Predicting the Outcomes of Professional Tennis Matches

Joss Peters

Master of Science
School of Informatics
University of Edinburgh
2017

Abstract

This project explores several probabilistic models for forecasting the outcomes of professional tennis matches. These models focus on addressing the effects of surface type and the variation of player skills through time. The models are trained and evaluated on an historical data set of approximately 45,000 tennis matches between 2000 and 2017. Within the models, free parameters are used to represent the skills of players and the characteristics of court surfaces. These parameters are fitted based upon the historical data using both maximum likelihood and variational approximate inference. The performance of the models is shown to be superior when compared against existing tennis prediction models from the literature. The results show that surface effects and the variation of player skills through time can be modelled more accurately using new approaches which have previously not been applied to tennis.

Acknowledgements

I am extremely grateful to my supervisor, Iain Murray, for his invaluable knowledge and technical guidance throughout the project. I would also like to give a special mention to my brother, Jools, for all the help he provided on the project. Finally, I would like to thank all of my family and friends who have supported me throughout my MSc.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Joss Peters*)

Table of Contents

1 Introduction **1**

2 Background **3**

2.1 The Game of Tennis 3

2.1.1 Court Surfaces 4

2.2 The Data Set 4

2.3 Betting in Tennis 6

2.3.1 Betting Odds and Implied Probability 6

2.3.2 Betting Strategies 7

2.4 Making and Scoring Predictions 7

2.4.1 Confidence of Predictions 7

2.4.2 Scoring Rules 8

2.4.3 Calibration 9

3 Review of Previous Work **10**

3.1 Point models 10

3.2 Regression and Neural Network Models 10

3.3 Pairwise Comparison Models 11

3.4 Discussion on Approaches 12

4 Models **13**

4.1 Modelling a Tennis Match using a Markov Chain 14

4.2 Baseline Point Model 16

4.3 Probabilistic Models 16

4.4 Vanilla Bradley-Terry Model 18

4.5 Free Parameter Point Model 18

4.6 Surface Factor Model 20

4.7 Time Series Model 20

5 Implementation 22

5.1 Evaluation Approach 22

5.1.1 Training and Test Set 23

5.2 Weighting Matches 24

5.2.1 Recency Weighting 24

5.2.2 Surface Weighting 24

5.2.3 Weighting in Probabilistic Models 25

5.2.4 Normalising Point and Game Outcomes 25

5.3 Fitting the Model Parameters 26

5.4 Penalised Maximum Likelihood Fitting 26

5.5 Approximate Inference Fitting 27

5.5.1 Variational Inference 28

5.5.2 Entropy 29

5.5.3 Cross-Entropy 29

5.5.4 Likelihood 29

5.5.5 Making Predictions using the Approximate Posterior 31

5.5.6 Parameter Initialisation 32

5.6 Fitting the Time Series Model 32

5.6.1 Joint Optimisation Model 33

5.6.2 Filtering Model 33

5.7 Confidence of Predictions 34

6 Results and Discussion 35

6.1 Baseline Results 36

6.1.1 Naive Predictions 36

6.1.2 Predictions using ATP rankings 36

6.1.3 Bookmakers 37

6.1.4 Point Model Results 37

6.1.5 Comparison of Baseline Calibration 37

6.2 Bradley Terry Model Results 38

6.2.1 Maximum Likelihood Fitting 38

6.2.2 Bayesian Fitting 44

6.3 Free Parameter Point Model Results 48

6.3.1 Maximum Likelihood Fitting 48

6.3.2 Bayesian Fitting 49

6.3.3 Comparison to Bradley-Terry Point Level Model 50

6.4 Surface Factor Model Results 51

6.5 Joint Optimisation Time Series Model Results 53

6.6 Filtered Time Series Model Results 56

6.7 Final Model and Test Set Results 59

6.8 Improving Profit 62

7 Conclusion **64**

7.1 Future Work 65

7.1.1 Expanding the Data Set 65

7.1.2 Combing the Surface Factor and Time Series Models 66

7.1.3 Additional Factors 66

A Appendix **67**

A.1 Variational Inference Likelihood Approximation 67

Bibliography **68**

Chapter 1

Introduction

In recent years, applying statistical analysis to sport has been rapidly growing to meet demand from coaches, media and gambling. The ability to accurately predict the outcome of a sporting event is something which many within the sporting world are fascinated by. For example, predictions can allow a media channel to provide more insightful coverage of a sporting tournament or allow tournament organisers to better match the skills of players giving closer and more exciting matches. Alternatively, the prediction models themselves can reveal something interesting about the characteristics of different playing styles, making them useful for coaching purposes. In recent years, the growth of online betting exchanges, such as Betfair, also provides a further and increasingly relevant motivator for research into prediction models. The large amount of historical data freely available, makes the sport of Tennis makes an appealing candidate for research.

The majority of existing research into tennis modelling focuses on pre-match prediction, where the goal is to predict the probability of either player winning prior to the match commencing. This is in contrast to in-play prediction, where the goal is to predict how the winning probabilities evolve during a match. The work presented in this paper focuses solely on the former of these two goals.

One of the most popular approaches to tennis modelling is hierarchical match models, such as those described by Knottenbelt et al. (2012) or Barnett and Clarke (2005). These models make use of the structure of the scoring system in tennis and model a match as a Markov chain with transition probabilities derived from historical player service statistics. Other research has explored the question of whether player ATP ranking points (the official tennis ranking system) are an effective basis for making predictions (Clarke and Dyte, 2000). Further work has demonstrated how alternative rating based models can be applied to tennis which give superior predictive performance in comparison to the ATP rankings (McHale and Morton, 2011). There is also wider research in the area of rating systems in other fields much of which could be applied to Tennis, for example Herbrich et al. (2007) or Glickman (2001). Other approaches to tennis prediction aim to utilise machine learning by creating sets of player related features and then fitting neural network or regression models. Sipko and Knottenbelt (2015) develop such a model, comparing it to hierarchical match models and claiming superior performance.

The goal of this project is to develop a series of models which will improve upon current state of the art models in tennis prediction. The models will aim to explicitly address two key factors within tennis modelling. Firstly, the effect of different court surface types and secondly, the variation of players skills through time. The paper begins with a review of current prediction models from literature in order to understand the strengths and weakness of the different approaches (Chapter 3). Following this, a description of the models trained during this project is provided, along with any relevant theoretical material and details of their implementation (Chapter 4 & 5). The results and findings of the project are detailed in Chapter 6, before conclusions and suggestions for future work are presented in Chapter 7.

Chapter 2

Background

2.1 The Game of Tennis

Tennis is a racquet sport which can be played both as singles (one vs one) and doubles (two vs two). Due to time constraints and inline with the majority of existing research, the scope of this project is restricted to modelling men's singles tennis. Tennis has a hierarchical structure: At the lowest level players compete for points which they accumulate in order to win games. Games are then accumulated in order to win sets and finally sets are accumulated in order to win the overall match. A point in tennis consists of one player (known as the server) serving to the opposing player (known as the receiver) in order to start a rally. The winner of the rally is then awarded the point, or in cases where the server fails to produce a valid serve in two attempts, it is awarded by default to the receiving player. To win a game a player must accumulate at least 4 points and at least two more than their opponent. Points in tennis are biased in favour of the server who has an attacking advantage. Typically the server, will win more than 60% of all points. As serving is rotated on a game by game basis this means that players are also strongly favoured to win games in which they are the serving player. There are no draws in tennis and matches are played as either best of 3 or best of 5 sets depending upon the tournament. Full information on the official rules of Tennis are published by the International Tennis Federation and can be found on-line [1].

Throughout the year, tennis players compete in a range of knock out format tournaments and are awarded ranking points for placing in these tournaments. These ranking

[1] http://www.itftennis.com/officiating/rulebooks/rules-of-tennis.aspx

points are then used to generate the official tennis rankings and determine qualification for future tournaments. The top tier of men's tennis is the ATP tour, which currently consists of 68 tournaments per year of varying degrees of prestige. Tournaments are typically played over the space of one or two weeks and several tournaments may happen simultaneously. The second tier of men's tennis is the Challenger tour which is of similar format to the ATP tour but at a lower level. The pool of players is continuous between the Challenger and ATP circuits. For example, a player may be competing in a mix of the top Challenger tournaments and low level ATP tournaments.

2.1.1 Court Surfaces

One important factor in tennis is that matches are played on a variety of different surfaces types: clay, carpet, hard court or grass. Each of these surface types affects the bounce of the ball slightly differently, which in turn influences the type of playing styles that are most successful on that surface. For example, clay due to its slower bounce typically results in points with long baseline rallies thus favouring players who are strong in that style. Accounting for surface type is therefore a critical aspect of tennis modelling.

2.2 The Data Set

The data set used for this project consists of approximately 45,000 men's singles ATP tour matches between the years 2000 and 2017. This data is freely available under a non commercial licence [1] and is obtained from GitHub [2] (Credited to Jeff Sackmann). Table 2.1 (page 5) summarises the key information which is contained about each match in the data. The data set also contains further side information such as player handedness, tournament seeding, tournament round, player age and length of match. The full list is omitted as none of this additional information is used by the models in this project. It is also possible to obtain more detailed historical information about each match. For example, point by point data is available showing the sequence in which points were won or lost within a match rather than just the overall totals. Data of this sort is available on-line [3] however modelling ordered sequences of points is not

[1] Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License - https://creativecommons.org/licenses/by-nc-sa/4.0/

[2] Jeff Sackman/ Tennis Abstract - http://www.tennisabstract.com, https://github.com/JeffSackmann

[3] For example, https://www.tennisbetsite.com/results.html

Table 2.1: Match Information

Description of Information
Date of the match
Name of the winning and losing player
The match score
Name of tournament
The surface type
Number of aces served in the match by each player
Number of double faults made by each player
Number of points served by each player
Number of valid 1st serves by each player
Number of 1st serve points by each player
Number of 2nd serve points won by each player
Number of games served by each player
Number of break points faced on serve by each player
Number of break points saved by each player
The ATP ranking points of each player

addressed within this project.

2.3 Betting in Tennis

Bets in tennis can primarily be placed in two types of markets: bookmakers or betting exchanges. The former is where a bookmaker offers odds and accepting customers place money directly against the bookmaker at these odds. In a betting exchange, customers instead offer odds and place bets against each other, with the exchange simply taking a small commission for each paired bet. Typically, more favourable odds can be found on exchange markets, however as there is limited historical data on these odds the models in this project will be compared against traditional bookmakers odds.

It is possible to place bets on a wide variety of events relating to different aspects of a tennis match, both before it commences and while it is being played. However, this project focuses only on bets placed on the overall outcome prior to the match starting.

2.3.1 Betting Odds and Implied Probability

Betting odds can be represented in either decimal or fractional format. Only decimal format is used in this research, however it is straightforward to convert between the two. Decimal format odds are simply a single number greater than 1 (for example 1.87). For a successful bet, the money received is given by $stake \times odds$. This is inclusive of the stake meaning profit is given by $stake \times (odds - 1)$. Odds can be used to infer an implied probability of the outcome in question happening:

$$p = \frac{1}{odds} \tag{2.1}$$

This represents the underlying probability that should exist in order for the odds to be completely fair. In an unbiased scenario, the implied probabilities from the odds of all possible outcomes of an event should sum to 1. However, for bookmaker odds the implied probabilities will almost always sum to over 1 as the bookmaker's have a built in margin that allows them to make long term profit. This discrepancy is known as over-round. For several of the performance metrics used in this project, the over-round is corrected by normalising the probabilities when comparing against the bookmaker odds.

2.3.2 Betting Strategies

Betting strategies aim to exploit cases where odds are undervalued. This is cases where the implied probability from the odds is less than the actual probability of the event in question happening. If one has a predictive model, a simple betting strategy is to place a unit bet whenever the model indicates that the offered odds are undervalued. Alternatively, one can use a more complex strategy, such as the Kelly criterion (Kelly, 1956). In this strategy the quantity of the bet is varied according to some function of the difference between the implied and actual probabilities. Of course, the success of any such strategy depends upon the accuracy of the models predictions in comparison to the accuracy of the implied probabilities from the odds themselves.

2.4 Making and Scoring Predictions

This project focuses on making probabilistic predictions on the overall outcome of each match. The predictions therefore take the form of a single value P_w, which is the probability assigned by a given model to the winning player. Since there are no draws and the probability of the losing player can be inferred as $P_l = 1 - P_w$, then a single value suffices for all predictions.

2.4.1 Confidence of Predictions

Probabilistic predictions, by their nature, account for uncertainty in classification tasks. However, in this case, it is essential to also consider the confidence of the probability value itself. To demonstrate why this is important consider a model predicting a winning probability of close to 0.5: In one case this could be due to the fact the model is confident about the skills of both players and expects an extremely tight match. In another case, the model may have almost no information about the skills of both players and simply output a probability to reflect this. Clearly there is big difference between these two cases since the later is likely to perform poorly if used in a betting strategy where the bookmakers have access to information that the model did not. It is therefore desirable to have a means to access the confidence of the probability values predicted by the models.

