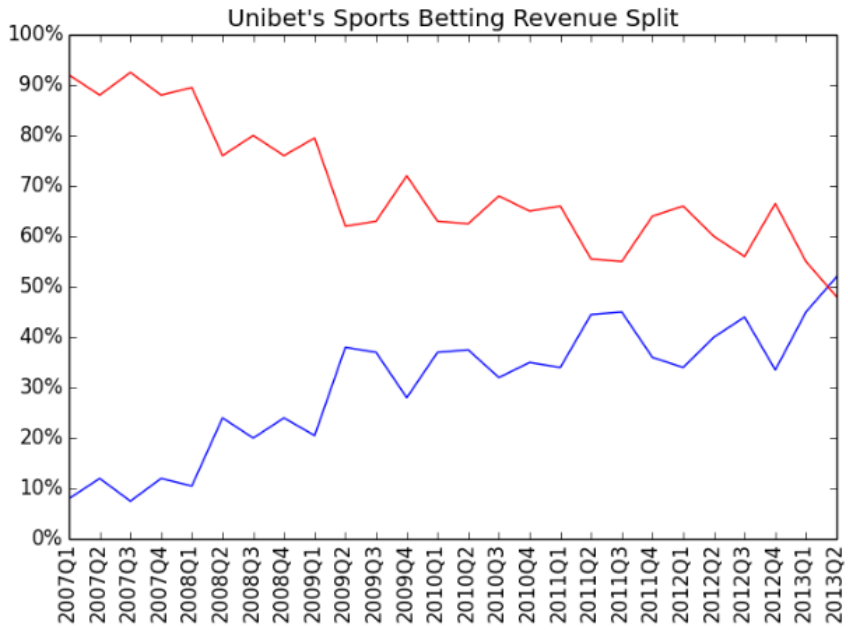


Figure 1.1: Revenue distribution of one particular bookmaker's (Unibet) football betting revenues between In-Play and Pre-Game football betting.

as shown on Figure 1.1.

There are two main styles of in-play betting: odds betting and spread betting. In odds betting, the events offered are similar to digital options in the sense that the bettor wins a certain amount if the event happens and looses a certain amount otherwise. Typical odds bets are whether one team wins the game, whether the total number of goals is above a certain number or whether the next goal is scored by the home team. In spread betting, the bets offered are such that the bettor can win or loose an arbitrary amount. A typical example is a bet called "total goal minutes" which pays the bettor the sum of the minute time of each goal. In this thesis we focus on odds betting, but most of the results can also be applied to spread betting. A study of spread betting containing analytical pricing formulae for various spread bets was published by [17].

In-play betting offers various types of events such as total goals, home and away goals, individual player goals, cards, corners, injuries and other events. In this thesis we focus on bets related to goal events only.

Throughout this thesis we refer to the value X_t of a bet as the price at which the bet can be bought or sold at time t assuming that the bet pays a fixed amount of 1 unit in case it wins and zero otherwise. This is a convenient notation from a mathematical point of view, however it is worth noting that different conventions are used for price indication in actual betting. The two most popular conventions are called fractional odds and decimal odds. Both of these conventions rely on the assumption that the bettor wagers a fixed stake when the bet is placed and enjoys a payoff in case the bet wins or no payoff in case it looses. Fractional odds is the net payoff of the bet in case the bet wins (that is, payoff minus stake), divided by the stake. Decimal odds is the total payoff of the bet in case the bet wins, divided by the stake. Therefore, the value of a bet X_t is always equal to the reciprocal of the decimal odds which is equal to the reciprocal of fractional odds plus one, formally:

$$X_t = \frac{1}{D_t} = \frac{1}{F_t + 1}, \tag{1.1}$$

where D_t denotes decimal and F_t denotes fractional odds. Most of the market data we used was originally represented as decimal odds, but they were converted to bet values using the above formula for all the charts and for the underlying calculations in this thesis.

It is also worth noting that bets can be bought or sold freely during the game. This includes going short which is referred to as lay betting. Mathematically this means that the amount held of a bet can be any real number.

In-play odds bets can either be purchased from a retail bookmaker at a

price offered by the bookmaker (for example Bet365, Paddy Power, William Hill, Ladbrokes, Unibet and others), but can also be traded on centralized marketplaces (operated for example by Betfair, Betdaq, Smarkets and others) where participants trade with each other through a limit order book, much like stocks are traded on exchanges.

1.1 Research objectives

The research objectives of this thesis are the following:

1. Football betting offers a unique insight into the interplay between a fundamental process, that is the football game itself and a market driven by the fundamental process, that is the betting market because both gameplay data and market data are widely available. This is in contrast with most situations in financial markets where the underlying fundamental processeses are usually hard to uncover and in some cases completely hidden. Therefore understanding how the football betting market is driven by the underlying fundamental process might yield useful insight into the interplay between financial fundamentals and asset prices.

2. To apply models, tools and results from financial mathematics and derivatives pricing to value, hedge and risk manage in-play football bets. Either apply existing theorems on a football betting setup or create new models specifically for football betting that are inspired by existing derivative pricing models.

3. Use the large amount of in-play data available on football games and build machine learning models to find relevant indicators and to predict

the future evolution and the outcome of the game.

1.2 Data description

I used two data sets. Data set 1 contains market prices of in-play bets while data set 2 contains high resolution gameplay data.

1.2.1 Data Set 1: Market Prices

Data set 1 contains market prices of in-play bets that are traded during the game. The time resolution of the data set is 1 second and it consists of a series of different types of bet prices covering 30 games of the 2012 UEFA European Championship. The types of the bets covered by the data set are:

- Match odds - which team will win the game, one of HOME, AWAY or DRAW
- Over/Under - whether the total number of goals scored by both teams is over or under a given value
- Correct score - correct number of goals scored by each team
- Next Goal - which team will score the next goal

I use this data in Chapters 3 and 4.

1.2.2 Data Set 2: Gameplay Data

Data set 2 contains high resolution in-play information from 2940 games in 7 leagues of the of the 2013/14 Season shown in Table 1.1. This data set contains sub-second resolution events for each game. Each event contains the following fields:

- The team and player associated with the event
- Sub-second resolution tine of the event during the game
- Location of the event on the field
- Type of the event

The types of events are one of:

- Pass, along with whether the pass was successful or not
- Goal scored
- Goal attempt saved
- Missed goal attempt
- Ball goes out of the field
- Corner awarded
- Card awarded, along with the type of card
- Clearance, which is when a player kicks the ball away from the goal they are defending

This data set is used in Chapters 5, 6 and 7.

1.3 Thesis structure

The structure of this thesis is the following.

Chapter 2 contains an overview of the relevant literature, theoretical background and methods used in the thesis.

Season	League	Game count
2013/14	English Football League Championship	562
2013/14	Spanish Segunda Division	412
2013/14	Brazilian Série A	508
2013/14	English Barclays Premier League	374
2013/14	French Ligue 1	386
2013/14	Italian Serie A	356
2013/14	Spanish La Liga	342
	TOTAL	2940

Table 1.1: List of the 7 Leagues of the 2013/2014 Season containing a total of 2940 games that the data set consists of.

In **Chapter 3** I introduce the *Constant Intensity Model* which is a risk-neutral valuation framework for pricing and hedging in-play football bets based on modelling scores by independent homogeneous Poisson processes with constant intensities. This model is in many sense the Black-Scholes model of football betting. The Fundamental Theorems of Asset Pricing are extended to this set up which enables us to derive arbitrage-free valuation formulae for contracts traded in the market.

In **Chapter 4** I present the *Local Intensity Model* which is an extension of the Constant Intensity Model by allowing for intensities that depend on the number of goals and time. I show that this model is in many respects similar to the Local Volatility model which is widely used in finance for option pricing. Borrowing the terminology from finance, I am referring to the model as the Local Intensity model and to the state and time dependent intensity function as the local intensity function. I am deriving formulae which are similar to Dupire's formula in the Local Volatility model. Calibration on real market data is also demonstrated.

In **Chapter 5** I introduce the *Microscopic Model* to describe the evolution of a football game. The model is an extension of the Local Intensity Model

Chapter	Model	Data set used	Goal Intensity Assumptions	Purpose of the model
Chapter 3	Constant Intensity	Market data	Constant intensity	Pricing and hedging in the risk-neutral measure
Chapter 4	Local Intensity	Market data	Depends on number of goals scored and time	Pricing and hedging in the risk-neutral measure
Chapter 5	Microscopic Model	Gameplay data	Depends on ball position, team holding the ball, number of goals scored, time	Prediction in the physical measure
Chapter 6	First Half Indicators Model	Gameplay data	Depends on 18-dimensional indicator vector observed at the end of the first half	Prediction in the physical measure

Table 1.2: A high level overview of the assumptions and data sets used by the different models and chapters.

by taking into account the position of the ball and the team holding the ball besides the number of goals scored and time. The model describes the evolution of these state parameters through a drift and diffusion coefficient for the ball position, a ball losing intensity for the team holding the ball and a usual goal intensity for the goals scored.

In **Chapter 6** I introduce the *First Half Indicators Model* which is the result of the study of various Machine Learning methods applied to football game data. The input of the model is an 18-dimensional indicator vector that is observed at the end of the first half of the game and the output of the model are goal intensities for the two teams that are assumed to be constant during the second half of the game.

In **Chapter 7** I use the Microscopic Model and the First Half Indicators Model to predict goal intensities in the second half of the game and I compared the predictive powers of the two models.

Table 1.2 summarizes the assumptions and data sets used in the different chapters and models. It is also worth noting that the purpose of the Constant Intensity and Local Intensity models are pricing and hedging in the risk-neutral measure where as the purpose of the Microscopic Model and the First Half Indicators Model is prediction in the physical measure.

1.4 Publication

Chapter 3 of this thesis has been submitted to the journal of Applied Mathematical Finance, it was reviewed, has been re-submitted and is currently awaiting the outcome of the second round of review. We are also planning to submit chapters 5 and 6 for publication by the final viva.

Chapter 2

Literature Review and Theoretical Background

This chapter contains an overview of the relevant literature, theorems and methods that are used throughout the subsequent chapters of the thesis.

2.1 Literature on Football Betting

The literature on football betting can be split into two main groups. The first group contains articles that study and model the distribution of the goals scored by the two teams at the end of the game, that is with no regard to the dynamics of the goal process during the game. The second group of articles study not only the full-time score distribution, but also the dynamics during the play which is referred to as in-play.

2.1.1 Full-time Score Distribution

One of the first articles studying the distribution of football scores is [4] which found that an independent Poisson distribution gives a reasonably accurate

description of football scores. It achieved further improvements by applying a bivariate Poisson distribution and it has been showed that it provides a better fit than the Poisson distribution. They found that the correlation coefficient between the scores of the home and away teams to be approximately equal to 0.2.

The Negative Binomial distribution has been used by [1] who found that it provides a better fit than the Poisson distribution. This has been confirmed by [2, 3] who studied a set of ball games including football games between 1954 and 1969 and also found that the Negative Binomial distribution provides a reasonably good fit.

The distribution of total scores as well as home and away scores has been studied by [49] in football games from 169 countries between 1999 and 2001. They found that the distributions do not follow the thin-tailed Poisson, nor the negative binomial distributions beyond the low scores, but heavy-tailed extremal distributions provide a better fit. In particular, the best fit of total scores was found to have a tail distribution with Gumbel α of 1, home and away scores were found to have a tail distribution with Fréchet α of 1.04 and 1.1, respectively.

The distribution of full time results of 6629 games between 1992 and 1995 has been studied by [5] and they suggested a modified Poisson distribution that accounts for the fact that the home and away scores are not independent. In particular, the proposed distribution function of the full time home and away scores is:

$$f_T(n_1, n_2) = \tau(n_1, n_2) \frac{e^{-\Lambda_1} \Lambda_1^{n_1}}{n_1!} \frac{e^{-\Lambda_2} \Lambda_2^{n_2}}{n_2!}, \tag{2.1}$$

where n_1 and n_2 are the full time scores of the home and away teams, Λ_1 and Λ_2 are the average number of goals scored by he home and away teams

during the game and the function $\tau(n_1, n_2)$ is responsible for the dependency between the scores of the two teams:

$$\tau(n_1, n_2) = \begin{cases} 1 - \Lambda_1 \Lambda_2 \rho & \text{if } n_1 = n_2 = 0, \\ 1 + \Lambda_1 \rho & \text{if } n_1 = 0,\ n_2 = 1, \\ 1 + \Lambda_2 \rho & \text{if } n_1 = 1,\ n_2 = 0, \\ 1 - \rho & \text{if } n_1 = 1,\ n_2 = 1, \\ 1 & \text{otherwise} \end{cases} \tag{2.2}$$

2.1.2 In-Play Score Dynamics

The distribution of in-play scores during the 90 minute interval of 4012 games between 1993 and 1996 has been studied by [23]. They found that goal scoring intensities depend on the game time with intensity increasing steadily as the game progresses. They also found that scoring intensities also depend on the current score. Therefore they proposed a model in which the goals scored by the two teams follow two Poisson processes with intensities that are functions of the current number of goals and time:

$$dN_t^i = d\mathcal{N}_t^{\lambda_i(N_t^1, N_t^2, t)},\ i \in \{1, 2\}, \tag{2.3}$$

where $i = 1$ refers to the home and $i = 2$ refers to the away team, N_t^i denotes the number of goals scored by team i, $\mathcal{N}_t^\lambda$ denotes a Poisson process of intensity λ and $\lambda_i(N_t^1, N_t^2, t)$ denotes the intensity function of team i which is a function of the current number of goals (N_t^1, N_t^2) and time t. The authors suggest a number of different functional forms for the intensities and find that the model can be applied both to explain the distribution of final scores and also to valuing in-play spread bets.

The standard homogeneous Poisson process has been applied by [17] to develop analytical valuation formulae for in-play spread bets on goals and also on corners.

A hierarchy of models suitable for in-play football betting has been studied by [42], starting from a homogeneous Poisson process with constant intensity, then introducing time-dependent intensities, finally introducing dependency on the current number of goals. He derived a valuation formula for in-play football bets in the general case when the intensity function is homogeneous in time:

$$\lambda_i (n_1, n_2, t) = \rho_i (n_1, n_2) \overline{\xi} (t), \ i \in \{1, 2\} \tag{2.4}$$

where $\rho_i (n_1, n_2)$ is the score-dependent component and $\overline{\xi} (t)$ is the time-dependent component.

The introduction of stochastic intensities has been suggested by [42] where the goal scoring intensities of the two teams depend not only on time and current score, but also on an independent Brownian driving process. In particular, he studied the case of the Cox-Ingerson-Ross process:

$$dx_t = \alpha (\theta - x_t) dt + \sigma \sqrt{x_t} dW_t, \tag{2.5}$$

where dW_t is a standard Brownian motion, α is the speed of mean reversion parameter, θ is the value of the mean that the process is reverting to and σ is the volatility parameter. A number of valuation formulae for in-play football bets have been derived within the stochastic intensity model.

Market rationality has been tested by [41] on the point spread betting market using match data for 1246 National Football League games between 1980 and 1985. They concluded that statistical tests cannot reject market

rationality while economic tests do reject market rationality. This implies that the betting market is reasonably efficient.

An efficiency test of point spread betting markets has been conducted by [39] using match data from 4219 NFL games between 1976 and 1994. They concluded that market inefficiencies are present, but they tend to dissipate over time.

2.2 Fundamental Theorems of Asset Pricing

The Fundamental Theorems of Asset Pricing form the basis of the risk-neutral framework of financial mathematics and derivative pricing. These theorems are relevant for pricing football bets because the bets are assets that are actually traded on markets and also because there are multiple kinds and forms of bets on the same underlying game in which sense the bets behave similar to derivative contracts.

The first Fundamental Theorem states that a market is arbitrage free if and only if there exists a probability measure under which the underlying asset prices are martingales. The second fundamental theorem states that the market is complete, (that is, any derivative product of the underlying assets can be dynamically replicated) if and only if the martingale measure is unique. These theorems have been developed by several authors, including [8], [10], [11], [35], [36], [37] and [12].

2.3 Overview of the Local Volatility model

Up until the crash of 1987, Black-Scholes implied volatilities of European options were constant across all strikes and maturities because the markets ac-

cepted the constant volatility assumption of the Black-Scholes model, where log-returns are assumed to follow the thin-tailed normal distribution. However, this changed after the crash of 1987 (see for example [19]), when the implied volatility surface became non-constant and the volatility smile appeared because markets dropped the log-normality assumption and started to use heavy-tailed distributions to price options. This became a problem because the Black-Scholes model suddenly became inconsistent with the market.

One of the solutions proposed was the introduction of the local volatility model by [31] and [45] which is still widely used today. The local volatility model allows the robust calibration of a stochastic process to any given implied volatility surface, and is therefore consistent with the whole range of European options prices. The fundamental assumption of the model is that volatility is a function of price and time. Formally, the price follows the following SDE:

$$dS_t = S_t \sigma_{LV}\left(S_t, t\right) dW_t + S_t r dt, \tag{2.6}$$

where $\sigma_{LV}\left(S_t, t\right)$ is the local volatility function and r is the risk-free rate.

One of the reasons for the popularity of the model is, that given a set of European option prices or Black-Scholes implied volatilities at the continuum of all strikes and maturities, the local volatility function can be determined in a straightforward way using the so-called Dupire formulas. The first formula describes the local volatility as a function of the European option prices:

$$\sigma_{LV}^2\left(K, T\right) = \frac{\partial C}{\partial T}\left[\frac{1}{2}K^2\frac{\partial^2 C}{\partial K^2} - (r-\gamma)K\frac{\partial C}{\partial K} - \gamma C\right]^{-1}, \tag{2.7}$$

where $C\left(K, T\right)$ is the price of a European option of strike K and maturity

T. The second formula connects the Black-Scholes implied volatlility surface with the local volatility surface:

$$\sigma_{LV}^2(K,T) = \frac{\partial w}{\partial T}\left[1 - \frac{k}{w}\frac{\partial w}{\partial k} + \frac{1}{4}\left(-\frac{1}{4} - \frac{1}{w} + \frac{k^2}{w^2}\right)\left(\frac{\partial w}{\partial k}\right)^2 + \frac{1}{2}\frac{\partial^2 w}{\partial k^2}\right]^{-1}, \tag{2.8}$$

where $w(K,T) = \sigma_{BS}^2(K,T)\,T$, $\sigma_{BS}(K,T)$ is the Black-Scholes implied volatility of a European option of strike K and maturity T and $k = \ln\left(e^{-rT}\frac{K}{S}\right)$ is the log-strike.

These formulas can be derived using the Black-Scholes formula, the Kolmogorov forward equation and a widely used result from [48] which states that the second derivative of undiscounted European call prices with respect to strike equal the marginal risk-neutral density function of the price. See for example [46] for details.

In order for these formulas to work, option prices or Black-Scholes implied volatilities are required for a continuum of strikes and maturities. However, prices are only quoted at a selected set of finite strikes (usually in the range of 5 to 20 quotes per maturity) and maturities (usually around 10 maturities starting from about 1 week up to a few years). This problem is solved in practice by fitting a parametric curve to the Black-Scholes implied volatility smile at each maturity and then interpolating these slices between maturities and thereby constructing the continuous Black-Scholes implied volatility surface. Dupire's formulas in Equations 2.7 or 2.8 can be used directly to compute the local volatility surface. Once the local volatility function is available, it can be used in equation 2.6 to generate a stochastic process which is consistent with all European option prices or Black-Scholes implied volatlities that were used in the calibration and at the same it time can also be used to price

exotic options with an arbitrary path dependent payoff.

2.4 Numerical methods for solving partial differential equations

A quick overview of the standard numerical method for solving partial differential equations is provided below which is used to solve the Microscopic Model in Chapter 5.

Numerical solution of partial differential equations is discussed by [51], [52] and [53] among others. For our purposes we are interested in parabolic equations of the following form which describe the time evolution of a marginal probability density function $f(x,t)$:

$$\frac{\partial}{\partial t} f(x,t) = -\frac{\partial}{\partial x}\left[\mu(x) f(x,t)\right] + \frac{1}{2}\frac{\partial^2}{\partial x^2}\left[\sigma^2(x) f(x,t)\right] \tag{2.9}$$

where $x \in [x_{min}, x_{max}]$ is the space variable and $t \in [0,T]$ is the time variable. The initial condition in time is a Dirac delta at an initial state x_0:

$$f(x, t=0) = \delta(x - x_0) \tag{2.10}$$

The boundary conditions in space are reflective.

2.4.1 Gradient descent method

The gradient descent method has been introduced by [80] among others. It is an iterative method for finding the minimum of a differentiable multivariate function $F(x)$, starting from an initial approximation x_0. The method consists of computing the sequence

$$x_{n+1} = x_n - \gamma \nabla F(x_n) \tag{2.11}$$

where $\nabla F(x_n)$ is the gradient of the function $F(x)$ and γ is a step size which is a parameter of the method. Besides ease of implementation and numerical stability, the advantage of the method lies in the fact that it doesn't involve costly computation of inverse Hessian matrices, like for example in case of Newton's method, therefore it is applicable for problems with high dimensions where inverse matrix computation would be unfeasible.

2.4.2 Finite difference operators and boundary conditions

In the finite difference method space is discretized by choosing a grid size Δx and setting $x_i = x_{min} + i\Delta x$ with $i \in [0, N-1]$ where $N = \frac{x_{max} - x_{min}}{\Delta x}$ is the number of grid points.

The first derivative $\frac{d}{dx}$ is approximated by the following finite difference operator $\frac{\Delta}{\Delta x}$:

$$\frac{df(x_i)}{dx} \approx \frac{\Delta f(x_i)}{\Delta x} = \frac{1}{\Delta x} \begin{cases} f(x_{i+1}) - f(x_i) & if\, i \in [1, N-2] \\ f(x_{i+1}) & if\, i = 0 \\ -f(x_i) & if\, i = N-1 \end{cases} \tag{2.12}$$

where the cases for the boundary values $i = 0$, $i = N-1$ are coming from the reflective boundary condition $f(x_{min}) = f(x_{max}) = 0$ in case of the first derivative.

The second derivative $\frac{d^2}{dx^2}$ is approximated by the following finite differ-

ence operator $\frac{\Delta^2}{\Delta x^2}$:

$$\frac{d^2 f(x_i)}{dx^2} \approx \frac{\Delta^2 f(x_i)}{\Delta x^2} = \frac{1}{\Delta x^2} \begin{cases} f(x_{i+1}) - 2f(x_i) + f(x_{i-1}) & if\, i \in [1, N-2] \\ f(x_{i+1}) - f(x_i) & if\, i = 0 \\ -f(x_i) + f(x_{i-1}) & if\, i = N-1 \end{cases} \tag{2.13}$$

where the cases for the boundary values $i = 0$, $i = N-1$ are coming from the reflective boundary condition $\frac{df}{dx}\big|_{x=x_{min}} = \frac{df}{dx}\big|_{x=x_{max}} = 0$ in case of the second derivative.

Time is discretized similarly to space by introducing a time step size of Δt and setting $t_j = j\Delta t$ with $j \in [0, M-1]$ where $M = \frac{T}{\Delta t}$ is the number of time steps.

The first order time derivative is approximated with the same finite difference operator as space, except for the boundary conditions:

$$\frac{df(t_j)}{dt} \approx \frac{1}{\Delta t} f(t_{j+1}) - f(t_j)\, for\ t \in [0, M-2] \tag{2.14}$$

2.4.3 Explicit Euler method

Within the explicit Euler method, Equation 2.9 is discretized as follows:

$$\frac{\Delta}{\Delta t} f(x_i, t_j) = -\frac{\Delta}{\Delta x} \left[\mu(x_i) f(x_i, t_j)\right] + \frac{1}{2} \frac{\Delta^2}{\Delta x^2} \left[\sigma^2(x_i) f(x_i, t_j)\right] \tag{2.15}$$

The advantage of the scheme is that the value of the function in the next time step $f(x_i, t_{j+1})$ can be obtained simply by ordering the terms:

$$f(x_i, t_{j+1}) = f(x_i, t_j) + \Delta t \left\{ -\frac{\Delta}{\Delta x} \left[\mu(x_i) f(x_i, t_j) \right] + \frac{1}{2} \frac{\Delta^2}{\Delta x^2} \left[\sigma^2(x_i) f(x_i, t_j) \right] \right\} \tag{2.16}$$

However, a major disadvantage is that in order to keep the solution numerically stable, the space and time step sizes must meet the following criteria:

$$\Delta x \leq \frac{\sigma^2}{2\mu} \tag{2.17}$$

$$\Delta t \leq \frac{\sigma^2}{4\mu^2} \tag{2.18}$$

In practice this might be a problem because for arbitrary values of σ and μ the grid size might exceed the memory capacity of the computer.

2.4.4 Implicit Euler method

Within the implicit Euler method, Equation 2.9 is discretized as follows:

$$\frac{\Delta}{\Delta t} f(x_i, t_j) = -\frac{\Delta}{\Delta x} \left[\mu(x_i) f(x_i, t_{j+1}) \right] + \frac{1}{2} \frac{\Delta^2}{\Delta x^2} \left[\sigma^2(x_i) f(x_i, t_{j+1}) \right] \tag{2.19}$$

Using this scheme the value of the function on the next time step $f(x_i, t_{j+1})$ appears on both sides, and therefore cannot be solved by simple ordering of terms but the equation must be solved instead as a numerical system. Our method of preference is the gradient method as described in section 2.4.1 because of it's ease of implementation. The main advantage of the implicit method is that it is always stable numerically, regardless of the step sizes Δt

and Δx.

2.5 Statistical Methods

The maximum likelihood method has been introduced by [82], a good overview is also available in [81]. It is a generic method for the estimation of parameters of a statistical model using observations. The method aims to find the model parameters that maximise the value of the likelihood function given the available observations. In practice it is usually advantageous to maximise the logarithm of the likelihood function, that is the log-likelihood function. In this section we summarise the log-likelihood functions of the Poisson distribution and the Poisson process.

2.5.1 Log-Likelihood of the Poisson Distribution

Proposition 1. *The log-likelihood ratio of two Poisson distributions with intensities $\lambda_0 T$, $\lambda_1 T$ and outcome k is:*

$$\ln \frac{\mathbb{P}\left[k|\lambda_1 T\right]}{\mathbb{P}\left[k|\lambda_0 T\right]} = -\left(\lambda_1 - \lambda_0\right) T + k \ln \frac{\lambda_1}{\lambda_0} \tag{2.20}$$

Proof. This follows directly form the definition of the Poisson distribution:

$$\mathbb{P}\left[k|\lambda T\right] = e^{-\lambda T} \frac{\left(\lambda T\right)^k}{k!} \tag{2.21}$$

□

2.5.2 Log-Likelihood of the Poisson Process

Proposition 2. *The log-likelihood ratio of two inhomogeneous Poisson processes with intensity functions $\lambda_0(t)$, $\lambda_1(t)$ and with a realised path that on*

a time interval $t \in [T_0, T_1]$ *had a total of* k *jumps at times* $\{t_i\}_{i=1}^k$ *is:*

$$\ln \frac{\mathbb{P}\left[\{t_i\}_{i=1}^k | \lambda_1(t)\right]}{\mathbb{P}\left[\{t_i\}_{i=1}^k | \lambda_0(t)\right]} = -\int_{T_0}^{T_1} (\lambda_1(t) - \lambda_0(t))\, dt + \sum_{i=1}^{k} \ln \frac{\lambda_1(t_i)}{\lambda_0(t_i)} \tag{2.22}$$

Proof. **Sketch of proof** Let us split the time interval $[T_0, T_1]$ into small time segments of length dt. The probability of no jump within a time segment $[t, t+dt]$ is equal to $\exp\left(-\int_t^{t+dt} \lambda(t')dt'\right)$ which in case dt is infinitesimally small is equal to $\exp(-\lambda(t)dt)$. The probability of one or more jumps is therefore $1 - \exp(-\lambda(t)dt)$ which again, in case dt is small is equal to $-\lambda(t)dt$. The likelihood of the whole path which is the joint probability of the realised events on the infinitesimal line segments is just the product of the probabilities because the events are independent of each other. That is, for all segments where no jump happened we have a term $\prod_{t\in[T_0,T_1]\setminus\{t_i\}_{i=1}^n} \exp(-\lambda(t)dt)$ which is equal to $\exp\left(-\int_{T_0}^{T_1} \lambda(t)dt\right)$ where technically the integral shouldn't contain the times of jumps $\{t_i\}_{i=1}^n$, but because those are a finite number of points they have no effect on the value of the integral and can be omitted. From the segments where a jump happened we have $\prod_{i=1}^n \lambda(t_i)dt$. The final result is the logarithm of the product of these two terms. □

2.5.3 Akaike Information Criterion - AIC

When using different models with different number of parameters, usually a model with a larger number of parameters provides a better fit, simply because more parameters allow more flexibility. Therefore, when model selection is performed, a penalty on the number of parameters is desirable in order to avoid over-fitting. The Akaike Information Criterion (AIC) is a commonly used estimator of the relative quality of a statistical model, see

among others [64, 65, 66]. According to the AIC, the preferred model is the one with the lowest AIC value which is defined as:

$$\text{AIC} = 2k - 2\mathcal{L}, \tag{2.23}$$

where k is the number of model parameters and $\mathcal{L}$ is the value of the log-likelihood. The number of model parameters k acts as a penalty term that discourages over-fitting.

2.5.4 Wilks' Theorem

Wilks' Theorem [67] states that the log-likelihood ratios of two nested models follow a χ^2 distribution. Let $f(x, \theta_1, \theta_2, ...\theta_h)$ denote the likelihood function of the more generic model of h parameters and let $f(x, \theta_{01}, \theta_{02}, ...\theta_{0m}, \theta_{m+1}, ...\theta_h)$ denote the likelihood of the more specific model which is a special case of the generic model by setting the first m parameters to the fixed values of $\theta_{01}, ...\theta_{0m}$. The maximum log-likelihood ratio of the nested models is defined as:

$$\mathcal{L} = \frac{\max_{\theta_{m+1},...\theta_h} f(x, \theta_{01}, \theta_{02}, ...\theta_{0m}, \theta_{m+1}, ...\theta_h)}{\max_{\theta_1,...\theta_h} f(x, \theta_1, ...\theta_h)} \tag{2.24}$$

Wilks' theorem states that if the sample size $n \to \infty$ is large, then the distribution of $-2\mathcal{L}$ is distributed like χ^2 with $h - m$ degrees of freedom.

2.6 Machine Learning

Machine learning is the study of a set of statistical models and algorithms that are used to perform specific tasks, relying on inference and patterns, rather than explicit instructions. Machine learning is classified in one of the following categories:

- **Supervised learning** methods are used in situations where both the input and desired output data is available and models are trained to produce the desired output on the given input.

  Feature selection methods aim to select a subset of the input variables that are most relevant in predicting or explaining the output variables, thereby reducing the dimensionality of the problem.

  Classification is the problem of identifying which category a an item belongs to, given a set of observed variables about the item.

  Regression is a set of statistical methods that aim to estimate the relationship among a set of variables.

- **Unsupervised learning** methods are used when only the input data is available and the method finds structure in the data.

- **Reinforcement learning** is concerned how a software agent makes decisions in an environment over time such that a given reward is maximised.

In this thesis I use supervised learing methods.

2.6.1 Correlation

The Pearson correlation coefficient is defined as:

$$\rho_{X,Y} = \frac{\text{cov}(X,Y)}{\sigma_X \sigma_Y} \tag{2.25}$$

where cov is the covariance and σ_X, σ_Y are the standard deviations of X and Y.

2.6.2 Random Forest of Trees

Decision trees are used for regression, classification and feature importance selection. The construction of the tree involves splitting the data set in two parts. The splitting is repeated in a recursive manner on both halves until the number of data points in the remaining parts becomes smaller than a pre-determined value at which point the tree is considered complete. The training of the model involves the construction of decision rules at each of the splits, according which the split will be performed. The decision rules are such that first one of the components of the multi-dimensional indicator vector x_j is selected and then a threshold value θ is determined. The split is performed so that data points with an indicator value less than the threshold, that is $x_j < \theta$ constitute one partition and $x_j \geq \theta$ constitute the other partition. During training each of these decision rules, that is the selection of the indicator component and the threshold value at each of the splits are constructed such that the outcome values y in the resulting partitions minimise a specific measure. A popular measure for discrete outcomes that I also used is the Gini impurity [76]:

$$\sum_{i=1}^{J} p_i \sum_{k \neq i} p_k = \sum_{i=1}^{J} p_i \left(1 - p_i\right) \tag{2.26}$$

where J is the number of classes, that is the number of distinct values of the outcome vector y and p_i is the frequency of the i.th outcome in the partition. Note that Gini impurity is essentially the sum of the binomial variances of the class frequencies. Gini impurity will therefore prefer splits where the different partitions have minimal variances.

Decision trees are prone to over-fitting. One way of tackling this problem is by constructing multiple trees by using random subsets of the data and

averaging out the results across the trees. I used the Extra-Trees method [75] that goes one step further in introducing randomness by choosing the splitting thresholds randomly from a set of candidates, rather than always choosing the most discriminative threshold.

2.6.3 K-Nearest Neighbors

The K-Nearest Neighbour model has been introduced by [77]. The training of the model doesn't involve any optimisation, instead every X and Y point of the data set is saved in a data structure. The output of the model for an arbitrary input x is determined by selecting a certain number $n_{neighbors}$ of nearest points from the data set that have the smallest distance $\|x - X\|_q$. The output of the model is determined by taking the average of the values of the nearest neighbor points, formally:

$$y(x) = \frac{1}{n_{\text{neighbors}}} \sum_{j \in S(x)} Y_j \tag{2.27}$$

where $S(x)$ is the set of indices j in the data set of the nearest neighbors, that is those $n_{\text{neighbors}}$ points that have the smallest distance $\|x - X_j\|_q$ where q is the exponent of the distance measure, that is $q = 2$ for Euclidean distance and $q = 1$ for L1 distance.

It is important to note that before applying this method, indicators need to be scaled to zero mean and unit variance. This is necessary because otherwise points along the dimension of an indicator that has lower variance would appear closer and therefore the results would be more sensitive to certain indicators simply due to their natural scaling.

Name	Function
Tanh	$\tanh(x)$
Sigmoid	$(1+e^{-x})^{-1}$
Softsign	$\frac{x}{1+\|x\|}$
ReLU	$\max(0,x)$
Linear	x

Table 2.1: List of commonly used activation functions for neural networks.

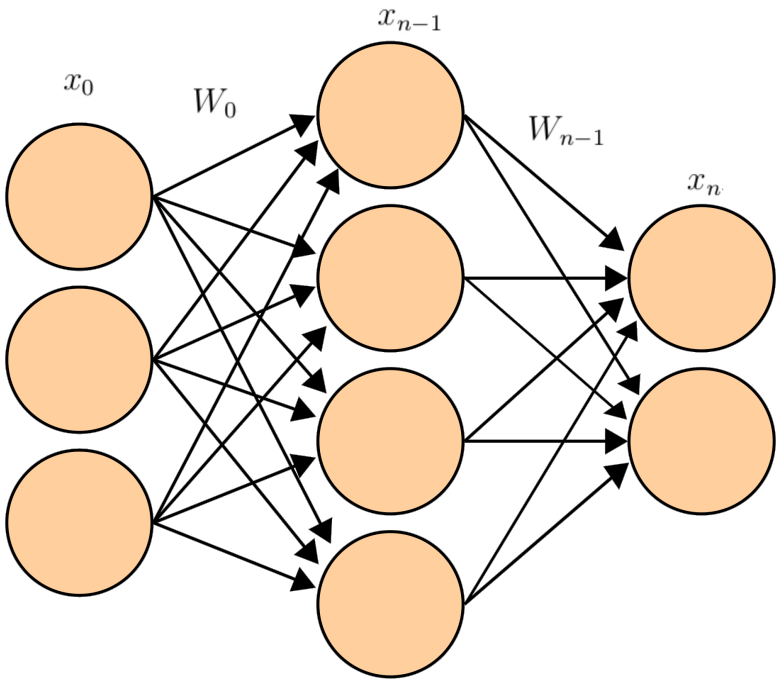


Figure 2.1: Overview of the Neural Network model.

2.6.4 Neural Networks

Feed-forward or non-recurrent Neural Networks consist of a set of layers chained together in a way such that the subsequent layers take their input from the output of previous layers with the input of the first layer being the overall input and the output of the last layer being the overall output to the model. Each layer consists of the following operations:

1. multiply the input vector x by a weight matrix W
2. add a scalar bias term b
3. apply an element-wise non-linear activation function $h(.)$

Formally, the output of neuron j in layer i is equal to:

$$x_{i+1}^{j} = h_i \left(b_i^j + \sum_k W_{i,k}^j x_i^k \right), \tag{2.28}$$

where x_i^k is the input from the k.th neuron of the previous layer, that is the output of the k.th neuron in the $i-1$.th layer. $W_{i,k}^j$ if the j.th row and k.th column in the weight matrix of the i.th layer. b_i^j is the bias term for the j.th neuron of the i.th layer. h_i is the activation function of the i.th layer. x_{i+1}^j is the output of the j.th neuron in the i.th layer. A list of commonly used activation functions is shown in Table 2.1.

Equation 2.28 can be written in vector form:

$$x_{i+1} = h_i \left(b_i + W_i x_i \right), \tag{2.29}$$

where x_i is the output vector of the $i-1$.th layer, W_i is the weight matrix of the i.th layer, b_i is the bias vector of the i.th layer and x_{i+1} is the output vector of the i.th layer.