2.4.2 Scoring Rules

Scoring rules are metrics which measure the accuracy of probabilistic predictions. The performance of models in this project are measured according to the following four scoring rules:

Classification Accuracy m_1

The percentage of matches where the winning player is assigned a probability of greater than 0.5. This metric is useful for providing an overall gauge of performance but doesn't address the quality of the probabilities themselves.

Average Probability m_2

The average probability assigned to the the winning player, given by:

$$m_2 = \frac{1}{N}\sum_{i=1}^{N} P_w{}^i, \tag{2.2}$$

where $P_w{}^i$ is the probability assigned by the model to the winning player of the i^{th} prediction and N is the total number of predictions.

Average Log Probability m_3

The average natural logarithm of the probability assigned to the winning player, given by:

$$m_3 = \frac{1}{N}\sum_{i=1}^{N} \log(P_w{}^i). \tag{2.3}$$

This metric is common in machine learning and probabilistic modelling and relates directly to a logistic loss cost function. This metric is important because it is the only one here which satisfies the mathematical criteria required to make it a proper scoring rule. Essentially, this means that the maximum score can be achieved only by predicting the true underlying probabilities. Proper scoring rules and their implications are addressed in detail by Gneiting and Raftery (2007).

Return on investment m_4

The return on investment evaluated against historical bookmaker odds. Based on a simple gambling strategy of placing a unit bet whenever the model predicts a probability greater than the implied probability from the bookmaker odds. This metric

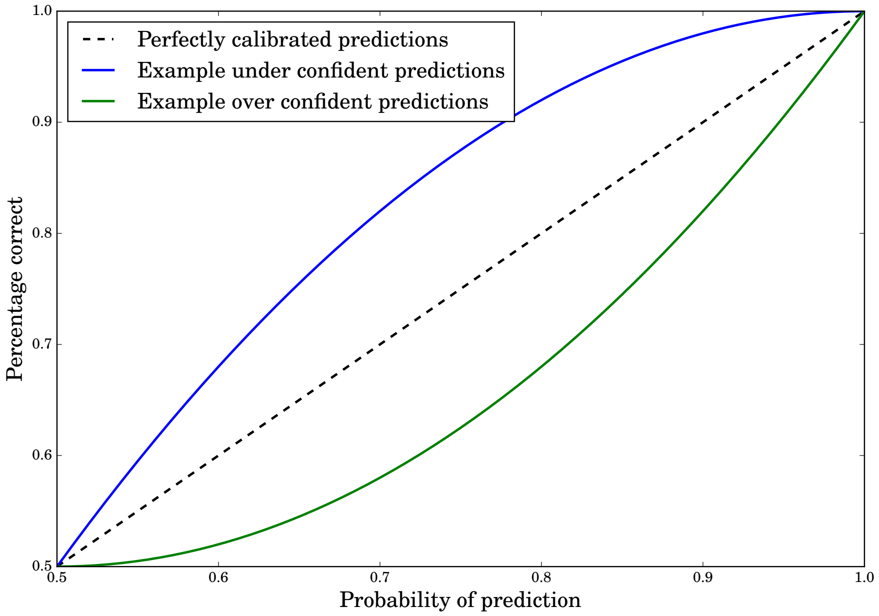

Figure 2.1: Example calibration plot. The horizontal axis shows relates to the models predictions. The vertical axis shows the percentage of time predictions of that probability were correct.

provides a comparison against the bookmaker predictions and is also entertaining to consider. However, more emphasis is placed on the other metrics as they have stronger theoretical basis.

2.4.3 Calibration

Calibration is the idea that a probabilistic prediction of 0.7 should be correct approximately 70% of the time. If, for example, a model makes thousands of predictions between 0.6 and 0.7 but only 50% of these were actually correct then clearly the model is poorly calibrated. Furthermore, these predictions can be described as over confident since they turned out to be successful a lower percentage of the time than their probabilities implied. A reverse scenario would describe under confident predictions. Examining calibration can be useful in order to better understand the behaviour and biases of different models. A calibration graph is a graph which shows a models predictions against the percentage of time they were actually correct. Figure 2.1 demonstrates some example calibration plots.

Chapter 3

Review of Previous Work

The majority of approaches to tennis match prediction fall into three categories: hierarchical/point models, paired comparison models, or regression/neural network models.

3.1 Point models

Point or hierarchical models focus on estimating the probability of players winning an individual point within a tennis match. Match winning probabilities are then derived using the assumption that points are independent and identically distributed (IID). This assumption has been shown to be incorrect but is argued to be a good approximation (Klaassen and Magnus, 2001). Historical player service statistics can be used to calculate point winning probabilities using simple equations (Barnett and Clarke, 2005). A short-coming of this approach is that the calculations suffer from bias because players face different opponents of varying skill levels. Knottenbelt et al. (2012) develop a common opponent averaging method in their point model aimed at addressing this issue. In this method, service probabilities are calculated using a specific subset of historical data, containing only matches where both players being modelled have played against the same opposing player.

3.2 Regression and Neural Network Models

Models in this category use sets of features to describe the characteristics of each player. A function is then used which takes the features of two players as an input and outputs the probability of one player winning. The function in question is a regression or neural network model with parameters learned through training on historical data. Early work in this area used regression models with ranking points or seeding as the main feature describing each player (Boulier and Stekler, 1999) (Clarke and Dyte, 2000). Additional features being side information such as player handedness, height, age and head to head wins. Recent work by Sipko and Knottenbelt (2015), has shown it is effective to incorporate features focusing on past playing statistics such as percentages of serves won, aces and double faults. These percentages can be calculated using the methods applied in point models, for example using the common opponent averaging approach.

3.3 Pairwise Comparison Models

Pairwise comparison refers to a process of comparing entities in pairs to determine which entity is preferred. A Bradley-Terry model (Bradley and Terry, 1952) is a popular method for pairwise comparison which has been applied to Tennis by McHale and Morton (2011). In this approach, each player is assigned a single positive free parameter representing their overall skill. The probability that one player beats another is then given by the simple relationship:

$$P_{ij}(r=1) = \frac{s_i}{s_i + s_j}$$

Where $P_{ij}(r=1)$ is the probability the player i wins against player j and s_i and s_j are the respective skill parameters of both players. Optimisation can be used to jointly solve for skills parameters of all players based on a fixed period of historical results. Although a Bradley-Terry model can be applied directly to the outcome of matches in Tennis, it has been shown to be more effective to instead model game outcomes (McHale and Morton, 2011). Match winning probabilities can then be derived based on the same IID assumption used in point models.

Rating systems such as ELO are also based on a similar underlying relationship to that of a Bradley-Terry model. However, these systems typically apply updates to players skills after every match making the skills dynamic through time. This is distinctly different to what is described above where skills are jointly optimised within a fixed period of time and assuming the skills to be constant within that period. Wider research in the area of rating systems has mostly been outside the domain of Tennis but could equally be applied here. For example, work on Microsoft's True Skill rating system (Herbrich et al., 2007).

3.4 Discussion on Approaches

Current neural network and hierarchical based models use input features such as player serve and ace probabilities. These features are not known, but are estimated by averaging various historical statistics. This contrasts with the process involved in the Bradley-Terry model implemented by McHale and Morton (2011), whereby player related free parameters are learned by the model. McHale and Morton only apply their model to game level data and thus unlike other models do not utilise lower level point information. However, the ideas behind their model could easily be extended to point level information. This would provide an alternative way to estimate many features used as inputs to other models. For example, different Bradley-Terry models could be trained to predict features such player ace, double fault and service percentages. One weakness of a Bradley-Terry model is that it does not allow for non-transitive relationships between players. This issue can be addressed by modelling players using multiple free parameters and has been explored in work outside of Tennis (Stern et al., 2009) (Stanescu, 2011).

Almost all of the tennis prediction models described so far share the same strategy in dealing with two important factors: The variation of players skills through time and surface effects. That strategy being to weight or filter matches in the input data according to their relevance. Matches that have been played most recently and on the same surface being classed as most relevant. Weighting or filtering by surface has been shown to provide improved performance compared to treating all surfaces the same (Sipko and Knottenbelt, 2015). However, it doesn't allow the model to learn characteristics of the surfaces themselves which may limit its ability to generalise between surface types. For example, in cases where the majority of the data for a given player is on one particular surface.

Chapter 4

Models

This chapter provides part of the technical description for models trained during this project and is structured as follows:

- Section 4.1 explains how a single tennis match can be modelled using a Markov chain. This is a component of models and is used to convert predictions of point or game winning probabilities into match winning probabilities.
- Section 4.2 describes a model from the literature which is used as a baseline for the project. This model estimates point winning probabilities by averaging some simple statistics from historical tennis matches. These are then converted to match winning probabilities using the Markov chain.
- Section 4.3 provides general discussion on probabilistic models. Its purpose is to provide clarity for the information presented in Sections 4.4 to 4.7.
- Section 4.4 describes the Bradley-Terry model, the simplest probabilistic model of match outcomes. We also apply it to model point and game outcomes and then use the Markov chain to convert the point or game predictions into match predictions.
- Sections 4.5 and 4.6 describe two models which extend the ideas of a Bradley-Terry model in order to model surface effects and distinct service probabilities. This is achieved by incorporating additional surface and player related parameters.
- Section 4.7 describes a method to model changes in players abilities over time. This is also based on the Bradley-Terry model, but considers the parameters of the model as changing over time rather than fixed.

4.1 Modelling a Tennis Match using a Markov Chain

A tennis match can be modelled using a Markov chain where the states correspond to the score and transition probabilities to the probability of a point being won or lost. Closed form equations can then be derived, which provide a way to convert point, game or set outcome probabilities into match outcome probabilities. This idea has been used heavily in previous work on Tennis modelling (e.g. Newton and Keller (2005)) and is also used within many of the models in this project. Figure 4.1 shows an explicit example of a Markov chain for a single game. This corresponds to the following closed form equation which relates the probability of winning a point to the probability of winning a game:

$$P_{game} = p_s^4 + 4p_s^4(1-p_s) + 10p_s^4(1-p_s)^2 + 20\frac{p_s^5(1-p_s)^3}{1-2p_s(1-p_s)}$$

Where P_{game} is the probability of the server winning the game and p_s is the probability of the server winning a point. Similar equations can be derived for set, tie break and match outcomes.

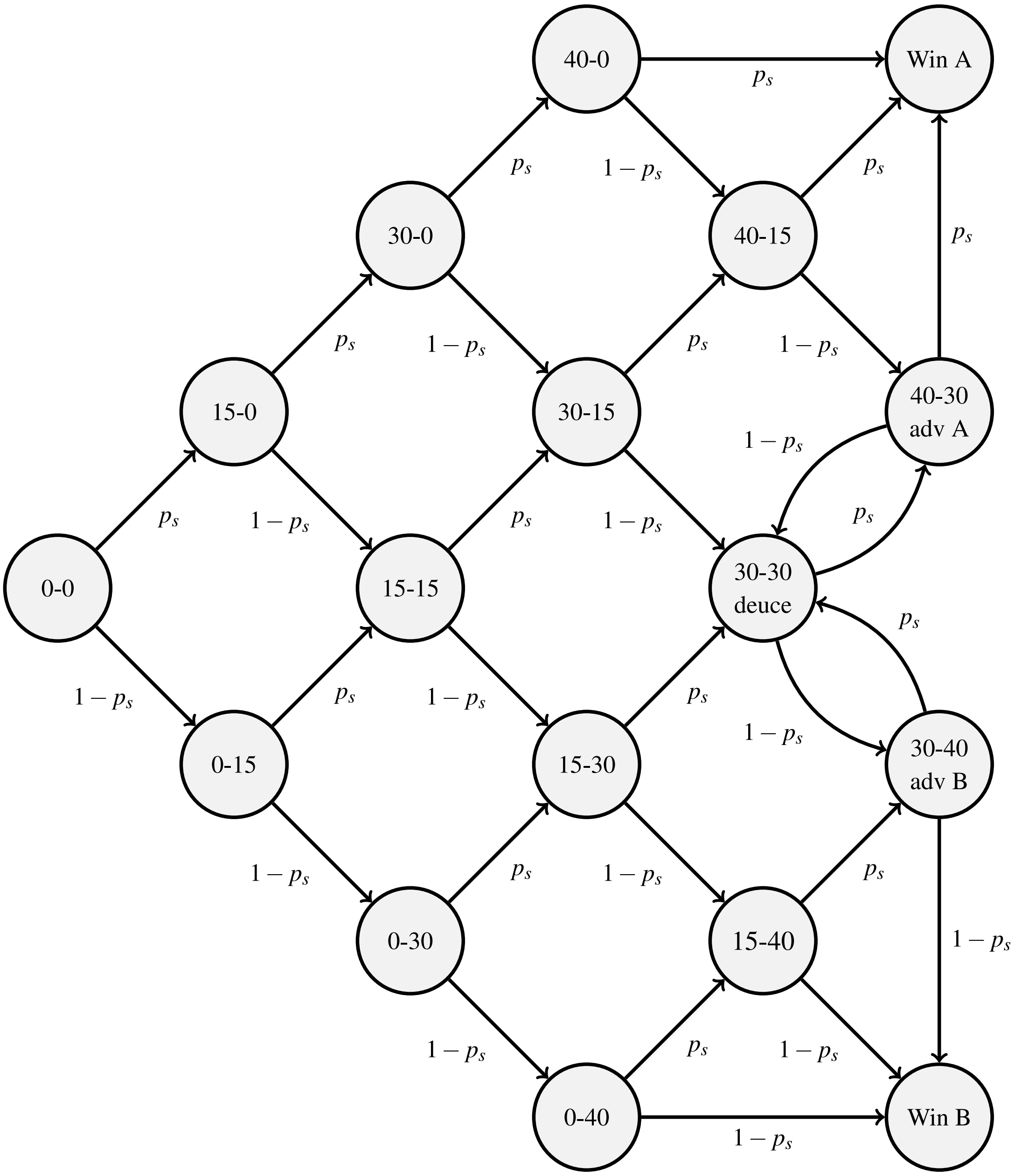

Figure 4.1: Markov Chain model of a Tennis Game. Adapted from Sipko and Knottenbelt (2015).