The overall output of a network that consists of n layers is therefore:

$$y = h_{n-1} \left(b_{n-1} + W_{n-1} h_{n-2} \left(\ldots b_1 + W_1 h_0 \left(b_0 + W_0 x \right) \right) \right) \tag{2.30}$$

where the input of the first layer is dented by $x = x_0$ and the output of the n.th layer is denoted by $y = x_n$. An overview of the Neural Network model's architecture is shown in Figure 2.1

2.6.4.1 Vanishing Gradients

Vanishing gradients is a common issue with neural networks. In case of a sigmoid or tanh activation function, if the input value to the activation function is very low or very high and is outside of the nonlinearity of the function then the function's derivative drops to zero which slows down training because gradients will have little effect on the weights. One common heuristics is to remedy the problem by initialising the weights using standard normal random variables, but scaling with $1/\sqrt{n}$ where n is the number of neurons, thereby resulting in a standard normal variable for the sum which is the initial input to the activation function which is therefore not expected to be either extremely low or high. This is the strategy that I chose, however additional, more sophisticated methods are available, see for example [72].

2.6.5 Regularisation

The aim of regularisation methods is to solve ill-posed problems or to avoid over-fitting of models. Two of the most widely used regularisation methods are Lasso and Elastic Net.

2.6.5.1 Lasso Regularisation

Lasso regularisation has been introduced in [63]. It is essentially a linear regression with an L1 penalty term on the regression coefficients. Formally, the minimizes the following expression:

$$\min_{b,w} \frac{1}{N} \sum_{i=1}^{N} (Y_i - b - X_i^T w)^2 + \alpha \sum_{j=1}^{p} |w| \tag{2.31}$$

where X_i and Y_i are the input and output vectors of the regression, w is the vector or coefficients and b is the bias term. α is the main parameter

that determines the strength of the regularisation. The Lasso regression is also used as a feature selection method because the penalty term tends to zero out the coefficients for terms that are less relevant.

2.6.5.2 Ridge Regression

Ridge regression, also known as Tikhonov regularisation is a linear regression with an L2 penalty term on the regression coefficients. Formally, the minimizes the following expression:

$$\min_{b,w} \frac{1}{N} \sum_{i=1}^{N} (Y_i - b - X_i^T w)^2 + \beta \sum_{j=1}^{p} |w|^2 \tag{2.32}$$

where X_i and Y_i are the input and output vectors of the regression, w is the vector or coefficients and b is the bias term. β is the main parameter that determines the strength of the regularisation.

2.6.5.3 Elastic Net Regularisation

Elastic Net regularisation is the combination of Lasso regularisation and Ridge regression in the sense that it introduces both an L1 and an L2 penalty term.

One way to avoid over-fitting is to use the Elastic Net method which adds both an L1 and an L2 penalty term to the log-likelihood. The objective function to minimise becomes:

$$\min_{b,w} \frac{1}{n} \sum_{i=1}^{n} (Y_i - b - X_i^T w)^2 + \alpha |w|_1 + \beta |w|_2 \tag{2.33}$$

where α is the factor controlling the strength of the L1 penalty term and β is the factor for the L2 term. The benefits of also using an L2 penalty term besides an L1 penalty term are that an L1 term alone tends to select

only one component from a set of highly correlated components. Most of the time this is desirable, however there might be cases where several of a highly correlated subset of components carry useful information in which case there is no way for an L1 regularisation alone to select more than one component, however with an additional L2 term this can be controlled with the parameter β. Furthermore, if the number of parameters p is higher than the number of samples n, Lasso selects only at most n coefficients before it saturates - this is not the case with Elastic Net.

Chapter 3

The Constant Intensity Model

In this chapter a risk-neutral valuation framework is developed for pricing and hedging in-play football bets based on modelling scores by independent homogeneous Poisson processes with constant intensities. The Fundamental Theorems of Asset Pricing are extended to this set up which enables us to derive arbitrage-free valuation formulae for contracts traded in the market. It is also described how to calibrate the model to the market and how transactions can be replicated and hedged. In order to refine the model two extensions are suggested to account for the non-constant nature of the Poisson intensities observed in the market: local intensity models, and stochastic intensity models.

3.1 Introduction

In this chapter we use independent homogeneous Poisson processes to model the scores of the two teams. We postulate a market of three underlying assets and show that within this model a unique martingale measure exists and therefore the market of in-play football bets is arbitrage-free and complete.

Homogeneous Poisson processes have been used before to model in-play football bets. However, to our knowledge it hasn't been shown before that the fundamental theorems of asset pricing apply to this setup and that the model results in an arbitrage-free and complete market that can be used to risk-manage and replicate a portfolio of bets.

3.2 An example game

In order to demonstrate our results let us here select the Portugal vs. Netherlands game from the UEFA Euro 2012 Championship which was played on the 22nd of June 2012. The reason for selecting this particular game is that the game had a rather complex unfolding with Netherlands scoring the first goal, but then Portugal taking the lead in the second half and finally winning the game. This made the odds fluctuate wildly during the game which makes it a good candidate to demonstrate how the model performs in an extreme situation. The number of goals as a function of game time is shown on Figure 3.1.

Figures 3.2 and 3.3 show historical market values of two bet types traded on a betting market called Betfair: Match Odds and Over-Under. Match Odds contains three bets: home team winning the game, away team winning the game and the draw. Over-Under contains bets on the total number of goals where Under X.5 is a bet that pays off if the total number of goals is equal or less than X. The dashed lines show the best buy and sell offers on the market while the continuous lines show the calibrated model values (see Section 3.5).

In case of Match Odds, the value of the bet for Netherlands winning the game jumped after Netherlands scored the first goal. When the scores

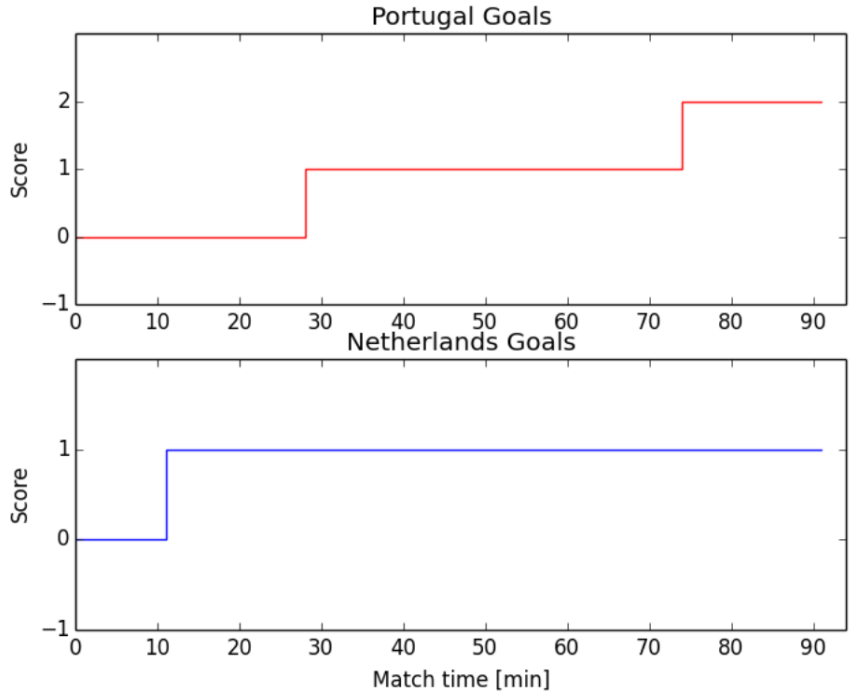


Figure 3.1: Scores of the two teams during the Portugal vs. Netherlands game on the 22nd of June, 2012. The half time results were 1-1 and the final results were a 2-1 win for Portugal.

became even after Portugal scored a goal, the value of the Draw bet jumped up and when Portugal took the lead by scoring the third goal, the value of the bet for Portugal winning the game jumped up. Finally, by the end of the game the value of the bet for Portugal winning the game converged to 1 and the value of the other bets went to zero.

In case of the Over-Under bets, trading ceased for the Under 0.5 bet after the first goal because this bet became worthless. By the end of the game, the value of the Under 3.5, 4.5, 5.5, 6.5 and 7.5 bets converged to 1 because the total number of goals was actually 3 and the values of the Under 0.5, 1.5 and 2.5 bets went to zero.

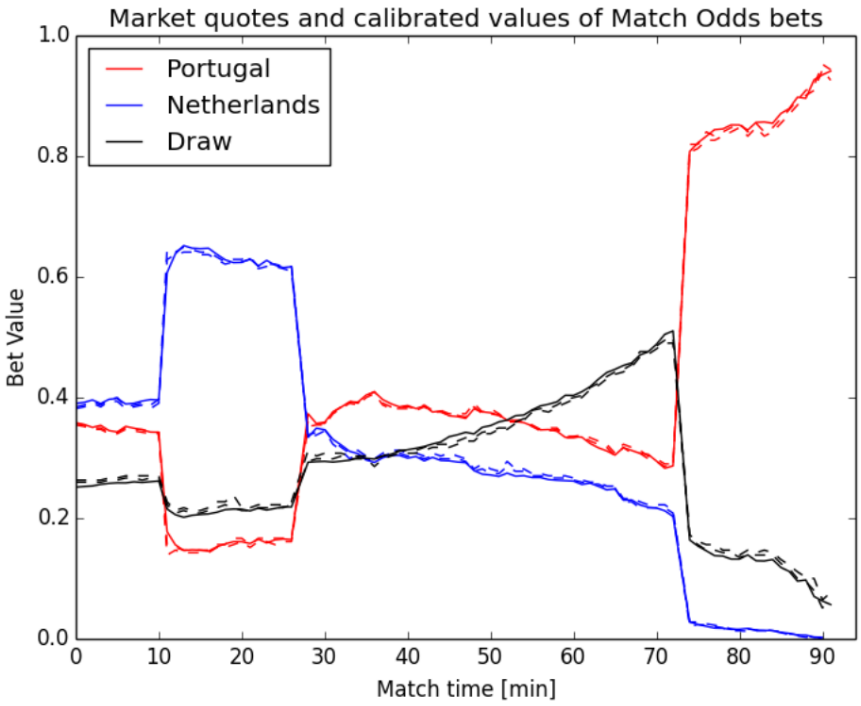


Figure 3.2: Values of the three Match Odds bets during the game: Draw (black), Portugal Win (red), Netherlands Win (blue). Dashed lines represent the best market buy and sell offers while the continuous lines represent the calibrated model values. Note hat the value of the Netherlands Win bet jumps up after the first goal because the chance for Netherlands winning the game suddenly increased. It jumped down for similar reasons when Portugal scored it's first goal and at the same time the value of the Portugal Win and Draw bets jumped up. By the end of the game, because Portugal actually won the game, the value of the Portugal Win bet reached 1 while both other bets became worthless.

3.3 Mathematical framework

In this section a risk-neutral valuation model for in-play football betting is presented. To do so, I follow the standard financial mathematical approach in which I start by assuming a probability space, then identify a market of tradable underlying assets and postulate a model for the dynamics of these assets. I show that the first and second fundamental theorems of asset pricing

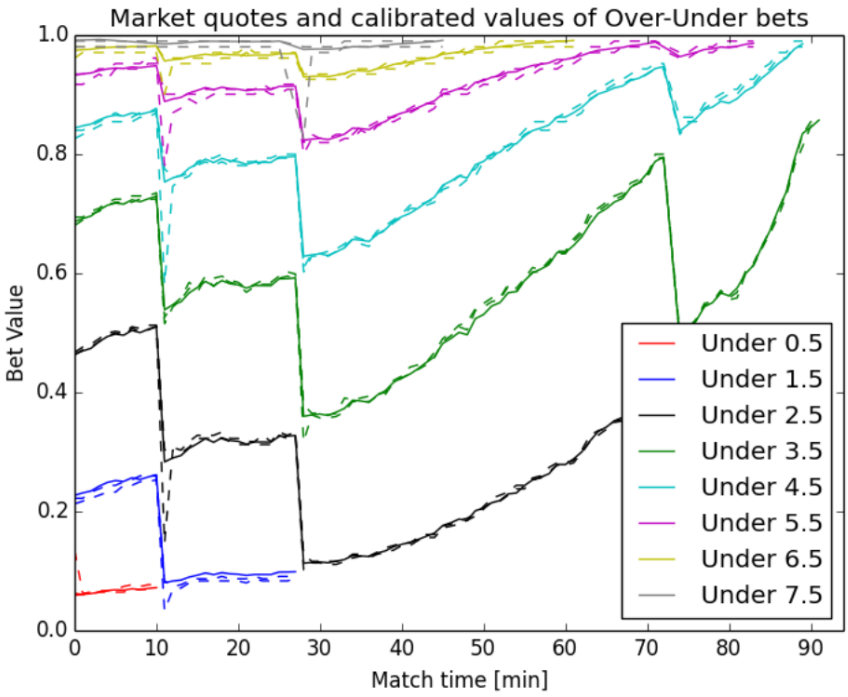


Figure 3.3: Values of Over/Under bets during the game. Under X.5 is a bet that pays off in case the total number of goals by the end of the game is below or equal to X. Marked lines represent the calibrated model prices while the grey bands show the best market buy and sell offers. Note that after the first goal trading in the Under 0.5 bet ceased and it became worthless. By the end of the game when the total number of goals was 3, all the bets up until Under 2.5 became worthless while the Under 3.5 and higher bets reached a value of 1.

apply to this market, that is the market is arbitrage-free and complete which means that all derivatives can be replicated by taking a dynamic position in the underlying assets.

In classical finance, the distinction between the underlying asset (which is usually a stock) and a derivative (which is usually an option on the stock) is natural. However, in football betting there is no such clear distinction because all bets are made on the scores, which is not a tradable asset. Therefore, in order to make the model work, I need to introduce two underlying assets, which at the end of the game have values equal to the number of goals scored

by the home and away teams. These assets are not among the most liquidly traded bets, therefore the choice might seem unusual. However, it is convenient from a mathematical point of view and towards the end of the section I will show that completeness applies in an extended sense in which a bet can not only be replicated from these artificial underlying assets, but also from any two linearly independent bets.

3.3.1 Setup

Let us consider a probability space $(\Omega, \mathcal{F},)$ that carries two independent Poisson processes N_t^1, N_t^2 with respective intensities μ_1, μ_2 and the filtration $(\mathcal{F}_t)_{t\in[0,T]}$ generated by these processes. Let time $t = 0$ denote the beginning and $t = T$ the end of the game. The Poisson processes represent the number of goals scored by the teams, the superscript 1 refers to the home and 2 refers to the away team. This notation is used throughout, the distinction between superscripts and exponents will always be clear from the context. The probability measure $\mathbb{P}$ is the real-world or physical probability measure.

I assume that there exists a liquid market where three assets can be traded continuously with no transaction costs or any restrictions on short selling or borrowing. The first asset B_t is a risk-free bond that bears no interests, an assumption that is motivated by the relatively short time frame of a football game. The second and third assets S_t^1 and S_t^2 are such that their values at the end of the game are equal to the number of goals scored by the home and away teams, respectively.

Definition 3. (model) The model is defined by the following asset dynamics:

$$\begin{aligned} B_t &= 1 \\ S_t^1 &= N_t^1 + \lambda_1 (T-t) \\ S_t^2 &= N_t^2 + \lambda_2 (T-t) \end{aligned} \tag{3.1}$$

where λ_1 and λ_2 are known real constants.

3.3.2 Risk-neutral pricing of bets

Definition 4. (trading strategy) A *trading strategy* is an $\mathcal{F}_t$-predictable vector process $\phi_t = (\phi_t^0, \phi_t^1, \phi_t^2)$ that satisfies $\int_0^t |\phi_s^i| \, ds < \infty$ for $i \in \{0, 1, 2\}$. The associated *value process* is denoted by

$$V_t^\phi = \phi_t^0 B_t + \phi_t^1 S_t^1 + \phi_t^2 S_t^2. \tag{3.2}$$

The trading strategy is *self-financing* if

$$V_t^\phi = V_0^\phi + \int_0^t \phi_s^1 dS_s^1 + \int_0^t \phi_s^2 dS_s^2. \tag{3.3}$$

where $\int_0^t \phi_s^i dS_s^i$, $i \in \{1, 2\}$ is a Lebesgue Stieltjes integral which is well defined according to Proposition 2.3.2 on p17 of [27].

Definition 5. (arbitrage-freeness) The model is *arbitrage-free* if no self-financing trading strategy ϕ_t exist such that $\mathbb{P}\left[V_t^\phi - V_0^\phi \geq 0\right] = 1$ and $\mathbb{P}\left[V_t^\phi - V_0^\phi > 0\right] > 0$.

Definition 6. (bet) A *bet* (also referred to as a *contingent claim* or *derivative*) is an $\mathcal{F}_T$-measurable random variable X_T.

Remark 7. In practical terms this means that the value of a bet is revealed at the end of the game.

Definition 8. (completeness) The model is *complete* if for every bet X_T there exists a self-financing trading strategy ϕ_t such that $X_T = V_T^{\phi}$. In this case it is said that the bet X_T is *replicated* by the trading strategy ϕ_t.

Theorem 9. *(risk-neutral measure) There exists a probability measure* $\mathbb{Q}$ *referred to as the risk-neutral equivalent martingale measure such that:*

(a) The asset processes B_t, S_t^1, S_t^2 *are* $\mathbb{Q}$*-martingales.*

(b) The goal processes N_t^1 *and* N_t^2 *in measure* $\mathbb{Q}$ *are standard Poisson processes with intensities* λ_1 *and* λ_2 *respectively (which are in general different from the* $\mathbb{P}$*-intensities of* μ_1 *and* μ_2*).*

(c) $\mathbb{Q}$ *is an equivalent measure to* $\mathbb{P}$*, that is the set of events having zero probability is the same for both measures.*

(d) $\mathbb{Q}$ *is unique.*

Proof. The proof relies on Girsanov's theorem for point processes (see Theorem 2 on p.165 and Theorem 3 on page 166 in [27]) which states that N_t^1 and N_t^2 are Poisson processes with intensities λ_1 and λ_2 under the probability measure $\mathbb{Q}$ which is defined by the Radon-Nikodym-derivative

$$\frac{d\mathbb{Q}}{d\mathbb{P}} = L_t, \tag{3.4}$$

where

$$L_t = \prod_{i=1}^{2} \left(\frac{\lambda_i}{\mu_i}\right)^{N_t^i} \exp\left[\left(\mu_i - \lambda_i\right) t\right]. \tag{3.5}$$

Then uniqueness follows from Theorem 8 on p.64 in [27] which states that if two measures have the same set of intensities, then the two measures must coincide. The Integration Theorem on p.27 of [27] states that $N_t^i - \lambda_i t$

are $\mathbb{Q}$-martingales, therefore the assets S_t^i are also $\mathbb{Q}$-martingales for $i \in \{1, 2\}$. Proposition 9.5 of [26] claims that $\mathbb{P}$ and $\mathbb{Q}$ are equivalent probability measures. The process of the bond asset B_t is a trivial martingale in every measure because it's a deterministic constant which therefore doesn't depend on the measure. □

Remark 10. Changing the measure of a Poisson process changes the intensity and leaves the drift unchanged. This is in contrast with the case of a Wiener process where change of measure changes the drift and leaves the volatility unchanged.

Theorem 11. *(arbitrage-free) The model is arbitrage-free and complete.*

Proof. This follows directly from the first and second fundamental theorems of finance. To be more specific, arbitrage-freeness follows from theorem 1.1 of [22] which states that the existence of a risk-neutral measure implies a so-called condition "no free lunch with vanishing risk" which implies arbitrage-freeness. Completeness follows from theorem 3.36 of [11] which states that the model is complete if the risk-neutral measure is unique. Alternatively it also follows from theorem 3.35 which states that the model is complete if the martingale representation theorem holds for all martingales which is the case according to Theorem 17, p.76 of [27]. □

Corollary 12. *The time-t value of a bet is equal to the risk-neutral expectation of its value at the end of the game, formally:*

$$X_t = \mathbf{E}^{\mathbb{Q}} \left[X_T | \mathcal{F}_t \right]. \tag{3.6}$$

Proof. This follows directly from Proposition 3.31 of [11]. □

Corollary 13. *The time-t value of a bet is also equal to the value of the associated self-financing trading strategy ϕ_t, formally:*

$$X_t = V_t^{\phi} = V_0^{\phi} + \int_0^t \phi_s^1 dS_s^1 + \int_0^t \phi_s^2 dS_s^2. \tag{3.7}$$

Proof. This follows directly from Proposition 3.32 of [11]. □

Definition 14. (linear independence) The bets Z_T^1 and Z_T^2 are *linearly independent* if the self-financing trading strategy $\phi_t^1 = (\phi_t^{10}, \phi_t^{11}, \phi_t^{12})$ that replicates Z_T^1 is $\mathbb{P}$-almost surely linearly independent from the self-financing trading strategy $\phi_t^2 = (\phi_t^{20}, \phi_t^{21}, \phi_t^{22})$ that replicates Z_T^2. Formally, at any time $t \in [0, T]$ and for any constants $c_1, c_2 \in \mathbb{R}$

$$c_1\phi_t^1 \neq c_2\phi_t^2 \quad \mathbb{P}\ a.s. \tag{3.8}$$

Theorem 15. *(replication) Any bet X_T can be replicated by taking a dynamic position in any two linearly independent bets Z_T^1 and Z_T^2, formally:*

$$X_t = X_0 + \int_0^t \psi_s^1 dZ_s^1 + \int_0^t \psi_s^2 dZ_s^2, \tag{3.9}$$

where the weights ψ_t^1, ψ_t^2 are equal to the solution of the following equation:

$$\begin{pmatrix} \phi_t^{11} & \phi_t^{12} \\ \phi_t^{21} & \phi_t^{22} \end{pmatrix} \begin{pmatrix} \psi_t^1 \\ \psi_t^2 \end{pmatrix} = \begin{pmatrix} \phi_t^1 \\ \phi_t^2 \end{pmatrix} \tag{3.10}$$

where $(\phi_t^{11}, \phi_t^{12})$, $(\phi_t^{21}, \phi_t^{22})$ and (ϕ_t^1, ϕ_t^2) are the components of the trading strategy that replicates Z_T^1, Z_T^2 and X_T, respectively. The integral $\int_0^t \psi_s^1 dZ_s^1$ is to be interpreted in the following sense:

$$\int_0^t \psi_s^1 dZ_s^1 = \int_0^t \psi_s^1 \phi_s^{11} dS_s^1 + \int_0^t \psi_s^1 \phi_s^{12} dS_s^2 \tag{3.11}$$

and similarly for $\int_0^t \psi_s^2 dZ_s^2$.

Proof. Substituting $dZ_t^1 = \phi_t^{11} dS_t^1 + \phi_t^{21} dS_t^2$, $dZ_t^2 = \phi_t^{12} dS_t^1 + \phi_t^{22} dS_t^2$ and Equation 3.7 into Equation 3.9 verifies the proposition.

□

3.3.3 European bets

Definition 16. (European bet) A *European bet* is a bet with a value depending only on the final number of goals N_T^1, N_T^2, that is one of the form

$$X_T = \Pi\left(N_T^1, N_T^2\right) \tag{3.12}$$

where Π is a known scalar function $\mathbb{N} \times \mathbb{N} \to \mathbb{R}$.

Example 17. A typical example is a bet that pays out 1 if the home team scores more goals than the away team (home wins) and pays nothing otherwise, that is $\Pi\left(N_T^1, N_T^2\right) = \mathbf{1}\left(N_T^1 > N_T^2\right)$ where the function $\mathbf{1}(A)$ takes the value of 1 if A is true and zero otherwise. Another example is a bet that pays out 1 if the total number of goals is strictly higher than 2 and pays nothing otherwise, that is $\Pi\left(N_T^1, N_T^2\right) = \mathbf{1}\left(N_T^1 + N_T^2 > 2\right)$.

Proposition 18. *(pricing formula) The time-t value of a European bet is given by the explicit formula*

$$X_t = \sum_{n_1=N_1^t}^{\infty} \sum_{n_2=N_2^t}^{\infty} \Pi\left(n_1, n_2\right) P\left(n_1 - N_t^1, \lambda_1\left(T-t\right)\right) P\left(n_2 - N_t^2, \lambda_2\left(T-t\right)\right), \tag{3.13}$$

where $P\left(N, \Lambda\right)$ is the Poisson probability, that is $P\left(N, \Lambda\right) = \frac{e^{-\Lambda}}{N!}\Lambda^N$ if $N \geq 0$ and $P\left(N, \Lambda\right) = 0$ otherwise.

Proof. This follows directly form Proposition 12 and Definition 16.

□

As it has been shown, the price of a European bet is a function of the time t and the number of goals (N_t^1, N_t^2) and intensities (λ_1, λ_2). Therefore, from now on I will denote this function by $X_t = X_t\left(N_t^1, N_t^2\right)$ or $X_t = X_t\left(t, N_t^1, N_t^2, \lambda_1, \lambda_2\right)$, depending on whether the context requires the explicit dependence on intensities or not.

Definition 19. (greeks) The *greeks* are the values of the following forward difference operators $(\delta_1,\ \delta_2)$ and partial derivative operator applied to the bet value:

$$\begin{aligned}
\delta_1 X_t\left(N_t^1, N_t^2\right) &= X_t\left(N_t^1 + 1, N_t^2\right) - X_t\left(N_t^1, N_t^2\right) && (3.14)\\
\delta_2 X_t\left(N_t^1, N_t^2\right) &= X_t\left(N_t^1, N_t^2 + 1\right) - X_t\left(N_t^1, N_t^2\right) && (3.15)\\
\partial_t X_t\left(N_t^1, N_t^2\right) &= \lim_{dt\to 0} \frac{1}{dt}\left[X_{t+dt}\left(N_t^1, N_t^2\right) - X_t\left(N_t^1, N_t^2\right)\right] && (3.16)
\end{aligned}$$

Remark. The forward difference operators δ_1, δ_2 play the role of Delta and the partial derivative operator ∂_t plays the role of Theta in the Black-Scholes framework.

Theorem 20. *(Kolmogorov forward equation) The value of a European bet $X\left(t, N_t^1, N_t^2\right)$ with a payoff function $\Pi(N_T^1, N_T^2)$ satisfies the following Feynman-Kac representation on the time interval $t \in [0, T]$ which is also known as the Kolmogorov forward equation:*

$$\partial_t X\left(t, N_t^1, N_t^2\right) = -\lambda_1 \delta_1 X\left(t, N_t^1, N_t^2\right) - \lambda_2 \delta_2 X\left(t, N_t^1, N_t^2\right) \quad (3.17)$$

with

$$X_T\left(T, N_T^1, N_T^2\right) = \Pi\left(N_T^1, N_T^2\right).$$

Proof. The proposition can be easily verified using the closed form formula from Proposition 18. Furthermore, several proofs are available in the literature, see for example Proposition 12.6 in [26], Theorem 6.2 in [25] or Equation 13 in [24]. □

Remark 21. Equation 3.17 also has the consequence that any portfolio of European bets that changes no value if either team scores a goal (Delta-neutral) does not change value between goals either (Theta-neutral). It is noted without a proof, that this holds for all bets in general.

Corollary 22. *The value of a European bet* $X\left(t, N_t^1, N_t^2, \lambda_1, \lambda_2\right)$ *satisfies the following:*

$$\frac{\partial}{\partial \lambda_i} X_t = (T-t)\,\delta_i X_t \tag{3.18}$$

where $i \in \{1, 2\}$.

Proof. This follows directly from Proposition 18. □

Proposition 23. *(portfolio weights) The components* (ϕ_t^1, ϕ_t^2) *of the trading strategy that replicates a European bet* X_T *are equal to the forward difference operators* (δ_1, δ_2) *of the bet, formally:*

$$\phi_t^1 = \delta_1 X\left(t, N_t^1, N_t^2\right) \tag{3.19}$$

$$\phi_t^2 = \delta_2 X\left(t, N_t^1, N_t^2\right). \tag{3.20}$$

Proof. Recall that according to Proposition 13, the time-t value of a bet is equal to $X_t = X_0 + \sum_{i=1}^{2} \int_0^t \phi_s^i dS_s^i$, which after substituting $dS_t^i = dN_t^i - \lambda_i dt$

becomes

$$\begin{aligned} X_t &= X_0 + \int_0^t \left(\phi_s^1 \lambda_1 + \phi_s^2 \lambda_2\right) ds \\ &\quad + \sum_{k=0}^{N_t^1} \phi_{t_k^1}^1 + \sum_{k=0}^{N_t^2} \phi_{t_k^2}^2, \end{aligned} \tag{3.21}$$

where I used $\int_0^t \phi_s^i dN_s^i = \sum_{k=0}^{N_t^i} \phi_{t_k^1}^i$ where $0 \leq t_k^i \leq t$ is the time of the k.th jump (goal) of the process N_t^i for $i \in \{1, 2\}$.

On the other hand, using Ito's formula for jump processes (Proposition 8.15, [26]), which applies because the closed form formula in Proposition 18 is infinitely differentiable, the value of a European bet is equal to

$$\begin{aligned} X_t &= X_0 + \int_0^t \partial_s X\left(s, N_s^1, N_s^2\right) ds \\ &\quad + \sum_{k=0}^{N_t^1} \delta_1 X\left(t_k^1, N_{t_k^1-}^1, N_{t_k^1-}^2\right) + \sum_{k=0}^{N_t^2} \delta_2 X\left(t_k^2, N_{t_k^2-}^1, N_{t_k^2-}^2\right), \end{aligned} \tag{3.22}$$

where t_k^i- refers to the fact that the value of the processes is to be taken before the jump.

Because the equality between Equations 3.21 and 3.22 hold for every possible jump times, the terms behind the sums are equal which proves the proposition.

□

3.4 Valuation Formulae

This section summarises a list of analytical formulae for the values of some of the most common in-play football bets. In the first sub-section I consider European bets, while the second sub-section contains non-European bets.

Bet type	**Payoff** $\Pi\left(N_T^1, N_T^2\right)$	**Bet value** $X_t\left(N_t^1, N_t^2, \lambda_1, \lambda_2\right)$
Home win	$\mathbf{1}\left(N_T^1 > N_T^2\right)$	$\sum_{k_1>k_2} \prod_{i=1}^{2} P\left(k_i - N_t^i, \Lambda_i\right)$
Away win	$\mathbf{1}\left(N_T^1 < N_T^2\right)$	$\sum_{k_1<k_2} \prod_{i=1}^{2} P\left(k_i - N_t^i, \Lambda_i\right)$
Draw	$\mathbf{1}\left(N_T^1 = N_T^2\right)$	$\sum_{k_1=k_2} \prod_{i=1}^{2} P\left(k_i - N_t^i, \Lambda_i\right)$
Correct Score	$\mathbf{1}\left(N_T^1 = K_1, N_T^2 = K_2\right)$	$\prod_{i=1}^{2} P\left(K_i - N_t^i, \Lambda_i\right)$
Over K	$\mathbf{1}\left(N_T^1 + N_T^2 > K\right)$	$\sum_{k=K+1}^{\infty} P\left(k - N_t^1 - N_t^2, (\Lambda_1 + \Lambda_2)\right)$
Under K	$\mathbf{1}\left(N_T^1 + N_T^2 < K\right)$	$\sum_{k=0}^{K-1} P\left(k - N_t^1 - N_t^2, (\Lambda_1 + \Lambda_2)\right)$
Odd	$\mathbf{1}\left(N_T^1 + N_T^2 = 1 \mod 2\right)$	$\exp\left[-\left(\Lambda_1 + \Lambda_2\right)\right] \cosh\left[\left(\Lambda_1 + \Lambda_2\right)\right]$
Even	$\mathbf{1}\left(N_T^1 + N_T^2 = 0 \mod 2\right)$	$\exp\left[-\left(\Lambda_1 + \Lambda_2\right)\right] \sinh\left[\left(\Lambda_1 + \Lambda_2\right)\right]$
Winning Margin	$\mathbf{1}\left(N_T^1 - N_T^2 = K\right)$	$\exp\left[-\left(\Lambda_1 + \Lambda_2\right)\right] \left(\frac{\Lambda_1}{\Lambda_2}\right)^{\frac{K - N_t^1 + N_t^2}{2}} \cdot B_{\lvert K - N_t^1 + N_t^2 \rvert}\left(2\sqrt{\Lambda_1 \Lambda_2}\right)$

Table 3.1: Valuation formulae for some of the most common types of in-play football bets. $\Pi\left(N_T^1, N_T^2\right)$ denotes the payoff function, that is the value of the European bet at the end of the game. $P\left(k, \Lambda\right)$ denotes the Poisson distribution, that is $P\left(k, \Lambda\right) = \frac{1}{k!} e^{-\Lambda} \Lambda^k$ and $\Lambda_i = \lambda_i \left(T - t\right)$ with $i \in \{1, 2\}$ for the home and the away team, respectively.

3.4.1 European Bets

The value of a European bet at the end of the game only depends on the final scores. The formulae of this section follow directly from Proposition 18. Table 3.1 summarises the payoffs and the values of the most common European bets.

Home and Away win bets win is the home or the away team scores more goals than the other. The Draw bet wins if the scores of the two teams are equal. Correct Score K_1-K_2 wins if the final result is K_1-K_2. Over K and Under K bets win if the total number of goals is over or under K. Odd and Even bets win if the total number of goals is an odd or even number.

The Winning Margin K bet wins if the difference between the home and away scores is equal to K. The value of this bet follows the Skellam distribution and $B_k\left(z\right)$ is the modified Bessel function of the first kind.

3.4.2 Non-European Bets

Bets in this category have a value at the end of the game that depends not only on the final score, but also on the score before the end of the game or the order of scores. We consider two popular bets in this category: Next Goal and Half Time / Full Time bets. Valuation of these bets follows from Corollary 12.

3.4.2.1 Next Goal

The Next Goal Home bet wins if the home team scores the next goal. The value of this bet is

$$X_t = \frac{\lambda_1}{\lambda_1+\lambda_2}\left[1-e^{-(\lambda_1+\lambda_2)(T-t)}\right]. \tag{3.23}$$

Similarly, the value of the Next Goal Away bet is equal to

$$X_t = \frac{\lambda_2}{\lambda_1+\lambda_2}\left[1-e^{-(\lambda_1+\lambda_2)(T-t)}\right]. \tag{3.24}$$

3.4.2.2 Half Time / Full Time

Half Time / Full Time bets win if the half time and the full time is won by the predicted team or is a draw. Given that there are 3 outcomes in each halves, there are 9 bets in this category. For example, the value of the Half Time Home / Full Time Draw bet before the end of the first half is equal to:

$$\begin{aligned} X_t = \sum_{k_1>k_2}\sum_{l_1=l_2} & P\left(k_1-N_t^1,\lambda_1\left(T_{\frac{1}{2}}-t\right)\right)P\left(k_2-N_t^2,\lambda_2\left(T_{\frac{1}{2}}-t\right)\right) \\ & \times P\left(l_1-k_1,\lambda_1\left(T-T_{\frac{1}{2}}\right)\right)P\left(l_2-k_2,\lambda_2\left(T-T_{\frac{1}{2}}\right)\right) \end{aligned} \tag{3.25}$$

In the second half, this bet either becomes worthless if the first half was not won by the home team or otherwise becomes equal to the Draw bet.

3.5 Model Calibration

In this section I discuss how to calibrate the model on historical market prices. I show that a unique equivalent martingale measure $\mathbb{Q}$ exists, that is, a set of intensities λ_1, λ_2 exist that are consistent with the prices observed on the market (see Propositions 9 and 11).

I apply a least squares approach in which I consider market prices of a set of bets and find model intensities that deliver model prices for these bets that are as close as possible to the market prices. Specifically, I minimize the sum of the square of the weighted differences between the model and market mid prices as a function of model intensities, using market bid-ask spreads as weights. The reason for choosing a bid-ask spread weighting is that I would like to take into account bets with a lower bid-ask spread with a higher weight because the price of these bets is more certain. Formally, the following expression is minimised:

$$R\left(\lambda_t^1, \lambda_t^2\right) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left[\frac{X_t^{i,MID} - X_t^i\left(\lambda_t^1, \lambda_t^2, N_t^1, N_t^2\right)}{\frac{1}{2}\left(X_t^{i,SELL} - X_t^{i,BUY}\right)}\right]^2}, \tag{3.26}$$

where n is the total number of bets used, $X_t^{i,BUY}$ and $X_t^{i,SELL}$ are the best market buy and sell quotes of the i.th type of bet at time t, $X_t^{i,MID}$ is the market mid price which is the average of the best buy and sell quotes, $X_t^i\left(N_t^1, N_t^2, \lambda_t^1, \lambda_t^2\right)$ is the model price of the i.th bet at time t, given the current number of goals N_t^1, N_t^2 and model intensity parameters λ_t^1, λ_t^2, see

Proposition 18. This minimization procedure is referred to as model calibration.

Calibration has been performed at a time step of 1 minute during the game, independently of any previous calibrations. I used the three most liquid groups of bets which in our case were Match Odds, Over / Under and Correct Score with a total of 31 bet types in these three categories.

The continuous lines on Figures 3.2 and 3.3 show the calibrated model prices while the dashed lines are the market buy and sell offers. It can be seen that the calibrated values are close to the market quotes, although they are not always within the bid-ask spread. As the measures of the goodness of the fit I use the optimal value of the cost function of Equation 3.26, which is the average distance of the calibrated values from the market mid prices in units of bid-ask spread. Its value is shown in Figure 3.4. I performed calibration for multiple games of the Euro 2012 Championship, the time average of the calibration errors for each game is shown in Table 3.2. The mean and standard deviation of the calibration errors across games is 1.57 ± 0.27 which is to be interpreted in units of bid-ask spread because of the weighting of the error function in Equation 3.26. This means, that on average, the calibrated values are outside of the bid-ask spread, but not significantly. Given that a model of only 2 parameters has been calibrated to a total of 31 independent market quotes, this is a reasonably good result.

Finally, the implied intensities, along with the estimated uncertainties of the calibration using the bid-ask spreads are shown in Figure 3.5. Contrary to our initial assumption, under which the intensities are constant, the actual intensities fluctuate over time. There seems to be an increasing trend in the goal intensities of both teams.

In order to better understand the nature of the implied intensity process,

Game	**Model vs Mkt in units of Bid-Ask**
Denmark vs. Germany	1.65
Portugal vs. Netherlands	1.18
Spain vs. Italy	2.21
Sweden vs. England	1.58
Italy vs. Croatia	1.45
Germany vs. Italy	1.50
Germany vs. Greece	1.34
Netherlands vs. Germany	1.78
Spain vs. Rep of Ireland	1.64
Spain vs. France	1.40
Average	1.57
Standard deviation	0.27

Table 3.2: Average calibration errors as shown in Figure 3.4 have been calculated for multiple games of the UEFA Euro 2012 Championship and are shown in this table. Note that the mean of the averages is just 1.57 with a standard deviation of 0.27 which shows that the model fit is reasonably good for all the games analyzed.

I assumed the logarithm of the sum of the intensities of the two teams to follow a Wiener process of constant drift and volatility, that is:

$$d \ln \left(\lambda_t^1 + \lambda_t^2 \right) = \mu dt + \sigma dW_t \tag{3.27}$$

where μ and σ are the drift and volatility of the process. Table 3.3 shows the results of the estimation for multiple games. The mean and standard deviation of the drift terms are $\mu = 0.55 \pm 0.16 \; 1/90min$ while the mean and standard deviation of the volatility terms are $\sigma = 0.51 \pm 0.19 \; 1/\sqrt{90min}$. The fact that implied goal intensities are increasing during the game is consistent with findings of [23] who found gradual increase of scoring rates by analyzing goal times of 4012 matches between 1993 and 1996.

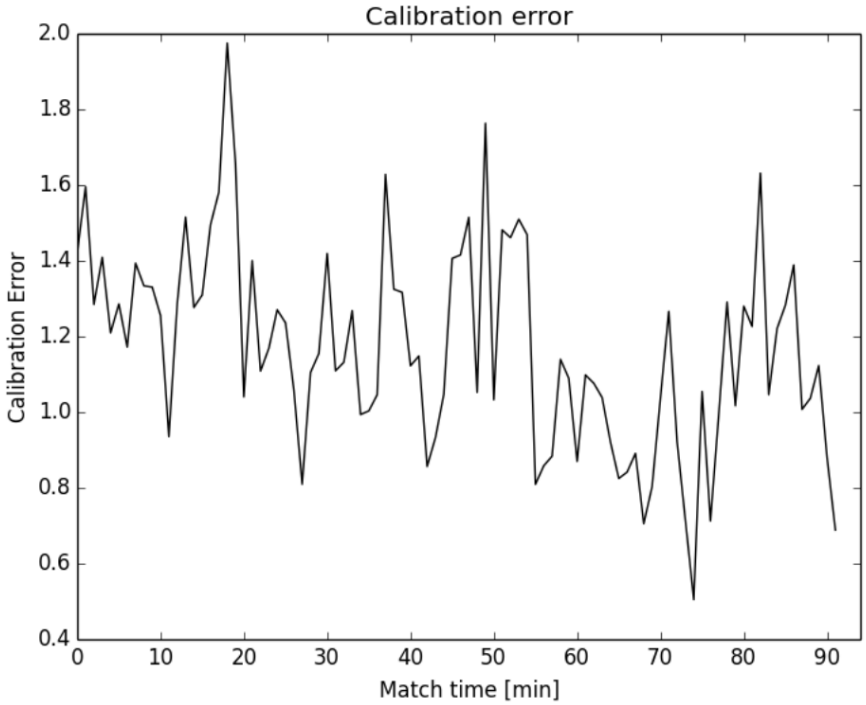


Figure 3.4: Calibration error during the game. Calibration error is defined as the average distance of all 31 calibrated bet values from the market mid prices in units of bid-ask spread. A formal definition is given by Equation 3.26. Note that the calibration error for this particular game is usually between 1 and 2 bid-ask spreads which is a reasonably good result, given that the model has only 2 free parameters to explain all 31 bet values.

3.6 Hedging

In this section I demonstrate market completeness and show that any bet can be replicated by dynamically trading in two linearly independent bets.

Recall Theorem 15, according to which any European bet X_t can be replicated by dynamically trading in two linearly independent instruments Z_t^1 and Z_t^2:

$$X_t = X_0 + \int_0^t \psi_s^1 dZ_s^1 + \int_0^t \psi_s^2 dZ_s^2 \tag{3.28}$$

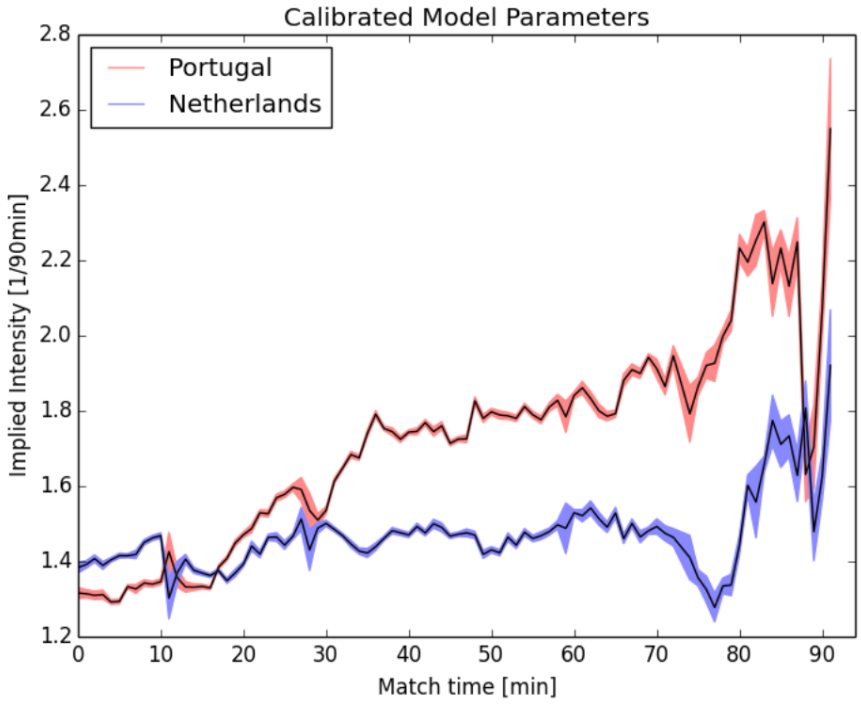


Figure 3.5: Calibrated model parameters, also referred to as implied intensities during the game. Formally, this is equal to the minimizer λ_t^1, λ_t^2 of Equation 3.26. The bands show the parameter uncertainties estimated from the bid-ask spreads of the market values of the bets. Note that the intensities appear to have an increasing trend and also fluctuate over time.

where the portfolio weights ψ_t^1, ψ_t^2 are equal to the solution of the equation:

$$\begin{pmatrix} \delta_1 Z_t^1 & \delta_1 Z_t^2 \\ \delta_2 Z_t^1 & \delta_2 Z_t^2 \end{pmatrix} \begin{pmatrix} \psi_t^1 \\ \psi_t^2 \end{pmatrix} = \begin{pmatrix} \delta_1 X_t \\ \delta_2 X_t \end{pmatrix} \tag{3.29}$$

where the values of the finite difference operators δ (Definition 19) can be computed using Proposition 18 using the calibrated model intensities. Equation 3.29 tells us that the change in the replicating portfolio must match the change of the bet value X_t in case either team scores a goal. This approach is analogous to delta hedging in the Black Scholes framework.

Game	**Drift** $[1/90min]$	**Vol** $[1/\sqrt{90min}]$
Denmark vs. Germany	0.36	0.28
Portugal vs. Netherlands	0.49	0.44
Spain vs. Italy	0.60	0.76
Sweden vs. England	0.58	0.59
Italy vs. Croatia	0.82	0.60
Germany vs. Italy	0.76	0.39
Germany vs. Greece	0.65	0.66
Netherlands vs. Germany	0.43	0.32
Spain vs. Rep of Ireland	0.32	0.78
Spain vs. France	0.48	0.25
Average	0.55	0.51
Standard deviation	0.16	0.19

Table 3.3: Average drift and volatility of total log-intensities estimated for multiple games of the UEFA Euro 2012 Championship. Note that the drift term is positive for all games which is consistent with the empirical observation of increasing goal frequencies as the game progresses.

The two bets that I use as replicating instruments are the bet for the home team winning the game (Z_t^1) and the bet for the away team winning the game (Z_t^2). The reason for this choice is that these are usually among the most liquidly traded bets with the lowest bid-ask spreads and are always linearly independent (see Definition 14).

It must be noted that if the scores become uneven and the game is close to finishing, the leading team is almost certain to win the game. In this case the values of these bets become insensitive to additional goals, because another goal does not significantly change the probabilities of winning. In this case the finite differences $\delta_i Z_t^j$, $i, j \in \{1, 2\}$ become small. If the contract X_t that is being replicated is still sensitive to additional goals, then the portfolio weights ψ_t^i can become very high which leads to instabilities in the replication. This is similar to the problem of hedging a digital option with the underlying stock when the stock price is close to strike and the option is

close to maturity. In order to avoid this problem, I perform replication only in those cases when the absolute number of any of the replicating instruments we need to hold in the portfolio is less than 50.

In case of the Correct Score 2-2 bet (X_t), the value of the bet along with the value of the replicating portfolio is shown in Figure 3.6. When computing the value of the replicating portfolios, I assumed no transaction costs, that is I assumed that the replicating instruments are traded on the market mid price. It can be seen that the value of the replicating portfolio matches the value of the original bet reasonably well. It seems that the main source of error is the difference of jump sizes at times when a goal is scored.

Figure 3.7 shows the jumps of contract values against the jumps of replicating portfolio values when a goal is scored. This plot contains all contracts and all goals of the Portugal - Netherlands game. It can be seen that the jumps of the original contract values are in line with the jumps of the replicating portfolio values with a correlation of 89%. Table 3.4 shows these correlations for multiple games. It can be seen that the correlations are reasonably high for all games with an average of 80% and a standard deviation of 19%.

3.7 Summary

In this chapter I have shown that the Fundamental Theorems of Asset Pricing apply to the market of in-play football bets if the scores are assumed to follow independent Poisson processes of constant intensities. I developed general formulae for pricing and replication. I have shown that the model of only 2 parameters calibrates to 31 different bets with an error of less than 2 bid-ask spreads. Furthermore I have shown that the model can also be used for

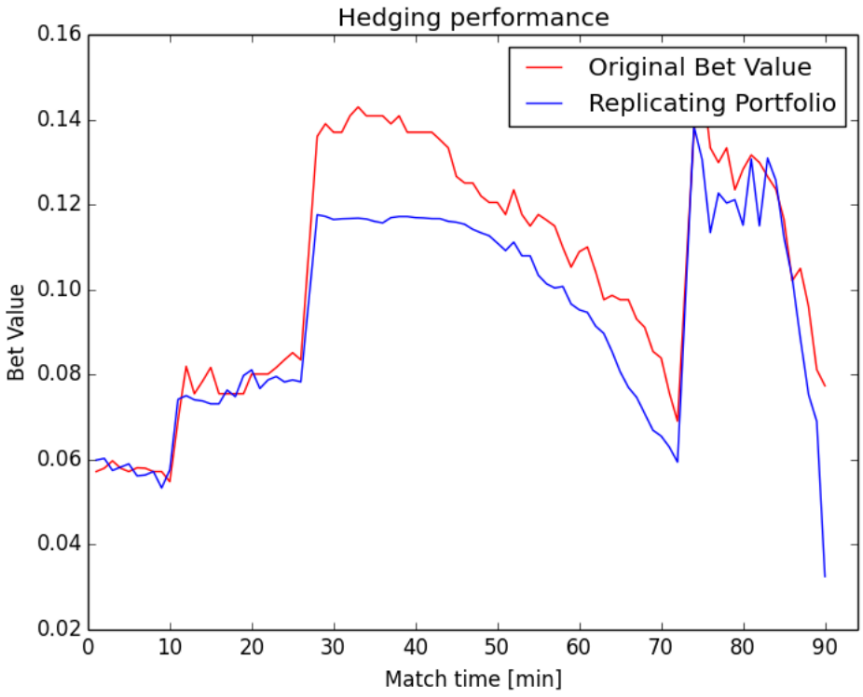


Figure 3.6: Values of the Correct Score 2-2 bet (red) and the replicating portfolio (blue). Note that since the final result of this game was 2-1, the value of this contract dropped to zero at the end of the game.

replication and hedging. Overall I obtained good agreement between actual contract values and the values of the corresponding replicating portfolios, however I point out that hedging errors can sometimes be significant due to the fact the implied intensities are in fact not constant. In order to account for this effect, two potential extensions of the model could be considered: local intensity models where intensities are deterministic functions of the state and stochastic intensity models where intensities change randomly.

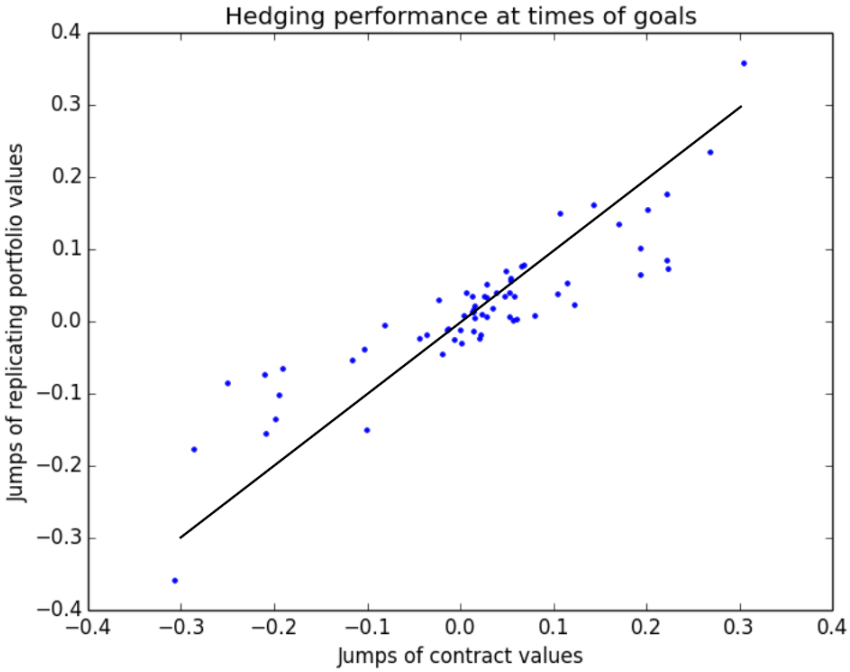


Figure 3.7: Changes of actual contract values (horizontal axis) and changes of values of the corresponding replicating portfolios (vertical axis) during times of goals. The changes are computed between the last traded price before a goal and the first traded price after a goal, for all goals. The figure contains all contracts that have been replicated. Note that the value changes of the replicating portfolios corresponds relatively well to the value changes of the original contracts because the blue points are lying close to the slope-1 black line.

Game	**Correlation**
Denmark vs. Germany	79%
Portugal vs. Netherlands	89%
Spain vs. Italy	97%
Italy vs. Croatia	47%
Spain vs. France	86%
Germany vs. Italy	99%
Germany vs. Greece	60%
Netherlands vs. Germany	93%
Spain vs. Rep of Ireland	98%
Sweden vs. England	50%
Average	80%
Standard deviation	19%

Table 3.4: Correlation between the jumps of bet values and jumps of replicating portfolios at times of goals for all bets of a game.

Chapter 4

The Local Intensity Model

In this chapter I present a model which is an extension of the Constant Intensity Model discussed in Chapter 3 by introducing intensities that depend on the number of goals and time. I show that this model is in many respects similar to the Local Volatility model which is widely used in finance for option pricing. Borrowing the terminology from finance, I am referring to the model as the Local Intensity model and to the state and time dependent intensity function as the local intensity function. The advantage of the Local Intensity model over the homogeneous Poisson model is that if the local intensity function is chosen appropriately, it can produce an arbitrary, not necessarily poissonian marginal distribution. Therefore the model can be perfectly calibrated to any empirically observed set of bet prices on the final score. I am deriving formulae that perform this perfect calibration which are similar to Dupire's formula in the Local Volatility model. Calibration on real market data is also demonstrated.

4.1 Local Intensity model in football betting

In this section I am considering the Local Intensity model for in-play football betting which is an extension to the homogeneous Poisson model by allowing for state and time dependent intensities. The necessity for the model comes from the fact that neither the historical realised goal frequencies, nor the market implied intensities are constant, the distribution of goals doesn't follow a homogeneous Poisson distribution but is more heavy-tailed. In fact, the situation is very similar to option markets where the Black-Scholes implied volatility is not constant, but depends on the option strike and therefore the distribution assumed by the market is more fat-tailed than a log-normal distribution. This becomes a practical problem when we are attempting to price an over-the-counter option or bet which is not traded on the market because it's unclear which implied volatility or intensity shall be used. The answer is that the model needs to be extended in a way such that it is consistent with all market observable prices, that is it is able to reprice the intensity or volatility smile and produces the more heavy-tailed distributions. The answer in finance was Dupire's local volatility model ([31] and [45]) which is an elegant extension of the Black-Scholes model: instead of assuming a constant volatility, the local volatility model assumes that the volatility is a function of spot and time, this function is referred to as the local volatility surface. The model became popular because Dupire derived a simple calibration formula to determine the shape of the local volatility surface from both the set of European call option prices and their smiling implied volatility surface. Using the calibrated local volatility surface in a partial differential equation (PDE) or Monte-Carlo simulation exactly reprices all European options and is widely used to price exotic options today across asset classes that are not traded on exchanges but rather directly between two parties

over the counter (OTC). Thereby the local volatility model can not only be calibrated to option prices that are consistent with the log-normal distribution of the Black-Scholes model, but to the prices observed in actual markets which are implied by more heavy-tailed distributions. In this chapter I show that the implied intensity smile also exist in football betting, both in historical and market implied goal frequencies, that is the distribution of actual scores is more heavy-tailed than the standard Poisson distribution (historical frequencies) and that the market is aware of this fact (market implied intensities). Then I introduce the local intensity model where the goal intensity is not constant, but depends on the number of goals and time. I derive the Dupire's calibration formula for football betting that delivers a local intensity surface from either bet prices or implied intensities. Finally I demonstrate that the model does correctly reprice the market intensity smile and therefore can be calibrated to more heavy-tailed distributions than the standard Poisson distribution. Throughout the chapter I remain in 1 dimension, that is I consider one team only for simplicity, but extension to two teams should be straightforward.

Formally, within the Local Intensity model, the number of goals follows the following SDE:

$$dN_t = d\mathcal{N}_t^{\lambda_{Loc}(N_t,t)}, \tag{4.1}$$

where N_t denotes the total number of goals scored by the two teams up to time t, $\mathcal{N}_t^{\lambda_{Loc}(N_t,t)}$ is a Poisson process of intensity $\lambda_{Loc(N_t,t)}$ and $\lambda_{Loc(N_t,t)}$ is the local intensity which is a function of the number of goals N_t and time t.

The local intensity function can be defined as the jump intensity of the process, conditional on the current number of goals and time:

Definition 24. (Local intensity) function is defined by:

$$\lambda_{Loc}(K,T) = \lim_{dt\to 0} \frac{1}{dt}\mathbb{P}\left[N_{t+dt} = N_t + 1 | t = T, N_t = K\right], \tag{4.2}$$

where N_t is the number of goals at time t.

The time evolution of the marginal distribution is described by the Kolmogorov forward equation which in case of the process 4.1 takes the following form:

$$\frac{\partial}{\partial T} f(K,T) = -\lambda_{Loc}(K,T) f(K,T) + \lambda_{Loc}(K-1,T) f(K-1,T), \tag{4.3}$$

where $f(K,T) = \mathbb{P}\left[N_t = K, t = T\right]$ is the marginal density function.

4.1.1 European Options vs. Football Bets

In finance, the most liquid options are European call and put options with a payoff function of $(S_T - K)^+$ at maturity, which is the reason for the Dupire formulae 2.7 and 2.8 being developed for these products. However, digital options with a payoff of $\Theta(S_T - K)$ (where $\Theta(x)$ is the Heaviside unit step function) can also be used in theory, Black-Scholes volatilities can also be implied from these products which are in general different from European implied volatilities and similar formulae can also be developed. Finally, in theory Arrow-Debreu or Dirac-like options with a payoff of $\delta(S_T - K)$ (where $\delta(x)$ is the Dirac delta) could also be used, but these products are not traded in practice.

However, the situation is somewhat different in football betting. European bets with a payoff of $(N_T - K)^+$ are not traded in practice. However, digital bets with a payoff of $\Theta(N_T - K)$ have a liquid market and are called

Over-Under bets, where "over" refers to digital calls and "under" refers to digital puts. Arrow-Debreu-like bets with a payoff of $\delta^K_{N_T}$ (where δ^j_i denotes the Kronecker delta) are also traded in practice and are called Correct Score bets. Because both of these classes of bets are traded, I am developing the Dupire formulae for both classes.

4.1.2 Over-Under bets

The most liquidly traded bets on the total score are called Over-Under bets. The payoff of these bets depends on the total number of goals N_T at a specific time of maturity T which is usually equal to either the half time or the full time. The payoff is equal to 1 (that is, the bet wins) if $N_T \leq K$ and 0 otherwise, where K is called the strike of the bet. Bets are available with a range of different strikes usually up to 8 at full time. The value of an Over-Under bet of strike K and maturity T is denoted by $F(K,T)$ and is equal to the marginal cumulative distribution function of N_T:

$$F(K,T) = \mathbb{P}[N_T \leq K] \tag{4.4}$$

The implied intensity of this bet is equal to the intensity of a Poisson distribution that has a cumulative distribution equal to the value of the bet.

Definition 25. (Implied intensity of an Over-Under) bet of strike K and maturity T is defined by $\Lambda^{OU}_{Impl}(K,T)$ that satisfies:

$$F(K,T) = e^{-\Lambda^{OU}_{Impl}(K,T)} \sum_{k=0}^{K} \frac{\left(\Lambda^{OU}_{Impl}(K,T)\right)^k}{k!} = \frac{\Gamma\left(K+1, \Lambda^{OU}_{Impl}(K,T)\right)}{K!}, \tag{4.5}$$

where $\Gamma(n,x)$ denotes the incomplete gamma function.

Dupire's calibration formulae 2.7 and 2.8 can be extended to the Local Intensity model as follows.

Proposition 26. ***(Dupire's formula for Over-Under bets)*** *The local intensity surface can be calibrated from prices of Over-Under bets using the following formula:*

$$\lambda_{Loc}\left(K,T\right)=-\frac{\frac{\partial}{\partial T}F\left(K,T\right)}{F\left(K,T\right)-F\left(K-1,T\right)} \tag{4.6}$$

Proof. The proposition can be proved by substituting Equation 4.6 directly into the Kolmogorov forward equation 4.3 and using

$$F\left(K,T\right)=\sum_{k=0}^{K}f\left(k,T\right), \tag{4.7}$$

which relates the marginal cumulative density to the distribution function.

□

Proposition 27. ***(Dupire's formula for Over-Under implied intensities)*** *The local intensity surface can be calibrated from the implied intensity surface using the following formula:*

$$\lambda_{Loc}\left(K,T\right)=-\frac{e^{-\Lambda_{Impl}^{OU}(K,T)}\left(\Lambda_{Impl}^{OU}\right)^{K}\left(K,T\right)\frac{\partial\Lambda_{Impl}^{OU}(K,T)}{\partial T}}{\Gamma\left(K+1,\Lambda_{Impl}^{OU}\left(K,T\right)\right)-K\Gamma\left(K,\Lambda_{Impl}^{OU}\left(K-1,T\right)\right)}, \tag{4.8}$$

Proof. The time derivative of the cumulative distribution in terms of implied intensity is:

$$\frac{\partial F\left(K,T\right)}{\partial T}=e^{-\Lambda_{Impl}^{OU}(K,T)}\frac{\left(\Lambda_{Impl}^{OU}\right)^{K}\left(K,T\right)}{K!}\frac{\partial\Lambda_{Impl}^{OU}\left(K,T\right)}{\partial T}. \tag{4.9}$$

Plugging this back to Equation 4.6 along with Definition 25 of implied intensity proves the formula.

□

Remark A first order expansion of the calibration formula if implied intensity changes slowly with strike can be obtained as follows:

$$\lambda_{Loc}(K,T) = \frac{\partial \Lambda^{OU}_{Impl}(K,T)}{\partial T}\left[1 + K\frac{\Lambda^{OU}_{Impl}(K-1,T) - \Lambda^{OU}_{Impl}(K,T)}{\Lambda^{OU}_{Impl}(K,T)}\right]^{-1} \tag{4.10}$$

4.1.3 Correct Score bets

Correct Score bets are also known as Arrow-Debreu securities and are also traded on the market, however the liquidity is somewhat less than the liquidity of Over-Under bets. The payout of a Correct Score bet is equal to 1 if the number of goals at a future time T is equal to the strike K and 0 otherwise.

The value of a Correct Score bet is equal to the probability $\mathbf{P}[N_T = K]$:

$$f(K,T) = \mathbb{P}[N_T = K], \tag{4.11}$$

Definition 28. (Implied intensity of a Correct Score bet) of strike K and maturity T is $\Lambda^{CS}_{Impl}(K,T)$ that satisfies

$$f(K,T) = e^{-\Lambda^{CS}_{Impl}(K,T)}\frac{\left(\Lambda^{CS}_{Impl}(K,T)\right)^K}{K!}. \tag{4.12}$$

Proposition 29. ***(Dupire's formula for Correct Score bets)*** *The local intensity surface can be calibrated from prices of Correct Score bets using the*

following formula:

$$\lambda_{Loc}\left(K,T\right)=-\frac{\frac{\partial}{\partial T}\sum_{k=0}^{K}f\left(k,T\right)}{f\left(K,T\right)} \tag{4.13}$$

Proof. This follows directly from 4.3. □

Proposition 30. ***(Dupire's formula for Correct Score implied intensities)***

The Local Intensity surface can be expressed as a function of Correct Score Implied Intensity surface as:

$$\lambda_{Loc}\left(K,T\right)=e^{\Lambda_{Impl}^{CS}(K,T)}\frac{K!}{\left(\Lambda_{Impl}^{CS}\left(K,T\right)\right)^{K}}\sum_{k=0}^{K}e^{-\Lambda_{Impl}^{CS}(k,T)}\frac{\left(\Lambda_{Impl}^{CS}\left(k,T\right)\right)^{k}}{k!}\times\left[1-\frac{k}{\Lambda_{Impl}^{CS}\left(k,T\right)}\right]\frac{\partial\Lambda_{Impl}^{CS}\left(k,T\right)}{\partial T} \tag{4.14}$$

Proof. This follows from substituting the following time derivative into 4.13:

$$\frac{\partial f\left(K,T\right)}{\partial T}=\frac{\partial\Lambda_{Impl}^{CS}\left(K,T\right)}{\partial T}\left[\frac{K}{\Lambda_{Impl}^{CS}\left(K,T\right)}-1\right]e^{-\Lambda_{Impl}^{CS}(K,T)}\frac{\left(\Lambda_{Impl}^{CS}\left(K,T\right)\right)^{K}}{K!} \tag{4.15}$$

□

4.2 Empirical demonstration

4.2.1 Historical realised intensity smile

A quick overview of historical distribution of full time total scores confirms the findings of [1, 3, 4, 49] that the distribution doesn't follow the Poisson

Total goals K	Number of games	Cumulative density $F(K,T)$	Implied intensity $\lambda_{Impl}(K,T)$
0	1962	7.70%	0.0285
1	4454	25.20%	0.0298
2	6181	49.47%	0.0300
3	5467	70.94%	0.0302
4	3844	86.04%	0.0302
5	1932	93.62%	0.0309
6	1026	97.65%	0.0309
7	365	99.08%	0.0318
8	163	99.73%	0.0317
9	49	99.92%	0.0321
10	15	99.98%	0.0325
11	3	99.99%	0.0352
12	2	100.00%	0.0364

Table 4.1: Historical distribution of full time $T = 90min$ total score in 25464 football games between 1996 and 2013. The Implied intensity $\lambda_{Impl}(K,T)$ column shows the Poisson intensities consistent with the corresponding Cumulative density $F(K,T)$. Note that implied intensities are not constant and show an increasing trend as a function of score which shows that score distribution is more heavy-tailed than the Poisson distribution.

distribution, but is more heavy tailed. Table 4.1 shows the distribution of full time $T = 90min$ total score in 25464 football games between 1996 and 2013. The cumulative density column $F(K)$ shows the ratio of games with full time total score less than or equal to the given score. Intensities $\lambda_{Impl}(K)$ have been implied using Definition 25 using a unit of [1/min]. Figure 4.1 shows the intensity values. Intensities are increasing with the number of goals which confirms that the historical distribution of full time total scores is more heavy-tailed than the Poisson distribution.

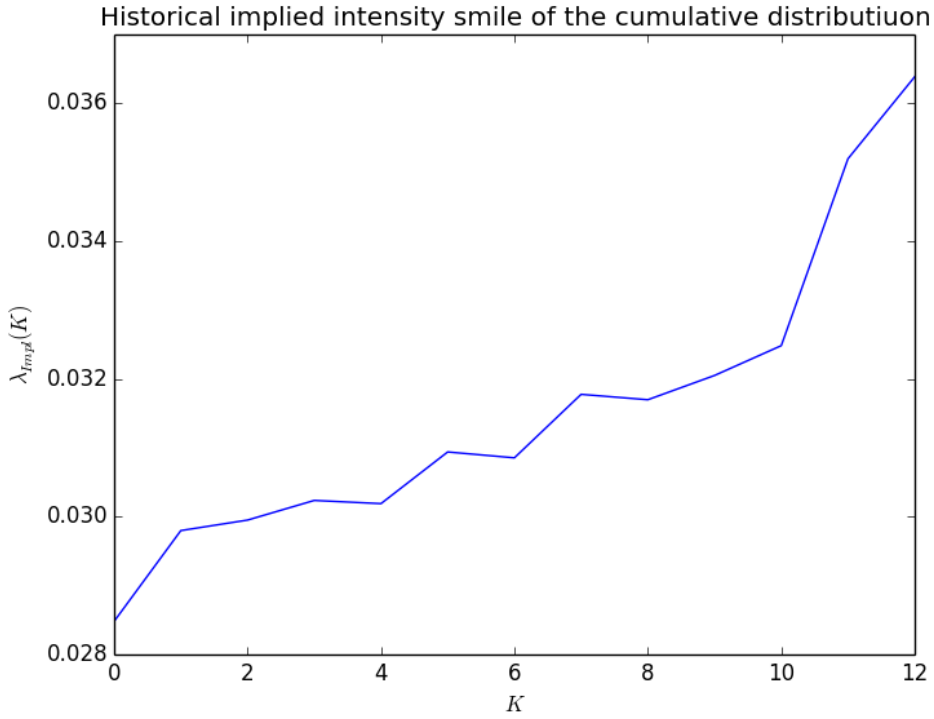


Figure 4.1: Historical implied intensities $\lambda_{Impl}(K,T)$ as a function of strike K. Intensities were implied from the historical cumulative distribution $F(K,T)$ of the full time $T = 90min$ distribution of 25464 football games between 1996 and 2013. Note that implied intensities are not constant and show an increasing trend as a function of score which shows that score distribution is more heavy-tailed than the Poisson distribution.

4.2.2 Market implied intensity smile

Betting markets seem to be aware of this fact. Table 4.2 shows in-play market prices of Over-Under bets at the 0th minute and 41st second (the time of the first market quote in the data set after the game started) of the Spain vs. Italy game of the 2012 UEFA European Championship which took place on the 10th of June 2012. The price columns show the bid and ask prices of Over-Under bets that pay out 1 if the full time score is less than or equal to the strike, that is $N_T \leq K$. The implied intensity columns show

Strike K	Bid Price $F^{Bid}(K,T)$	Ask Price $F^{Ask}(K,T)$	Bid Intensity $\lambda^{Bid}_{Impl}(K,T)$	Ask Intensity $\lambda^{Ask}_{Impl}(K,T)$
0	12.82%	15.15%	0.0220	0.0202
1	36.31%	37.04%	0.0232	0.0229
2	62.96%	63.29%	0.0233	0.0232
3	81.97%	82.76%	0.0236	0.0231
4	92.31%	92.86%	0.0240	0.0235
5	97.09%	98.04%	0.0245	0.0223
6	99.01%	99.09%	0.0249	0.0245

Table 4.2: In-play market prices of Over-Under bets at the 0th minute and 41st second of the Spain vs. Italy game of the 2012 UEFA European Championship which took place on the 10th of June 2012. The price columns show the bid and ask prices of Over-Under bets that pay out 1 if the full time score is less than or equal to the strike, that is $N_T \leq K$. The implied intensity columns show the corresponding Poisson implied intensities from both the bid and the ask quotes in units of [1/min]. Note that implied intensities are not constant, but show an increasing trend as a function of strike. This confirms that the market is pricing these bets by assuming a distribution of final scores which is more heavy-tailed than the Poisson distribution.

the corresponding Poisson implied intensities from both the bid and the ask quotes in units of [1/min]. Figure 4.2 shows the implied intensities as a function of strike. Note that implied intensities are not constant, but show an increasing trend as a function of strike. This confirms that the market is pricing these bets by assuming a distribution of final scores which is more heavy-tailed than the Poisson distribution.

In order to demonstrate the method, I calibrated a local intensity surface $\lambda_{Loc}(K,T)$ using the market implied volatility smile $\lambda_{Impl}(K,T)$ shown in Figure 4.2. Because the implied volatilities are only available at a single maturity $T = 90min$ which corresponds to the full time of the game, the first step is to interpolate the implied intensity smile to all maturities $t < T$ before the full time. For simplicity I used a constant interpolation per strike, but if for example implied intensities at half time are also available, then a

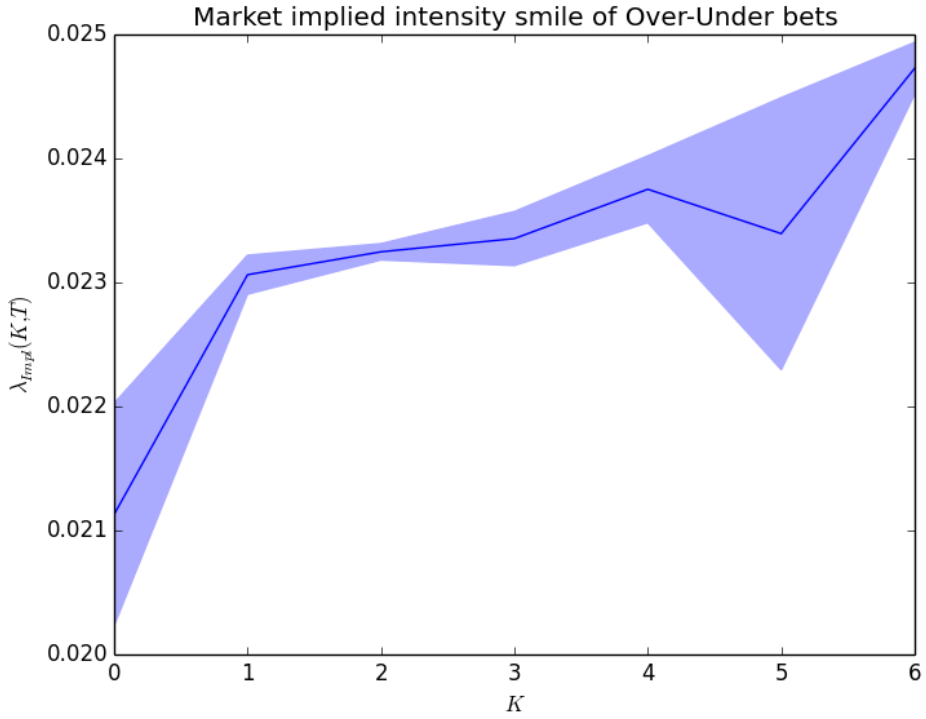


Figure 4.2: Implied intensity smile of bid and ask quotes of in-play Over-Under bets at the 0th minute and 41st second of the Spain vs. Italy game of the 2012 UEFA European Championship which took place on the 10th of June 2012. Note that implied intensities are not constant, but show an increasing trend as a function of strike. This confirms that the market is pricing these bets by assuming a distribution of final scores which is more heavy-tailed than the Poisson distribution.

linear interpolation can also be used which matches the half time implied intensities. The interpolated implied intensity surface is shown in Figure 4.3.