4.2 Baseline Point Model

The point model described by Barnett and Clarke (2005) is implemented as a baseline for comparing the performance of later models. This model predicts match winning probabilities by converting service probabilities using the Markov chain described in the previous section. Barnett and Clarke provide equations for estimating the service probabilities based on combining simple averages from historical data:

$$f_{ij} = f_t + (f_i - f_{av}) - (g_j - g_{av})$$

$$f_i = a_i b_i + (1 - a_i) c_i$$

$$g_i = a_{av} d_i + (1 - a_{av}) e_i$$

Where f_{ij} is the probability of player i winning a point on their serve against player j and:

a_i	percentage of first serves in play for player i
b_i	percentage of points won on first serve given that first serve is in for player i
c_i	percentage of points won on second serve for player i
d_i	percentage of points won on return of first serve for player i
e_i	percentage of points won on return of second serve for player i
f_i	percentage of points won on serve for player i
g_i	percentage of points won on return for player i
a_{av}	tour average first serve percentage across all players
f_{av}	tour average percentage of points won on serve across all players
g_{av}	tour average percentage of points won on return across all players
f_{av}	tournament average percentage of points won on serve

All of these percentages (a-g) are set based on observed fractions from previous matches. Barnett and Clarke use percentages provided by the ATP which are based on the 70 most recent previous matches. However, in this project the percentages are calculated as a weighted average from 3 years of previous data. The weighting given to each data point in the calculation is determined by an exponential recency function which is discussed in Section 5.2.

4.3 Probabilistic Models

The models that we will discuss in Sections 4.4 to 4.7 can all be classed as probabilistic models. These models share a common methodology and structure which is reviewed in this section. The models can be broken down as consisting of:

- Observed variables. These are the outcomes of points, games or matches. A single outcome will be given the notation r and a set of outcomes D.
- Unobserved variables. These are the parameters of the model itself which are chosen as part of the model design. These will be represented generally by the notation w.
- A relationship defining the probability of observing an outcome given the parameters of the model $P(r|w)$. This is also chosen as part of the model design.
- A prior probability distribution on the parameters of the model $P(w)$. A Gaussian or uniform prior is used in all of the models in this project.

Given the parameters of the model, predictions can be made for future outcomes based upon the relationship $P(r|w)$. However, as the parameters are unknown, the goal of the process is to infer them based upon a set of observed results. Predictions can then be made based upon their inferred settings. For some data D, on a set of n outcomes, the posterior probability of the model parameters can be expressed as:

$$P(w|D) = \frac{P(D|w)P(w)}{P(D)}.$$

$P(D|w)$ is the likelihood of the model given the data and $P(D)$ is marginal likelihood of the model found through normalisation. Under the assumption that outcomes are independent, the likelihood can be expressed as:

$$P(D|w) = \prod_{i=1}^{N} P(r^i|w).$$

A prediction for a new match r^{n+1} can be made by taking the expectation of $P(r^{n+1}|w)$ with respect to the posterior distribution $P(w|D)$. However, for the models in this project determining this expectation and the normalisation constant of the posterior distribution $P(w|D)$ requires doing integrals which are not tractable. Instead, two separate methods will be used in order to obtain approximate results. The first method is to use a point estimate of model parameters at their most probable settings. This method is referred to as penalised maximum likelihood fitting. The second method is to approximate the full posterior distribution using an approximate inference technique. This is referred to as the Bayesian approach.

In this paper, the probabilistic models are discussed in two parts. The first part, which is discussed in this chapter, covers the model design, its parameters and how predictions can be made given the parameters. The second part covers how the parameters of the model are fitted from the data using the two different methods mentioned above. This is covered in Chapter 5.

4.4 Vanilla Bradley-Terry Model

The second model implemented is the Bradley-Terry model described by McHale and Morton (2011). This is used as a further baseline, but is also extended to predict point level outcomes. Additionally, McHale and Morton only explore fitting the parameters of their model using maximum likelihood, whereas in this project the parameters are also fitted using a Bayesian approach. The model is based on the previously given relationship:

$$P(r_{ij} = 1|s_i, s_j) = \frac{s_i}{s_i + s_j}. \tag{4.1}$$

Where $r_{ij} = 1$ indicates a win for player i against player j and s_i and s_j are the player skills which are the parameters of the model. Equation 4.1 can be re-parametrised as follows in order to constrain the skills to only have positive values:

$$P(r_{ij} = 1|s_i, s_j) = \frac{e^{s_i}}{e^{s_i} + e^{s_j}} = \frac{1}{1 + e^{-(s_i - s_j)}} = \sigma(s_i - s_j), \tag{4.2}$$

where σ is the logistic sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$. For some data consisting of a set of results $D = \{r^1, ...r^n\}$ for a set of players $M = \{1, ..., m\}$ with skills $S = \{s_1, ..., s_m\}$ the likelihood of the model given the data can be expressed as:

$$P(D\,|S) = \prod_{k=1}^{n} \sigma(s_w^k - s_l^k), \tag{4.3}$$

where s_w^k and s_l^k are the respective skills of the winning and losing player of the $k^t h$ outcomes. For a given set of the player skills, predictions can be made using Equation 4.2. Details on the methods used to fit the parameters from the data are provided in Chapter 5.

4.5 Free Parameter Point Model

Although the Bradley-Terry model described above can be applied to point level outcomes, it requires treating points generally with no distinction between the service points of each player. The model in this section extends the ideas of a Bradley-Terry model in order to predict distinct service probabilities for each player. This means that we split the points into two classes, one for points where a player is serving and one for points where a player is receiving.

In this model, each player is represented by two parameters: one which represents their attacking skill and one which represents their defensive skill. The parameters of the model are therefore a set of attacking skills $S = \{s_1,...,s_m\}$ and a set of defensive skills $B = \{b_1,...,b_m\}$. In tennis, serving and receiving serves can be considered attacking and defensive parts of the game respectively. Based on this, a players chance of winning a point on their serve in this model is assumed to depend only upon their own attacking strength and their opponents defensive strength. It is recognised that this assumption is an idealisation since a receiving player may use attacking elements of their game in returning or where a rally develops. However, the assumption is used in order to reduce the complexity of the model. The probability of player i winning a point against player j on their serve is defined as:

$$P(r_{ij} = 1|s_i, b_j) = \sigma(s_i - b_j). \tag{4.4}$$

With r_{ij} now referring specifically to the probability of player i winning a point on serve against player j rather than winning a point generally. Note that $r_{ij} \neq 1 - r_{ji}$ which would otherwise be true in the general case. The likelihood of the data can now be represented as:

$$P(D|S) = \prod_{k=1}^{n} \left[P({r_{ij}}^k|{s_i}^k, {b_j}^k)\; P({r_{ji}}^k|{b_i}^k, {s_j}^k) \right]. \tag{4.5}$$

Where:

$$P({r_{ij}}^k|{s_i}^k, {s_b}^k) = \sigma({s_i}^k - {b_j}^k)^{{r_{ij}}^k} \left(1 - \sigma({s_i}^k - {b_j}^k)\right)^{(1-{r_{ij}}^k)}.$$

Given the attacking and defensive skills of two players, Equation 4.4 can be used to predict the probability of either player winning a point on their serve. The Markov chain described in Section 4.1 can then be used convert these probabilities into match winning probabilities. As with the previous model, the fitting of the parameters is discussed in Chapter 5.

4.6 Surface Factor Model

This model focuses on developing an approach to explicitly model the effects of surface. Inspiration is taken from work on rating systems with multiple factors (Stanescu, 2011). Each player is defined as having a vector of k different positive skills $s = [s_1, ..., s_k]$. These skills are arbitrary, but could for example represent a players strength in different areas such as serving, baseline rallies or net play. Each surface is defined as having a vector of k positive weights $w = [w_1, ..., w_k]$. These weights describe the characteristics of the surface and indicate how important different aspects of a players game are for playing on it. The overall skill exhibited by a player on a particular surface is given by $w^T s$. The probability of a player i winning against player j on surface s is then defined as:

$$P(r_{ijs} = 1 | s_i, s_j, w_s) = \sigma(s_i w_s^T - s_j w_s^T). \tag{4.6}$$

The likelihood for a set of matches can then be expressed as: (using notation previously defined)

$$P(D\ |S, W) = \prod_{k=1}^{n} \sigma({s_i}^k {w_s^T}^k - {s_j}^k {w_s^T}^k), \tag{4.7}$$

where W is a set of weight vectors for all surfaces. Both the set of player skill vectors S and the set of surface weight vectors W constitute the parameters of the model and are both learned as part of the fitting process. Given a set of fitted parameters, Equation 4.6 can be used to make predictions for future outcomes.

4.7 Time Series Model

The models previously described are aimed at a scenario where one jointly optimises the model parameters assuming them to be constant within a fixed period of matches. In this time series model, the aim is to explicitly model the parameters as varying over time. Skill models of this nature have been developed by both Herbrich et al. (2007) and Glickman (2001). The model described in this section relies closely on ideas and equations from both of these papers.

This model is based directly upon the Bradley-Terry model described in Section 4.4. However, the model is extended by considering each player as having distinct skills at different points in time and by defining how these distinct skills are related. The probability of player i beating player j at time t is defined using the relationship from a simple Bradley-Terry model:

$$P(r_{ijt} = 1 | s_{it}, s_{jt}) = \sigma(s_{it} - s_{jt}), \tag{4.8}$$

where s_{it} is now the skill of player i specifically at time t. A prior skill is also defined for all players of $s_0 = N(s_0; 0, {\sigma_0}^2)$. We now wish to consider that the skill of a player at time t is dependant upon their skill at adjacent points in time (s_{t+1} and s_{t-1}). This is modelled by assuming a Gaussian drift between the skills at time t and $t+1$:

$$s_{t+1} = \alpha s_t + (\sqrt{1-\alpha^2})v, \tag{4.9}$$

where $\alpha = [0, 1]$ and v is a Gaussian which is equal to the prior s_0:

$$v = s_0 = N(v; 0, {\sigma_0}^2).$$

Equation 4.9 describes how the skill parameters are expected to change over time. The effect of this relationship is that if no data is seen, the beliefs about the parameters gradually drift back towards the prior. This can be argued to be a realistic assumption, because the players being modelled are top level athletes and their skills can be expected to decrease if they do not regularly compete in tournaments. Equation 4.9 is similar to that used by Herbrich et al. (2007) and Glickman (2001). However, they both apply a slightly different relationship such that only the variance and not the mean of players skills drifts over time.

The parameter α in Equation 4.9 controls the rate at which the drift occurs and effects the flexibility of the skills through time. If α is close to 1, then the skills of players are presumed to change only very slowly. At the other extreme, if α is 0, then skills at adjacent time points are independent. The value of α is set as part of the fitting process.

Chapter 5

Implementation

This chapter discusses the implementation of the models and is structured as follows:

- Section 5.1 discusses the overall process used to evaluate the performance of the models on the historical data set.
- Section 5.2 discusses surface and recency weighting and how these are incorporated into the models.
- Section 5.3 provides details on the two methods used to fit the parameters of the probabilistic models described in the previous chapter.
- Section 5.6 details relating to the implementation of the time series model. This model is implemented using two different approaches, one based on jointly optimising player skills for multiple points in time and one based on applying filtering updates.
- Section 5.7 discusses how we define a confidence measure for the predictions made by the models in this project.

5.1 Evaluation Approach

In this project, an online iterative process is used to generate historical predictions for each model. These predictions are then scored based upon the metrics discussed in Chapter 2. This approach is similar to that used in previous work on tennis prediction (McHale and Morton, 2011).