Using the calibration formula in Equation 4.8, the implied intensity surface can be transformed to a local intensity surface which is shown in Figure 4.3

The local intensity function shown in Figure 4.4 can be used directly in a Monte-Carlo method to generate paths according to the SDE 4.1. The generated Monte-Carlo paths are consistent with the original implied volatility

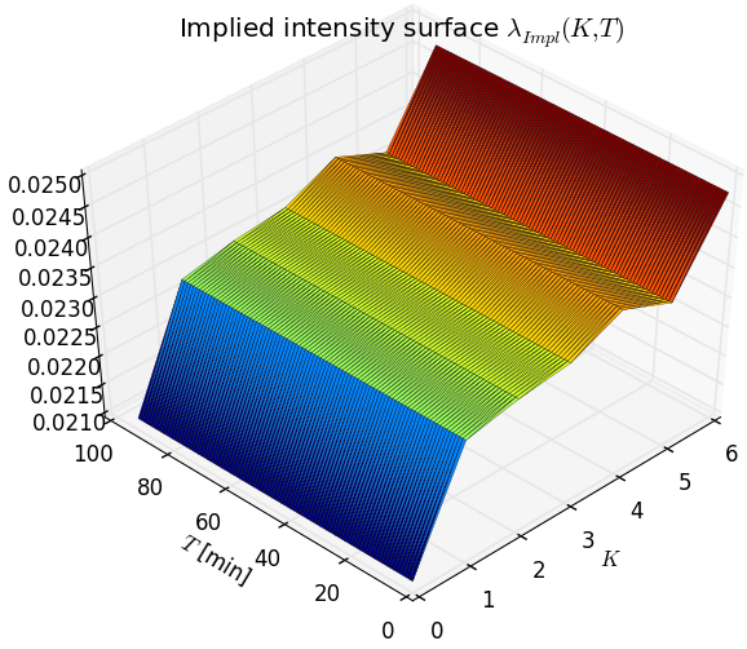


Figure 4.3: Implied intensity surface generated from the implied intensity smile shown on Figure 4.2 by applying a constant interpolation to all maturities.

surface, that is, the values of all Over-Under bets within the Monte-Carlo simulation are equal to the original values of these bets. Figure 4.5 shows the implied intensity smile of the Over-Under bets which have been priced in a Monte-Carlo simulation of 10000 paths using the calibrated local intensity surface shown in Figure 4.4. Note that the repriced implied intensities match the original implied intensities shown in Figure 4.2.

Figure 4.6 summarises the steps of the method.

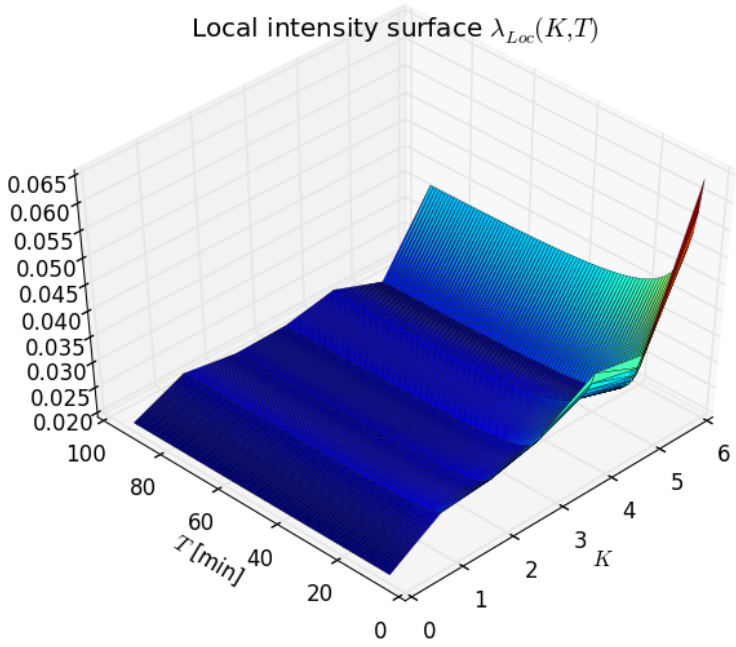


Figure 4.4: Local intensity surface generated from the implied intensity surface shown on Figure 4.3 by applying the calibration formula 4.8.

4.3 Summary

In this chapter I introduced the Local Intensity model for pricing total goal over-under bets in in-play football betting. The need for the model was motivated by the observation of the so-called ”intensity smile” effect, that is the fact that implied intensities of different strikes are non-constant which contradicts the Constant Intensity Model. This is similar to the ”volatility smile” effect in option markets. The Local Intensity model itself was motivated by Dupire’s Local Volatility model by the introduction of a local intensity surface that is the function of time and the current number of goals. Dupire’s calibration formulae have been extended to the Local Intensity model and it has been demonstrated that the model can be perfectly calibrated to an

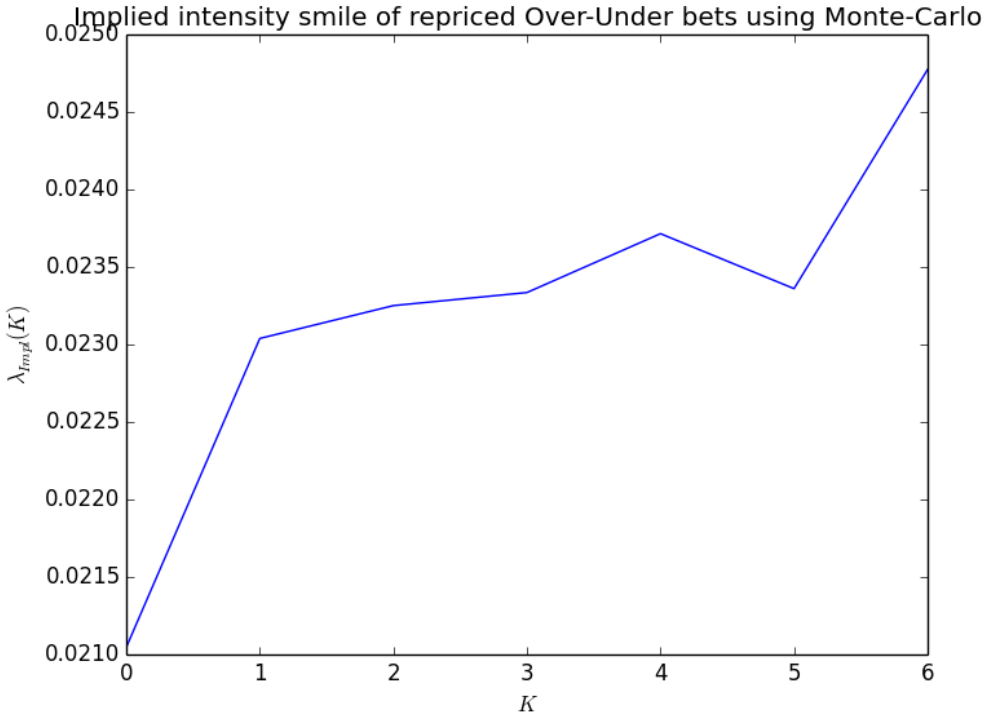


Figure 4.5: Implied intensity smile of the Over-Under bets within the Monte-Carlo simulation of 10000 paths using the calibrated local intensity surface shown in Figure 4.4. Note that the repriced implied intensities match the original implied intensities shown in Figure 4.2.

arbitrary intensity smile.

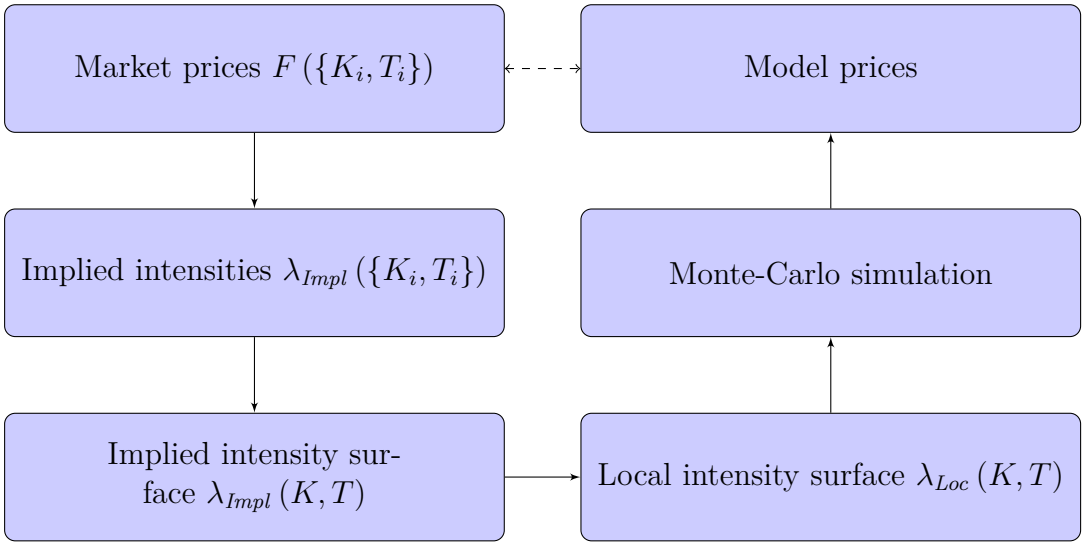


Figure 4.6: Summary of the steps involved in calibrating the Local Intensity model. Market prices of Over-Under bets $F(\{K_i, T_i\})$ are first transformed to implied intensities $\lambda_{Impl}(\{K_i, T_i\})$ which are then interpolated into an implied intensity surface $\lambda_{Impl}(K, T)$. Local intensity surface $\lambda_{Loc}(K, T)$ is constructed using the calibration formula 4.8 which in turn can be used to run a Monte-Carlo simulation to price an arbitrary bet. The prices computed in this way are going to be consistent with the original prices of Over-Under bets.

Chapter 5

Microscopic Model of Football

I propose a Microscopic Model to describe the evolution of a football game. The model is an extension of the Local Intensity Model by taking into account the position of the ball and the team holding the ball besides the number of goals scored and time. The model describes the evolution of these state parameters through a drift and diffusion coefficient for the ball position, a ball losing intensity for the team holding the ball and a usual goal intensity for the goals scored. I found that the initial ball position and the team holding the ball has an effect on goal scoring intensities for about the first 30 seconds, beyond the initial state becomes less relevant and scoring intensities approach their stationary values.

5.1 Model definition

I propose a model for the microscopic dynamics of a football game. Formally, the state of the game within the model is characterized by three random variables: the position of the ball on the field, the team holding the ball and the goals scored by the two teams. These quantities are assumed to

be interdependent. I use standard stochastic processes to build the model. Therefore, the position of the ball is driven by a 2-dimensional Brownian motion characterized by a drift and volatility parameter. The two teams are assumed to loose the control of the ball according to a Poisson point process of a given intensity parameter. Finally, the goals scored by the two teams are driven by two Poisson processes, one for each team with it's own intensity. These driving Brownian and Poisson processes are assumed to be independent, however, all of their parameters are assumed to be dependent on the current state and time, this dependence links the three processes.

5.1.1 Formal Model Definition

Definition 31. Within the model, the state variables evolve according to the following stochastic difference equations:

$$dX_t = \mu dt + \sigma d\mathcal{W}_t \tag{5.1}$$

$$dU_t = -2U_t d\mathcal{N}_t^{\nu} \tag{5.2}$$

$$dN_t^u = d\mathcal{N}_t^{\lambda_u}, \tag{5.3}$$

where t denotes the time elapsed since the beginning of the game.

The state is described by three variables: X_t, U_t and N_t^u. X_t denotes the 2-dimensional coordinates of the ball at time t, with $X_t \in [(0,0),(100,100)]$, where the line $(0,0)-(0,100)$ is the home team's goal line and $(100,0)-(100,100)$ is the away team's goal line. $U_t \in \{-1,1\}$ denotes the team owning the ball at time t with 1 referring to the home team and -1 referring to the away team, however depending on context sometimes I explicitly refer the two teams by *HOME* and *AWAY*. N_t^u denotes the total number of goals

awarded to team u at time t.

The driving stochastic processes $d\mathcal{W}_t$, $d\mathcal{N}_t^{\nu}$ and $d\mathcal{N}_t^{\lambda_u}$ are all independent of each other. $d\mathcal{W}_t$ is a standard 2-dimensional Brownian process, $d\mathcal{N}_t^{\nu}$ is a standard Poisson process of intensity ν and $d\mathcal{N}_t^{\lambda_u}$ are two standard Poisson processes of intensity λ_u for $u \in \{-1, 1\}$.

The model parameters are μ, σ, ν, λ_u, where μ is the 2-dimensional drift vector of the ball, σ is the 2-by-2 upper triangular volatility matrix of the ball, ν is the intensity of losing the ball to the opponent and λ_u are the two team's goal scoring intensities. In the general case, all of the model parameters are functions of all of the state and time, that is (X_t, U_t, N_t^u, t). This introduces the dependency between the state parameters.

The initial conditions intuitively follow from the rules of the football game: X_0 is equal to the middle of the field which is $(50, 50)$, U_0 is a Bernoulli random variable with equal probabilities for the two teams and N_0^u are both equal to zero. The boundary conditions for the ball position X_t are reflective at the boundaries of the field.

5.1.2 Effects not accounted for by the Microscopic Model

There are a few effects that are not considered by this model. First, in a real football game, the ball is passed between several players, this model doesn't take the players into account, it only models the actual position of the ball and assumes it to be a continuous diffusion process. Second, a real football game is interrupted from time to time for several reasons, such as the ball going out of the field for a throw-in, corner kick, injury, penalty cards and so on; none of these events are modelled and the game is assumed to progress continuously. Third, goals scored during free kicks and penalty kicks are modelled the same way as goals scored during regular game play.

t	N_t^{AWAY}	N_t^{HOME}	$X_{x,t}$	$X_{y,t}$	U_t
13:9	0	0	66.9	37.7	HOME
13:10	0	0	70.5	52.2	HOME
13:11	0	0	69.7	35.8	HOME
13:13	0	0	72.8	8.1	HOME
13:16	0	0	67.2	36.7	HOME
13:18	0	0	71.2	39	HOME
13:21	0	0	98.2	37.3	HOME
13:22	0	0	92.3	47.2	HOME
14:20	0	1	50	49.7	AWAY
14:21	0	1	47	49.1	AWAY
14:23	0	1	65	65.2	AWAY
14:25	0	1	69.2	37.4	AWAY
14:27	0	1	61.7	7.9	AWAY
14:30	0	1	58.1	18.8	AWAY

Table 5.1: Example time series of the state variables t, X_t, U_t and N_t for the Spain vs. Italy game of the 2012 UEFA European Championship.

5.2 Estimation of model parameters

5.2.1 Historical football game data

Our data set consists of the state variables U, X and N recorded at irregularly spaced discrete times t for all of the 31 games of the 2012 UEFA European Championship. An example of the data set is shown in Table 5.1 which reports the time around the first goal of the Spain vs. Italy game. The state variables are recorded at a rate not more frequent than a second.

5.2.2 Estimation of model parameters from historical data

In order to estimate the model parameter surfaces $\mu, \sigma, \nu, \lambda^u$ that are each functions of the state parameters (X, U, N, t), I am discretising the state space, assuming that the parameter surfaces are constant within the discrete

bins and estimating the values of the parameters in the bins.

To discretize the position X, I split the football field by a uniform 10-by-10 grid to regions of equal size. Similarly, in order to discretize time t, I split each game in 4 equal time intervals, such that both the first and the second half consists of two equal intervals. The scores N and the team U don't need to be discretised because they are already discrete variables. In this way I generated a total of 5167 discrete states for every possible state vector that appears in any of the 31 games.

Because the parameters are assumed to be constant within the bins, Equations 5.1, 5.2 and 5.3 become independent. The maximum likelihood estimates of the parameters in a single bin are:

$$\mu = \frac{\sum_{i=1}^{n} \Delta X_{t_i}}{\sum_{i=1}^{n} \Delta t_i} \tag{5.4}$$

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} \Delta X_{t_i} \Delta X_{t_i}^T \Delta t_i - \mu\mu^T \frac{1}{n}\sum_{i=1}^{n} \Delta t_i \tag{5.5}$$

$$\nu = \frac{\sum_{i=1}^{n} \Delta U_i}{\sum_{i=1}^{n} \Delta t_i} \tag{5.6}$$

$$\lambda^u = \frac{\sum_{i=1}^{n} \Delta N_i^u}{\sum_{i=1}^{n} \Delta t_i} \tag{5.7}$$

where the changes of state are simply the difference between two consecutive states:

$$(\Delta X, \Delta U, \Delta N, \Delta t)_i = (X, U, N, t)_{i+1} - (X, U, N, t)_i \,. \tag{5.8}$$

Using these equations the parameters can be estimated in all of the discrete state bins using the recorded state variables from all of the 31 games. Because I did not differentiate between games and used all of the available

data, this resulted in the parameter values for an "average" game.

5.2.3 Dependencies of model parameters on state variables

A priori I assumed that the model parameters depend on every state parameter. However, in order to understand whether this is really the case or whether some parameters don't depend strongly on some state variables which might potentially allow for simplifications in the model, I show the estimated parameter values as a function of each state variable in the following subsections. The first subsection shows the dependency on ball coordinate X, the second subsection shows the dependency on game time t, the third subsection shows the dependency on total goals $N^{HOME} + N^{AWAY}$ and the fourth subsection shows the dependency on goal difference $N^{HOME} - N^{AWAY}$. The dependency on the team U is shown explicitly because the parameters are shows for both teams in each subsection.

5.2.3.1 Model parameters as a function of ball coordinate

This subsection shows the dependency of the model parameters with respect to the ball coordinate X and the team holding the ball U.

Figures 5.1, 5.2, 5.3 and 5.4 show the calibrated parameter surfaces as a function of ball coordinate X and the team holding the ball U. The HOME team's gate is always on the left of the figures, that is x = 0 and the AWAY team's gate is at the right of the figures, that is x = 100. Dependencies on the number of goals N^u and time t have been averaged on these figures.

Figure 5.1 shows the x and y component of the ball drift parameter μ as a function of ball position X and the team holding the ball U. It can be seen that μ_x is positive when the home team holds the ball and it is negative when

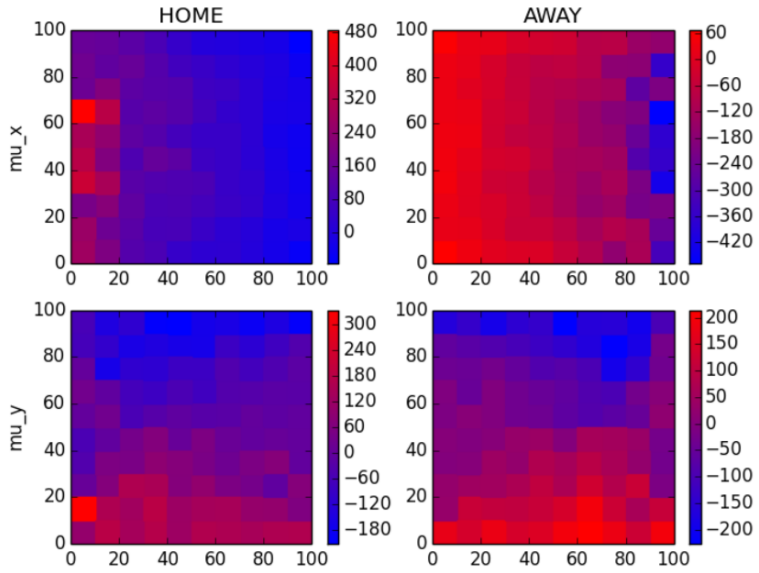


Figure 5.1: Ball drift μ_x and μ_y as a function of ball coordinate X and team holding the ball U. The units are in [1% of the length of the football field] per minute. Therefore a value of 100 means one football field length per minute.

the away team hold the ball which corresponds to the fact that each team tries to bring the ball closest to the opponent's gate. It can also be seen that the absolute value of mu_x decreases as the ball gets closer to the opponent's gate which can be explained by the opponent slowing down the ball as it gets closest to it's gate. The y component of the ball drift parameter μ_y is such that the ball tries to stay at the middle of the field, regardless of which team has the ball.

Figure 5.2 shows the xx, xy and yy components of the square of the ball volatility parameter σ, that is the components of the variance matrix as a function of ball position X and the team holding the ball U. It can be seen that the xx components σ_x is higher towards the team's own gate and decreases gradually as the ball gets closer to the opponent's gate. The ball gets "frozen" in a way as it hits the defense. The yy term σ_y is more or less

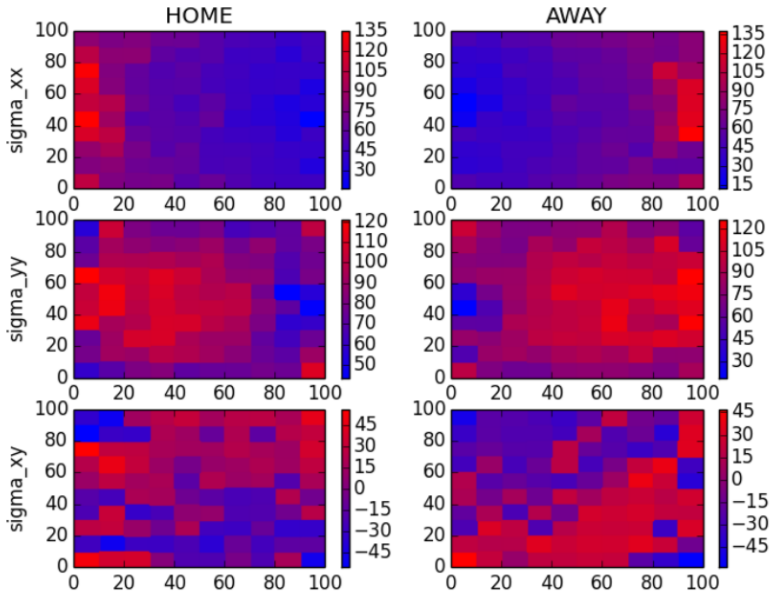


Figure 5.2: Components of the ball volatility matrix σ_{xx}, σ_{yy} and σ_{xy} as a function of ball coordinate X and team holding the ball U. The units are in [1% of the length of the football field] per $\sqrt{minute}$. Therefore a value of 100 corresponds to a standard deviation of one football field over the course of one minute.

constant, however it also decreases somewhat towards the opponent's gate. The cross term σ_{xy} is close to zero.

Figure 5.3 shows the ball losing intensity ν as a function of ball position X and the team holding the ball U. It shows the counter-intuitive result that a team is more likely to loose the ball when it's closer to it's own gate.

Figure 5.4 show the goal scoring intensity λ^u as a function of ball position X and the team holding the ball U. The top left figure shows the intensity for the home team when it has the ball. It can be seen that intensity is zero almost everywhere, except near to the away team's gate. The same can be seen on the bottom right figure that shows the goal scoring intensity for the away team when it has the ball. The top right and bottom left figures show the intensities of own goals, it can be seen that the values are significantly

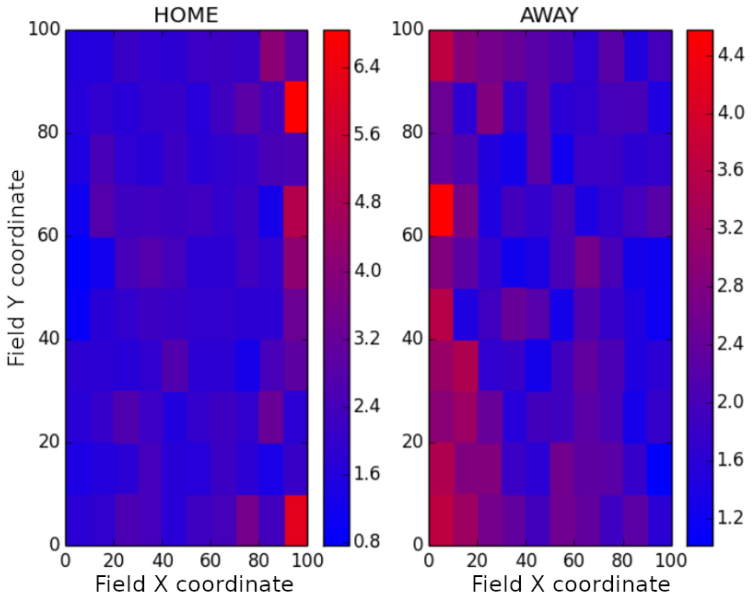


Figure 5.3: Ball losing intensity ν as a function of ball coordinate X and team holding the ball U. The units are in 1 per minute. Therefore a value of 2 means losing the ball twice a minute on average.

lower, but still non-zero.

Figure 5.5 shows the ball drift μ as arrows and the volatility σ as ellipses. It can be seen that each team attempts to push the ball towards the opponent along the x axis and towards the center of the field along the y. The drift along the x axis towards the opponent's gate becomes weaker as the ball gets closer to the gate while the push towards the center along the y axis is roughly the same regardless of the distance from the gate. The ball volatilities don't seem to depend heavily neither on the position, nor the direction or the team.

Figure 5.6 shows the goal scoring intensity λ as a function of ball position on a Voronoi plot. Each registered ball position point is considered from the data set and the set of points is split into two subsets: those where a goal happened within 30 seconds and those where a goal didn't happen. As expected, most of the time no goal happened, therefore there are more points

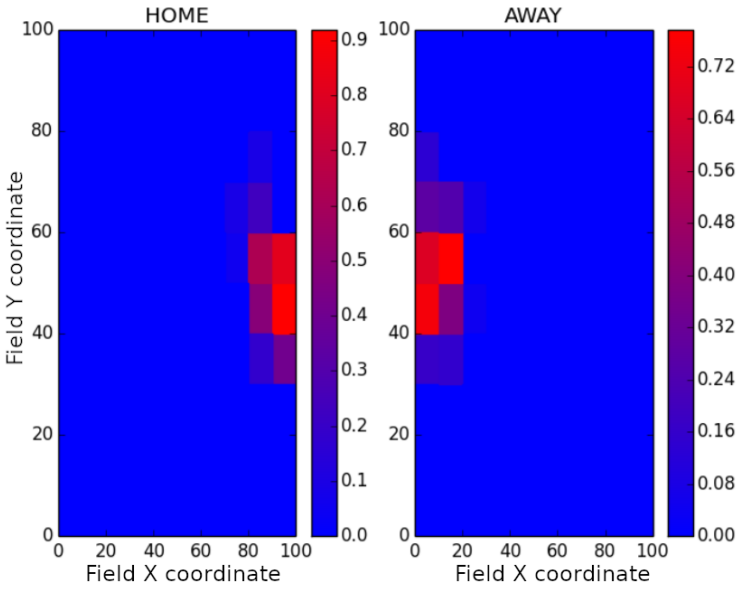


Figure 5.4: Goal scoring intensities λ^{HOME} and λ^{AWAY} as a function of ball coordinate X and team holding the ball U. The units are in 1 per minute. Therefore a value of 0.1 means scoring a goal every 10 minutes on average. Note that around the gates the goal scoring intensity reaches about 1 per minute, which means that on average one goal happens every minute, if the ball is close to the gate which is much higher than the overall goal scoring intensity of a game, but most of the time the ball is not close to the gate.

in the set with no goals. Using the set of points where a goal did happen, a Voronoi tesselation has been constructed which constitutes the cells shown in the image, with each cell belonging to one point that resulted in a goal within 30 seconds. Then, the points that are in the same cell but haven't resulted in a goal have been counted for each cell. Given the fact that now we are left with cells that each contain one point that resulted in a goal and a variable number of no-goal points, we can assume that the probability of a goal from a point within a given cell is equal to the reciprocal of the number of points within the cell. Finally, this probability has been divided by the 30 seconds time frame to come up with goal intensities for each cell.

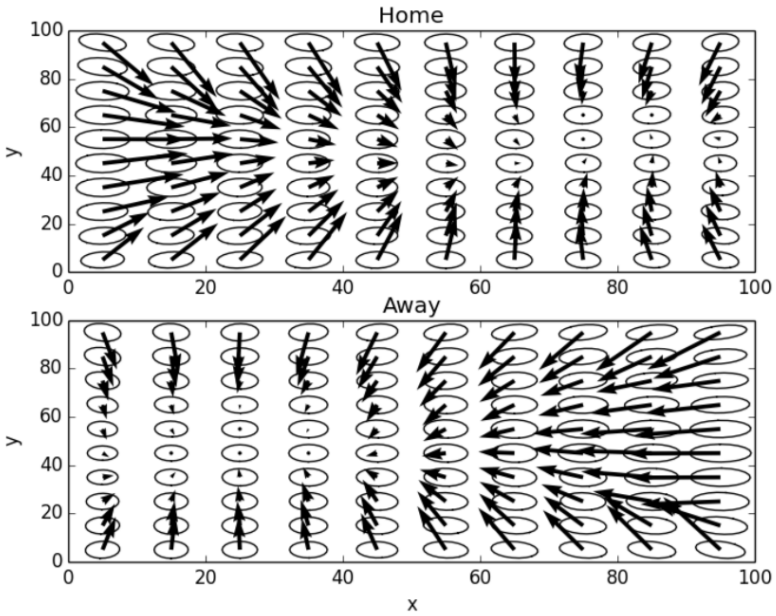


Figure 5.5: The arrows represent the drift parameter μ where the ellipses represent the volatility parameter σ as a function of ball coordinate X and team holding the ball U.

As expected, scoring intensities are high around the opponent's gate and are negligible elsewhere.

5.2.3.2 Model parameters as a function of game time

This subsection shows the dependency of the model parameters with respect to the game time t and the team holding the ball U.

Figures 5.7, 5.8, 5.9 and 5.10 show the parameter surfaces as a function of game time t and the team holding the ball U. Dependencies on ball position X and the number of goals N have been averaged on these figures.

Figures 5.7 shows the x and y component of the ball drift parameter μ as a function of time t and the team holding the ball U. The away team's drift in the x direction decreases by around 10% during the game which can be explained by the fact that players get more tired towards the end of the

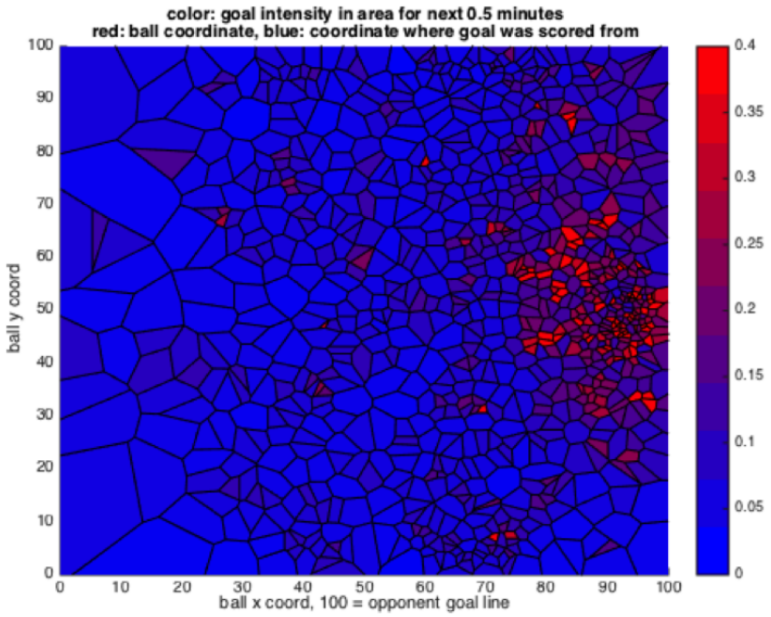


Figure 5.6: Goal scoring intensities of the HOME team λ^{HOME} as a function of ball coordinate X. This image has been constructed by considering each registered ball position in the data set. These set of ball position points have been split into two subsets: those where a goal happened within 30 seconds (A) and those where a goal didn't happen (B). Using the subset of points where a goal did happen (A), a Voronoi tesselation has been constructed, these cells are shown in the image. By the definition of Voronoi tesselation, each of the cells contains exactly one points from subset A, that is from which a goal happened within 30 seconds. These Voronoi cells also contain a variable number of points from subset B, that is points where a goal didn't happen within 30 seconds. The reciprocal of the total number of points within a cell can be assumed to be proportional to the goal scoring probability from within a Voronoi cell because is the average number of goals per point. This has been used to determine the color scale of each cell.

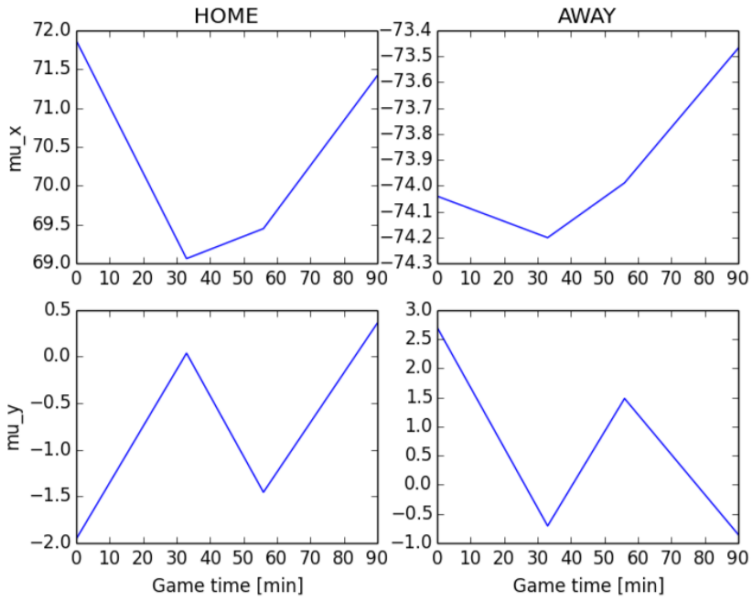


Figure 5.7: x and y components of the drift parameter μ as a function of game time t and team holding the ball U. The units are in [1% of the length of the football field] per minute. Therefore a value of 100 means one football field length per minute.

game. However, the home team's drift in the x direction doesn't change significantly. Drifts in the y direction are not significantly different than zero.

Figure 5.8 shows the xx, xy and yy components of the square of the ball volatility parameter σ, that is the components of the variance matrix as a function of time t and the team holding the ball U. Volatilities don't change significantly during the game, but there is a slight increase in the xx component for both teams.

Figure 5.9 shows the ball losing intensity ν as a function of time t and the team holding the ball U. Ball losing intensities are again relatively stable as a function of game time with a slight decrease for both teams as time goes by.

Figure 5.10 show the goal scoring intensity λ^u as a function of time t

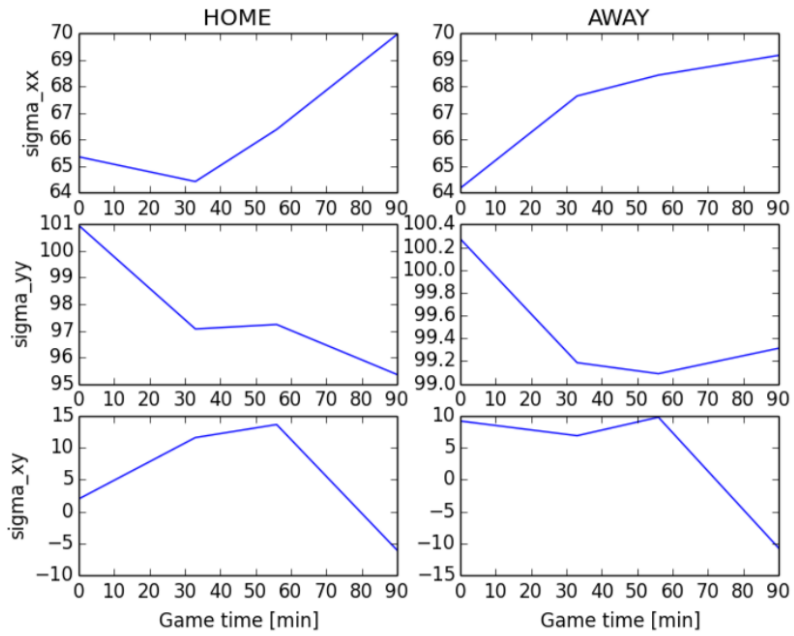


Figure 5.8: Components of the ball variance matrix σ_{xx}^2, σ_{yy}^2 and σ_{xy}^2 as a function of game time t and team holding the ball U. The units are in [1% of the length of the football field]^2 per minute. Therefore a value of 10000 corresponds to a standard deviation of one football field over the course of one minute.

and the team holding the ball U. This parameter has the strongest time dependence with the goal intensity of both teams steadily increasing during the game by about 50% for the home team and about 100% for the away team. The trends for own goals cannot be reliably determined from the data.

5.2.3.3 Model parameters as a function of total goals

This subsection shows the dependency of the model parameters with respect to the total goals scored by the two teams $N^{HOME} + N^{AWAY}$ and the team holding the ball U.

Figures 5.11, 5.12, 5.13 and 5.14 show the parameter surfaces as a function of total goals scored by the two teams $N^{HOME} + N^{AWAY}$ and the team holding

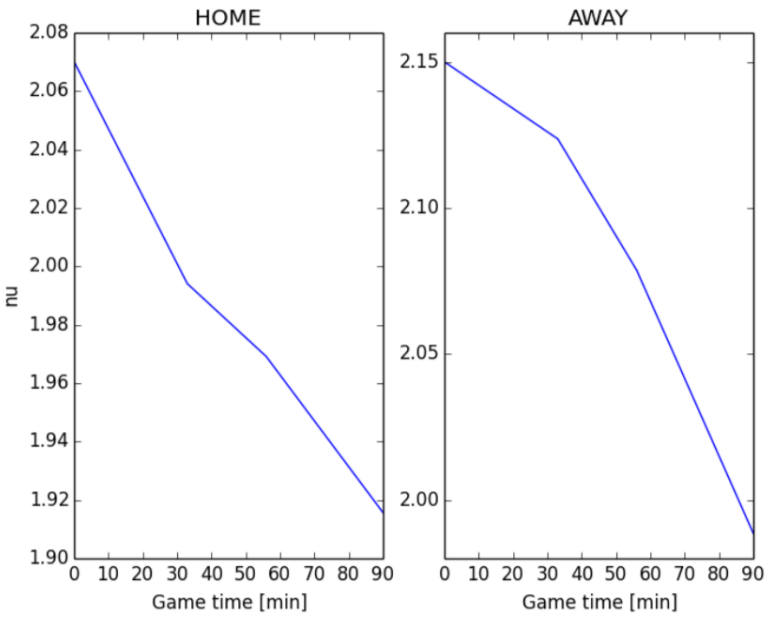


Figure 5.9: Ball losing intensity ν as a function of game time t and team holding the ball U. The units are in 1 per minute. Therefore a value of 2 means losing the ball twice a minute on average.

the ball U. Dependencies on ball position X and time t have been averaged on these figures.

Figures 5.11 shows the x and y component of the ball drift parameter μ as a function of total goals $N^{HOME} + N^{AWAY}$ and the team holding the ball U. The results are similar as shown on Figure 5.7: the away team's drift in the x direction decreases during the game while the y component is close to zero and the home team's drift is stable.

Figure 5.12 shows the xx, xy and yy components of the square of the ball volatility parameter σ, that is the components of the variance matrix as a function of total goals $N^{HOME} + N^{AWAY}$ and the team holding the ball U. Volatilities don't change significantly with the number of total goals.

Figure 5.13 shows the ball losing intensity ν as a function of total goals $N^{HOME} + N^{AWAY}$ and the team holding the ball U. Ball losing intensities

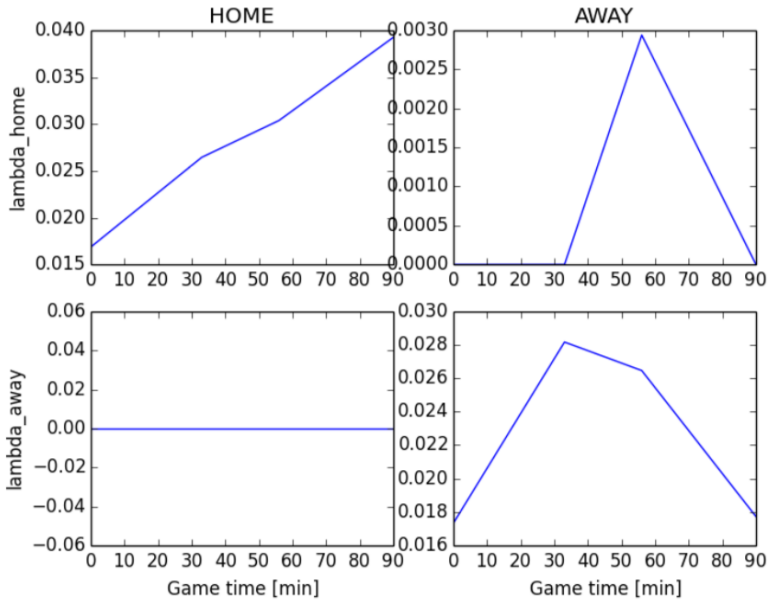


Figure 5.10: Goal scoring intensities λ^{HOME} and λ^{AWAY} as a function of game time t and team holding the ball U. The units are in 1 per minute. Therefore a value of 0.1 means scoring a goal every 10 minutes on average.

tend to decrease somewhat as the number of total goals increase.

Figure 5.14 show the goal scoring intensity λ^u as a function of total goals $N^{HOME} + N^{AWAY}$ and the team holding the ball U. The home team's goal scoring intensities tend to be unchanged as the number of goals increase, but the away team's goal scoring intensities are increasing.

The results are similar to the dependencies on game time shown on Figures 5.7, 5.8, 5.9 and 5.10 which can be explained by the fact that the number of goals tend to increase as time goes by, therefore these two parameters have a string positive correlation.

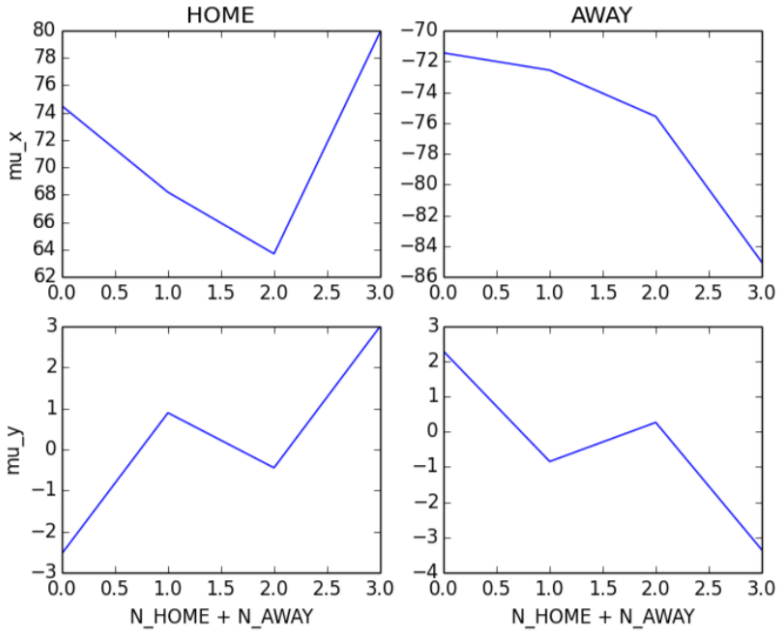


Figure 5.11: x and y components of the drift parameter μ as a function of total goals scored by both teams $N^{HOME}+N^{AWAY}$ and team holding the ball U. The units are in [1% of the length of the football field] per minute. Therefore a value of 100 means one football field length per minute.

5.2.3.4 Model parameters as a function of goal difference

This subsection shows the dependency of the model parameters with respect to the total goals scored by the two teams $N^{HOME} - N^{AWAY}$ and the team holding the ball U.

Figures 5.18, 5.16, 5.17 and 5.15 show the parameter surfaces as a function of goal difference $N^{HOME} - N^{AWAY}$ and the team holding the ball U. Dependencies on ball position X and time t have been averaged on these figures.

Figures 5.18 shows the x and y component of the ball drift parameter μ as a function of goal difference $N^{HOME} - N^{AWAY}$ and the team holding the ball U. The x components of the drift depend on the goal difference such

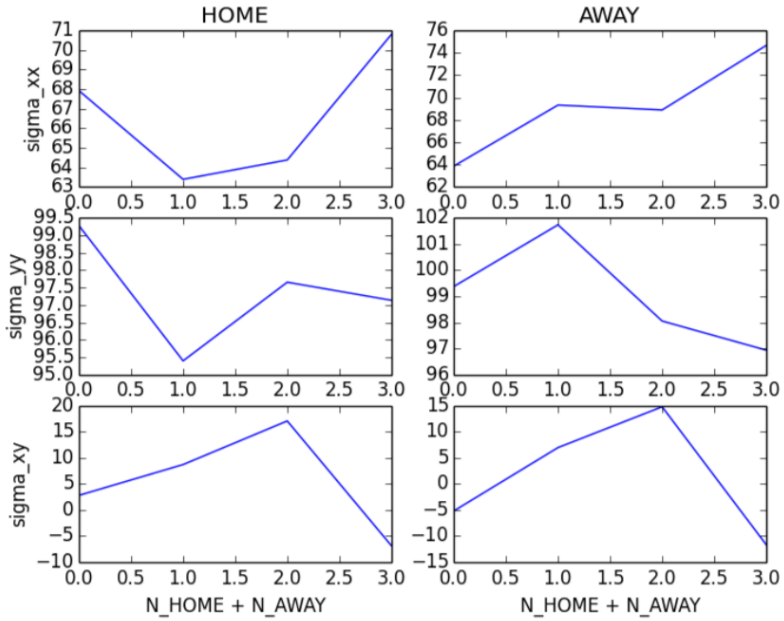


Figure 5.12: Components of the ball variance matrix σ^2_{xx}, σ^2_{yy} and σ^2_{xy} as a function of total goals scored by both teams $N^{HOME} + N^{AWAY}$ and team holding the ball U. The units are in [1% of the length of the football field]^2 per minute. Therefore a value of 10000 corresponds to a standard deviation of one football field over the course of one minute.

that both teams tend to push the ball faster towards the opponent's gate when they are leading. The y components are not affected significantly.

Figure 5.16 shows the xx, xy and yy components of the square of the ball volatility parameter σ, that is the components of the variance matrix as a function of goal difference $N^{HOME} - N^{AWAY}$ and the team holding the ball U. The volatility in the x direction tends to increase for a team if that team is leading and decrease otherwise. However, the volatility in the y direction behaves in the opposite way: it tends to decrease when a team is leading. The cross term xy is unaffected.

Figure 5.17 shows the ball losing intensity ν as a function of goal difference $N^{HOME} - N^{AWAY}$ and the team holding the ball U. Ball losing intensities

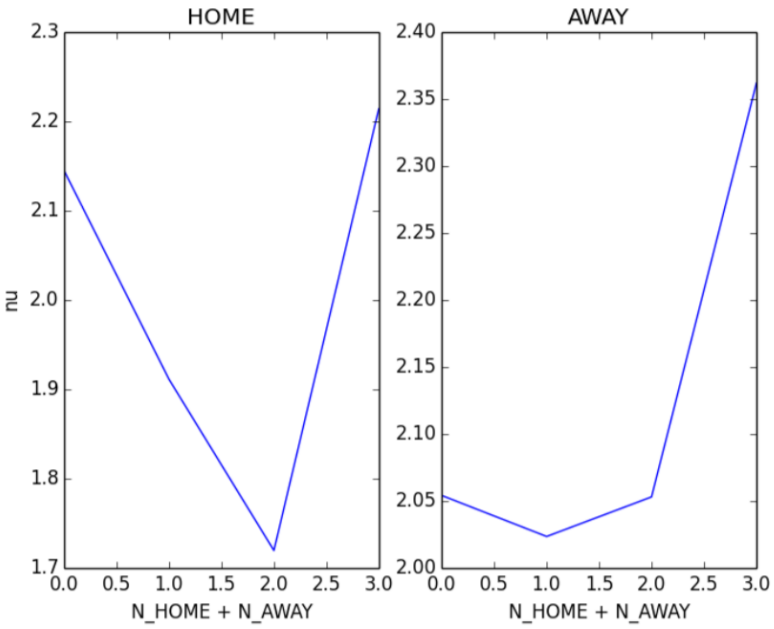


Figure 5.13: Ball losing intensity ν as a function of total goals scored by both teams $N^{HOME} + N^{AWAY}$ and team holding the ball U. The units are in 1 per minute. Therefore a value of 2 means losing the ball twice a minute on average.

tend to increase for the leading team.

Figure 5.15 show the goal scoring intensity λ^u as a function of goal difference $N^{HOME} - N^{AWAY}$ and the team holding the ball U. Goal scoring intensities tend to increase for the leading team.

5.2.3.5 Summary of the model parameter dependencies on state parameters

Table 5.2 shows a summary of the model parameter dependencies on the state parameters. Empty cells mean no significant dependency. The ball drift in the x direction μ_x is such that it pushes the ball towards the opponent's gate and is stronger when a team is leading. The ball drift in the y direction μ_y is such that it pushes the ball towards the middle of the field. The ball volatility

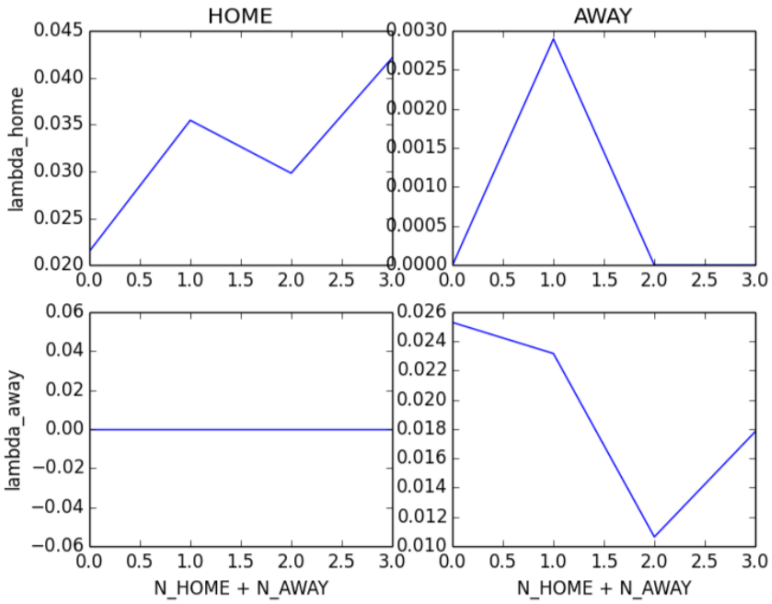


Figure 5.14: Goal scoring intensities λ^{HOME} and λ^{AWAY} as a function of total goals scored by both teams $N^{HOME} + N^{AWAY}$ and team holding the ball U. The units are in 1 per minute. Therefore a value of 0.1 means scoring a goal every 10 minutes on average.

in the x direction σ_{xx} decreases towards the opponent's gate and increases if the team is leading. The ball volatility in the y direction σ_{yy} slightly decreases if a team is leading. The ball volatility in the cross direction σ_{xy} is not significant. The ball losing intensity ν increases towards the opponent's gate and increases if a team is leading. The goal scoring intensities are nonzero around the opponent's gate, increase if a team is leading and also increase with time and with the total number of goals.

5.2.4 Time scaling properties of ball distance

In order to assess whether the calibrated model is in line with the statistical features of the data set, I calculated the time scaling properties of the ball distance empirically using the raw data set and also within the model, using

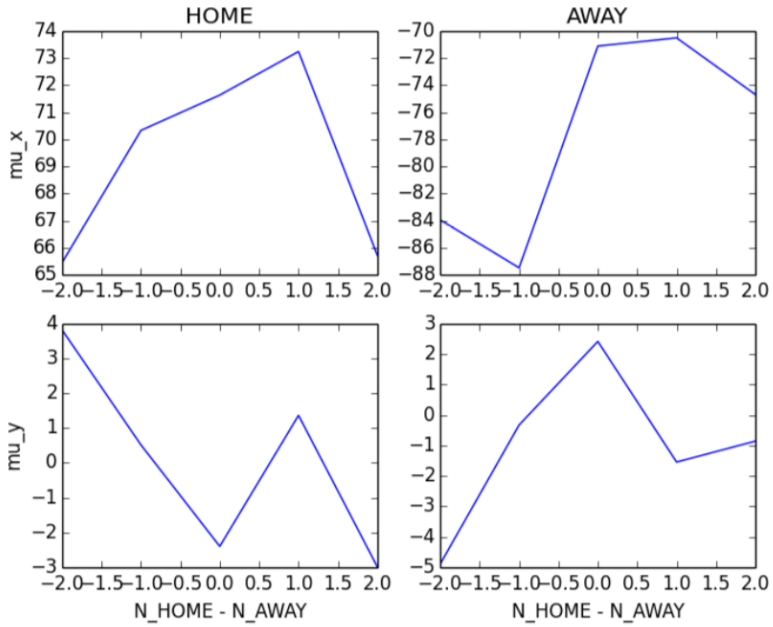


Figure 5.15: x and y components of the drift parameter μ as a function of goal difference $N^{HOME} - N^{AWAY}$ and team holding the ball U. The units are in [1% of the length of the football field] per minute. Therefore a value of 100 means one football field length per minute.

the estimated model parameters. The results are shown in Figure 5.19.

From the empirical data set I selected situations where the ball was within a 10 by 10 square at the center of the field (where the size of the whole field is 100 by 100 units) and observed the time evolution of the ball coordinate X_t for the next minute. There were a total of 494 such situations in all of the 31 games. The red curve shows the mean distance from the center as the function of time and the red bands show the 5% and 95% percentiles.

In order to generate the curve within the model, I performed the same number of Monte-Carlo simulations as the number of samples in the empirical data set, that is 494. In each simulation the ball started at the center of the field and I used the model parameters calibrated from all of the 31 games. The blue curve shows the mean distance from the center as the function of

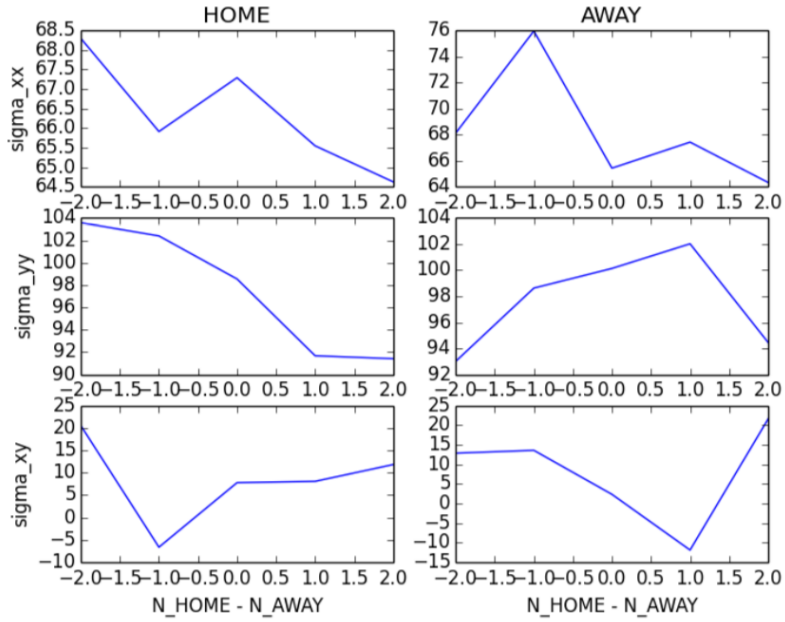


Figure 5.16: Components of the ball variance matrix σ_{xx}^2, σ_{yy}^2 and σ_{xy}^2 as a function of goal difference $N^{HOME} - N^{AWAY}$ and team holding the ball U. The units are in [1% of the length of the football field]^2 per minute. Therefore a value of 10000 corresponds to a standard deviation of one football field over the course of one minute.

time and the blue bands show the 5% and 95% percentiles.

The empirical and the simulated curves are in reasonable agreement.

5.3 Simplifying the model

Using the findings in the previous section I am able to simplify the model by introducing the following restrictions in the dependencies of the model parameters on the state variables:

- The ball losing intensities ν only depend on the team U and are otherwise constant.

- The ball drifts mu_x and mu_y only depend on position X and team

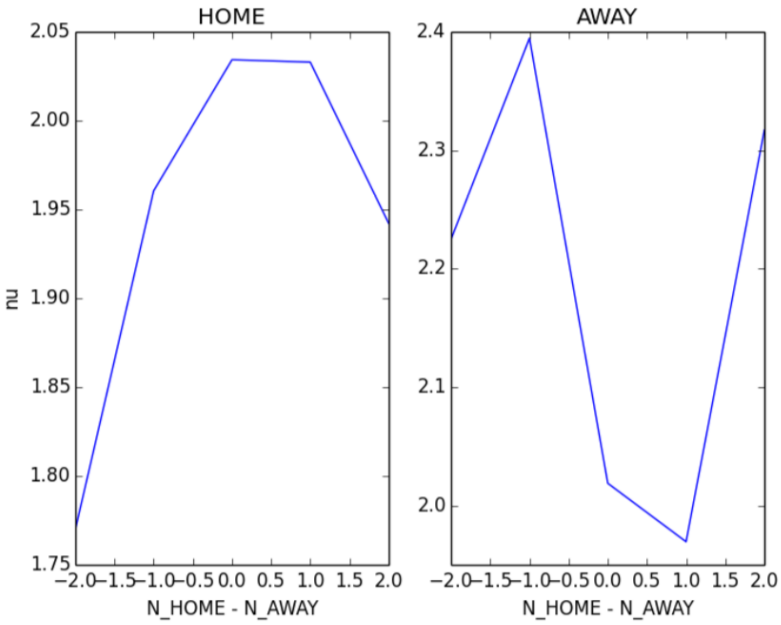


Figure 5.17: Ball losing intensity ν as a function of goal difference $N^{HOME} - N^{AWAY}$ and team holding the ball U. The units are in 1 per minute. Therefore a value of 2 means losing the ball twice a minute on average.

U such that the ball is pushed towards the opponent's gate in the x direction and towards the middle to the field in the y direction.

- The ball volatilities σ_x and σ_y only depend on the team U and are otherwise constant. The volatility in the cross direction σ_{xy} is zero.

- The goal intensities λ^u depend on the team U, position X and time t, but don't depend on the number of goals N. The dependency with position X is Gaussian around the opponent's gate with a standard deviation of 15 units and the dependency in time is linear.

With these restrictions, the formal dependencies of the model parameters on the state variables simplify to the following:

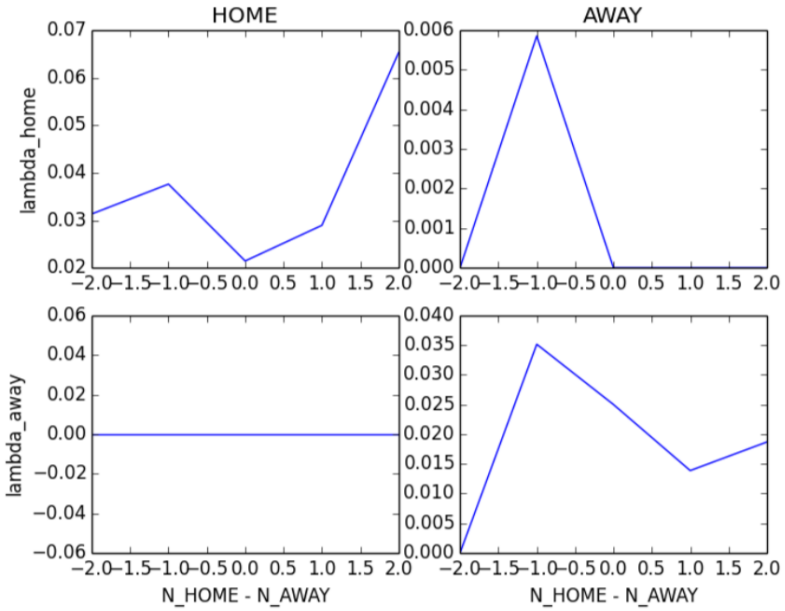


Figure 5.18: Goal scoring intensities λ^{HOME} and λ^{AWAY}as a function of goal difference $N^{HOME} - N^{AWAY}$ and team holding the ball U. The units are in 1 per minute. Therefore a value of 0.1 means scoring a goal every 10 minutes on average.

$$\nu(U) = \nu^U \tag{5.9}$$

$$\mu_x(X,U) = \begin{cases} \mu_x^U \frac{100-X_x}{100} & if\ U = HOME \\ -\mu_x^U \frac{X_x}{100} & if\ U = AWAY \end{cases} \tag{5.10}$$

$$\mu_y(X,U) = \mu_y^U \frac{50 - X_y}{100} \tag{5.11}$$

$$\sigma_x(U) = \sigma_x^U \tag{5.12}$$

$$\sigma_y(U) = \sigma_y^U \tag{5.13}$$

$$\sigma_{xy} = 0 \tag{5.14}$$

$$\lambda^U(X,U,t) = \lambda_0^U \lambda_X^U(X)\,\lambda_t(t) \tag{5.15}$$

$$\lambda_X^U(X) = \begin{cases} \exp\left[-\frac{(X_x-100)^2+(X_y-50)^2}{2\cdot 15^2}\right] & if\ U = HOME \\ \exp\left[-\frac{X_x^2+(X_y-50)^2}{2\cdot 15^2}\right] & if\ U = AWAY \end{cases} \tag{5.16}$$

$$\lambda_t(t) = 1 + \lambda_t \frac{t}{T} \tag{5.17}$$

$$\tag{5.18}$$

Parameter	Ball position and Team X and U	Total score $N^{HOME} + N^{AWAY}$
Ball losing intensity ν	Higher near to opponent's gate	Slight decrease
Ball drift μ_x	Pushes ball towards opponent	-
Ball drift μ_y	Pushes ball towards the middle	-
Ball volatility σ_x	Decreases towards opponent's gate	Slight increase
Ball volatility σ_y	-	-
Ball volatility σ_{xy}	-	-
Scoring intensity λ^u	Higher near to opponent's gate.	Increases

Parameter	Score difference $N^{HOME} - N^{AWAY}$	Game time t
Ball losing intensity ν	Increases if team is leading	Slight decrease
Ball drift μ_x	Increases if team is leading	-
Ball drift μ_y	-	-
Ball volatility σ_x	Increases if team is leading	Slight increase
Ball volatility σ_y	Slight decrease if team is leading	-
Ball volatility σ_{xy}	-	-
Scoring intensity λ^u	Increases if team is leading	Increase

Table 5.2: Summary of the dependencies of parameters μ, σ, ν and λ^u on the state parameters X, U, N and t.

This parametrisation has a total of 13 scalar parameters. The values of these parameters have been estimated using least squares such that the simplified parameter surfaces are as close as possible to the estimated parameter surfaces in the discrete bins of the previous section. The list of the parameters along with their estimated values are shows in Table 5.3.

5.4 Solving the simplified model

Using this parametrisation, the original Equations 5.1, 5.2 and 5.3 describing the evolution of state within the model simplify to the following SDEs:

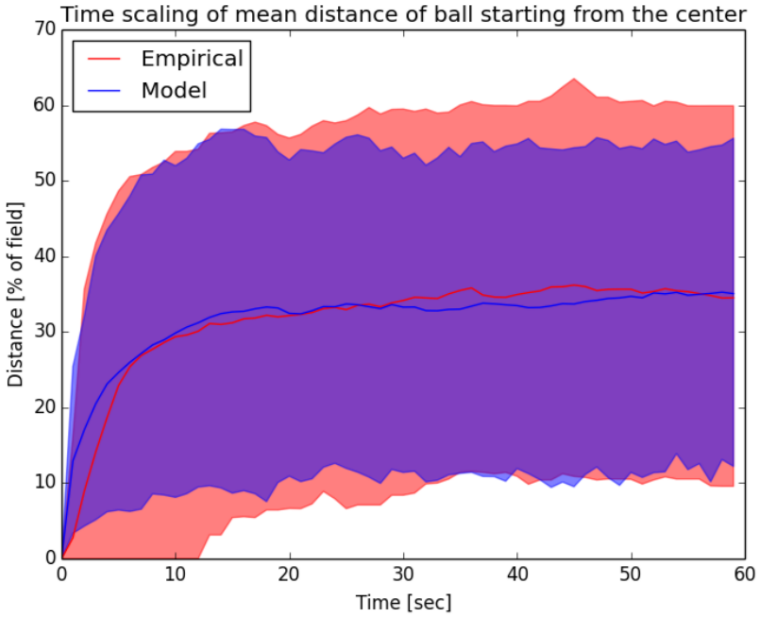


Figure 5.19: Time scaling properties of the ball distance. The red curve shows the mean distance of the ball from the center as a function of time within the empirical data set, conditional on the ball being at the center at time zero. The blue curve shows the same within the calibrated model, using a Monte-Carlo simulation. The bands show the 5% and 95% percentiles. The size of the whole field is 100 by 100.

Parameter	Value
ν^{HOME}	3.02
ν^{AWAY}	3.15
μ_x^{HOME}	284.42
μ_x^{AWAY}	275.13
μ_y^{HOME}	261.39
μ_y^{AWAY}	252.30
σ_x^{HOME}	55.19
σ_x^{AWAY}	54.23
σ_y^{HOME}	53.75
σ_y^{AWAY}	53.81
λ_0^{HOME}	157.36
λ_0^{AWAY}	126.36
λ_t	0.32

Table 5.3: List of calibrated model parameters.

$$
\begin{aligned}
dU_t &= -2U_t d\mathcal{N}_t^{\nu(U)} && (5.19)\\
dX_{x,t} &= \mu_x\left(X_x, U\right) dt + \sigma_x\left(U\right) d\mathcal{W}_{x,t} && (5.20)\\
dX_{y,t} &= \mu_y\left(X_y, U\right) dt + \sigma_y\left(U\right) d\mathcal{W}_{y,t} && (5.21)\\
dN_t^U &= d\mathcal{N}_t^{\lambda^U(X,U,t)}, && (5.22)
\end{aligned}
$$

In this section I attempt to solve this set of equations analytically. Solving Equation 5.19 is relatively easy because it's a point process jumping between two states with constant intensity. However, solving Equations 5.20 and 5.21 is more complicated. As it is pointed out in detail in section 5.4.2.1, these equations are similar to the Ornstein-Uhlenbeck process. However, we have a process for each team and these two processes are mixed together by the point process jumping between the two teams which introduces a complication that I was unable to find an analytical solution for. As a result I used standard numerical techniques to solve these equations.

5.4.1 Distribution of the team U_t

Equation 5.19 describing the team dynamics becomes independent of the other state parameters. The corresponding Kolmogorov forward equation is:

$$
\frac{\partial}{\partial t} f_U\left(u,t\right) = -\nu\left(u\right) f_U\left(u,t\right) + \nu\left(-u\right) f_U\left(-u,t\right), \tag{5.23}
$$

where $f_U\left(u,t\right) = \mathbb{P}\left[U_t = u\right]$ is the marginal probability that the ball is held by team u at time t. It can be seen that the solution to equation 5.23 is:

$$f(u,t) = Ae^{-(\nu(1)+\nu(-1))t} + \frac{\nu(-u)}{\nu(1)+\nu(-1)}, \tag{5.24}$$

where A is defined by the initial condition at $t = 0$ such that $A = f(u,0) - \frac{\nu(-u)}{\nu(1)+\nu(-1)}$. It can also be seen that in the stationary case the probability that team u has the ball, or in other words the average ball possession of team u is equal to:

$$f(u, t \to \infty) = \frac{\nu(-u)}{\nu(1)+\nu(-1)}. \tag{5.25}$$

5.4.2 Computing the marginal density of team U_t and position X_t

According to Equations 5.20 and 5.21 the ball coordinates X_x and X_y become independent of each other and only depend on themselves and U. Therefore the distribution in the two spatial dimensions become independent of each other and can be computed as two one-dimensional distributions, rather than one two-dimensional distribution.

5.4.2.1 Similarity with the Ornstein-Uhlenbeck process

Because of the choice of $\mu_x(X_x, U)$ and $\mu_y(X_y, U)$, the processes for X_x and X_y are both special cases of the Ornstein Uhlenbeck process [57], also known as the Vasicek model [59] which is defined by the following SDE:

$$dx_t = \kappa(\theta - x_t)\,dt + \sigma\,dW_t \tag{5.26}$$

One of the reasons for the popularity of the Ornstein-Uhlenbeck process is that the marginal distribution has the following analytical closed form solution (see for example [18] or [58]):

$$f(x,t) = \sqrt{\frac{\kappa}{\pi\sigma^2(1-e^{-2\kappa t})}} \exp\left\{\frac{-\kappa}{\sigma^2}\left[\frac{(x-x_0e^{-\kappa t})^2}{1-e^{-2\kappa t}}\right]\right\} \tag{5.27}$$

which is valid for $\theta = 0$ assuming that the process started at x_0 at $t = 0$.

Our case differs in two significant ways. First, the boundaries at $x = 0$ and $x = 100$ are reflective, both for X_x and X_y. Ornstein-Uhlenbeck processes with reflective boundaries have been studied by [61], [62] and [60] among others, therefore this obstacle could be solved. However, the second difficulty in our case is that the parameters κ and θ depend on the team U and are therefore randomly changing between two different values. I was unable to find any treatment for this case in the literature, and unable to come up with an analytical solution, therefore I reverted to classical numerical methods to solve the equations.

5.4.2.2 Numerical solution

The Kolmogorov forward equations describing the time evolutions of the marginal densities of the ball positions X_y and X_y are:

$$\begin{aligned}
\frac{\partial}{\partial t} f_{X_x}(x_x,u,t) &= -\frac{\partial}{\partial x_x}\left[\mu_x(x_x,u) f_{X_x}(x_x,u,t)\right] + \frac{1}{2}\frac{\partial^2}{\partial x_x^2}\left[\sigma_x^2(x_x,u) f_{X_x}(x_x,u,t)\right] \\
&\quad -\nu(u) f_{X_x}(x_x,u,t) + \nu(-u) f_{X_x}(x_x,-u,t) \qquad (5.28) \\
\frac{\partial}{\partial t} f_{X_y}(x_y,u,t) &= -\frac{\partial}{\partial x_y}\left[\mu_y(x_y,u) f_{X_y}(x_y,u,t)\right] + \frac{1}{2}\frac{\partial^2}{\partial x_y^2}\left[\sigma_y^2(x_y,u) f_{X_y}(x_y,u,t)\right] \\
&\quad -\nu(u) f_{X_y}(x_y,u,t) + \nu(-u) f_{X_y}(x_y,-u,t) \qquad (5.29)
\end{aligned}$$

where $f_{X_x}(x_x,u,t)$ is the joint marginal distribution of the x ball coordinate and the team holding the ball, that is

$$f_{X_x}(x_x, u, t) = \lim_{dx_x \to 0} \frac{1}{dx_x} \mathbb{P}\left[X_{x,t} \in (x_x, x_x + dx_x), U_t = u\right] \tag{5.30}$$

and $f_{X_y}(x_y, u, t)$ is similarly the joint marginal distribution of the y ball coordinate and the team holding the ball.

The joint marginal distribution of the x and y ball coordinate along with the team holding the ball is:

$$f_X(x_x, x_y, u, t) = \frac{f_{X_x}(x_x, u, t) f_{X_y}(x_y, u, t)}{f_U(u, t)} \tag{5.31}$$

The partial differential equations 5.28 and 5.29 can be solved numerically as discussed in Section 2.4. In order to assess the performances of both the explicit and implicit methods, I solved for the marginal distribution of the ball's x coordinate at the end of the game for the home team, using both methods. In the explicit method I needed to use 500,000 time steps to avoid the stability issues. The implicit method was stable with an arbitrarily low number of time steps, however if the number of time steps was below 15,000, then the solution didn't converge to the explicit method's solution. Figure 5.20 shows the solutions obtained with both methods, using different number of time steps in the implicit method. Because the solution became stable from above 15,000 time steps in the implicit method, I decided to use the implicit method with 20,000 time steps which corresponds to roughly 4 steps per second.

Figure 5.21 shows the time evolution of the marginal joint probability density function of the ball coordinate and the team having the ball $f_X(x_x, x_y, u, t)$. The figures on the left show the density given that the home team has the ball, that is $f_t(X, U = HOME)$ and the figures on the right

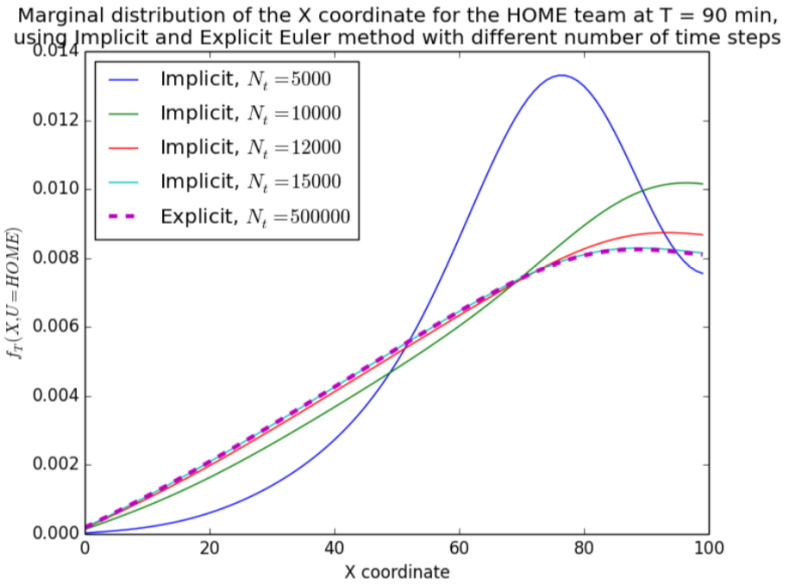


Figure 5.20: Comparison of convergence of the marginal distribution of the X ball coordinate between the Implicit and the Explicit methods with various numbers of time steps. Note that the Implicit method converges even if the number of time steps is very small, however with very small number of time steps it doesn't converge to the correct distribution. Convergence to the correct distribution for the Implicit method can be achieved by increasing the number of time steps. The Explicit method on the other hand doesn't converge at all with small number of time steps. Convergence can only be be achieved by increasing the number of time steps to extremely high values in which case the solution converges to the correct solution.

show the density for the away team $f_t(X, U = AWAY)$. Initially, at $t = 0$ the ball was at the center $X_0 = (50, 50)$ of the field and was held by the $U_0 = HOME$ team, this is shown by the figures in the top row corresponding to the first second in which case most of the density is on the left, around the center. As time goes by, more and more density propagates to the away team because of the non-zero ball losing intensity ν^{HOME} of the home team. Because the drifts in the x direction μ_x^U are opposite for the two teams, each team tries to push the ball towards the opponent's gate which is shown in the figures in the bottom row corresponding to 120 seconds. At this point

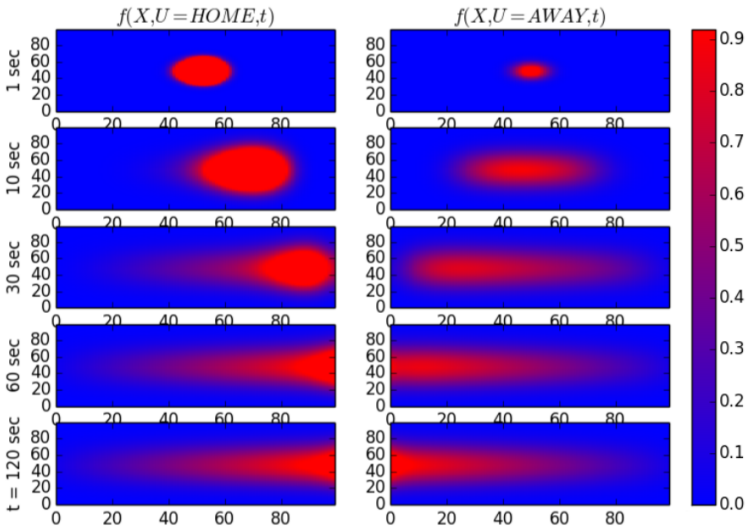


Figure 5.21: Marginal distribution of the ball coordinate X_t as a function of time, conditional on the home (left) or away (right) team holding the ball such that at $t = 0$ the home team had the ball. The colors represent the value of the probability density function, with red for high probabilities and blue for low probabilities. It can be seen that shortly after start almost all the probability is concentrated to the middle of the field and HOME team still holding on to the ball, however there is already a small probability for AWAY team gaining possession. As time goes by, the probability distribution approaches the stationary state in which the total probability is roughly split equally between the two teams, however the ball has a higher chance to being closer to the opponent's gate, whichever team is holding the ball.

the marginal distribution reached the stationary state.