The process itself consists of moving through the data making predictions for one tournament round of matches at a time. For each batch of predictions, the model is completely refitted based upon the 3 years of data directly previous. This ensures that predictions for all matches are only made based upon information that would have been available at the time they were played. A down side to this approach, is that the overall evaluation process is expensive due to the number of optimisations that need to be performed. To evaluate the full set of data (predictions from 2005 to 2017) requires approximately 3,800 optimisations using this approach. It is possible to perform the optimisations in parallel and this has been implemented for some models in this project. We train all of the models in less than 8 hours, but some are split over as many as 20 cores. There is scope for further improvement by initialising the parameters of each optimisation based on the fitted parameters from the optimisation previous in sequence. However, this was not applied due to it not integrating easily with the parallel implementation.

The filtered time series model which is discussed in Section 5.6.2 is the only model which differs slightly from what is described above. In this model, player skills are continuously updated so that they are always based upon the full history of matches rather than just 3 years. The updates also have to be performed sequentially and therefore cannot be performed in parallel.

5.1.1 Training and Test Set

The data is split into two sets: A training set consisting of matches from years 2005 to 2015 and a test set consisting of matches from years 2016 to 2017. The training set is used for the majority of the model evaluations and is the basis for making model choices. The test set is reserved until the end and is only evaluated once for each model. Its purpose is to show how well the performance of each model generalises to data outside the training set.

It is common in machine learning to also use a third validation set for cross validating model choices. However, this is not necessary in this project due to the nature of the on line fitting process described above.

	Clay	Hard	Carpet	Grass
Clay	1			
Hard	0.1	1		
Carpet	0.1	1	1	
Grass	0.01	0.5	0.5	1

Table 5.1: Surface Weightings

5.2 Weighting Matches

Weighting provides a straightforward method to account for the effects of time and surface type in models which do not otherwise explicitly address these factors. For models in this project, weightings are applied according to match recency and surface type following the method used by McHale and Morton (2011). This section discusses how these weightings are determined and how they are applied to the different models.

5.2.1 Recency Weighting

For each match, a recency weighting w_r is calculated according to an exponential decay function of the following form:

$$w_r = \left(\frac{1}{2}\right)^{\frac{t}{\lambda}},$$

where t is the difference in days between the match and those to be predicted and λ is the half life of the decay. In this project, the half life is determined by using a grid search of values to optimise the online predictive performance measured on the training set.

5.2.2 Surface Weighting

For each match, a surface weighting is applied according to Table 5.1. The values in this table are selected based upon those used by both McHale and Morton (2011) and Sipko and Knottenbelt (2015). An alternative approach to this would be simply splitting the data by surface. The model for each surface would then have less data, but only data for the correct surface. This would correspond to setting the weights for other surfaces to zero. For the models in project, we find that a non-zero weighting for other surfaces results in the superior performance.

5.2.3 Weighting in Probabilistic Models

Weighting in a probabilistic model translates to raising the factor in the likelihood expression to the power of the weighted value. For a set of results R, this changes the general form of the likelihood to:

$$P(R|w) = \prod_{i=1}^{N} P(r^i|w)^{w^i},$$

where w^i is the weight associated with the i^{th} match and other notation is as previously defined. When fitting model parameters by maximum likelihood discussed in Section 5.4, the overall effect of applying weightings in this manner is that the contribution of each data point to the cost function is scaled by its respective weight. When fitting the models parameters using a Bayesian approach discussed in Section 5.5, the weighting effects the quantity of evidence associated with each match.

5.2.4 Normalising Point and Game Outcomes

In this project, models are trained based on different levels of match information: Points, games or overall matches. When modelling games or points the totals of wins and losses involved are much higher in comparison to when modelling matches. This has a knock on effect on likelihood expression and the final confidence that the model has about the skills of players. To illustrate, consider a match where player i lost to player j and in that match player i won 40 points and player j won 60 points. If modelling match outcomes then the likelihood would be expressed as:

$$P(r|s_i, s_j) = \sigma(s_i - s_j)^0 \sigma(s_j - s_i)^1 \tag{5.1}$$

However, if modelling point outcomes the likelihood relating to the same match would be expressed as:

$$P(r|s_i, s_j) = \sigma(s_i - s_j)^{40} \sigma(s_j - s_i)^{60} \tag{5.2}$$

Comparing the two equations demonstrates the difference in the amount evidence the model believes it has seen from just one match of data. In the later case, it is clear that the model will become confident about the skills of players after seeing far fewer matches. Furthermore, in tennis the number of points played in each match varies a significant amount. This means that some matches will effectively contribute to the models estimate of player skills more heavily than others. In order to avoid this effect, we choose to apply a normalising weighting to every match of data when modelling point or game outcomes. This weighting is defined as one divided by the total number of points or games won in the respective match. Applying this normalisation to the example above changes the likelihood expression to the following:

$$P(r|s_i, s_j) = \sigma(s_i - s_j)^{0.4}\sigma(s_j - s_i)^{0.6} \tag{5.3}$$

It is possible that applying some alternative softening to the counts of point and games in matches may provide better results. This was not something explored in this project and is therefore highlighted as an area to be investigated in future work.

5.3 Fitting the Model Parameters

This section discusses the two different approaches which are used to fit the parameters of the models presented in Sections 4.4 to 4.7.

5.4 Penalised Maximum Likelihood Fitting

Penalised maximum likelihood fitting is closely related to maximum a posteriori (MAP) estimation. A MAP estimate is where the parameters of a model are estimated as a point value at a mode of the posterior probability distribution. For a posterior probability distribution of the form:

$$P(w|D) \propto \prod_{i=1}^{N} P(r^i|w)P(w), \tag{5.4}$$

The MAP estimate can be found by maximising this expression with respect to the parameters w. Equivalently, we can minimise the negative log of this expression, which is more stable. The log of Equation 5.4 is:

$$-\log\big(P(w|D)\big) \propto -\left[\sum_{i=1}^{N} \log\big(P(r^i|w)\big)\right] - \log\big(P(w)\big). \tag{5.5}$$

When a uniform prior is used, MAP estimation is known as maximum likelihood fitting since it is equivalent to maximising the likelihood term alone. When a non-uniform prior is used, it can be referred to penalised maximum likelihood fitting, as the parameters are regularised in some way according to the prior. When the prior is a spherical Gaussian, the effect of the prior is directly equivalent to what is known as L2 regularisation. This is where the log likelihood is maximised alone, but with an added regularisation term of the form:

$$\beta \sum_{i=1}^{K} w_i^2,$$

where β is a regularisation constant equivalent to the precision of the Gaussian prior. In this project, models are fitted using both L2 regularised and un-regularised maximum likelihood. For the relevant models, Equation 5.5 is maximised by taking derivatives with respect to the model parameters and optimising using a quasi-Newton optimiser package from the scipy library (Jones et al., 2001–).

5.5 Approximate Inference Fitting

Approximate inference methods are a class of techniques which can be used to approximate posterior distributions when exact learning and inference is intractable. Variational methods are a subclass of these techniques which provide an analytical approximation by matching the true posterior to an alternative distribution of tractable form.

In this project, a variational method is used to fit approximate Gaussian posteriors to the models in Sections 4.4 to 4.7. We derive this method based upon stochastic variational inference (SVI) (Hoffman et al., 2013). However, due to the size of our data set and our success using batch fitting when obtaining MAP estimates, we derive a non-stochastic method. This allows us to use the same quasi-Newton optimiser package that we use in MAP fitting. The variational procedure is then reasonably fast when coded in simple python.

Related work on fitting Bayesian based Bradley-Terry type models has prominently used Assumed Density Filtering (ADF) or Expectation Propagation (EP) (e.g. Birlutiu and Heskes (2007), Stanescu (2011) or Herbrich et al. (2007)). However, we chose a method which is based upon SVI because it can be more easily applied to extended models with additional parameters. We also ruled out MCMC Sampling methods, as these would have been too slow due to the number fits that had to be performed in each model evaluation.

5.5.1 Variational Inference

For a model with parameters w the posterior distribution over the parameters given some data (D) can be represented as:

$$P(w|D) = \frac{P(D|w)P(w)}{P(D)}$$

$P(D|w)$ is the likelihood of the model given the data, $P(D)$ is marginal likelihood of the model found through normalisation and $P(w)$ is a prior distribution on the model parameters. For the models described in Sections 4.4, 4.5 and 4.7, the likelihood is a product of terms and each term is a sigmoid function containing some linear combination of the model parameters. The likelihood can therefore be represented generally as:

$$P(D|w) = \prod_{i=1}^{N} \sigma(w^T x^i) \tag{5.6}$$

The goal of the variational procedure is to approximate $P(w|D)$ with a multivariate Gaussian of the form $Q(w) = N(w; m, V)$ and for a Gaussian prior $P(w) = N(w; m_0, \Sigma)$. This is achieved by optimising the parameters of the approximate posterior (m and V) in order to minimise the following objective function:

$$J = \Big\langle \underbrace{\log(N(w;\, m, V))}_{entropy} \Big\rangle_{N(w;\, m,V)} - \Big\langle \underbrace{\log(P(D|w))}_{likelihood} \Big\rangle_{N(w;\, m,V)} - \Big\langle \underbrace{\log(P(w))}_{cross-entropy} \Big\rangle_{N(w;\, m,V)} . \tag{5.7}$$

This objective function arises from taking the Kullback-Leibler divergence between the approximate and true posterior distributions. Minimising J translates to maximising a lower bound on the logarithmic marginal likelihood of the model. Equation 5.7 can be minimised by taking gradients with respect to m and V and then using a gradient method. Evaluating the cost and gradients of the entropy and cross-entropy expectations can be performed analytically. However, it is not possible to evaluate the likelihood expectation analytically, due to the nature of the sigmoid terms within it. The procedure instead uses an estimate of the gradients and cost for this term. This is discussed below along with the results for the entropy and cross-entropy terms. The maths used in the derivation of the many of the equations presented relies upon results on multivariate Gaussians (Petersen et al., 2008) and error function integrals (Ng and Geller, 1969).

5.5.2 Entropy

Cost:

$$\Big\langle \log(N(w;\ m,V)) \Big\rangle_{N(w;\ m,V)} = -\frac{1}{2}\log(|V|) + const$$

Gradients:

$$\nabla_{L_{diag}} = -\frac{1}{L_{diag}} \qquad \nabla_V = -\frac{1}{2}V^{-1} \qquad \nabla_m = 0$$

Where L is the cholesky factor $V = LL^T$. Due the constraints on the covariance terms it is more stable to optimise with respect to L when considering the full covariance matrix. When only fitting a diagonal covariance, it is more convenient to consider gradients with respect to V directly.

5.5.3 Cross-Entropy

Cost:

$$-\Big\langle \log(N(w;\ m_0,\Sigma)) \Big\rangle_{N(w;\ m,V)} = \frac{1}{2}\mathrm{Tr}(\Sigma^{-1}V) + \frac{1}{2}(m_0 - m)^T\Sigma^{-1}(m_0 - m) + const$$

Gradients:

$$\nabla_L = \Sigma^{-1}L \qquad \nabla_V = \frac{1}{2}\Sigma^{-1} \qquad \nabla_m = -\Sigma^{-1}(m_0 - m)$$

Where Tr is the trace operator.

5.5.4 Likelihood

For a likelihood of the form in Equation 5.6, the corresponding expectation term in Equation 5.7 can be expressed as follows:

$$-\Big\langle \log(P(D|w)) \Big\rangle_{N(w;\ m,V)} = -\sum_{i=1}^{N} \Big\langle \log(\sigma(w^T x^i)) \Big\rangle_{N(w;\ m,V)}. \tag{5.8}$$

In the original SVI procedure, an unbiased estimate of the cost and gradients is obtained by using a sample from the current posterior $N(w;\ m,V)$. This is evaluated on a random subset of the N likelihood terms at each iteration. This stochastic procedure is efficient and extremely scalable allowing it to be applied to large data sets (Hoffman et al., 2013). However, for this project we find it is more effective to use an alternative approach where the estimates of the cost and gradients are deterministic. These estimates are more expensive at each iteration but allow a quasi-Newton method to be used instead of gradient descent. This is more efficient overall since the number of iterations required to converge to a solution is far smaller. Additionally, the process is more stable and an arbitrary level of accuracy can be achieved.

5.5.4.1 Deterministic Estimate of Likelihood Cost and Gradients

This section summarises how closed form estimates of the likelihood expectations in Equation 5.8 and their gradients can be obtained. Each of the N terms in Equation 5.8 are of the form:

$$\Big\langle \log(\sigma(w^T x)) \Big\rangle_{N(w;\, m,V)}.$$

By performing a change of variables this can be reduced to a one dimensional expectation of the form:

$$\Big\langle \log(\sigma(\mu + \tau v)) \Big\rangle_{N(v;0,1)},$$

where $\mu = m^T x$ and $\tau = \sqrt{(x^T V x)}$. The goal is now to obtain gradients of this expectation with respect to τ and μ, from which the desired gradients of m and V can be easily obtained using the chain rule. The gradient with respect to μ can be shown as.

$$\begin{aligned}
\nabla_\mu \Big\langle \log(\sigma(\mu+\tau v)) \Big\rangle_{N(v;0,1)} &= \Big\langle \nabla_\mu \log(\sigma(\mu+\tau v)) \Big\rangle_{N(v;0,1)} \\
&= \Big\langle \frac{\sigma(\mu+\tau v)(1-\sigma(\mu+\tau v))}{\sigma(\mu+\tau v)} \Big\rangle_{N(v;0,1)} \\
&= \Big\langle \sigma(-\mu-\tau v) \Big\rangle_{N(v;0,1)} \\
&= \int_{-\infty}^{\infty} \sigma(-\mu-\tau v) N(v;0,1)\, dv \qquad (5.9)
\end{aligned}$$

Likewise the gradient with respect to τ can be shown as:

$$\begin{aligned}
\nabla_\tau \Big\langle \log(\sigma(\mu+\tau v)) \Big\rangle_{N(v;0,1)} &= \Big\langle v\sigma(-\mu-\tau v) \Big\rangle_{N(v;0,1)} \\
&= \int_{-\infty}^{\infty} v\sigma(-\mu-\tau v) N(v;0,1)\, dv \qquad (5.10)
\end{aligned}$$

The integrals in Equations 5.9 and 5.10 cannot be computed analytically. However, an approximation can be found by using the fact that the sigmoid function can be approximated by the following expression (Crooks, 2009):

$$\sigma(x) \approx \frac{1}{2}\Big(1 + \operatorname{erf}\big(\frac{\sqrt{\pi}}{4}x\big)\Big),$$

where erf is the error function $\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$. By substituting the sigmoid function in Equations 5.9 and 5.10 with this approximation, both the integrals can be computed in closed form. Performing these integrals gives the following estimates for the gradients of τ and μ:

$$\nabla_{\mu}\Big\langle \log(\sigma(\mu+\tau v))\Big\rangle_{N(v;0,1)} = 0.5+0.5\,\text{erf}\left[\frac{-\mu\sqrt{\pi}}{4\sqrt{1+\frac{\pi}{8}\tau^2}}\right] \tag{5.11}$$

$$\nabla_{\tau}\Big\langle \log(\sigma(\mu+\tau v))\Big\rangle_{N(v;0,1)} = -\frac{\frac{\sqrt{\pi}}{4}\sigma}{\sqrt{1+\frac{\pi}{16}\tau^2}}\exp\left[-\frac{0.5\frac{\pi}{16}\mu^2}{1+\frac{\pi}{16}\tau^2}\right]\frac{1}{\sqrt{2\pi}} \tag{5.12}$$

In principle, the optimisation could be performed based on the gradient information alone. However, evaluating the cost was a requirement for the optimiser package used in this project. A cost estimate can be obtained by integrating back upwards from the approximation of the gradients. This gives what the terms inside the expectation would have been if the approximate gradients were actually the true gradients. Performing this integral leads to the following approximation:

$$\Big\langle \log(\sigma(x))\Big\rangle_{N(v;\,0,1)} \approx \Big\langle -a\exp(-\frac{1}{2}bx^2) - 0.5x\text{erf}(cx) + 0.5x\Big\rangle_{N(v;\,0,1)}, \tag{5.13}$$

where:

$$a=\frac{2}{\pi},\quad b=\frac{\pi}{8},\quad c=\frac{\sqrt{\pi}}{4},\quad x=\mu+\tau v.$$

All of the terms inside expectation on the right hand side of Equation 5.13 can be computed analytically, which gives a closed form estimation of the cost for each likelihood term. The results are provided in the appendix.

Equation 5.13 arises from first approximating the gradients and then working backwards to estimate the cost. However, we find that once the form of the approximation is known it can be improved by refitting the parameters a, b and c. This results in new values of:

$$a=0.692310,\quad b=0.358114,\quad c=0.443113. \tag{5.14}$$

Which gives a closer fit for:

$$\log(\sigma(x)) \approx a\exp(-\frac{1}{2}bx^2) - 0.5x\text{erf}(cx) + 0.5x. \tag{5.15}$$

This fits for all x with maximum error of 0.0035. Note that the gradient estimates given in Equations 5.11 and 5.12 relate to the original values of a, b and c.

5.5.5 Making Predictions using the Approximate Posterior

To make predictions in a Bayesian model we take the expectation of $P(r=1|w)$ (the prediction given the parameters) with respect to the posterior distribution $P(w|D)$. For the models in this project $P(r=1|w)$ can be represented in the form $\sigma(w^T x)$ and $P(w|D)$ is the fitted approximate Gaussian posterior of the form $N(w;\ m,V)$. The expectation can therefore be expressed as:

$$\left\langle \sigma(w^T x) \right\rangle_{N(w;\ m,V)} = \int_{-\infty}^{\infty} \sigma(w^T x) N(w;\ m,V) dw. \tag{5.16}$$

By performing a change of variables this can be reduced to a one dimensional integral of the form:

$$\int_{-\infty}^{\infty} \sigma(\mu + \tau v) N(v; 0, 1) dv, \tag{5.17}$$

where $\mu = m^T x$ and $\tau = \sqrt{(x^T V x)}$. This integral cannot be computed analytically but it can approximated by the following expression (Crooks, 2009):

$$\int_{-\infty}^{\infty} \sigma(\mu + \tau v) N(v; 0, 1) dv = \sigma\left(\frac{\mu}{\sqrt{1 + \frac{\pi}{8}\tau^2}} \right) \tag{5.18}$$

5.5.6 Parameter Initialisation

With the exception of the surface factor model, there is a unique solution when fittings the parameters of all of the models. This means that the solution reached is the same regardless of the initialisation used at the start of the optimisation. For these models we therefore simply initialise the skills to zero at the start of every optimisation. In the Bayesian case we initialise the posterior covariance and mean to be equal to the prior covariance and mean.

In the surface factor model there is an issue with symmetry because the ordering of the parameters within the player and surface vectors is arbitrary. This creates a problem that the gradients of the parameters within the vectors are always equal if they are initialised to zero. To avoid this problem, we initialise the parameters of this model randomly at the start of each optimisation by drawing from a standard normal distribution.

5.6 Fitting the Time Series Model

In this project, the time series model described in Section 4.7 is implemented in two different ways. Both of these approaches are discussed as follows:

5.6.1 Joint Optimisation Model

In this approach, a fixed 3 year window of historical data is split into n time periods and all matches within a time period are treated as if they happened at the same time. Each player is therefore considered as having a distinct skill for each of these time periods. The skills of all players at all time periods are then jointly fitted as part of one optimisation. The dependencies between player skills at neighbouring time steps are captured by adding the appropriate correlation terms within the prior precision matrix. These terms can be derived based upon the relationship of how the skills are expected to drift over time (equation 4.9, page 21). For constant time steps, the prior precision matrix on the skills of the same player at different points in time has the following form.

$$\begin{pmatrix} c & a & & & & 0 \\ a & b & a & & & \\ & a & b & \ddots & & \\ & & \ddots & \ddots & a & \\ & & & a & b & a \\ 0 & & & & a & c \end{pmatrix}$$

Where:

$$a = \frac{1-\alpha}{{\sigma_0}^2(1-\alpha^2)}, \quad b = \frac{1+\alpha^2}{{\sigma_0}^2(1-\alpha^2)}, \quad c = \frac{1}{{\sigma_0}^2(1-\alpha^2)}$$

Once the prior precision matrix is derived, the fitting of the parameters can be done using the variational procedure described above.

5.6.2 Filtering Model

This approach is similar to that used in traditional rating systems and described by Glickman (2001): We move through the historical data sequentially and successively update the skills of players based on short time periods of matches. For each time period, the posterior distribution of player skills is approximated based on the matches in that period, which are treated as if they all happen at the same time. A drift is then applied to the skills according to Equation 4.9 (page 21) and the result used as the prior for the next time period. This process is repeated sequentially through all of the data. At each step, the fitting is performed using the variational procedure described in Section 5.5. Predictions can be made based upon the skill parameters of the playTers at the most recent time period.

A key difference between this approach and the joint optimisation approach is that the fitting itself never involves skills of the same player from different points in time. Instead, the skills between time steps are only linked through the fact that the posterior at one time step becomes the prior for the next time step. This approach has the advantage that it is easy to update the skills of players as matches happen. Additionally, the optimisations involved in this approach are far cheaper which allows much finer periods of time to be considered. We update the skills after every round of a tournament. By contrast, in the joint optimisation approach, due to the cost of the optimisation we only consider breaking a 3 year span into a maximum of 4 time periods.

5.7 Confidence of Predictions

The confidence of a prediction is the uncertainty associated with the predicted probability value and is discussed in Section 2.4.1. One way this can be defined is to assign a measure of the quantity of information which the data used to train the model contains about both players in a predicted match. If recency weighting is used in the model then this can be achieved by considering the combined weight of matches for each player in the historical data (Sipko and Knottenbelt, 2015). However, in this project, as we fit our models using a Bayesian approach, we can define a measure of confidence which is based on the uncertainty present in the fitted parameters. Predictions in the Bayesian models are made based upon the previously given expectation:

$$P(r=1|D,x) = \left\langle \sigma(w^T x) \right\rangle_{N(w;\ m,V)}, \tag{5.19}$$

where $\mu = m^T x$ and $\tau = \sqrt{(x^T V x)}$. We can perform a further change of variables with respect to the distribution $P(w) = N(w;\ m,V)$ in order to obtain a distribution with respect to $P(p = \sigma(w^T x))$. This results in a Logit-normal distribution $P(p)$ with a mean corresponding to the prediction and variance which provides a measure of the uncertainty of the predicted probability value. Typically this variance will be larger if τ^2 is greater and vice versa. Therefore in this project we define an approximate measure of confidence for each prediction as:

$$\frac{1}{\tau^2}. \tag{5.20}$$

Chapter 6

Results and Discussion

This chapter presents the project results and findings. The chapter is structured as follows:

- Section 6.1 discusses the performance of the model baselines.
- Sections 6.2 to 6.6 present the respective results for each of the models described Sections 4.4 to 4.7 in Chapter 4.
- Section 6.7 presents the results from evaluating the final model on the previously unseen test set.
- Section 6.8 discusses how return of investment can be maximised by considering the confidence of predictions.

For the results which are presented in Sections 6.1 to 6.6, all of the models are evaluated on the training set of data. Unless otherwise specified, the scores across all models in these sections are based on predictions for the same subset of matches. This subset contains all matches for ATP tournaments between the start of 2005 to the end of 2015 where both players have played a minimum of 5 matches in the previous 12 months of data. Approximately 26,700 matches.

All of the models are evaluated using the online process described in Section 5.1. In this process, all predictions are made based on the model only having seen data which is older than the match itself. This means that if the model parameters are over fitted, it will simply be reflected directly in the performance scores. It is possible that hyper-parameters may be over fitted. However, a final comparison is performed on a held out test set in Section 6.7 to confirm this is not the case.

Model	Accuracy	Average Probability	Average Log Probability	ROI
Naive Predictions	50.00%	0.500	−0.690	−12.56%
Using ATP Rankings	65.72%	0.557	−0.618	−14.93%
Point Model	67.15%	0.588	−0.605	−10.38%
Bookmakers	70.28%	0.607	−0.565	–

Table 6.1: Baseline performance on the training set consisting of approximately 26,700 predictions from 2005 to 2015. For all metrics higher is better.

6.1 Baseline Results

In this project, models are compared against four baselines: The first is predictions derived from bookmakers odds. The second is predictions made based upon player ATP ranking points. The third is predictions from a point model from the literature described in Section 4.2. The last is naive predictions which are made given no information about the matches or players. The performance of each of these baselines is shown in Table 6.1.

6.1.1 Naive Predictions

The scores for the Naive predictions are based on predicting random probabilities drawn from a uniform distribution. This is with the exception of the third metric, which is based upon a constant prediction of 0.5. The purpose of this baseline is simply to provide a sense check for the performance of other models.

6.1.2 Predictions using ATP rankings

ATP ranking points can predict the winner of a match but can't be used to make probabilistic predictions directly. Instead the scores for predictions made using the ATP rankings in Table 6.1 are based on a logistic regression model fitted to player ranking points. This approach is implemented by following previous work by McHale and Morton (2011) and the results obtained are consistent with those in that paper.

6.1.3 Bookmakers

The bookmaker results are based on averaged historical odds from 5 different bookmakers. For metrics 2 and 3, the scores relate to normalised implied probabilities. The odds are not adjusted when calculating the return on investment for other models.

If someone chose to only bet on the bookmakers favourite then the ROI would be -3.62%. In contrast, betting only on the bookmakers underdog gives a ROI of -15.20%. This bias poses an issue when using return on investment as a metric for comparing models. A model that favours making over confident predictions will tend to bet more often on the favourite and thus may appear to score better in this category.