5.4.3 Computing the goal intensity

Because the goal intensity in Equation 5.15 is independent of the number of goals and only depends on the team, ball position and time, the expectation of the goal intensity can be calculated given the marginal density $f_X\left(x_x, x_y, u, t\right)$ by:

$$\begin{aligned}\lambda^u\left(t|t_0,x_0,y_0,u_0\right) &= \mathbb{E}\left[\lambda^u\left(X,U,t\right)|U(t_0)=u_0,X_x(t_0)=x_0,X_y(t_0)=y_0\right] \\ &= \lambda_0^u\lambda_t\left(t\right)\mathbb{E}\left[\lambda_X^u\left(X\right)|U(t_0)=u_0,X_x(t_0)=x_0,X_y(t_0)=y_0\right] \\ &= \lambda_0^u\lambda_t\left(t\right)\int_{x_x,x_y}\lambda_X^u\left(x_x,x_y,t\right)f_X\left(x_x,x_y,u,t|x_0,y_0,u_0,t_0\right)dx_xdx_y \quad (5.32)\end{aligned}$$

Note that in terms of dimensions, the λ_0 parameter is the only intensity-like parameter in the sense that it's dimension is reciprocal of time while the other two parameters λ_t and λ_X are dimension-less.

Practically, all that needs to be done is to compute the dot product of the position-dependent component $\lambda_X^U\left(X\right)$ in Equation 5.16 by $f_X\left(x_x,x_y,u,t\right)$ and then multiply by the time-dependent component $\lambda_t\left(t\right)$ in Equation 5.17 and the finally multiply by the team's intensity parameter λ_0^U.

Figure 5.22 shows the goal scoring intensities for the two teams as a function of time, given that initially the ball was held by the $U = HOME$ team and started from different X_x coordinates while the $X_y = 50$ coordinate was always at the middle. The top figure shows the goal intensity of the home team λ_t^{HOME} and the bottom figure shows the goal intensity of the away team λ_t^{AWAY}. The home team's gate is located at $X_x = 0$ and the away team's gate is located at $X_x = 100$. The purple lines show the intensities when the ball started from right in front of the away team's gate. It can be seen that the home team's goal scoring intensity is initially high and gradually decreases as the distribution of the ball the ball drifts away from the gate or gets lost by the home team. At the same time, the away team's goal scoring intensity is practically zero for about 20 seconds and starts to increase gradually. This is because the away team needs time to gain the ball and bring it to the home team's gate at the opposite of the field before they have a chance to score the

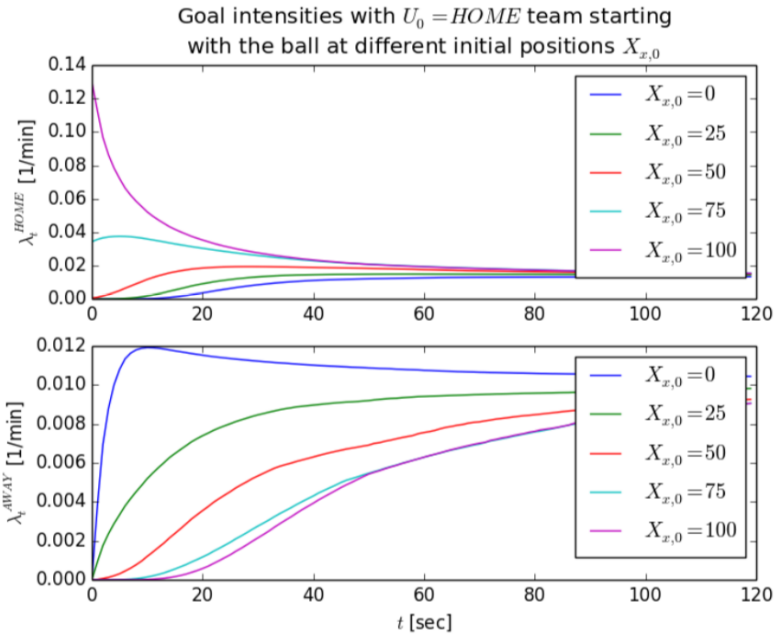


Figure 5.22: Home and away goal intensities $\lambda(U,t)$ for different starting positions $X_{x,0}$, given that at $t = 0$ the ball was held by the home team. Note that the home team's goal intensity starts form a high value in case $X_{x,0} = 100$ because in this case the ball is starting from right in front of the opponent's gate. On the other hand, when $X_{x,0} = 100$, that is when the ball starts from the home team's own gate, the goal intensity is initially zero because the team needs time to first bring the ball over to the opponent's gate in order to have a chance to score a goal. The away team's goal intensity starts at zero regardless of the initial position of the ball, because first they need to gain possession in order to have a chance to score a goal. Then, the away team's goal intensity rises faster or slower, depending on whether the ball started from right in front of the opponent's gate $X_{x,0} = 0$, or their own gate $X_{x,0} = 100$. In both cases the goal intensities converge to a stationary average intensity, after about two minutes which is the characteristic time during which the ball position and the team holding the ball has an effect on the near term goal intensity.

goal. The red curve corresponds to the ball starting from the center to the field. Both teams need time to increase their goal scoring intensity because the ball needs to get to their opponent's gate, however the away team needs somewhat more time because they also need to gain the ball from the home team. The blue curve corresponds to the ball starting from right in front of the home team's gate. Both team's goal intensities start at zero. However, the home team's intensity increases slower than the away team's intensity because the home team needs more time to bring the ball to the away team's gate at the opposite of the field. The away team's intensity increases faster because the ball is already in front of the home team's gate, all they need is to gain possession. After about two minutes, the marginal probability density of the ball position and the team having the ball $f_t(X, U)$ converges to the stationary state which makes the goal scoring intensities of both teams converge to the average value, which is independent of the initial condition.

This figure summarises the main difference between the constant intensity model and the microscopic model that takes into account the ball position and the team having the ball. It shows the ball position and the team having the ball is relevant for roughly the first 30 seconds in terms of affecting the goal scoring intensities. Beyond this time horizon the initial state dissipates and the intensities reach their stationary values.

5.5 Comparison of the full and the simplified model

In order to judge how well the simplified semi-analytic model performs compared to the full model, I performed Monte-Carlo simulation on the full model and estimated the time evolution of the marginal joint probability density

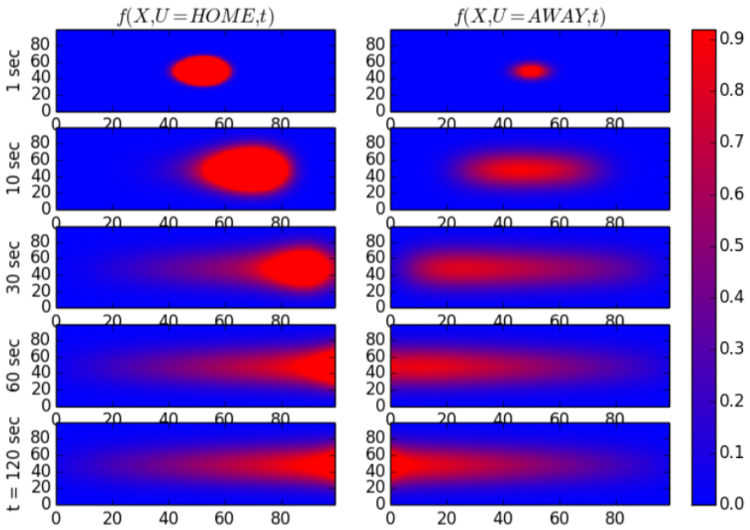


Figure 5.23: Marginal distribution of the ball coordinate X_t as a function of time, conditional on the home (left) or away (right) team holding the ball such that at $t = 0$ the home team had the ball, using Monte-Carlo simulation on the full model with 1e6 paths as defined in Section 5.1 and estimated in Section 5.2. The colors represent the value of the probability density function, with red for high probabilities and blue for low probabilities. It can be seen that the marginal density estimated from the full model evolves in a similar fashion to that of the simplified model's semi-analytic solution shows in Figure 5.21 which is an indication that the assumptions of the simplified model are reasonable.

function of the ball coordinate and the team having the ball $f_X(x_x, x_y, u, t)$ which is shown in Figure 5.23. The full model was defined in Section 5.1 and the model parameters were estimated in Section 5.2. Because the model is not analytically tractable I used Monte-Carlo simulation consisting of 1e6 paths to calculate the probability density function. It can be seen that the marginal density estimated from the full model evolves in a similar fashion to that of the simplified model's semi-analytic solution shows in Figure 5.21 which is an indication that the assumptions of the simplified model are reasonable.

5.6 Summary

In this chapter I suggested a model for the in-play dynamic of not only the number of goals scored (1), but also two additional features: the position of the ball (2) and the team holding the ball (3). These three state variables are driven by a set of stochastic differential equations, such that the scores and the team holding the ball are driven by jump processes and the ball position is driven by a Wiener process. In the first part of the chapter the parameters of these processes were allowed to depend freely on all of the state variables and after studying these generic parameter surfaces I was able to simplify the model greatly by parametrising these dependencies using a model of just 13 scalar parameters. I then showed that this simplified model can be solved using an efficient semi-analytic approach by solving the partial differential equation numerically that discribes the distribution of the state parameters.

I found that goal scoring intensities change over time and depend on the initial starting position of the ball and the initial team holding the ball. This is in line with intuitive expectations: if a team manages to bring the ball close to the opponent's gate, then this team's goal scoring intensity is high in the near future while if the ball is far away from the opponent's gate, the goal scoring intensity is lower. Given an initial starting position, over time the goal scoring intensities of both teams converge to stationary values. I found that this convergence happens in about 30 seconds, this is the time window in which the initial ball position is relevant and during which near term goal intensities can be predicted. Outside of this window the initial position has no predictive power and goal scoring intensities reach converge to their average values.

Chapter 6

Predicting Second Half Scores from First Half Features using Machine Learning Methods

In this chapter I study an in-play football data set containing high resolution information about various events during the game, such as goals, goal attempts, passes, ball position on the field, penalty cards, corners and so on. I construct features observed in the first half of the game (X) and consider the number of goals scored in the second half (Y). I run a number of feature selection methods to determine the most significant features. I then consider various predictive models and perform model selection and validation. In line with previous chapters, the goals are predicted in terms of Poisson intensities. The main difference to previous chapters is that in this chapter I follow a black box machine learning approach and consider a larger number of state variables, instead of considering stochastic models of specific forms that are driven by a lower number of parameters.

6.1 Introduction

In this chapter I use a data set that contains high resolution in-play information about various events, such as the location of the ball on the field, passes, cards, corners, goal attempts, goals and so on. Using these events I construct an 18-dimensional feature vector at the end of the first half of the game denoted by the vector X that only contains events that happened during the first half of the game. I use these features to to predict the two team's Poisson goal intensities λ in the second half of the game using a model that is in general denoted by $f(X, \theta)$ with θ being the parameters of the model. The Poisson goal intensities are same as defined in Chapter 3 apart from the fact that in this section the intensity is scaled to the whole 45 minute interval of the second half whereas in Chapter 3 intensity was quoted in units of $1/minute$. The actual number of goals scored by the two teams in the second half is denoted by a 2-vector Y and is assumed to follow independent Poisson distributions with an intensity 2-vector of λ:

$$\lambda = f(X, \theta) \tag{6.1}$$

$$Y \sim \text{Poisson}(\lambda) \tag{6.2}$$

An overview of the chapter is reported in Figure 6.1.

6.2 Overview of the data set

The data set I used contains a total of 2940 games played in the 2013/2014 Season of the 7 Leagues described in Table 6.1.

The data for each single game consists of a set of game level information

Variable	
X	18-dimensional vector of indicators determined during the first half of the game
λ	2-dimensional second half Poisson goal intensity vector
Y	2-dimensional outcome of the number of goals scored in the second half of the game

Feature selection methods
Correlation
Lasso
Random forest of trees

Prediction Methods	**Parameters**	**Hyperparameters**
K-Nearest Neighbors		Number of nearest neighbors, distance measure
Linear / Elastic Net	Weights, biases	Regularization factors
Neural Network	Weights, biases	Number of layers, number of neurons, activation function, regularization factors

Datasets	**Used for fitting**
Training	Optimize model parameters
Validation	Optimize model hyperparameters
Testing	No fitting, one final check

Performance measures	
Poisson Log-likelihood	Used for optimizing model parameters and hyperparameters
Table of frequencies	Can be interpreted intuitively, not used for optimization
Akaike Information Criterion	Used for model selection

Figure 6.1: A high level overview of the chapter showing the different variables, data sets, methods and performance measures used. A detailed description of all feature variables is shown in Table 6.2.

League	Number of games
English Football League Championship	526
Spanish Segunda Division	482
Brazilian Série A	386
English Barclays Premier League	410
French Ligue 1	408
Italian Serie A	416
Spanish La Liga	312

Table 6.1: List of the 7 Leagues of the 2013/2014 Season containing a total of 2940 games that the data set consists of.

such as the time of the game, the names of home and away teams and the list of players in both teams. This is then followed by a list of relevant events that happened during the game. Each event contains the millisecond resolution timestamp of the event, the game time of the event, such as first or second half, minutes and seconds, the relevant team and player associated with the event, the 2-dimensional coordinates of the event on the field, the type of the event and additional auxiliary information depending on the event type.

6.3 Constructing feature set and outcome variables

6.3.1 X - Feature Set - First Half Events

I constructed a total of 9x2 features for each game that consisted of the number of the 9 types of events reported in Table 6.2 that occurred in the first half of the game for the each of the two teams. These 9 event types as shown in Table 6.2 covered all available events in the data set. GOAL is the number of goals scored, PASS_OWN denotes the number of pass events in the team's own half, PASS_OPP denotes the number of pass events in the opponent's

Event Type	Description
GOAL	Goals scored by home / away teams
PASS_OWN	Any successful or unsuccessful passes on the team's own half of the field
PASS_OPP	Any successful or unsuccessful passes on the opponent's half of the field
ATTEMP_SAVED	Shot by home / away teams towards the goal of the opponent that were saved
MISS	Shot by home / away teams that goes wide or over the goal
CLEARANCE	Player of home / away team under pressure hits the ball clear of the defensive zone or / and out of play
CORNER_AWARDED	Corners awarded to home / away teams
CARD	Red, yellow or 2nd yellow card obtained by home / away teams
OUT	Ball goes out for a throw-in or a goal kick

Table 6.2: Types of events that were used as features.

half, ATTEMP_SAVED denotes the number of opponent goal attempts that were saved by the team, MISS denotes the number of missed shots that went wide or over the goal, CLEARANCE is the number of times the team was able to hit the ball away from it's defensive zone, CORNER_AWARDED is the number of corners awarded to the team, CARD is the number of cards obtained by the team and OUT is the number of times the ball went out for a throw-in or a goal kick. A summary of every feature is show in Table 6.2.

6.3.2 Y - Target Variables - Second Half Results

The outcome variables are a vector of 2 elements: the number of goals scored by the HOME and AWAY teams, respectively in the second half of the game.

6.4 Feature Selection

As a first step I ran a number of standard feature selection methods on the training set to separate the relevant features from the irrelevant ones.

It is important to note that in all the tests, for both the X and Y variables I constructed the differences between the HOME and the AWAY features. That is, for example the PASS_OWN feature in this section is the difference between the number of PASS_OWN events by the HOME team and the AWAY team. Similarly, for the Y outcomes, GOAL means the difference between the number of goals scored by the home and away teams in the second half. The reason I use differences rather than just the bare values is that in this way I can decrease the number of variables by a factor of two while at the same time preserve the most important link between outcomes and features. When I perform prediction in the next section I use the full set of variables, not just the differences.

6.4.1 Correlation

As a first attempt to determine the predictive power of the features I calculated correlation (see Section 2.6.1) between the X feature and the Y outcome variables.

The results are reported in Figure 6.2 and in Table 6.3.

Not surprisingly, PASS_OWN, PASS_OPP, CORNER_AWARDED, ATTEMPT_SAVED, MISS and GOAL events have a positive correlation. The explanation for PASS events is that the team with more passes has a higher ball possession, therefore more chance to score a goal. CORNER_AWARDED has a positive correlation, because it's easier to score a goal from a corner. ATTEMPT_SAVED and MISS are saved / missed near-goal events assigned

	Corr(X,Y)
PASS_OPP	20%
CORNER_AWARDED	14%
PASS_OWN	12%
ATTEMPT_SAVED	11%
CLEARANCE	-10%
MISS	10%
GOAL	10%
CARD	-10%
OUT	1%

Table 6.3: Correlations between first half features and second half scores ordered by decreasing absolute correlation.

to the attempting team, therefore the team with more attempts will eventually score more. GOALs in the first half naturally correlate positively with GOALs in the second half. It is interesting to note that first half GOALs have the lowest positive correlation with second half GOALs and PASS events have the strongest positive correlation.

The correlation with OUT events is practically negligible, apparently it is not relevant how many times the ball went off the field form any of the teams.

CARD and CLEARANCE events have a negative correlation with the simple explanations that a red or yellow CARD would decrease a team's chance to score a goal. The CLEARANCE event is assigned to the team which is under pressure in the sense that it is in a situation where it is likely to receive a goal, the more such situations are, the more likely it is that the team will actually receive a goal.

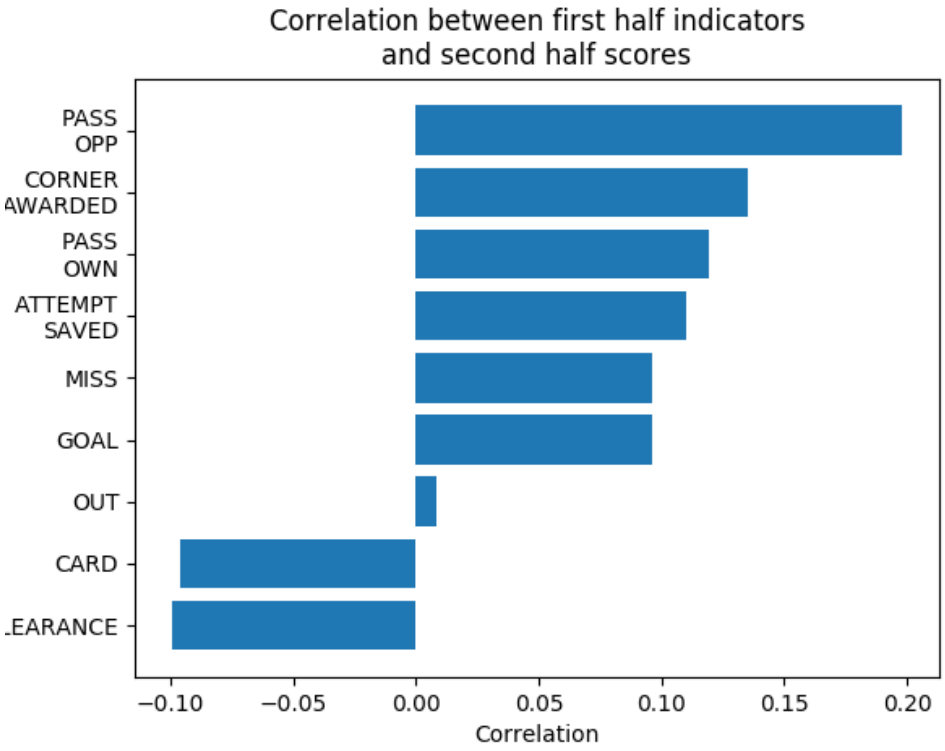
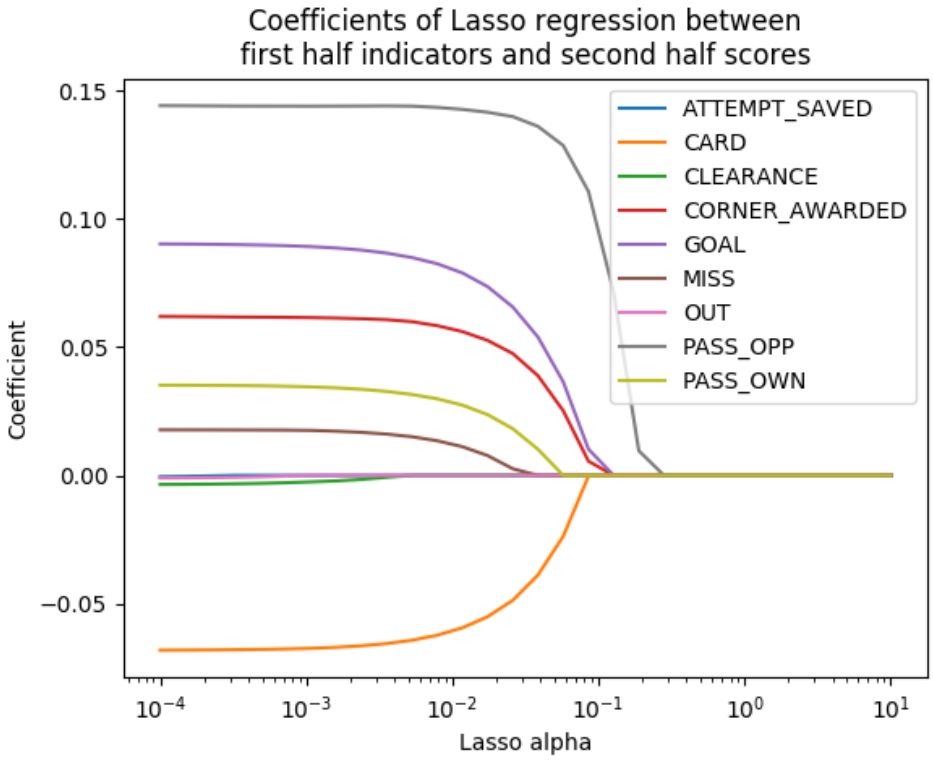


Figure 6.2: Correlations between first half HOME - AWAY feature differences and second half HOME - AWAY score differences.

6.4.2 Lasso Regression

I also performed Lasso regression (see Section 2.6.5.2) to select relevant features. By running the regression multiple times and increasing the value of the L1 penalty factor α, the coefficient of the least relevant feature goes to zero first while more relevant features keep a non-zero coefficient until the value of α increases to the point where eventually all coefficients become zero.

Figure 6.3 shows the values of the estimated coefficients as a function of the L1 penalty factor α. Note that the signs of the coefficients of the various features are consistent with the signs of the correlations in the previous section, that is all features have positive coefficients, except for CARD and CLEARANCE.



0.350.35

Figure 6.3: Values of Lasso coefficients of first half features as a function of L1 penalty factor α when regressing against second half scores. Note that the least relevant feature's coefficients go to zero first.

Figure 6.4 shows the values of α where each of the feature coefficients go to zero, the same data is reported in Table 6.4.

6.4.3 Random Forest of Trees

I performed another feature selection by using Random Forest of Trees (see Section 2.6.2). I created 250 random decision trees using Gini impurity. The more significant an feature is the higher it appears in the tree, therefore measuring the average height of each feature gives the feature importances for this method. The results are reported in Figure 6.5 and in Table 6.5 and are consistent with the results obtained with the other two methods.

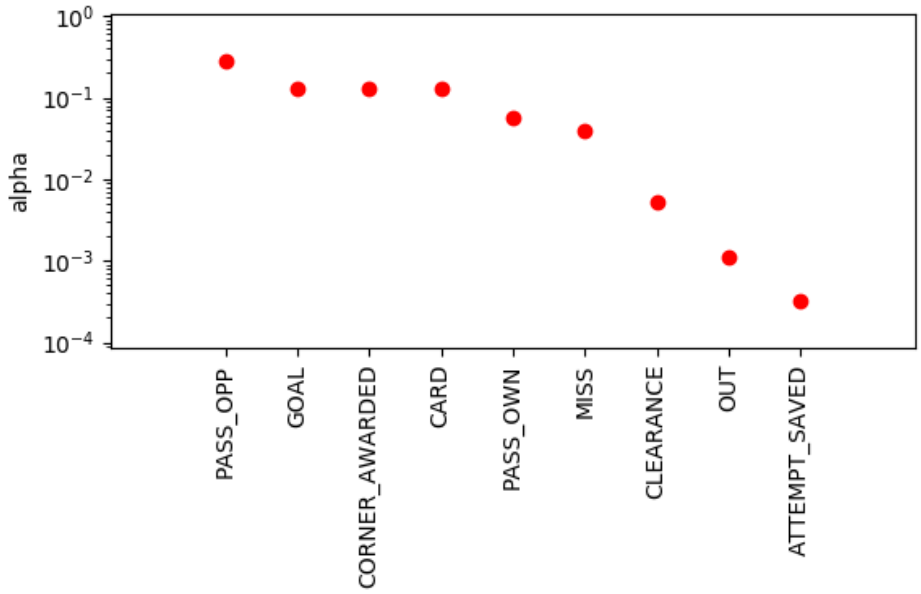


Figure 6.4: Lasso L1 penalty factor α values where the corresponding feature coefficients dropped to zero. Features with the highest values are the most relevant.

6.4.4 Overall Feature Rank

As we have seen, the three feature selection methods gave reasonably consistent results. To come up with a final ranking of feature importances I calculated the average of each feature's rank across the different methods. The overall rank of importances is PASS_OPP, CORNER_AWARDED, PASS_OWN, GOAL, CARD, ATTEMPT_SAVED, CLEARANCE, MISS and OUT. The overall feature ranks are reported in Table 6.6 which are found to be consistent among the three methods.

X	$\ln(\alpha)$
PASS_OPP	-1.3
CARD	-2.1
CORNER_AWARDED	-2.1
GOAL	-2.1
PASS_OWN	-2.9
MISS	-3.3
CLEARANCE	-5.2
OUT	-6.8
ATTEMPT_SAVED	-8.0

Table 6.4: Values of the log of Lasso L1 penalty factors α where the corresponding feature coefficients dropped to zero. Features with the highest values are the most relevant. The table shows $\ln(\alpha)$ values because the range of the α values varies greatly.

X	Importance
PASS_OPP	0.14
CORNER_AWARDED	0.13
GOAL	0.13
CARD	0.12
PASS_OWN	0.11
ATTEMPT_SAVED	0.10
CLEARANCE	0.10
MISS	0.08
OUT	0.07

Table 6.5: Average heights of each feature in the Random Forest of Trees method by building 250 random decision trees using Gini impurity measure.

6.5 Predicting Second Half Scores from First Half In-Play Feature

In this section I use different methods to predict the number of goals in the second half of the game based on features from the first half. Because scores are inherently random, I am not trying to predict the exact number of goals. Instead I make the same assumption as in Chapter 3, that is, that

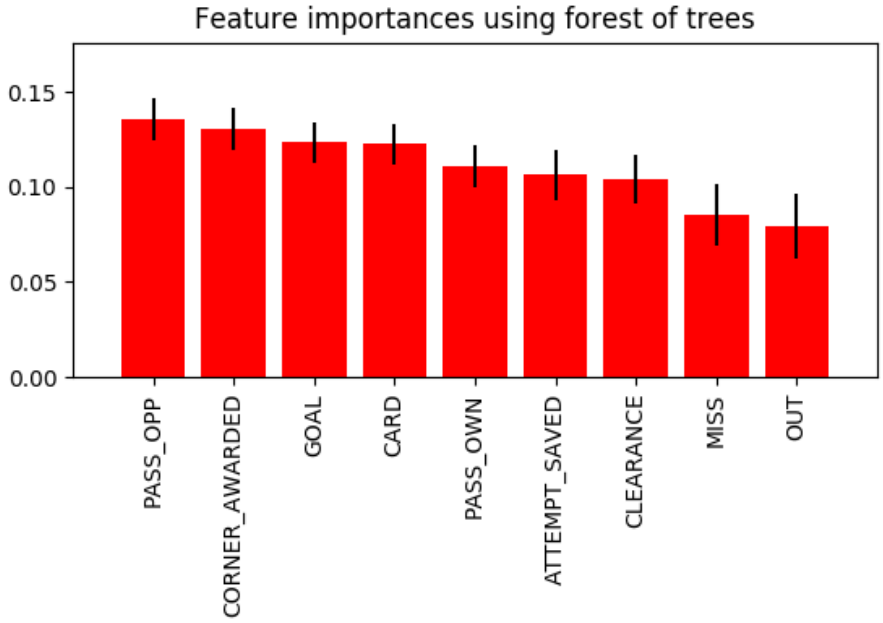


Figure 6.5: Average heights of each feature in the Random Forest of Trees method by building 250 random decision trees using Gini impurity measure.

the two team's scores follow independent Poisson distributions. The main assumption in this section is that the intensities are functions of the first half features, formally:

$$\lambda_i^M = f^M(X_i, \theta), \tag{6.3}$$

where f^M is the model function of model M, θ is the vector of model parameters, X_i is the vector of first half features for game i and λ_i is the predicted 2-vector of the home and away goal intensities for the second half of game i.

The predicted probability for a certain number of goals Y in the second

X	Correlation	Lasso	Forest	AVERAGE
PASS_OPP	1	1	1	1.0
CORNER_AWARDED	2	3	2	2.7
PASS_OWN	3	5	5	4.3
GOAL	7	4	3	4.6
CARD	8	2	4	4.6
ATTEMPT_SAVED	4	9	6	6.3
CLEARANCE	5	7	7	6.3
MISS	6	6	8	6.6
OUT	9	8	9	8.6

Table 6.6: Comparison of the feature importance ranks determined by different methods.

half then follows the independent Poisson distribution which has the following likelihood function:

$$\mathbb{P}\left(Y,\lambda\right) = e^{-\lambda_H}\frac{\lambda_H^{Y_H}}{Y_H!}e^{-\lambda_A}\frac{\lambda_A^{Y_A}}{Y_A!} \tag{6.4}$$

where the λ denotes second half intensities, Y denotes second half scores and the superscripts H and A refer to the home and away teams, respectively.

I use log-likelihood ratio over the Constant model (Section 6.5.1) as the primary measure of goodness of model fit which according to Section 2.5.1 has the following form:

$$\begin{aligned}
\mathcal{L}^M &= \ln\prod_i \frac{\mathbb{P}\left(Y_i,\lambda_i^M\right)}{\mathbb{P}\left(Y_i,\lambda^{Const}\right)} && (6.5)\\
&= \sum_i Y_{H,i}\ln\frac{\lambda_{H,i}^M}{\lambda_H^{Const}} - \left(\lambda_{H,i}^M - \lambda_H^{Const}\right) && (6.6)\\
&+ \sum_i Y_{A,i}\ln\frac{\lambda_{A,i}^M}{\lambda_A^{Const}} - \left(\lambda_{A,i}^M - \lambda_A^{Const}\right) && (6.7)\\
& && (6.8)
\end{aligned}$$

where the product index i goes through the set of games in the data set, $\mathcal{L}$ is the value of the log-likelihood ratio and M refers to the model being used. $\lambda_{H,i}^{M}$ and $\lambda_{A,i}^{M}$ are the second half home and away intensities predicted by model M for game i. λ_{H}^{Const} and λ_{A}^{Const} are the constant intensities that don't depend on the game i. $Y_{H,i}$ and $Y_{A,i}$ are the number of goals scored in the second half of game i.

The reason for using likelihood ratios over the Constant model is that in this way I can use the Constant model as a base-line and quickly determine whether a specific model is doing at least as good as the trivial Constant model.

6.5.0.1 Table of Frequencies

Log-likelihoods are theoretically sound measures of model goodness and are suitable to perform model fitting by maximising their value, see for example [82] or [81]. However, the actual log-likelihood value is not very informative in terms of how well the model fits. As an alternative measure of goodness I decided to use so-called table of frequencies which here are only used to check the results and not for any fitting or optimisation purposes.

A table of frequencies is, in some sense, similar to a stochastic confusion matrix because it compares predicted probabilities of the possible outcomes versus the actual frequency of each outcome. Outcomes in my case are the number of goals scored by the two teams in the second half and the number of the possible outcomes is therefore relatively high: considering only 0 to 4 possible goals for each team, the number of possible outcomes would be 25 and a 25-by-25 matrix would not be very informative. What I do instead is that I decrease the number of possible outcomes by condensing them into one of 3 possible outcomes depending on which team won the second half:

HOME, if the home team scored more goals, AWAY if the away team did score more goals and DRAW if the number of goals scored are equal. In this way the table of frequencies becomes a 3-by-3 matrix.

The table is then constructed in the following way. Each game in the data set has a certain second half outcome and also a probability predicted by the model for each of the 3 possible outcomes. The games are grouped by their actual outcomes in 3 groups, with each group corresponding to one column of the table. Within each group, the average of the predicted probabilities of the games are computed which results in a 3-vector that constitute the rows of the table. For example, the cell in the HOME column and the DRAW row of the table contains the average predicted probabilities of a DRAW for all games that eventually had a HOME win outcome. In case of a hypothetically perfect predictor that always predicts the actual outcome with 100% probability, the table would be equal to an identity matrix. In a practical case we expect that a good predictor gives higher chance for the actual outcome, rather than to other outcomes.

6.5.0.2 Training set, Validation set, Parameter fitting, Hyper-parameter tuning

The model parameters θ are split in two groups. The first group is referred to as regular model parameters and denoted by $\theta_{regular}$. For example in case of a Neural Network the weights of the network constitute the regular parameters. The second group is referred to as hyper-parameters and is denoted by θ_{hyper}. For example in case of Neural Networks, these constitute the number of neurons in each layer, the number of layers, the type of activation function used and so on. An overview of the parameters and hyper-parameters for each model are reported in Figure 6.1.

At the same time, the data set is randomly split into two groups of equal size: training set and validation set. These sets are fixed throughout the chapter. There is actually a third independent set referred to as test set, but the test set is not used for determining parameter values, but only as an independent final check of the results in the Conclusions section. The training and the validation sets contain 1470 games and the test set contains 1093 games.

In this way parameter fitting becomes a two stage process. First, for a given value of the hyper-parameter θ_{hyper}, I determine the regular parameter $\theta_{regular}$ by maximising the log-likelihood over the training set:

$$\theta_{regular}\left(\theta_{hyper}\right) = \arg\max_{\theta_{regular}} \mathcal{L}(\theta_{hyper}, \theta_{regular}, X_i, Y_i, i \in TRAIN) \tag{6.9}$$

This maximisation is referred to as parameter fitting in the conventional sense that involves a gradient-based optimisation of the parameter values.

The second step is to select the best performing hyper-parameter. This is again a maximisation of the log-likelihood, but this time the maximisation is performed on the validation set, instead of the training set:

$$\begin{aligned} \theta &= \left(\theta_{hyper}, \theta_{regular}\left(\theta_{hyper}\right)\right) && (6.10) \\ &= \operatorname*{argmax}_{\theta_{hyper}} \mathcal{L}(\theta_{hyper}, \theta_{regular}\left(\theta_{hyper}\right), X_i, Y_i, i \in VALIDATE) && (6.11) \end{aligned}$$

An important difference between regular parameter fitting and hyper-parameter selection is that log-likelihoods are usually not continuous as a function of hyper-parameters, therefore gradient-base optimisation methods cannot be employed and I therefore rather perform a simple maximum selec-

tion over a set of possible values.

6.5.1 Constant Model - The Benchmark

I used the Constant Model as a benchmark which assumes that each game behaves the same way in the sense that the goal intensities are constant regardless of the values of the features. This is similar to the Constant Intensity model introduced in Chapter 3 with the slight difference that in Chapter 3 the constancy assumption is emphasized as a function of other state variables of the same game where in this chapter the constancy is across all games and as a function of all feature values. I used this model as a baseline model to compare more sophisticated models in subsequent sections. Formally, the model's parameters are only the constant intensities λ^{Const} which at the same time are the predicted intensities for all games, regardless of the values of the feature vectors X_i:

$$f^{\text{Const}}(X_i, \theta = \lambda^{\text{Const}}) = \lambda^{\text{Const}}, \tag{6.12}$$

The model estimation becomes a trivial task: the maximum likelihood estimate of the constant Poisson intensity is the average of the number of goals.

$$\lambda^{\text{Const}} = \frac{1}{n_{\text{train}}} \sum_{i \in \text{train}} Y_i \tag{6.13}$$

where n_{train} is the size of the whole training set. This can be easily verified by differentiating Equation 6.4 wrt. a constant λ. For our training set, the value of the constant intensities are:

	AWAY	DRAW	HOME
p_HOME	38.65%	38.65%	38.65%
p_DRAW	37.43%	37.43%	37.43%
p_AWAY	23.92%	23.92%	23.92%

Table 6.7: Table of frequencies for the constant model. Columns correspond to the actual outcome in the training set and rows contain the average predicted probabilities of those outcomes using the model parameters calibrated on the training set which in case of the constant model are the same constant intensities in Equation 6.15 for all games. As expected, the predicted probabilities are the same for all outcomes. Also, note that HOME win probabilities are overall higher than AWAY win probabilities which is referred to as the 'home effect'.

$$\lambda_{\mathrm{H}}^{\mathrm{Const}} = 0.8363 \tag{6.14}$$

$$\lambda_{\mathrm{A}}^{\mathrm{Const}} = 0.5890 \tag{6.15}$$

where the superscript H refers to the HOME and A to the AWAY team. Note that these intensities are the overall goal intensities for the second half of the game, that is they are to be interpreted in terms of average number of goals per 45 minutes.