6.1.4 Point Model Results

The results for the point model in Table 6.1 relate to the model described in Section 4.2. In this model, service probabilities are estimated by combining observed fractions from previous matches. A half-life of 400 days was used in the decay function for weighting each match of data within these estimations. This half-life was selected based on a grid search of values and optimising with respect to accuracy. Results from literature for the same model give 67.27% and 0.605 for metrics 1 and 2 respectively, based on using 12 months of back data and no recency weighting (Spanias, 2014). This appears better than the results obtained here, however the time span of data used is smaller. Our model evaluated on the same time span produces 67.89% and 0.595 for the same metrics.

6.1.5 Comparison of Baseline Calibration

Figure 6.1 shows a plot of the calibration for 3 of the baseline models. The bookmaker model appears the best calibrated out of the 3, although its predictions are slightly under confident for larger probabilities. The point model displays the opposite trend and tends to give over confident predictions. The ATP ranking points model is the worst calibrated of the 3 and is both over and under confident in different regions.

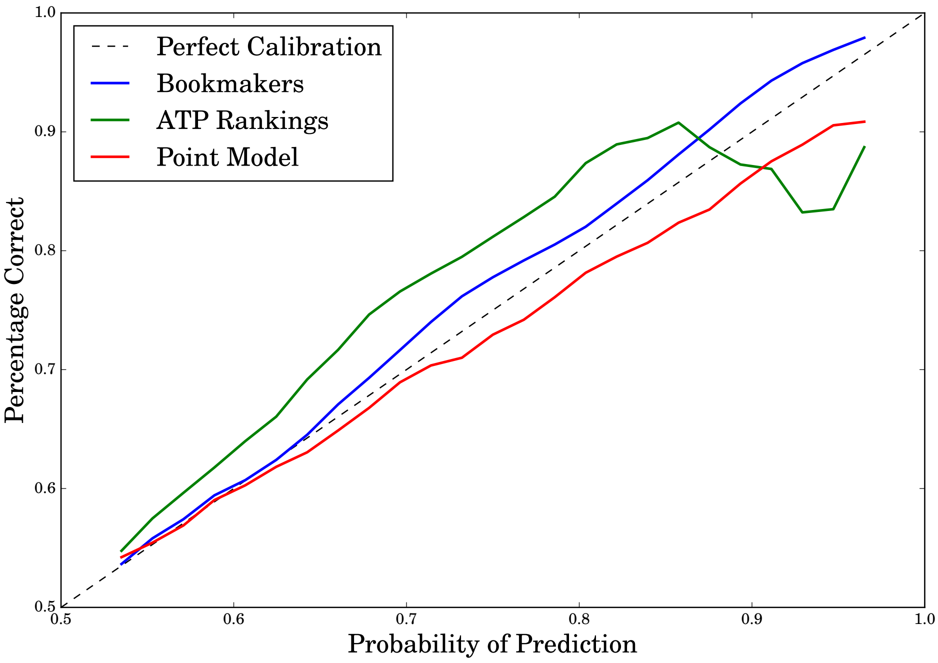

Figure 6.1: Calibration of baseline models. Evaluated as a moving average in bins of 0.07. The x-axis relates to the models predictions and the y-axis to the percentage of time predictions of that probability were correct

6.2 Bradley Terry Model Results

This section presents the results for the Bradley-Terry model described in Section 4.4. The experiments relating to this model had three primary aims:

- Firstly, to compare the performance of models trained based on point level and game level data.
- Secondly, to explore whether regularisation provided an improvement in models fitted with maximum likelihood.
- Thirdly, to compare the predictive performance of models fitted using maximum likelihood and the same models fitted using approximate inference.

6.2.1 Maximum Likelihood Fitting

All of the results in this sub-section are for Bradley-Terry models fitted using maximum likelihood as described in Section 5.4.

6.2.1.1 Determining Optimal Half Life

Prior to carrying out the main experiments, a grid search was performed in order to determine the most suitable half life for the weight decay function. Table 6.2 provides a summary of the model performance for some of the values tested.

Half Life (days)	Accuracy	Average Probability	Average Log Probability	ROI
30	63.62%	0.623	−1.012	−06.67%
60	65.20%	0.609	−0.741	−08.61%
150	66.96%	0.604	−0.638	−10.49%
360	67.52%	0.602	−0.621	−10.32%
600	67.28%	0.606	−0.619	−10.09%
inf	67.08%	0.607	−0.621	−09.62%

Table 6.2: Performance of Match Bradley-Terry Model for a selection of different half life values. The actual search was performed in steps of 30 in a range of 30 to 1000 days.

Based on the results a half-life of 360 days was chosen to be used in all applicable models. It is observed in Table 6.2 that the different performance metrics are not all maximised by the same half life value. Most notably, return on investment improves for short half life values, in spite of the clearly poorer model accuracy. An explanation for this is that a short model half life puts the focus on the most recent information, which better captures the current form of players. This means that some predictions will be more accurate but at the expensive of the predictions as a whole being more noisy. For return of investment, this is likely to be a more favourable trade off since poor predictions are not penalised as heavily compared to other metrics. For example, log probability can have penalisations of potentially unlimited size. By contrast, for return on investment, poor predictions simply receive a score equivalent to random predictions.

6.2.1.2 Regularisation

L2 regularisation was explored for point, game and match level Bradley-Terry Models. Figure 6.2 shows plots of the model performance for each of the different Bradley-Terry models according to a range of regularisation values. With the exception of metric 2, a very small amount of regularisation seems to improve performance.

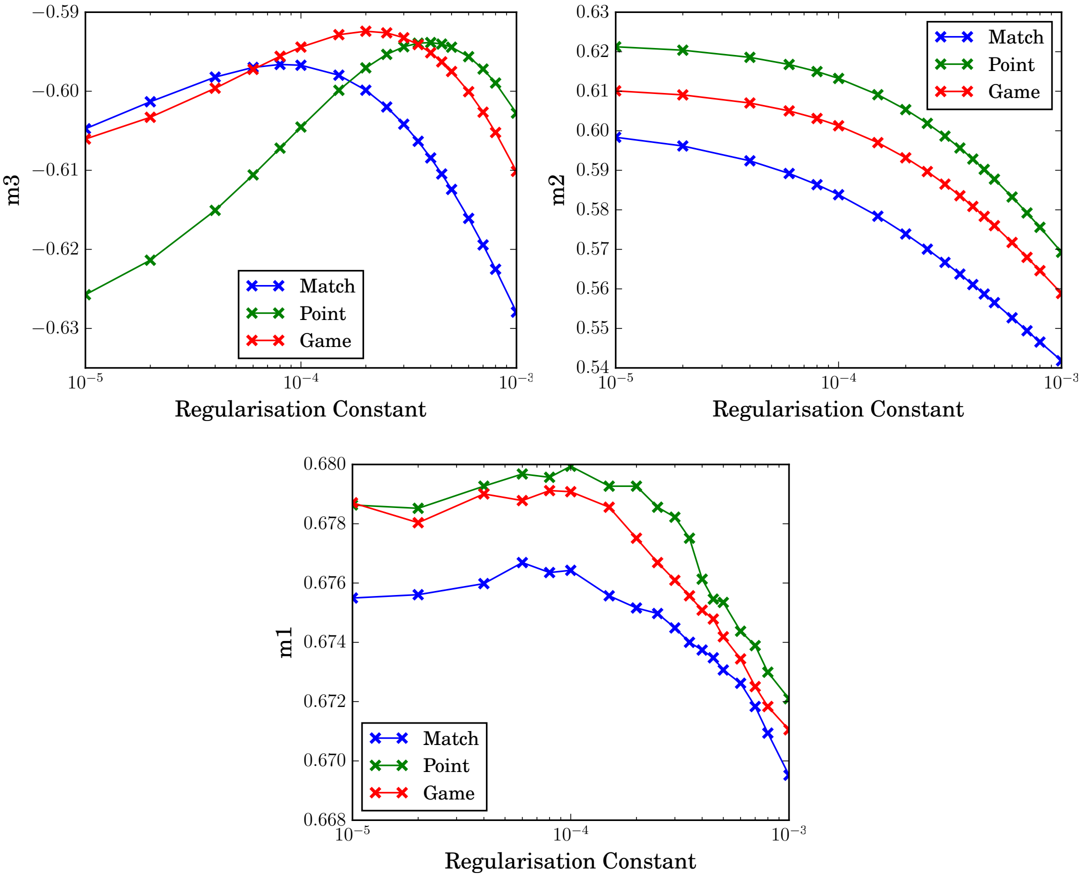

Figure 6.2: Regularised Bradley Terry Models. Left to right Accuracy (m_1), Average Probability (m_2), Average Log Probability (m_3). For all metrics higher is better.

6.2.1.3 Comparing Point, Game and Match Models

From Figure 6.2, both the point and game based models appear superior to the match model across all metrics. However, the difference between the game and point models is inconclusive as the metrics disagree on which model is the best performing. Furthermore, the differences in the scores themselves are small. In order to show whether the differences shown in Figure 6.2 are consistent over time, we plot both accuracy and log probability as a 12 month moving average for predictions between 2005 and 2015 (Figure 6.3). It can be seen that the performance of all of the models over time is extremely varied. However, a large part of this variation can be attributed to random trends in the number of upsets throughout different years. This explanation is supported by the fact that the performances fluctuations seen in the Bradley-Terry models are also mirrored in the bookmakers baseline. The curves for the point and game models frequently intersect throughout the 11 year period. This shows that neither model

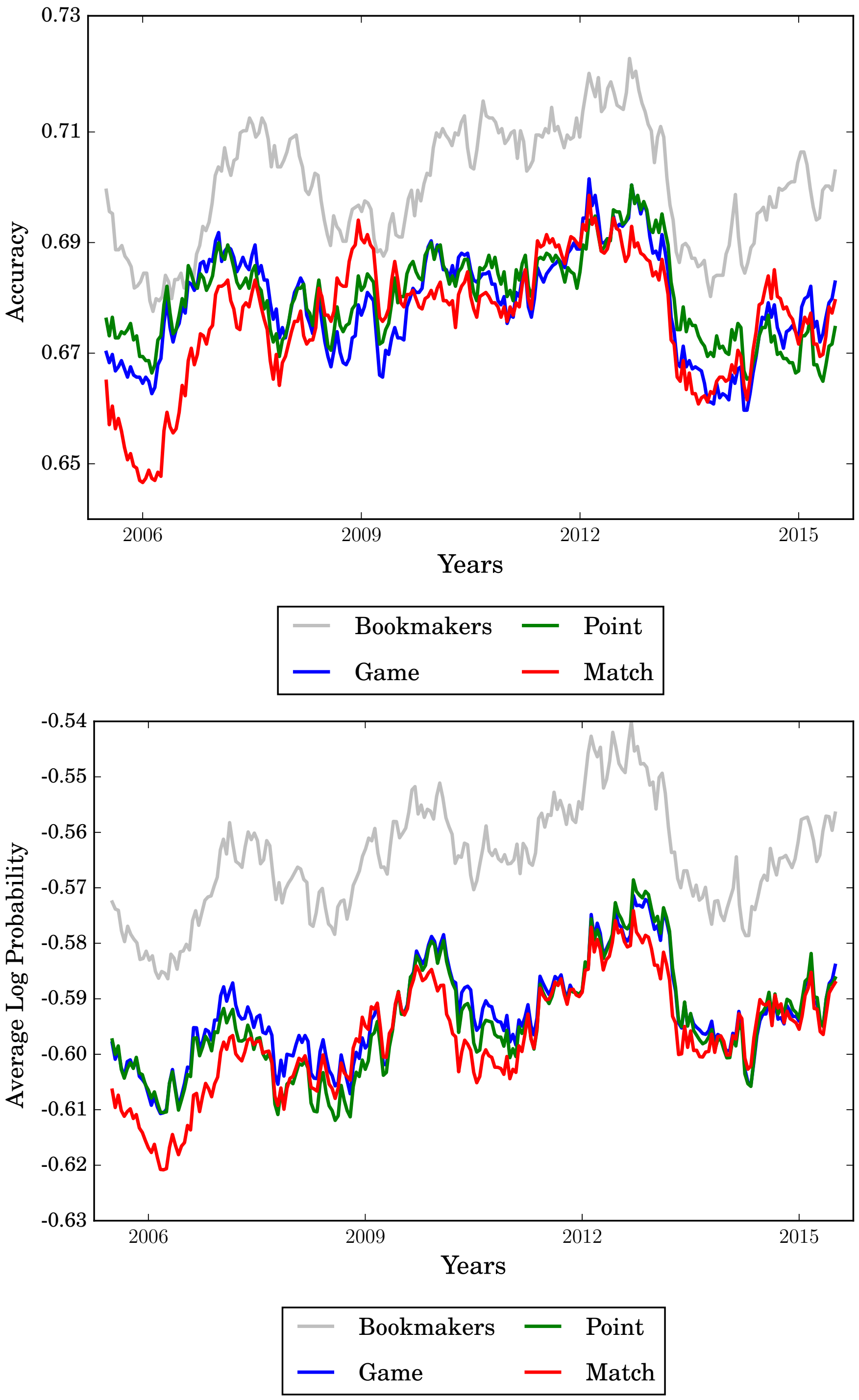

Figure 6.3: Model Accuracy (top) and Average Log Probability (bottom) over time based on a 12 month moving window. For both metrics higher is better. Each model is regularised according to the optimal regularisation constant for that model.

is consistently better than the other, suggesting that the overall performance of both models is essentially the same. However, examining the discrimination between the two models (Figure 6.3) shows they still disagree on approximately 6% of matches.