Table 6.7 shows the table of frequencies for the constant model. The values in each column are the same due to the fact that the constant model predicts the same probabilities for all games, regardless of their feature values or outcomes.

6.5.2 K-Nearest Neighbors

As a second, more sophisticated model I used the K-Nearest Neighbour model introduced by [77]. Refer to section 2.6.3 for an overview of the model.

It is important to note that before applying this method, features need to be scaled to zero mean and unit variance. This is necessary because otherwise points along the dimension of an feature that has lower variance would appear closer and therefore the results would be more sensitive to certain features simply due to their natural scaling. The scaling factors have been determined on the training set and the same factors have been applied on the validation set in order to make sure that all model parameters, including the scaling factors are determined using the training set alone.

I fitted the model using two different distance measures: L1 and L2 (Euclidean). I used a different number of nearest neighbors going up to the size of the whole training data set which is 1460. I also used a different number of features from 1 to 9, in the order of relevance as reported in Table 6.6 of the previous section. The results are reported in Tables 6.8 and 6.9 for the L2 and L1 norms respectively. The best results are achieved by using 150 nearest neighbors and 4 out of 9 features in both cases. Euclidean L2 distance performs somewhat better, than the L1 distance. The table of frequencies for the best results are reported in Tables 6.10 and 6.11.

6.5.3 Linear Model

Moving up in complexity I next used a linear model where the log-intensity is a linear function of the features with the model parameters being the weights and the biases of the model:

$$\lambda^{\text{Linear}}(X) = f^{\text{Linear}}(X, \theta = \{w_H, w_A, b_H, b_A\}) \tag{6.16}$$

$$= \begin{cases} \exp(X^T w_H + b_H) \\ \exp(X^T w_A + b_A) \end{cases} \tag{6.17}$$

Number of features	Number of nearest neighbors: 25	100	150	200	250	300	500	750	1000	1460
1	-15.7	27.8	28.9	31.9	30.4	30.1	31.4	24.6	18.9	0.0
2	-5.5	31.6	33.4	30.2	29.9	28.4	28.5	24.7	18.4	0.0
3	-6.6	28.2	29.6	29.2	30.7	31.7	29.2	24.7	18.4	0.0
4	-25.7	18.9	26.2	27.3	29.4	29.3	27.6	23.7	17.3	0.0
5	-13.7	20.6	25.5	26.3	29.1	27.8	27.9	25.7	19.4	0.0
6	-15.2	24.8	25.5	27.9	27.2	28.2	29.4	25.3	19.1	0.0
7	-15.9	27.7	24.6	27.2	29.0	28.4	25.8	23.4	18.1	0.0
8	-23.1	18.7	25.0	27.6	29.7	29.6	25.1	22.9	18.8	0.0
9	-30.8	19.9	22.2	22.4	26.1	25.0	22.0	21.9	16.9	0.0

Table 6.8: Log-likelihood ratio of the Nearest Neighbor model over the Constant model on the validation set with L2 (Euclidean) distance. Columns correspond to the number of nearest neighbors. Rows correspond to the number of features used in the order reported in Table 6.6. Note that the last columns corresponds to 1460 nearest neighbors which is the complete data set and therefore is equivalent to the constant model, hence the log-likelihood ratio in the last column is equal to zero. The maximum value corresponds to 150 nearest neighbors and 2 out of 9 features.

where X is the column-vector of the features of an individual game, w_H, w_A are the vectors of linear coefficients for the HOME and AWAY teams and b_H, b_A are the scalar bias terms for the HOME and AWAY teams.

The model can be trained by maximising the log-likelihood on the training set. It is important to note that the model is linear in the sense that log-intensities are linear functions of the features. However, due to the fact that intensities are not linear and also to the fact that the Poisson log-likelihood function is also non-linear, this model is not an ordinary linear regression and in order to maximise the log-likelihood a non-linear optimisation needs to be performed. Maximising on the training set without applying any regularization would introduce over-fitting. I therefore apply both L1 and L2 regularisation on the weights as explained hereafter.

Number of features	Number of nearest neighbors									
	25	100	150	200	250	300	500	750	1000	1460
1	-15.7	27.8	28.9	31.9	30.4	30.1	31.4	24.6	18.9	0.0
2	-5.5	31.6	33.4	30.2	29.9	28.4	28.5	24.7	18.4	0.0
3	-6.6	28.2	29.6	29.2	30.7	31.7	29.2	24.7	18.4	0.0
4	-25.7	18.9	26.2	27.3	29.4	29.3	27.6	23.7	17.3	0.0
5	-13.7	20.6	25.5	26.3	29.1	27.8	27.9	25.7	19.4	0.0
6	-15.2	24.8	25.5	27.9	27.2	28.2	29.4	25.3	19.1	0.0
7	-15.9	27.7	24.6	27.2	29.0	28.4	25.8	23.4	18.1	0.0
8	-23.1	18.7	25.0	27.6	29.7	29.6	25.1	22.9	18.8	0.0
9	-30.8	19.9	22.2	22.4	26.1	25.0	22.0	21.9	16.9	0.0

Table 6.9: Log-likelihood ratio of the Nearest Neighbor model over the Constant model on the validation set with L1 distance. Columns correspond to the number of nearest neighbors. Rows correspond to the number of features used in the order reported in Table 6.6. Note that the last columns corresponds to 1460 nearest neighbors which is the complete data set and therefore is equivalent to the constant model, hence the log-likelihood ratio is equal to zero. The maximum value corresponds to 150 nearest neighbors and 2 out of 9 features.

	AWAY	DRAW	HOME
p_HOME	35.64%	38.32%	39.54%
p_DRAW	37.85%	37.54%	37.21%
p_AWAY	26.51%	24.14%	23.26%

Table 6.10: Table of frequencies of the Nearest Neighbour model with L2 (Euclidean) distance using 150 nearest neighbors and 4 out of 9 features.

6.5.3.1 Elastic Net Regularization

I used the Elastic Net method which adds both an L1 and an L2 penalty term to the log-likelihood and thereby can avoid over-fitting. The objective function to minimise becomes:

$$\mathcal{L}^{\text{Linear}}\left(w_H, w_A\right) + \alpha\left(\|w_H\|_1 + \|w_A\|_1\right) + \beta\left(\|w_H\|_2 + \|w_A\|_2\right) \tag{6.18}$$

	AWAY	DRAW	HOME
p_HOME	35.55%	37.73%	38.83%
p_DRAW	38.02%	37.67%	37.35%
p_AWAY	26.42%	24.59%	23.82%

Table 6.11: Table of frequencies of the Nearest Neighbour model with L1 distance using 250 nearest neighbors and 2 out of 9 features.

The coefficients w_k and the biases b_k are determined by minimising the objective function over the training set. The L1 and L2 regularization factors α and β are determined by cross-validating against the validation set. That is the maximisation is performed on the training set for multiple values of the regularization factors and then the factors that result in the highest log-likelihood on the validation set are chosen. The log-likelihood ratios on the validation set are reported in Table 6.12 and in Figure 6.6. The optimal log-likelihood ratio on the validation set was reached with $\alpha = 1000$ and $\beta = 0$ with a log-likelihood ratio of 37.4. As the values of the regularization factors increase, the value of the log-likelihood ratio goes to zero. This is because a high enough regularization factor forces all weights to zero and only the bias terms remain in which case the model degrades into the Constant model.

To demonstrate the effect of over-fitting and the necessity for hyper-parameter tuning on the validation set, the log-likelihood ratios on the training set are shown in Table 6.13. As expected, the highest log-likelihood ratio on the training set is achieved with zero regularization factors.

It is worth noting that the L1 and L2 regularizations have a similar effect in the sense that a similar maximum log-likelihood ratio can be achieved by using either one of the regularizations. However it is also worth noting that using L2 only yields a slightly lower result of 36.5 than the optimum value of 37.4 that was achieved by using L1 only.

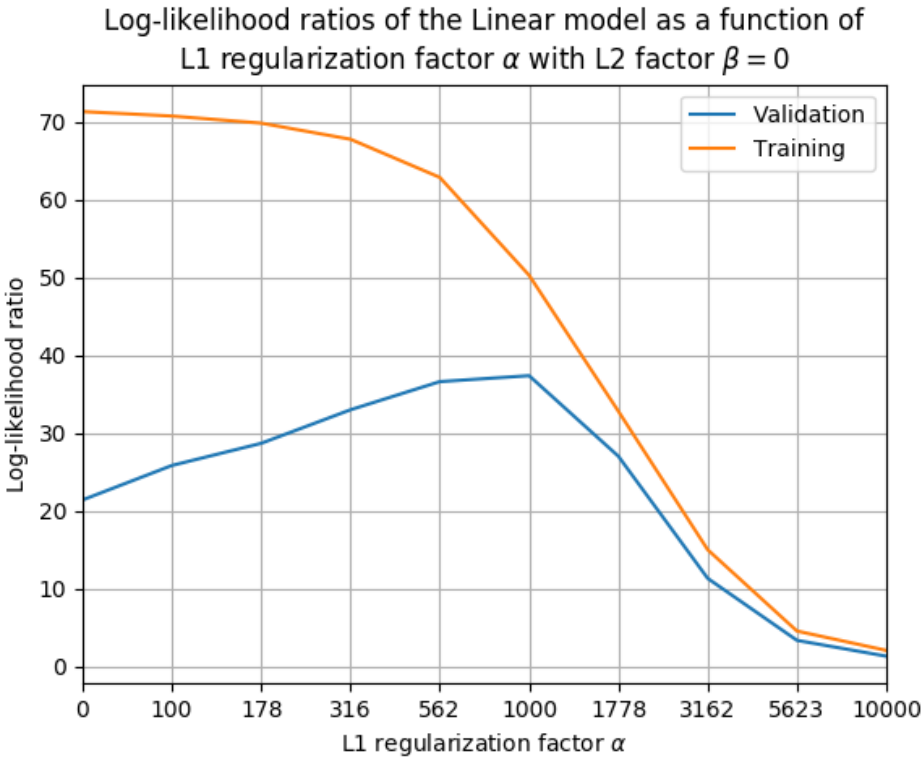


Figure 6.6: Log-likelihood ratios of the Linear model over the Constant model as a function of the L1 regularization factor α with L2 regularization factor of $\beta = 0$. Small values of the regularization factor correspond to over-fitting, that is the training set has the highest log-likelihood ratio, but the validation set performs poorly. By increasing the regularization factor there is an optimum at which the validation set has the maximum log-likelihood ratio and the training set still performs reasonably well. Any further increase after this point will deteriorate performance.

L1 regularization factor α \ L2 regularization factor β	0	10	100	1000	10000	100000	1000000
0	21.4	21.4	22.0	26.5	36.5	16.8	2.3
100	25.8	25.9	26.3	29.7	36.7	16.2	2.2
178	28.7	28.8	29.2	31.8	36.7	15.6	2.1
316	33.0	32.9	33.0	34.9	36.4	14.7	2.0
562	36.6	36.1	36.2	37.2	34.7	12.9	1.7
1000	37.4	36.3	36.5	36.0	29.5	9.6	1.3
1778	27.0	29.0	28.2	24.3	19.2	5.4	0.7
3162	11.3	13.3	11.9	9.7	7.3	3.5	0.5
5623	3.3	3.3	1.1	2.2	3.1	1.8	0.1
10000	1.3	0.4	2.0	1.4	-0.0	1.2	-0.2

Table 6.12: Log-likelihood ratios of the Linear model over the Constant model on the validation set. Rows and columns correspond to the values of α and β, that is the L1 and L2 regularization factors, respectively. The optimum value is achieved at $\alpha = 1000$ and $\beta = 0$. Note that sufficiently large regularization factors result in zero weights except for the bias term, in which case the model becomes equivalent to the Constant Model, hence the log-likelihood ratio is zero.

6.5.4 Neural Networks

Finally, the most complex model I consider are Neural Networks as introduced in Section 2.6.4 to predict second half goal intensities. The network's input is the 2 x 9 dimensional feature vector X and the output is the 2-dimensional intensity vector λ^{NN}. The last layer's activation function was always exponential in order to ensure positivity of the intensities. Then the formal model definition according to Equation 2.30 is:

$$\begin{aligned} \lambda^{\text{NN}}(X) &= f^{\text{NN}}(X, \theta = \{W_i, b_i, i \in [0, n-1]\}) \\ &= \exp\left(b_{n-1} + W_{n-1} h_{n-2}\left(\ldots b_1 + W_1 h_0\left(b_0 + W_0 X\right)\right)\right) \end{aligned} \quad (6.19)$$

where the regular model parameters consist of the bias vectors b_i and the

L1 regularization factor α	L2 regularization factor β = 0	10	100	1000	10000	100000	1000000
0	71.3	71.3	71.3	70.9	59.9	21.5	2.9
100	70.7	70.7	70.7	69.8	58.2	20.5	2.7
178	69.8	69.9	69.8	68.7	56.8	19.7	2.6
316	67.8	67.7	67.5	66.4	54.0	18.3	2.4
562	62.9	62.6	62.7	61.5	48.6	15.9	2.1
1000	50.3	50.0	50.2	49.3	37.3	11.7	1.6
1778	32.8	33.7	33.1	29.7	22.1	6.9	0.9
3162	15.0	16.7	13.9	12.2	8.2	5.1	0.9
5623	4.5	3.5	2.3	2.1	3.0	2.7	0.1
10000	2.0	1.0	2.1	1.8	-0.4	1.5	-0.1

Table 6.13: Log-likelihood ratios of the Linear model over the Constant model on the training set. Rows and columns correspond to the values of α and β, that is the L1 and L2 regularization factors, respectively. As expected, maximum log-likelihood ratio on the training set is achieved with no regularization, but this is over-fitting which is demonstrated by the fact that log-likelihood ratio on the validation set is sub-optimal with zero regularization factors as reported in Table 6.12

	AWAY	DRAW	HOME
p_HOME	36.60%	38.34%	39.77%
p_DRAW	37.68%	37.42%	37.01%
p_AWAY	25.72%	24.24%	23.22%

Table 6.14: Table of frequencies of the Linear model at the optimum fit of L1 factor $\alpha = 1000$ and L2 factor $\beta = 0$.

weight matrices W_i of each layer and $h_i(.)$ denote the element-wise non-linear activation functions.

The regular parameters of the model are the weights and the biases while the hyper-parameters are the number of layers, the number of neurons in each layer and the types of activation functions used.

6.5.4.1 Over-fitting and Regularization

Because the number of parameters in a Neural Network is much higher than in case of a Linear model, over-fitting is an even more serious issue and must be addressed. There are several widely used methods, a few of which are early stopping [68], dropout [69], noise injection [70] and L1 or L2 regularization [71] that was already introduced in section 6.5.3.1.

I decided to use L1 regularization on the weights of the network that is essentially the inclusion of a penalty term in the cost function:

$$\mathcal{L}^{\mathrm{NN}}\left(W_0, W_1, \ldots W_{n-1}\right) + \sum_{i \in [0,n-1]} \alpha \left\|W_i\right\|_1 \tag{6.20}$$

Note that only the weights are regularized and the bias terms are allowed to fit freely.

6.5.5 Results

I used the Tensorflow library [78] to train the network and used the built-in Adam optimisation method [74] to maximise the log-likelihood values. As usual, I trained on the training set and then checked the log-likelihood ratio against the Constant model on the Validation set.

6.5.5.1 Learning Rate

Learning rate is one of the most important hyper-parameters, therefore I decided to check it's effect on the results first. I created a network consisting of 1 hidden layer with 8 neurons in that layer, plus the input layer with 18 neurons and the output layer of 2 neurons. The activation function in the hidden layer was tanh. The results are reported in Figure 6.8. Note that the learning rate doesn't have a significant effect on the log-likelihood ratio in

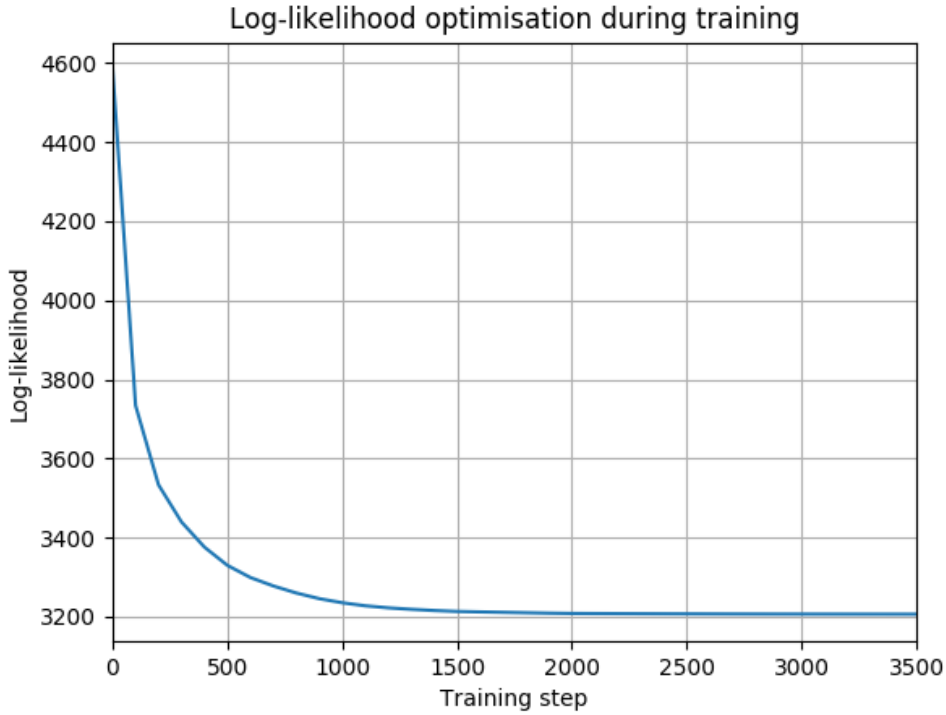


Figure 6.7: Log-likelihood optimisation on the training set as a function of the number of training steps. Because the optimisation library is only able to perform minimisation and not maximisation, I was doing minimisation of negative log-likelihoods which is the reason for the values decreasing and not increasing as training progresses. Learning rate is 0.001, L1 regularisation factor α is 200, 1 hidden layer with 8 neurons and the activation function is tanh.

the range $\alpha < 300$, therefore I decided to use the value of 0.001 from here on. Figure 6.7 shows the Log-likelihood optimisation during training as a function of training steps with learning rate set to 0.001 and L1 regularisation factor set to 200.

6.5.5.2 Number of Neurons

The effect of the number of neurons on the Log-likelihood ratio is reported in Figure 6.9 using 1 hidden layer and a tanh activation function. It can be

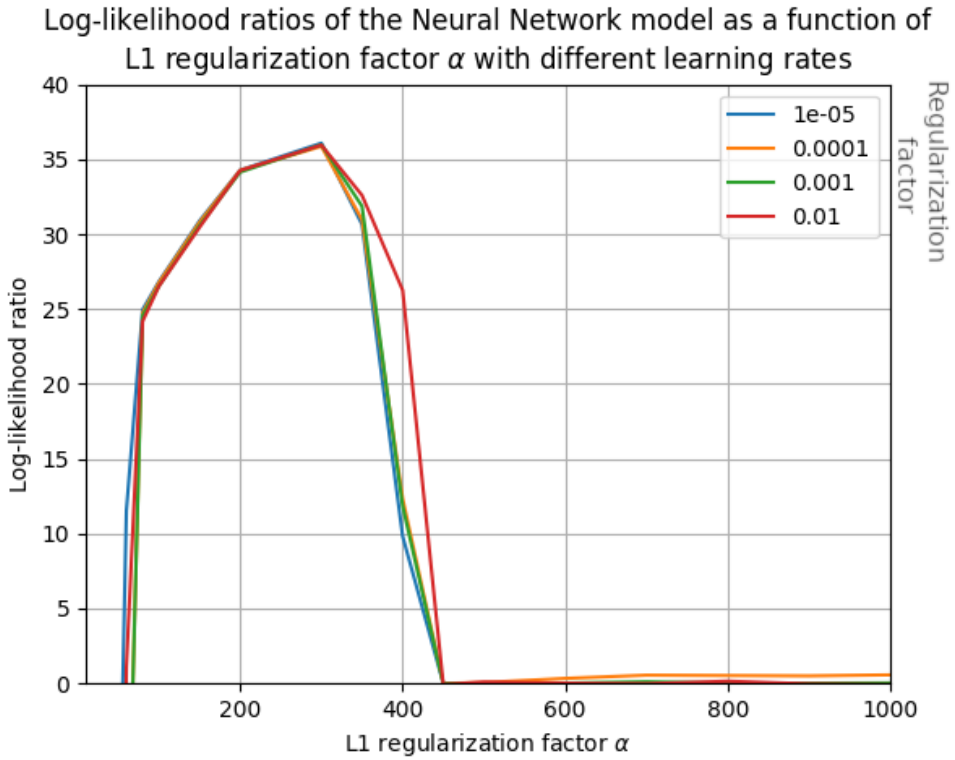


Figure 6.8: Log-likelihood ratios of the Neural Network model over the Constant model as a function of the L1 regularization factor α using various learning rates. There is 1 hidden layer with 8 neurons and the activation function is tanh. Note that the learning rate doesn't seem to significantly affect the log-likelihood ratio.

seen that the maximum is achieved in case of 16 neurons in the hidden layer which together with the 18 neurons in the input layer and 2 neurons in the output layer makes a total of 36 neurons in the network.

6.5.5.3 Number of Layers

I created networks consisting of multiple layers and ran the Log-likelihood ratios using 1 to 4 layers, with 8 neurons in each layer. The results are reported in Figure 6.10. The optimum is with 1 layer and adding more layers actually degrades performance.

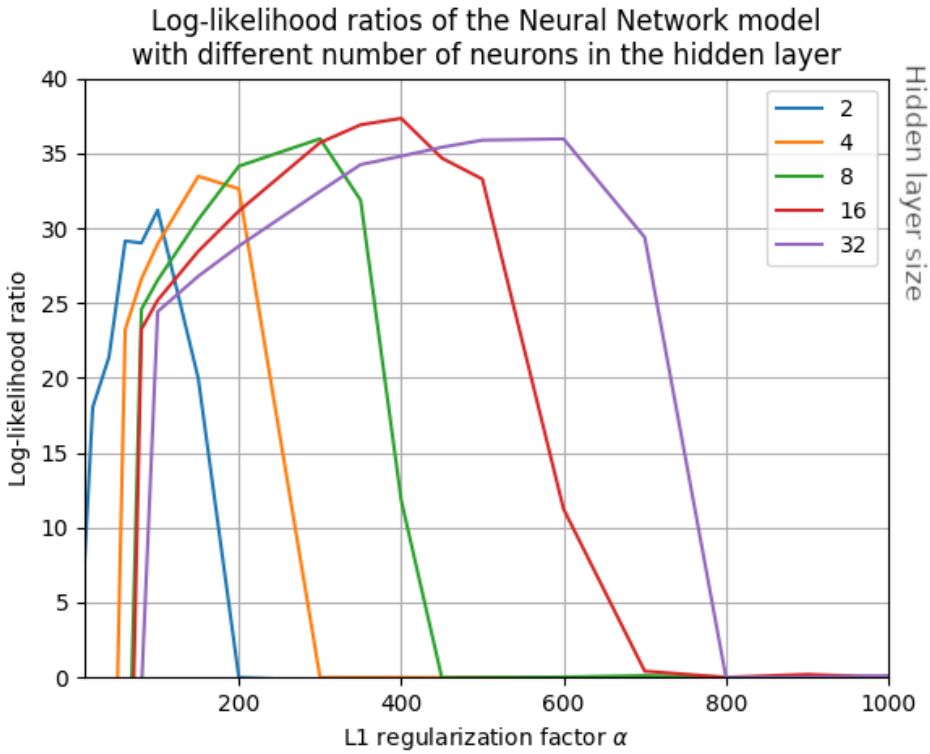


Figure 6.9: Log-likelihood ratios of the Neural Network model over the Constant model as a function of the L1 regularization factor α using different number of neurons in the hidden layer. Learning rate is 0.001, there is 1 hidden layer and the activation function is tanh.

6.5.5.4 Activation Function

I compared the effect of using various activation functions in the hidden layer using 1 hidden layers with 8 neurons. The difference is not significant, although tanh and ReLU seem to perform best. The results are reported in Figure 6.11.

6.5.5.5 Best Architecture

The overall best performing architecture was able to reach a log-likelihood ratio of 37.33 and consisted of 1 hidden layers with 16 neurons in the hidden

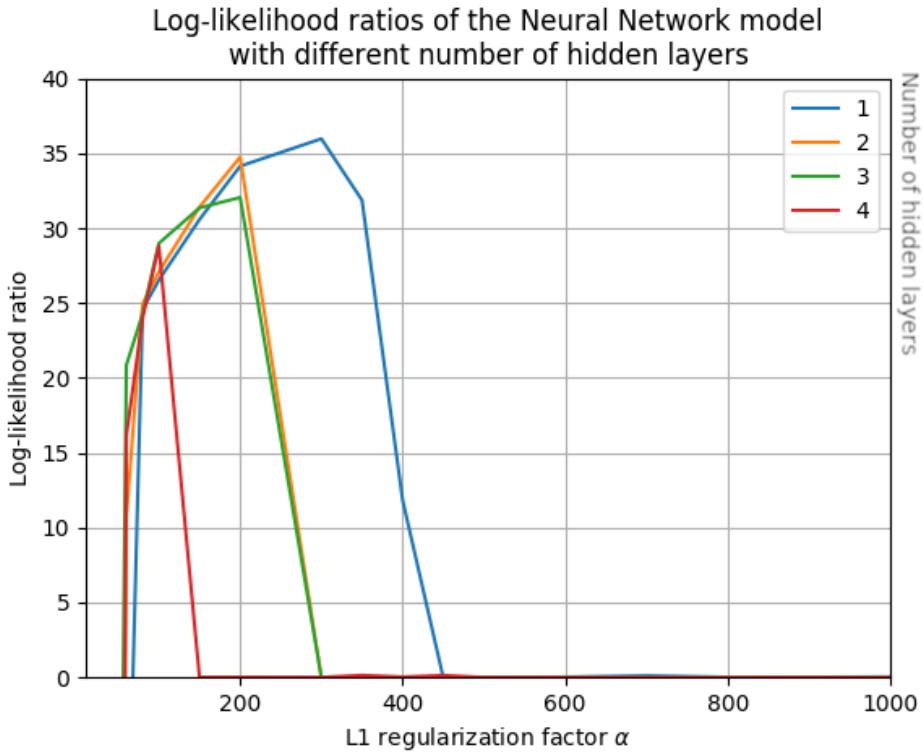


Figure 6.10: Log-likelihood ratios of the Neural Network model on the validation set as a function of the L1 regularization factor α using different number of hidden layers. Learning rate is 0.001, there are 8 neurons in each layer and the activation function is tanh.

	AWAY	DRAW	HOME
p_HOME	36.32%	38.27%	39.73%
p_DRAW	37.69%	37.42%	36.92%
p_AWAY	25.99%	24.31%	23.35%

Table 6.15: Table of frequencies of the Neural Network model using the best performing architecture.

layer using a tanh activation function with an L1 regularization factor of $\alpha = 400.00$. The corresponding table of frequencies is reported in Table 6.15.

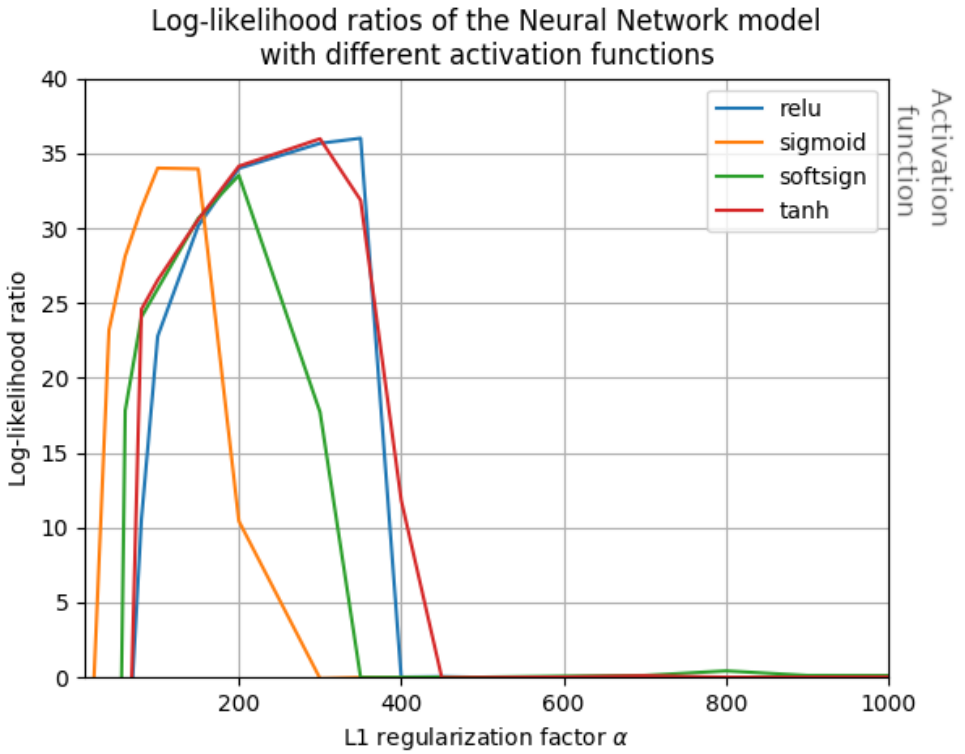


Figure 6.11: Log-likelihood ratios of the Neural Network model on the validation set as a function of the L1 regularization factor α using various activation functions. Learning rate is 0.001, there are 8 neurons in 1 hidden layer.

6.5.6 Summary

Table 6.16 shows the log-likelihood ratios achieved by the different models, using the best performing hyper-parameter values in case of each model in the $\mathcal{L}$ Validate row. All the models performed similarly with the Nearest Neighbor model being the worst performing model achieving a log-likelihood ratio of 33.44. The Linear model had a slightly higher log-likelihood ratio of 37.36 than the Neural Network model of 37.33. Although the difference is very small and the Linear model seems performing only slightly better than the Neural Network model. One explanation might be that my data sets

were of size of 1460 games which may just be too small for training the 320 parameters of the network.

When I take into account the number of parameters and use the Akaike Information Criterion (see Section 2.5.3) to decide between the Linear and the Neural Network model, the Linear model is a clear winner because the number of parameters k is much lower than that of the Neural Network model. In case of the Nearest Neighbor model I assumed the number of parameters k to be equal to 1, which is the number of nearest neighbors.

Table 6.16 also shows Wilks' p-values according to Section 2.5.4. In my case the base-line constant model has $m = 2$ parameters for the two team's constant intensities and the number of parameters of the alternative model is denoted by $h = k$, therefore the degrees of freedom of the χ^2 distribution is simply equal to $k - 2$. The Nearest Neighbor and the Linear models have high p values of close to 100% while the Neural Network model has a low p value with close to 0% due to the large number of parameters.

Throughout the chapter the training and the validation set was fixed. In order to avoid spurious results due to the specific choice of these sets, I created 5 additional training and validation sets which contain the same games and are of the same size as the original sets, but are randomly re-shuffled. In this way I created 5 cross-validation sets. I re-trained every model on the training sets and then took the log-likelihood ratios from the validation sets for those same hyper-parameters that were found to be best performing in the original validation set. In this way I obtained 5 log-likelihood ratios corresponding to the 5 cross-validation sets for each model. I calculated the minimums, maximums and averages of these 5 log-likelihood ratio values for each model and the results are reported in the $\mathcal{L}$ Cross-Validate Min, Max, Avg rows of Table 6.16. By looking at the $\mathcal{L}$ Cross-Validate Avg row, it can be seen

that the results are in line with the results found using the original set shown in row $\mathcal{L}$ Validate: the Linear model performs the best followed closely by the Neural Network model, followed by the K-Nearest Neighbor model. This confirms that the original results are not due to the specific choice of the original train and validation sets, but are robust to random changes in the train and validation sets.

As a final check, I ran all the trained models against a test set that I never used during training and hyper-parameter tuning in the earlier sections. The test set consisted of a total of 1093 games from the 2013/2014 season of the following championships: Dutch Eredivisie, German Bundesliga, Russian Premier League and Norwegian Tippeligaen. The log-likelihood ratios over the constant model for the best performing architectures for the various models are show in the $\mathcal{L}$ Test row of Table 6.16. The log-likelihood ratios are somewhat lower than the ones achieved on the validation set. This is indeed expected since those values correspond to the maximum across all hyper-parameters considered, therefore those values are optimised whereas the values on the test set are not. The differences are reasonably low so that we can be confident that no serious over-fitting happened in the hyper-parameter tuning phase.

There is a discrepancy in the table of frequencies (Tables 6.10, 6.11, 6.14 and 6.15), namely that all models perform poorly when predicting the DRAW outcome in the sense that the predicted probability of a DRAW isn't higher for games that have a DRAW outcome than for games that have a HOME or AWAY win outcome. This is an interesting discrepancy that can be explained through the limitations of the constant intensity model and the fitting process. Recall Figures 4.1 and 4.2 that show that the goal distribution is more heavy-tailed than the Poisson distribution, that is higher goal numbers

	K-Nearest Neighbor	Linear	Neural Network
$\mathcal{L}$ Validate	33.44	37.36	37.33
k	1	36	320
AIC	64.88	2.72	-565.34
Wilks' p-value	100.00%	99.98%	0.00%
$\mathcal{L}$ Cross-Validate Min	29.11	35.52	34.71
$\mathcal{L}$ Cross-Validate Avg	35.18	37.28	36.63
$\mathcal{L}$ Cross-Validate Max	44.43	39.64	39.33
$\mathcal{L}$ Test	29.67	30.44	33.14

Table 6.16: The maximum log-likelihood ratios achieved by the different models on the Validation set, on the 5-fold Cross-Validation set and on the final Test set along with the number of parameters k and the corresponding AIC values and Wilks' p-values.

have a higher chance than predicted by the constant Poisson model. However throughout this chapter we used a constant Poisson parametrization for the goal distribution, that can be responsible for such discrepancies. Furthermore, when fitting the models, we were maximising the log-likelihood defined in Equation 6.5, instead of minimizing some kind of discrepancy in the table of frequencies. In other words, the table of frequencies was never a fititng target, it's rather a side-effect kind of observation of goodness that can have discrepancies. This can be seen by the fact that even when we look at the simplest possible model, the constant model and we look at the actual training set that was used for fitting, rather than the validation set, even in this case the HOME, AWAY and DRAW probabilities predicted by the model after the fit are slightly different than the emprical frequencies in the dataset. In other words, the fit doesn't reproduce the actual table of frequencies because that wasn't the fitting target. In this case the predicted HOME, DRAW and AWAY probabilities were 38.65%, 37.43% and 23.92% respectively as seen in Table 6.7. In contrast, the actual training set had 39.29% of games ending in a HOME win, 35.55% enidng in a DRAW and

25.16% ending in an AWAY win. It can be seen that even in the simplest case (constant model, fitting on the training set) the table of frequencies underestimates the probability of a DRAW. This can explain the discrepancy seen across all models of being a poor predictor of a DRAW outcome.

6.6 Conclusions

In this chapter I studied an in-play football data set containing high resolution information regarding various events that happen during the game. I constructed features from the first half of the game (X) and considered the number of goals scored in the second half (Y). First I ran various feature selection methods to determine the most significant features which turned out to be the number of passes on the opponent's half, corners awarded, number of passes in the team's own half and number of goals scored in the first half. Then I used maximum likelihood to quantify goodness of fit using various models and to predict the Poisson goal intensity of the second half based on the values of the features in the first half. I used the Constant intensity model as a benchmark. The Linear model with Elastic Net regularization turned out to perform the best in terms of likelihood on the validation set, slightly outperforming both the K-Nearest Neighbor and the Neural Network models. However, in terms of Akaike information criterion, the best performing model turned out to be the K-Nearest Neighbor model followed by the Linear model, followed by the Neural Network model. The reason for this is that the K-Nearest Neighbour model had the lowest number of parameters with $k = 0$. However this might be slightly misleading, because although the model itself doesn't have any parameters that were optimised, the trained model contains a copy of the whole training set, therefore deter-

mining the number of parameters k in case of the K-Nearest Neighbor model is slightly ambiguous and therefore this result must be interpreted with care. In terms of the Linear and Neural Network models there is no such ambiguity, although the two models performed similarly in terms of log-likelihood, in terms of Akaike Information criterion the Linear model clearly outperforms the Neural Network model because it has much lower number of parameters.

In conclusion, the Linear Model together with parameter regularization turned out to be the most effective method among the methods that were investigated in this chapter.

Chapter 7

Comparing the Predictive Powers of the Microscopic and the First Half Indicators Models

In this chapter I used the Microscopic Model introduced in Chapter 5 and the First Half Indicators Model introduced in Chapter 6 to predict goal intensities in the second half of the game and I compared the predictive powers of the two models.