	Game Model Correct	Game Model Incorrect
Point Model Correct	65.0%	3.0%
Point Model Incorrect	2.9%	29.1%

Table 6.3: Discrimination between Point and Game Models. Both with regularisation of 0.0001. The matrix shows the percentage of matches which were correctly predicted by both models, incorrectly predicted by both models or correctly predicted by one and incorrectly predicted by the other.

6.2.1.4 Calibration of Bradley-Terry Models

Figure 6.4 shows calibration plots for regularised and unregularised Bradley-Terry models. It can be seen that all of the unregularised models are over confident. However, this bias is corrected with regularisation, at which point all of the models become similarly well calibrated. For each of the models, the regularisation value that produces the best calibration, coincides with the value which is optimal for the third performance metric (average log probability).

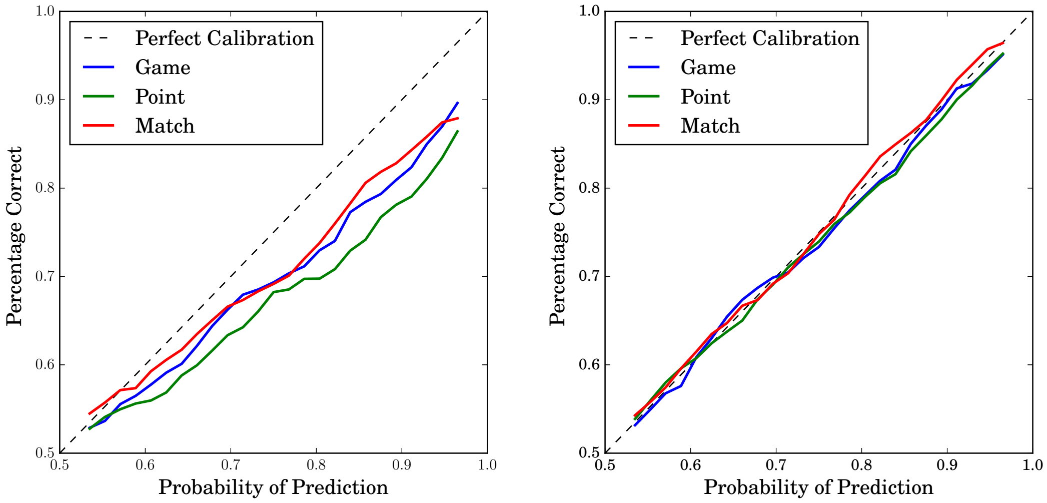

Figure 6.4: Calibration of point, game and match Bradley-Terry models. Left is for unregularised models and right is regularised. The regularisation is chosen specific to each model according to what is optimal in each case.

6.2.1.5 Aggregating Point, Game and Match Models

Each of the point, game and match models capture different information relevant to prediction. Aggregating the predictions from these models can provide superior performance compared to the individual counterparts. Aggregating is achieved by simply averaging the predictions from separately trained point, game and match models. The performance of this aggregated model is summarised in Table 6.4. It can be seen that the aggregated model does improve upon the performance of any of the individual models. Figure 6.5 shows log probability as a moving average over time. The improvement is reasonably consistent over time, giving confidence that it would also be present in future data. However, the size of the improvement itself is small.

Model	Accuracy	Average Probability	Average Log Probability	ROI
Match	67.64%	0.584	−0.597	−13.93%
Game	67.91%	0.601	−0.594	−09.26%
Point	67.99%	0.613	−0.605	−06.05%
Aggregated	68.09%	0.599	−0.591	−10.08%

Table 6.4: Summary of performance of for different Bradley Terry Models including aggregated model. All with regularisation of 0.0001.

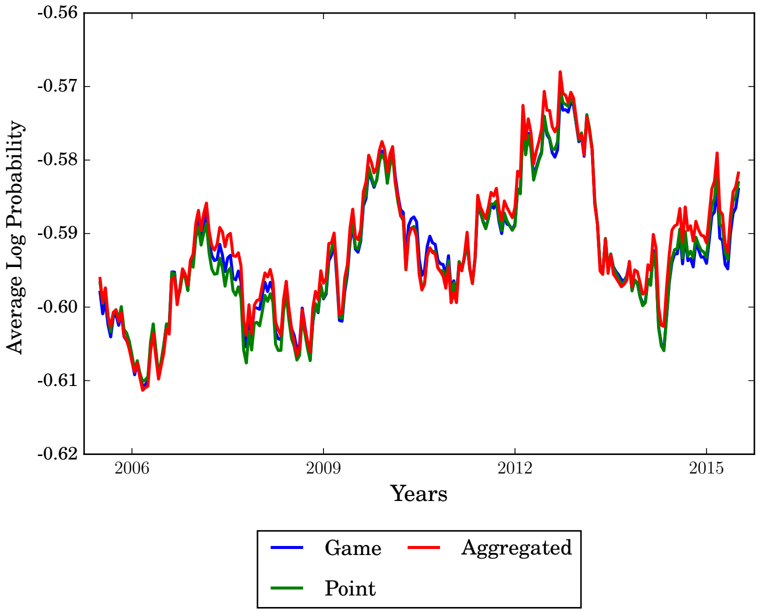

Figure 6.5: Average Log Probability over-time for aggregated model

6.2.2 Bayesian Fitting

This section discusses results from applying a Bayesian approach to fitting the parameters in point and match based Bradley-Terry models. For each model, the variational procedure described in Section 5.5 was used to approximate posterior densities of player skills. Predictions were then made based upon the posterior densities using the method described in Section 5.5.5. Figure 6.6 provides a demonstration of marginal posterior skill densities obtained in the match level model. It can be seen that the model is able to learn the skills of some players better than others. This is expected as some players have a greater number of matches in the data than others. A further observation is that the uncertainty present in all of the skills is high.

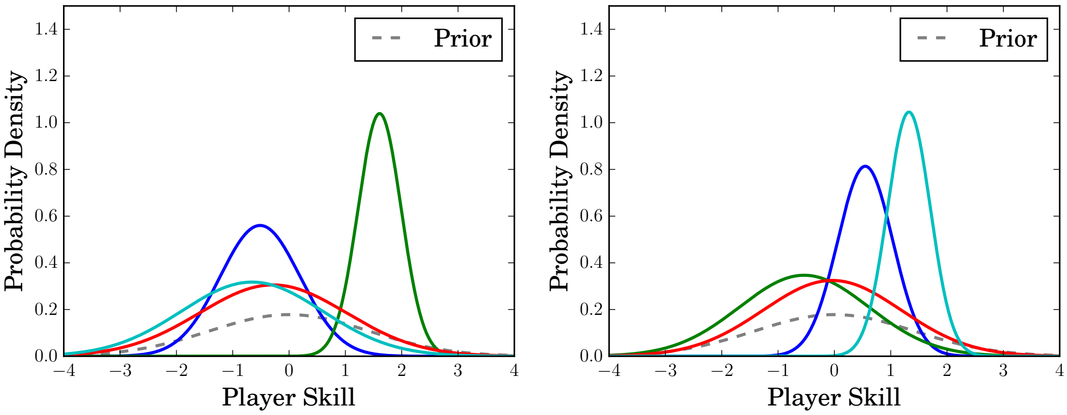

Figure 6.6: Example marginal posterior skill densities for a random subset of players in the Match level Bradley-Terry model. Based on a prior variance of 2.

6.2.2.1 Determining Optimal Prior Variance

A grid search was performed in order to determine the optimal value of prior variance. Figure 6.7 shows the model performance in terms of accuracy and average log probability for a range of prior variance values. It can be seen that a prior variance of around 2 produces the highest accuracy in both models. The variance values that produce optimal performance in terms of average log probability are smaller than those which are optimal for accuracy. This trend is more extreme for the point level Bradley-Terry model. The differences observed between the point and match level models can be attributed to the fact that we expect less extreme differences in skills within the point level model and therefore a tighter prior. This is because low skilled players win a much larger percentage of points against high skilled players than they do matches. However, even though there is smaller differences between the fitted skills in the point level model the final predictions may still be equally confident. This is because in the point level model the fitted skills are used to predict point winning probabilities which are then converted to match winning probabilities using the Markov chain described in Section 4.1. In the Markov chain, small differences in point winning probabilities result in much larger differences in match winning probabilities. By contrast, in the match level model, the fitted skills are used to predict the match winning probabilities directly.

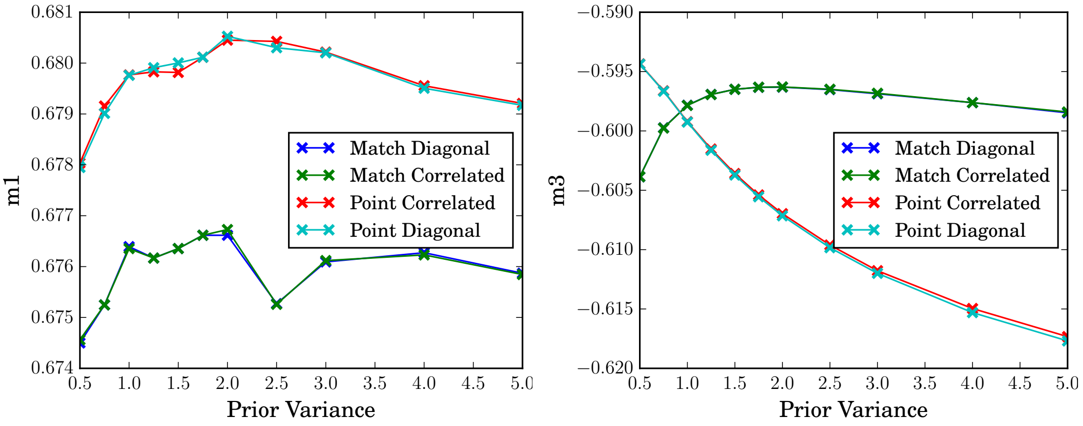

Figure 6.7: Model performance vs prior variance for point and match level Bayesian Bradley-Terry models. Accuracy left and average log probability right. The posteriors are fitted for both

6.2.2.2 Comparison of Correlated vs Diagonal Covariance

From Figure 6.7 it can be seen that the predictive performance of models where the skills are fitted using a full covariance is almost identical to when the skills are fitted using a diagonal covariance. In order to further investigate, we plot a histogram of the differences in predictions between a model where a full covariance is used and a model where a diagonal covariance is used (Figure 6.8). Additionally, we also plot a histogram of the values of the off diagonal terms in an example full covariance fit (Figure 6.8). From these plots it can be seen that: Firstly, although there are non-zero covariance terms, the magnitude of these terms is extremely small in comparison the marginal variances (diagonal terms) which can be seen in Figure 6.6. Secondly, the predictions between full and diagonal covariance models typically differ by less than 0.1%. Both these points suggest that the approximate posterior fitted when a full covariance matrix is used is almost identical to that fitted in the diagonal case. This could be because the true posterior simply isn't shaped in a manner that can be fitted any better with a full covariance than a diagonal covariance. It is not clear why this would be the case, since intuitively one would expect the skills of players that have played against each other to be correlated in some way. It should be noted that during the optimisation, a lower cost is achieved when fitting the full covariance compared to the diagonal covariance but the difference is extremely small. There is scope for further work to better understand how the shape of the true posterior differs from that of the approximate posterior. This would provide further insight into why using the full covariance matrix has almost no effect on the final predictions in this model.

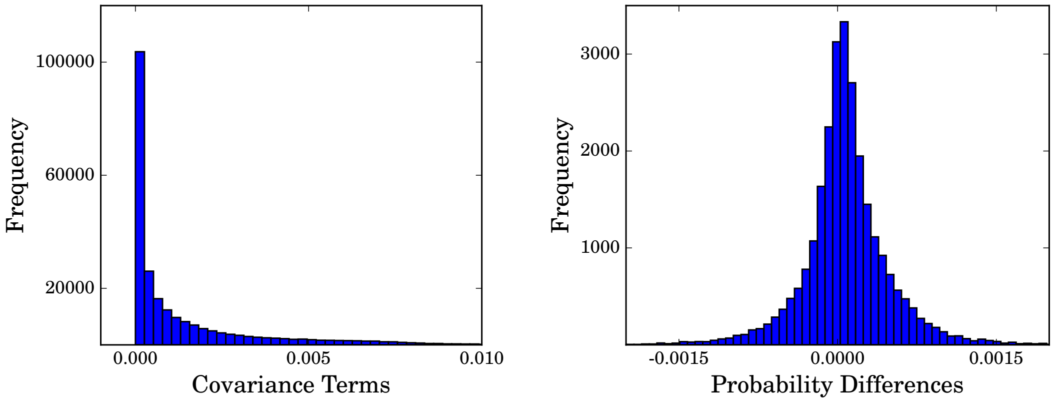

Figure 6.8: Histogram of off diagonal terms in a fitted full covariance matrix (left). Histogram of the probability differences in predictions between full and diagonal covariance fitted models (right). Both for match level model with a prior of 2.