7.1 Data Set and Calibration

I perform prediction on the second halves of the same set of 30 games of the UEFA Euro 2012 Championship that I used in Section 5.2.

7.1.1 Microscopic Model

To calibrate the Microscopic Model, I use the data from the first halves of the 30 games of the UEFA Euro 2012 Championship and perform the same calibration procedure outlined in Section 5.3 with the only difference that I only use data from the first halves of the games to determine the model parameter values which I then use to predict the intensities in the second half.

7.1.2 First Half Indicators

To fit the parameters of the First Half Indicators Model, I use the set of the same 1470 games from Section 6 that were used to train the model. The hyper-parameters are the ones of the best performing model according to Section 6.5.6. Specifically, I use a Linear Model with Elastic Net Regularisation with $\alpha = 1000$ and $\beta = 0$.

7.2 Prediction

7.2.1 Microscopic Model

Recall Equation 5.32 according to which the predicted goal intensity for time t, conditional on an initial state at time t_0 is defined by

$$\lambda^u\left(t|t_0, x_0, y_0, u_0\right) = \mathbb{E}\left[\lambda^u\left(X, U, t\right)|U(t_0) = u_0, X_x(t_0) = x_0, X_y(t_0) = y_0\right] \tag{7.1}$$

where x_0, y_0 and u_0 are the initial positions of the ball and the team initially holding the ball at time t_0. Team u's goal intensity at time $t > t_0$ is

the probability that the team scores a goal in the infinitesimal time interval $[t, t + dt]$, divided by the length of the time interval dt. The intensity within the model can be calculated using Equation 5.32 after calibrating the model parameters using the first half of the game.

As time t_0 goes by in the second half of the games, the initial values x_0, y_0, u_0 are updated continuously and on each update we are able to calculate predicted goal intensities for all future times $t > t_0$ of the game. As illustrated in Figure 5.22, this prediction is expected to be more precise for the near future, that is for t close to t_0 than for times further away. Betting markets are aware of this and therefore an in-play delay of about 8 seconds is introduced depending on the market which is applied on all bets submitted to the market. Therefore I introduced a delay parameter denoted by Δt which is equal to the difference between the time t_0 when the state of the game was observed and the time t that the intensity was predicted for. That is $\Delta t = t - t_0$. I kept this delay constant for the whole of the second half, but ran the prediction with multiple different values. Ultimately, in this way I was able to calculate the time series of predicted intensities for the two teams defined in Equation 7.1 which from now on is denoted by the short-hand notation $\lambda^u_{\Delta t}(t)$:

$$\lambda^u_{\Delta t}(t) = \lambda^u \left(t + \Delta t | t, X_x(t), X_y(t), U(t)\right) \tag{7.2}$$

where now t is the time when the state is observed and the prediction is made, Δt is the time delay that the prediction is made for in the future and X_x, X_y, U denote the state of the game that is updated continuously as the game progresses.

7.2.2 First Half Indicators

The predicted goal intensities in the First Half Indicators model are constant for the whole duration of the second halves of the games, depending on the indicator values observed at the end of the first half.

7.3 Evaluation of the Predictive Power

Recall that the log-likelihood ratio of two Poisson Processes according to Equation 2.22 in Section 2.5.2 is:

$$\ln \frac{\mathbb{P}\left[\{t_i\}_{i=1}^{k} | \lambda_1(t)\right]}{\mathbb{P}\left[\{t_i\}_{i=1}^{k} | \lambda_0(t)\right]} = -\int_{T_0}^{T_1} (\lambda_1(t) - \lambda_0(t))\, dt + \sum_{i=1}^{k} \ln \frac{\lambda_1(t_i)}{\lambda_0(t_i)} \tag{7.3}$$

where $\lambda_1(t)$ is the Microscopic model's predicted goal intensity coming from Equation 7.2 and $\lambda_0(t)$ is the goal intensity coming from the First Half Indicators Model which for a specific game is constant during the whole of the second half.

7.4 Results

The home team's goal intensities predicted by the two models for the second half of the Portugal - Netherlands game of the 2012 European UEFA Championship using various delay values are shown in Figures 7.1, 7.2, 7.3, 7.4. As the delay values are increasing from 2 to 40 seconds, the variability of the Microscopic Model intensities are decreasing because the initial state is becoming less relevant.

Table 7.1 shows the log-likelihood ratios of the Microscopic Model with various delay values over the First Half Indicators model for 30 games of the UEFA Euro 2012 Championship. The log-likelihood ratios were calculated using Equation 2.22. Figure 7.5 shows the average log-likelihood ratios across all games as a function of delay along with the 10% - 90% confidence interval. Note that smaller delay values result in a higher log-likelihood ratio and as the delay approaches 30 seconds, the log-likelihood ratio value drops to close zero. That is, for small delay values the Microscopic Model is able to outperform the First Half Indicators model because the state of the game is still relevant for the time the predictions are being made for. As the delay grows, the initial state of the game becomes less relevant and the intensities approach their stationary values, as also shown in Figure 5.22. In this regime the Microscopic Model has no edge over the First Half Indicators model which is the reason for the average log-likelihood ratios dropping to negative values.

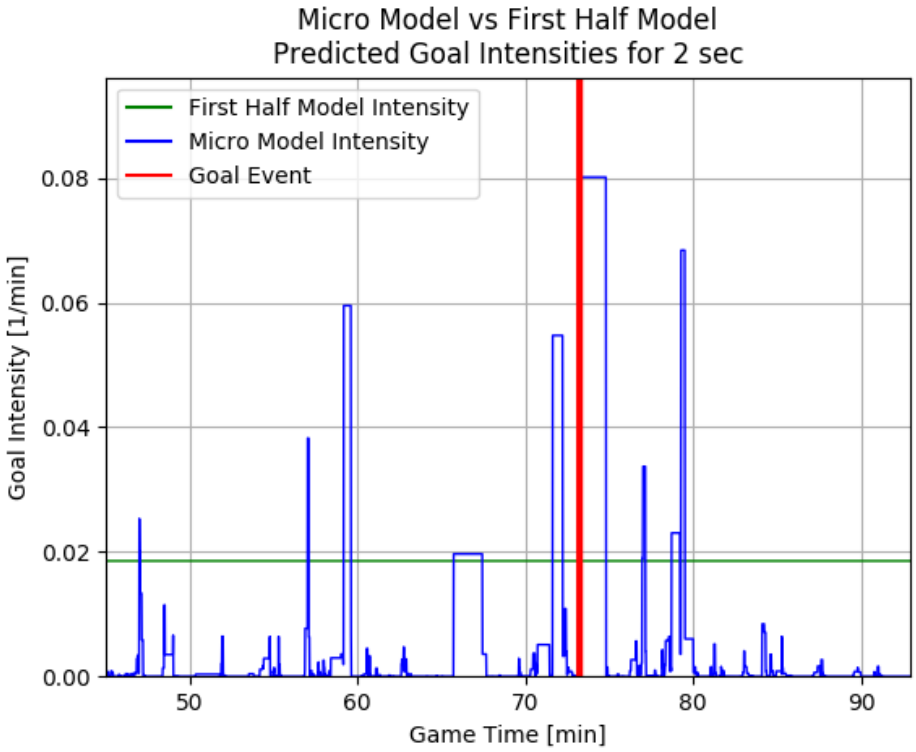


Figure 7.1: Predicted second half goal intensities for the HOME team of the Portugal - Netherlands 2012 UEFA Championship game as a function of time. Microscopic Model intensities were predicted with a delay of 2 seconds. First Half Model intensities were predicted with the best performing Elastic Net model using indicator values from the first half of the game. Actual goal events are show with vertical red lines.

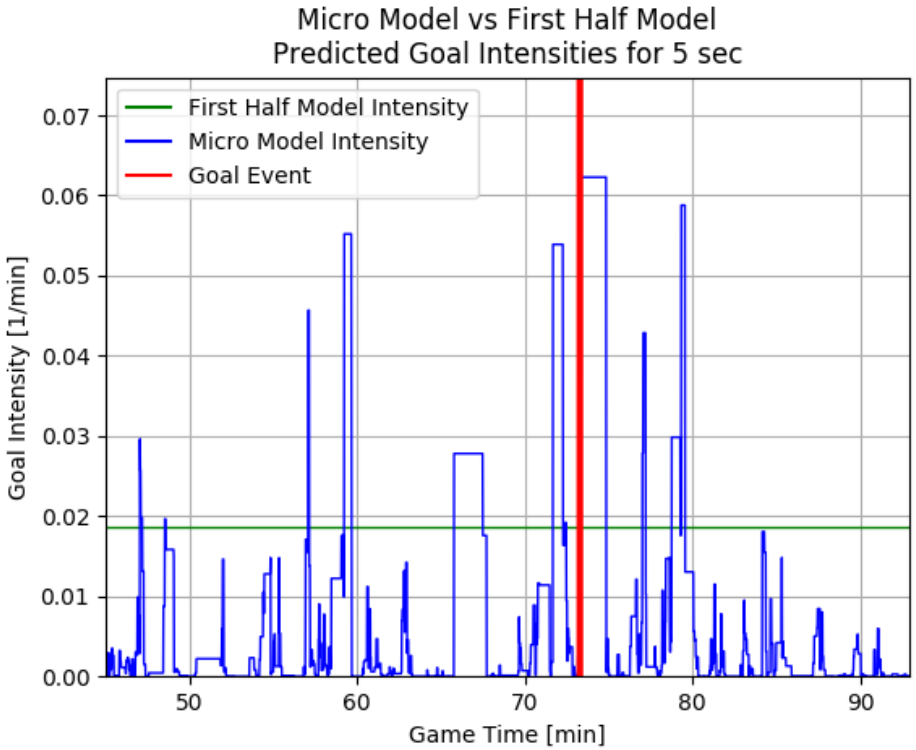


Figure 7.2: Predicted second half goal intensities for the HOME team of the Portugal - Netherlands 2012 UEFA Championship game as a function of time. Microscopic Model intensities were predicted with a delay of 5 seconds. First Half Model intensities were predicted with the best performing Elastic Net model using indicator values from the first half of the game. Actual goal events are show with vertical red lines.

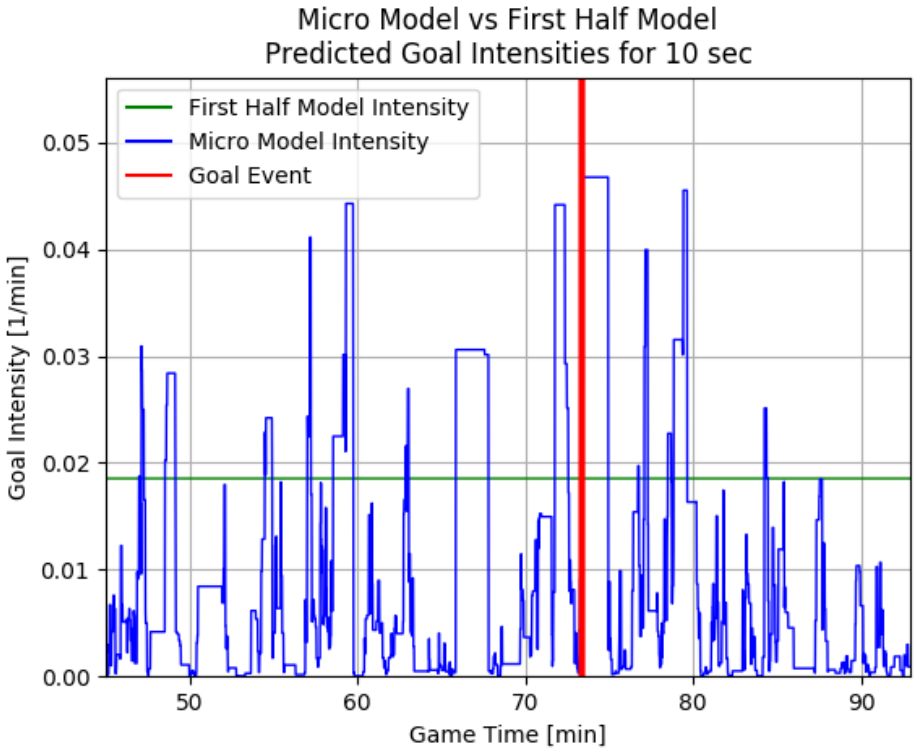


Figure 7.3: Predicted second half goal intensities for the HOME team of the Portugal - Netherlands 2012 UEFA Championship game as a function of time. Microscopic Model intensities were predicted with a delay of 10 seconds. First Half Model intensities were predicted with the best performing Elastic Net model using indicator values from the first half of the game. Actual goal events are show with vertical red lines.

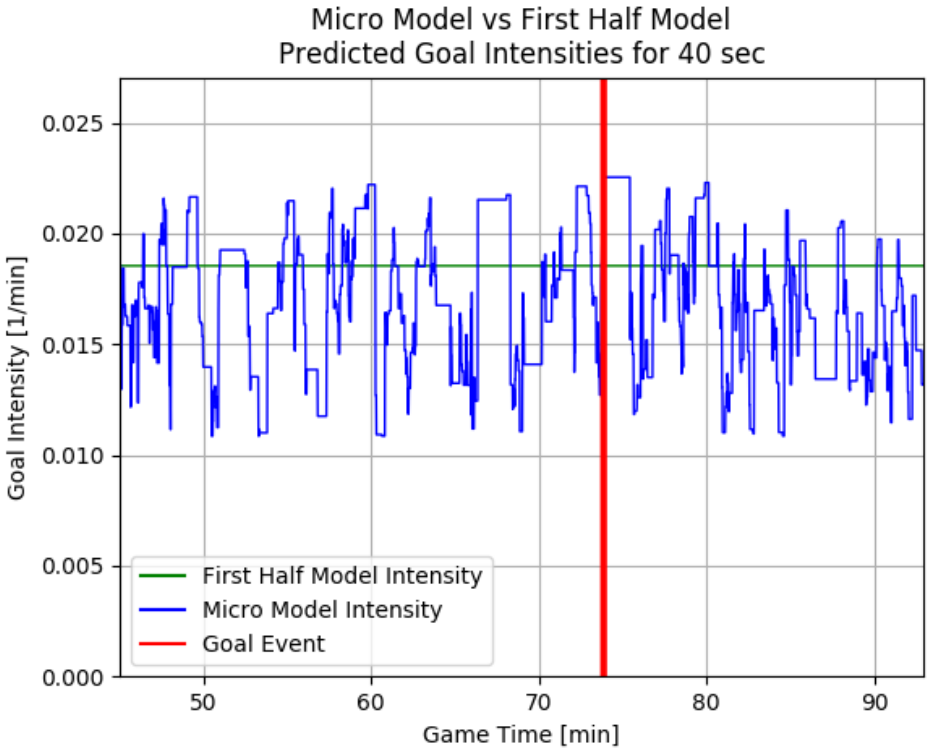


Figure 7.4: Predicted second half goal intensities for the HOME team of the Portugal - Netherlands 2012 UEFA Championship game as a function of time. Microscopic Model intensities were predicted with a delay of 40 seconds. First Half Model intensities were predicted with the best performing Elastic Net model using indicator values from the first half of the game. Actual goal events are show with vertical red lines.

	2	5	10	20	40	55
Euro 2012 - Poland v Greece	2.16	1.65	1.04	0.34	-0.34	-0.59
Euro 2012 - Russia v Czech Republic	5.39	4.76	3.85	2.64	1.25	0.67
Euro 2012 - Netherlands v Denmark	0.97	0.75	0.45	0.12	-0.15	-0.24
Euro 2012 - Germany v Portugal	3.10	2.61	2.06	1.42	0.73	0.46
Euro 2012 - Spain v Italy	4.27	3.66	2.91	1.97	0.96	0.56
Euro 2012 - Rep of Ireland v Croatia	3.11	2.60	2.02	1.32	0.62	0.35
Euro 2012 - France v England	1.35	1.11	0.81	0.49	0.23	0.14
Euro 2012 - Ukraine v Sweden	5.73	4.70	3.52	2.10	0.59	-0.01
Euro 2012 - Greece v Czech Republic	2.68	2.25	1.70	1.01	0.28	0.07
Euro 2012 - Poland v Russia	1.56	1.42	1.06	0.52	-0.08	-0.33
Euro 2012 - Denmark v Portugal	3.87	3.23	2.42	1.44	0.38	-0.04
Euro 2012 - Netherlands v Germany	2.08	1.82	1.43	0.91	0.31	0.06
Euro 2012 - Italy v Croatia	2.91	2.39	1.79	1.08	0.36	0.08
Euro 2012 - Spain v Rep of Ireland	5.51	4.70	3.66	2.37	0.95	0.40
Euro 2012 - Sweden v England	4.47	4.43	3.61	2.37	0.95	0.36
Euro 2012 - Ukraine v France	2.65	2.23	1.61	0.77	-0.23	-0.65
Euro 2012 - Greece v Russia	1.24	1.01	0.77	0.52	0.24	0.13
Euro 2012 - Czech Republic v Poland	2.15	1.72	1.20	0.55	-0.12	-0.37
Euro 2012 - Portugal v Netherlands	1.93	1.50	0.97	0.32	-0.37	-0.63
Euro 2012 - Denmark v Germany	2.80	2.38	1.83	1.13	0.43	0.16
Euro 2012 - Croatia v Spain	3.51	2.93	2.31	1.67	0.96	0.67
Euro 2012 - Sweden v France	4.32	3.61	2.80	1.86	0.82	0.39
Euro 2012 - England v Ukraine	3.94	3.40	2.82	2.15	1.45	1.19
Euro 2012 - Czech Republic v Portugal	3.32	2.83	2.24	1.54	0.80	0.52
Euro 2012 - Germany v Greece	7.82	6.46	4.84	2.91	0.74	-0.16
Euro 2012 - Spain v France	3.19	2.74	2.19	1.50	0.81	0.55
Euro 2012 - England v Italy	1.01	0.79	0.51	0.20	-0.10	-0.21
Euro 2012 - Portugal v Spain	1.12	0.92	0.63	0.23	-0.10	-0.19
Euro 2012 - Germany v Italy	2.65	2.16	1.59	0.95	0.28	0.06
Euro 2012 - Spain v Italy	4.17	3.56	2.83	1.94	0.95	0.56

Table 7.1: Log-likelihood ratios of the Microscopic Model with various delay values in seconds (columns) over the First Half Indicators with the best performing Elastic Net Model for various games of the UEFA Euro 2012 Championship (rows).

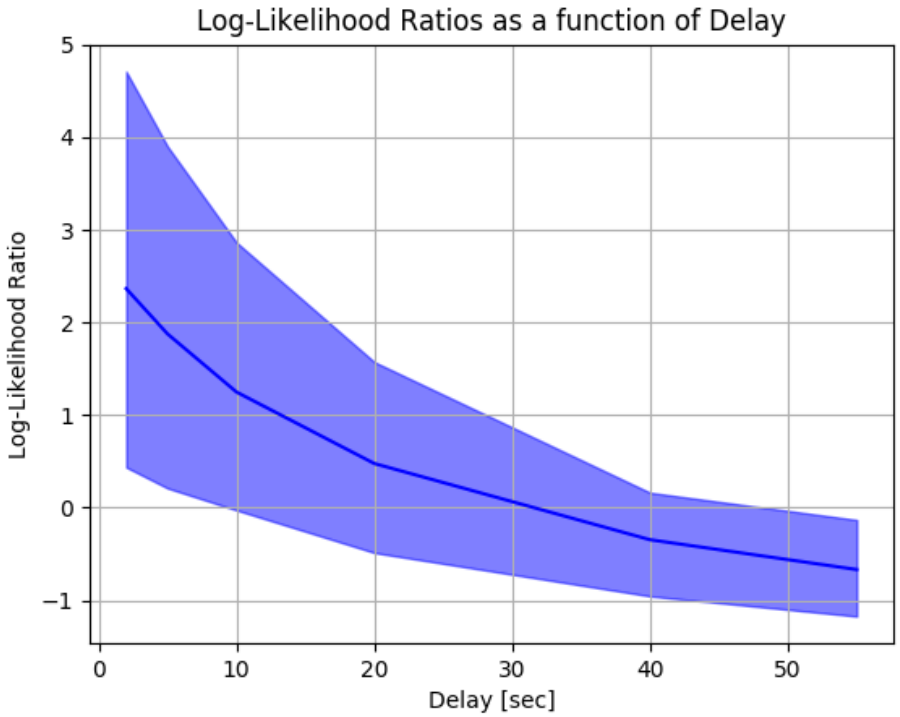


Figure 7.5: Average with 10% to 90% confidence intervals of the Log-likelihood ratios of the Microscopic Model over the First Half Indicators with the best performing Elastic Net Model for a total of 30 games of the UEFA Euro 2012 Championship as a function of Microscopic Model Delay. Note that in case of small delay the Microscopic Model clearly outperforms the First Half Indicators Model because for small delays the state of the game is relevant and this is only taken into account by the Microscopic Model. As the delay increases, the state becomes less relevant and the edge over the First Half Indicators model degrades.

7.5 Summary

In this chapter I performed prediction using the Microscopic Model and the First Half Indicators Model and compared the predictive powers of the two models using log-likelihood ratios. In case of the Microscopic Model, the initial state (number of goals, ball position, team holding the ball) was updated continuously during the second half as the game progressed and the gal intensity was predicted starting from the current initial state into a specific delay time Δt into the future. On the other hand, in case of the First Half Indicators model, the value of the 18-dimensional indicator vector was observed once at the end of the first half, a goal intensity was predicted and the same constant intensity was used during the whole of the second half. Therefore, as expected, for small delay values, up to 30 seconds the Microscopic Model did outperform the First Half Indicators Model. As the delay increased above 30 seconds, the initial state became less relevant and the predictive power of the Microscopic Model over the First Half Indicators Model degraded.

Chapter 8

Conclusions and Further Research

This chapter summarizes the main results of the thesis. The limitations are also being discussed, followed by a set of possible improvements and suggestions for future research.

8.1 Summary

This thesis applies financial mathematics and machine learning to develop models for valuing and risk-managing in-play football bets and to predict the outcomes of football games using in-play data.

Overall we were able to apply existing methods form financial mathematics and to develop new models inspired by existing models used for pricing financial derivatives. We were also able to build predictive models using machine-learning methods.

The findings of the thesis are the following:

The **Constant Intensity Model** in Chapter 3 provides a risk-neutral

hedging and pricing framework for in-play football bets. Two homogeneous Poisson processes have been used to model the goals scored by the two teams. It has been shown that the Fundamental Theorems of Asset Pricing can be extended to this setting and that the market under this model is arbitrage free and complete. Delta-hedging has been demonstrated using in-play market data and was found to perform reasonably well.

The **Local Intensity Model** in Chapter 4 addresses the issue of the so-called intensity smile. The Constant Intensity model prices all bets types with the same intensity and in practice fails to reprice Over/Under bets of different strikes because the implied intensity is not a constant, but rather an increasing function of strike. This is very similar to the strike-dependent nature of the Black-Scholes volatility, that is the volatility smile. I showed that a similar approach that has been used in finance to deal with this effect can also be used in in-play football betting. Specifically, taking Dupire's local volatility model I have shown that introducing a time and score dependent local intensity surface, a consistent model can be constructed that is able to reprice any implied intensity smiles. Dupire's calibration formulae have been derived for the local intensity model and calibration has been demonstrated on in-play betting market data.

The **Microscopic Model** in Chapter 5 attempts to make the next step by modelling the in-play dynamic of not only the goals scored, but two additional and important features as well: the position of the ball and the team holding the ball. These state variables are driven by a set of stochastic differential equations, such that the scores and the team holding the ball are driven by jump processes and the ball position is driven by a Wiener process. In the first part of the chapter the parameters of these processes were allowed to depend freely on all of the state variables and after studying

these generic parameter surfaces I was able to simplify the model greatly by using parametrisations and decrease the number of model parameters to just 13. I then showed that this simplified model can be solved using an efficient semi-analytic approach so that the PDE describing the distribution of the state parameters can be solved on a grid.

The **First Half Indicators Model** in Chapter 6 attempts to forecast the results of the second half of the game using indicators observed at the end of the first half. First I used feature selection methods to select relevant features and the performed model selection to find the model from a set of standard machine learning methods with the highest predictive power. The best performing model was found to be a linear model with Elastic Net regularisation.

In Chapter 7 I compared the predictive powers of the Microscopic Model and the First Half Indicators Model and found that the Microscopic Model is able to outperform the First Half Indicators Model, but only if the delay is under 30 seconds because after this the initial position of the ball and the team holding the ball becomes practically irrelevant.

8.2 Future Research

The thesis undeniably has a number of limitations and there is room for further improvement. A list of potential directions of future research can be summarised as:

- The Local Intensity model in Chapter 4 could be enhanced by:
  - Consider alternative approaches to the constant interpolation of implied intensities from T to all times $t < T$.

  - Extend from one dimension to two so that it can be used to price bets sensitive to the goals scored by individual teams, not only to bets on the total score.

- Consider Stochastic Intensity models where the intensity is driven by a separate stochastic process, in the spirit of the Heston model, such as for example the model suggested by [42].

- The Microscopic Model in Chapter 5 could be enhanced by:

  - Improving the calibration of the model such that the calibration is not a two-step process of first fitting non-parametric surfaces and then fitting the model parameters on these surfaces but rather a one step process that yields model parameters directly.

  - Solving the model in a more efficient way. The current solution is semi-analytic in the sense that some variables can be solved analytically such as the team possessing the ball, but others are solved on a grid using numerical PDE methods such as the distribution of the ball position. A fully analytical solution would improve the computational performance dramatically and therefore would make calibration and solution more efficient, however it is unclear whether this is possible.

- The First Half Indicators model in Chapter 6 could be enhanced by:

  - Apply the classification model in a more dynamic way by using more frequently updated data, for example once every minute, rather than only training the model at the end of the first half.

  - Constructing additional indicators that have potentially higher predictive power.

– Using alternative methods, for example support vector machines or deep recurrent neural networks.

8.3 Closing Remarks

The sports betting industry becomes bigger and bigger every year. Betting providers are increasingly relying on quant skills and there is actually movement between quant roles in the traditional financial sector and betting companies. Although the regulatory and legal framework for sports betting and financial derivatives is vastly different, bets are in many sense sililar to derivatives and the techniques developed for pricing and risk-managing derivatives can be transferred and applied to the betting industry. This technology transfer is useful not only as a pure intellectual exercise, but can have practical benefits by making the betting markets more efficient and thereby serving both the betting companies and retail customers as well.

In this thesis we were able to show that a number of results from financial mathematics and asset pricing can be transferred and applied to in-play football bets. We were also able to developed predictive models based on fundamental in-play data.

Overall, the in-play football betting market proved to be an interesting research topic that provided a unique opportunity to observe the interplay between the fundamental processes driving a market and the prices of tradable assets in the market.

Bibliography

[1] MJ Moroney. Facts from figures. *London, Pelican*, page 3, 1951.

[2] Charles Reep and Bernard Benjamin. Skill and chance in association football. *Journal of the Royal Statistical Society. Series A (General)*, 131(4):581–585, 1968.

[3] Charles Reep, Richard Pollard, and Bernard Benjamin. Skill and chance in ball games. *Journal of the Royal Statistical Society. Series A (General)*, pages 623–629, 1971.

[4] MJ Maher. Modelling association football scores. *Statistica Neerlandica*, 36(3):109–118, 1982.

[5] Mark J Dixon and Stuart G Coles. Modelling association football scores and inefficiencies in the football betting market. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 46(2):265–280, 1997.

[6] Dimitris Karlis and Ioannis Ntzoufras. Analysis of sports data by using bivariate poisson models. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 52(3):381–393, 2003.

[7] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *The Journal of Political Economy*, pages 637–654, 1973.

[8] John C Cox and Stephen A Ross. The valuation of options for alternative stochastic processes. *Journal of Financial Economics*, 3(1):145–166, 1976.

[9] John C Cox, Stephen A Ross, and Mark Rubinstein. Option pricing: A simplified approach. *Journal of Financial Economics*, 7(3):229–263, 1979.

[10] J Michael Harrison and David M Kreps. Martingales and arbitrage in multiperiod securities markets. *Journal of Economic Theory*, 20(3):381–408, 1979.

[11] J Michael Harrison and Stanley R Pliska. Martingales and stochastic integrals in the theory of continuous trading. *Stochastic processes and their applications*, 11(3):215–260, 1981.

[12] Kerry Back and Stanley R Pliska. On the fundamental theorem of asset pricing with an infinite state space. *Journal of Mathematical Economics*, 20(1):1–18, 1991.

[13] D Dyte and SR Clarke. A ratings based poisson model for world cup soccer simulation. *Journal of the Operational Research Society*, pages 993–998, 2000.

[14] SJ Chapman. The kelly criterion for spread bets. *IMA Journal of Applied Mathematics*, 72(1):43–51, 2007.

[15] Elmar Bittner, Andreas Nußbaumer, Wolfhard Janke, and Martin Weigel. Self-affirmation model for football goal distributions. *EPL (Europhysics Letters)*, 78(5):58002, 2007.

[16] Peter Carr and Dilip Madan. Towards a theory of volatility trading. *Option Pricing, Interest Rates and Risk Management, Handbooks in Mathematical Finance*, pages 458–476, 2001.

[17] AD Fitt, CJ Howls, and M Kabelka. Valuation of soccer spread bets. *Journal of the Operational Research Society*, 57(8):975–985, 2005.

[18] Tomas Björk. *Arbitrage theory in continuous time*. Oxford university press, Oxford, 2004.

[19] John C Hull. *Options, futures, and other derivatives*. Pearson Education India, 2006.

[20] Mark HA Davis. Martingale representation and all that. In *Advances in Control, Communication Networks, and Transportation Systems*, pages 57–68. Springer, New York, 2005.

[21] Pierre Del Moral. *Feynman-Kac Formulae*. Springer, New York, 2004.

[22] Freddy Delbaen and Walter Schachermayer. A general version of the fundamental theorem of asset pricing. *Mathematische annalen*, 300(1):463–520, 1994.

[23] Mark Dixon and Michael Robinson. A birth process model for association football matches. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 47(3):523–538, 1998.

[24] Willy Feller. On the integro-differential equations of purely discontinuous markoff processes. *Transactions of the American Mathematical Society*, 48(3):488–515, 1940.

[25] Sheldon M Ross. *Introduction to probability models*. Academic press, Massachusetts, 2006.

[26] Peter Tankov. *Financial modelling with jump processes*. CRC Press, London, 2004.

[27] Pierre Brémaud. *Point processes and queues*, volume 30. Springer, New York, 1981.

[28] David Lando. On Cox processes and credit risky securities. *Review of Derivatives Research*, 2(2-3):99–120, 1998.

[29] David R Cox. Some statistical methods connected with series of events. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 129–164, 1955.

[30] Steven L Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6(2):327–343, 1993.

[31] B. Dupire. Pricing with a smile. *Risk*, 7(1):18–20, 1994.

[32] Steven E Shreve. *Stochastic calculus for finance II: Continuous-time models*, volume 11. Springer, New York, 2004.

[33] GamblingCompliance. In-play tracker - summer 2013. 2013.

[34] Darrell Duffie. Credit risk modeling with affine processes. *Journal of Banking and Finance*, 29(11):2751–2802, 2005.

[35] J Michael Harrison and Stanley R Pliska. A stochastic calculus model of continuous trading: complete markets. *Stochastic processes and their applications*, 15(3):313–316, 1983.

[36] Chi-Fu Huang. Information structure and equilibrium asset prices. *Journal of Economic Theory*, 35(1):33–71, 1985.

[37] Darrell Duffie. *Security markets: Stochastic models*, volume 19821. Academic Press New York, 1988.

[38] John C Cox, Jonathan E Ingersoll Jr, and Stephen A Ross. A theory of the term structure of interest rates. *Econometrica: Journal of the Econometric Society*, pages 385–407, 1985.

[39] Philip K Gray and Stephen F Gray. Testing market efficiency: Evidence from the nfl sports betting market. *The Journal of Finance*, 52(4):1725–1737, 1997.

[40] Linda M Woodland and Bill M Woodland. Market efficiency and the favorite-longshot bias: The baseball betting market. *The Journal of Finance*, 49(1):269–279, 1994.

[41] John Gandar, Richard Zuber, Thomas O'Brien, and Ben Russo. Testing rationality in the point spread betting market. *The Journal of Finance*, 43(4):995–1008, 1988.

[42] Benoit Jottreau. How do markets play soccer? 2009.

[43] Anthony Neuberger. Volatility trading. *London Business School working paper*, 1990.

[44] Emmanuel Derman and Iraj Kani. Riding on a smile. 7(32):9, 1994.

[45] Emanuel Derman, Iraj Kani, and Neil Chriss. Implied trinomial tress of the volatility smile. *The Journal of Derivatives*, 3(4):7–22, 1996.

[46] Jim Gatheral. *The volatility surface: a practitioner's guide*, volume 357. John Wiley & Sons, 2011.

[47] Roger W Lee. The moment formula for implied volatility at extreme strikes. *Mathematical Finance*, 14(3):469–480, 2004.

[48] Douglas T Breeden and Robert H Litzenberger. Prices of state-contingent claims implicit in option prices. *Journal of business*, pages 621–651, 1978.

[49] J Greenhough, PC Birch, Sandra C Chapman, and George Rowlands. Football goal distributions and extremal statistics. *Physica A: Statistical Mechanics and its Applications*, 316(1):615–624, 2002.

[50] Robert F Pawula. *Generalizations and extensions of the Fokker-Planck-Kolmogorov equations*. PhD thesis, California Institute of Technology, 1965.

[51] J. W. Tomas. *Numerical Partial Differential Equations: Finite Difference Methods*. Springer-Verlag, New York, 1995.

[52] Gordon D Smith. *Numerical solution of partial differential equations: finite difference methods*. Oxford university press, 1985.

[53] William H Press. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[54] John L Kelly Jr. A new interpretation of information rate. *Information Theory, IRE Transactions on*, 2(3):185–189, 1956.

[55] Leonard C MacLean, Edward O Thorp, and William T Ziemba. *The Kelly capital growth investment criterion: theory and practice*, volume 3. world scientific, 2011.

[56] Peter Smoczynski and Dave Tomkins. An explicit solution to the problem of optimizing the allocations of a bettor's wealth when wagering on horse races. *Mathematical Scientist*, 35(1), 2010.

[57] George E Uhlenbeck and Leonard S Ornstein. On the theory of the brownian motion. *Physical review*, 36(5):823, 1930.

[58] Kalok C Chan, G Andrew Karolyi, Francis A Longstaff, and Anthony B Sanders. An empirical comparison of alternative models of the short-term interest rate. *The journal of finance*, 47(3):1209–1227, 1992.

[59] Oldrich Vasicek. An equilibrium characterization of the term structure. *Journal of financial economics*, 5(2):177–188, 1977.

[60] Amy R Ward and Peter W Glynn. Properties of the reflected ornstein–uhlenbeck process. *Queueing Systems*, 44(2):109–123, 2003.

[61] LM Ricciardi and L Sacerdote. On the probability densities of an ornstein-uhlenbeck process with a reflecting boundary. *Journal of Applied Probability*, pages 355–369, 1987.

[62] Gareth O Roberts, Omiros Papaspiliopoulos, and Petros Dellaportas. Bayesian inference for non-gaussian ornstein–uhlenbeck stochastic volatility processes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(2):369–393, 2004.

[63] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[64] Hirotogu Akaike. Information theory and an extension of the maximum likelihood principle. In *Selected Papers of Hirotugu Akaike*, pages 199–213. Springer, 1998.

[65] Hirotugu Akaike. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974.

[66] Hirotugu Akaike. Prediction and entropy. In *Selected Papers of Hirotugu Akaike*, pages 387–410. Springer, 1985.

[67] Samuel S Wilks. The large-sample distribution of the likelihood ratio for testing composite hypotheses. *The Annals of Mathematical Statistics*, 9(1):60–62, 1938.

[68] Lutz Prechelt. Automatic early stopping using cross validation: quantifying the criteria. *Neural Networks*, 11(4):761–767, 1998.

[69] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[70] Lasse Holmstrom and Petri Koistinen. Using additive noise in back-propagation training. *IEEE Transactions on Neural Networks*, 3(1):24–38, 1992.

[71] Tomaso Poggio and Federico Girosi. Networks for approximation and learning. *Proceedings of the IEEE*, 78(9):1481–1497, 1990.

[72] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth*

International Conference on Artificial Intelligence and Statistics, pages 249–256, 2010.

[73] Tomaso Aste and T Di Matteo. Sparse causality network retrieval from short time series. *Complexity*, 2017, 2017.

[74] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arxiv. org, 2014.

[75] Leo Breiman et al. Arcing classifier (with discussion and a rejoinder by the author). *The annals of statistics*, 26(3):801–849, 1998.

[76] Antonio D'Ambrosio and Valerio A Tutore. Conditional classification trees by weighting the gini impurity measure. In *New Perspectives in Statistical Modeling and Data Analysis*, pages 273–280. Springer, 2011.

[77] Evelyn Fix and Joseph L Hodges Jr. Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, California Univ Berkeley, 1951.

[78] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[79] John Frank Charles Kingman. *Poisson processes*, volume 3. Clarendon Press, 1992.

[80] Elijah Polak. *Optimization: algorithms and consistent approximations*, volume 124. Springer Science & Business Media, 2012.

[81] John Aldrich et al. Ra fisher and the making of maximum likelihood 1912-1922. *Statistical science*, 12(3):162–176, 1997.

[82] RA Fisher. On an absolute criterion for fitting frequency curves. *Statistical Science*, 12(1):39–41, 1997.