6.2.2.3 Comparison Bayesian vs Maximum Likelihood

Table 6.5 provides a summary of the predictive performance of point and match level Bradley-Terry models fitted using both a Bayesian approach and penalised maximum likelihood. The respective regularisation constants and prior variances are chosen in order to maximum accuracy in each case.

Model	Accuracy	Average Probability	Average Log Probability	ROI
Match ML	67.64%	0.584	-0.597	-13.16%
Match Bayesian	67.66%	0.587	-0.596	-13.93%
Point ML	67.99%	0.613	-0.606	-6.05%
Point Bayesian	68.05%	0.616	-0.607	-6.01%

Table 6.5: Performance summary of Bayesian and Maximum Likelihood (ML) fitted point and match level Bradley-Terry models.

It can be seen that the Bayesian fitted models appear very slightly superior to the penalised maximum likelihood models. However, the difference is extremely small and is therefore unlikely to justify the additional cost and complexity associated with implementing the Bayesian case. As a further comparison we plot histograms of the differences in predictions between a the maximum likelihood and Bayesian based models (Figure 6.9). This shows that the probabilities predicted by the respective models typically differ by less than 2%. This further highlights the similarity of the final predictions produced by both methods.

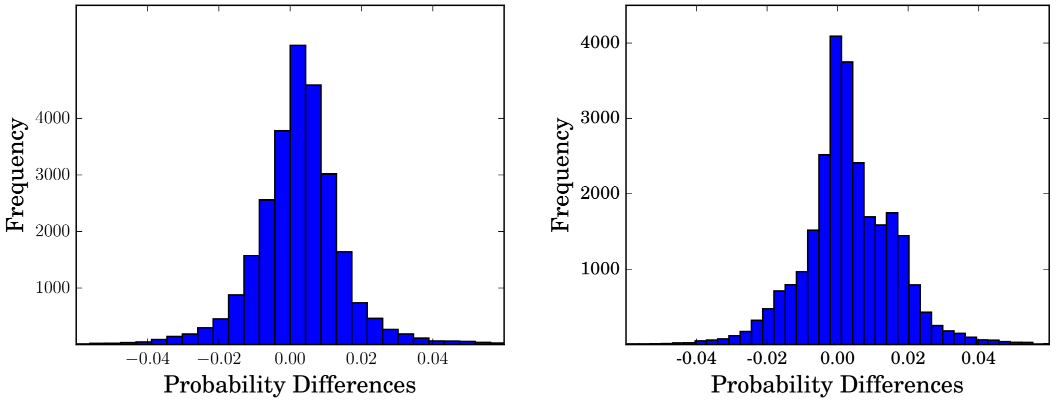

Figure 6.9: Histogram of probability differences between Bradley-Terry models fitted using maximum likelihood and approximate inference. Point level model left and match level model right. A prior variance of 2 and diagonal covariance is used in both Bayesian models and regularisation constant of 0.0001 in the maximum likelihood models.

6.3 Free Parameter Point Model Results

This section discusses the results for the Free Parameter Point model described in Section 4.5. The model parameters were fitted using both a Bayesian approach and penalised maximum likelihood. As with the previous model, the optimal settings for the prior variance and L2 regularisation constant were determined based on a grid search. The aim of experiments relating to this model was to show whether it improved upon the performance of point level Bradley-Terry model.

6.3.1 Maximum Likelihood Fitting

Figure 6.10 shows a plot of the performance of the Free Parameter Point model across a range of regularisation constants in the first 3 performance metrics. For comparison the point based Bradley-Terry model is also shown. The Free Parameter Point model has the maximum score in all 3 metrics, however is not consistently better than the

Bradley-Terry model across all regularisation values. Overall the performance of both models is similar.

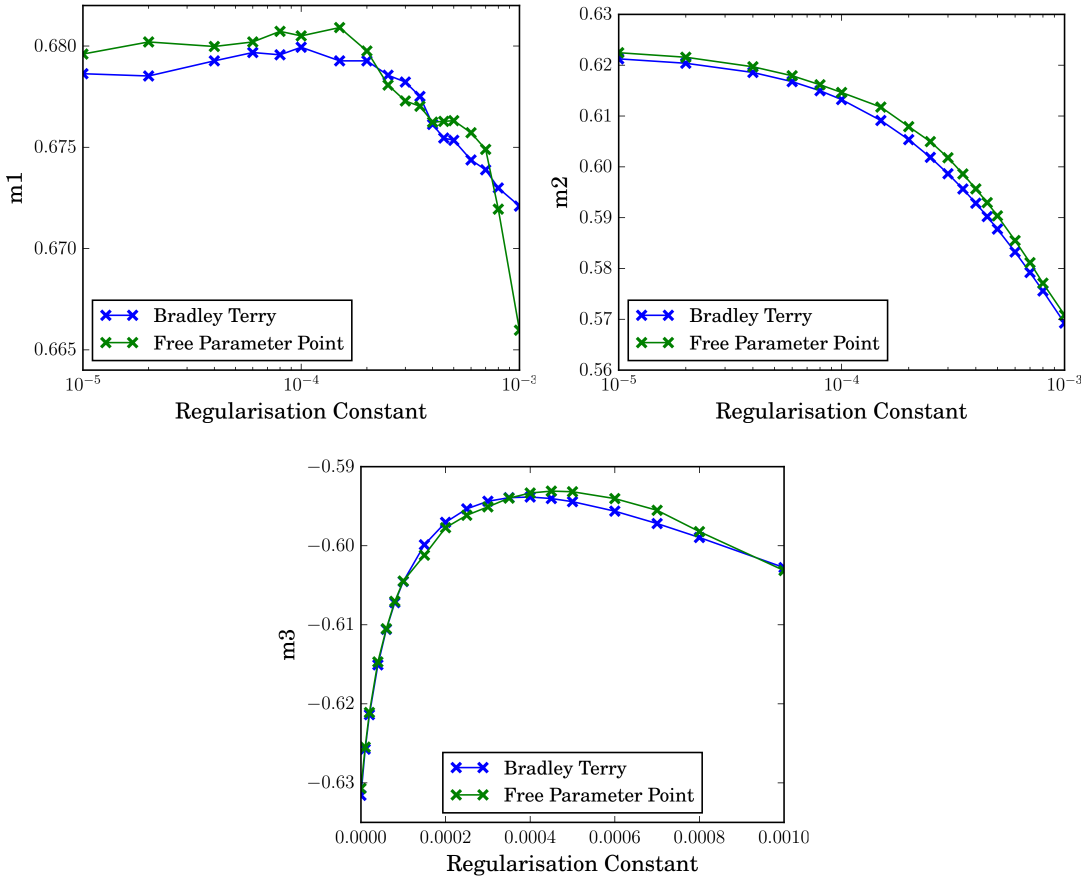

Figure 6.10: Performance of regularised Free Parameter Point model and point level Bradley-Terry model. Accuracy (top left), average probability (top right) and average log probability (bottom). For all metrics higher is better.

6.3.2 Bayesian Fitting

Figure 6.11 shows the results of the prior variance grid search for the Bayesian fitted model. Again, the point based Bradley-Terry model is also shown for comparison. Due to the similarity in performance of models fitted using full and diagonal covariance matrices, the results for diagonal covariance are omitted. The differences in performance between the two models are consistent with those observed in the maximum likelihood case. However, here the Free Parameter Point model is consistently superior across almost all prior variance values tested.

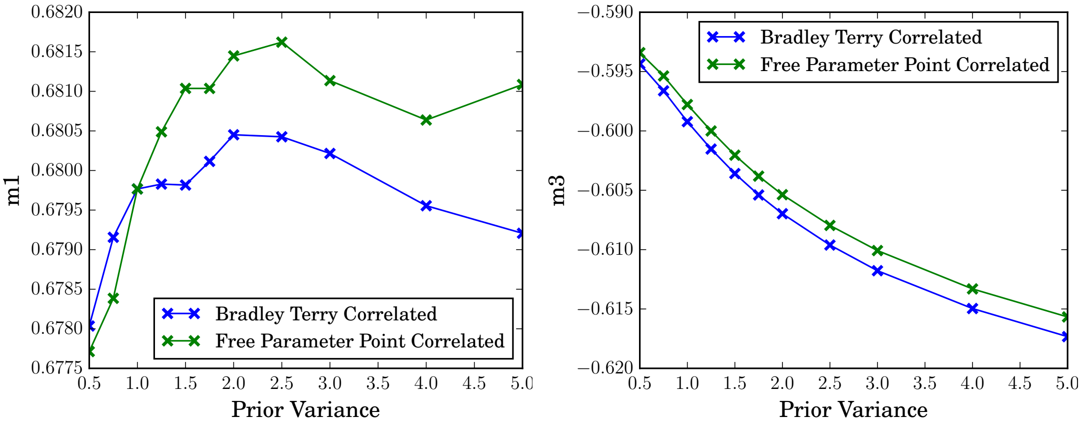

Figure 6.11: Performance of Bayesian Free Parameter Point model and point level Bayesian Bradley-Terry model. Accuracy (left) and average log probability (right). For all metrics higher is better.

6.3.3 Comparison to Bradley-Terry Point Level Model

Both the point level Bradley-Terry model and free parameter point model predict point winning probabilities which are converted to match winning probabilities using the Markov chain. However, the free parameter point model has two free parameters per player and predicts distinct probabilities for a player winning a point on their own serve and on their opponents serve. In contrast, the point level Bradley-Terry model only has one free parameter per player and simply predicts the probability of a player winning any point rather than splitting the points into two classes. Of the two models, we would expect the point level Bradley-Terry model to have inferior performance. This is because it is based on more simplistic assumptions and also has less parameters so should be less flexible in general. The results, which can be seen in Figures 6.10 and 6.11, support this hypothesis as the performance of the free parameter point model does appear slightly superior. Table 6.6 also provides a summary of the performance of both models for Bayesian and maximum likelihood fitted cases. However, the difference in performance is very small which suggests that there is minimal gains from modelling points in two classes as opposed to generally. Alternatively, the free parameter point model is too simplistic to properly exploit the potential performance gains from modelling points in two classes.

Model	Accuracy	Average Probability	Average Log Probability	ROI
ML Bradley Terry	67.99%	0.613	−0.605	−6.05%
Bayesian Bradley Terry	68.03%	0.617	−0.610	−6.00%
ML Free Parameter Point	68.05%	0.615	−0.605	−5.78%
Bayesian Free Parameter Point	68.16%	0.618	−0.608	−5.60%

Table 6.6: Performance summary of Free Parameter Point model and point level Bradley Terry model. For all metrics higher is better. Maximum likelihood (ML) models are for a regularisation constant of 0.0001 and Bayesian models for a prior variance of 2.5.

6.4 Surface Factor Model Results

This section discusses the results for the surface factor model described in Section 4.6. Within this model, each player and also each surface is represented by a vector of positive parameters. Player's then exhibit an overall skill on each surface based on their own parameters and the parameters of that surface. This is a new approach to addressing the effects of surface type in tennis. Surface weighting, described in Section 5.2, is the dominant approach used in previous work. The aim of the results presented in this section is to compare the performance of the two different approaches.

We explored fitting models with different numbers of player and surface parameters and compared the performance of these models against a standard surface weighted Bradley-Terry model. A summary of the performance of each of these model is given in Table 6.7. It can be seen that the model with only a single factor is the worst performing of all the models, including the surface weighted model. This is expected since this model is equivalent to a standard Bradley-Terry model without any surface weighting. The model accuracy improves with the addition of a second factor but larger numbers of factors appear to make little difference.

The results presented in Table 6.7 are all based on a regularisation constant of 0.0001. This was chosen with consideration to the results in the previous section and kept constant across all models in order to provide a fair comparison. However, because models differ in the number of parameters the effect of the same regularisation constant may

Model	Accuracy	Average Probability	Average Log Probability	ROI
Surface Weighted	67.99%	0.613	−0.605	−6.05%
1 factor	67.67%	0.613	−0.612	−6.93%
2 factors	68.16%	0.620	−0.613	−5.87%
3 factor	68.19%	0.621	−0.614	−5.85%
4 factors	68.19%	0.618	−0.614	−5.47%

Table 6.7: Performance of surface factor models for different numbers of factors. The surface weighted model relates to a standard Bradley-Terry model with surface weighting. All models are based on point level information and are fitted using penalised maximum likelihood with a regularisation constant of 0.0001. Predictions for the surface factor models are averaged from 10 repeated optimisations with different random initialisations.

not be equivalent. We therefore consider the performance of several of the models across a full range of regularisation values. This can be seen in Figure 6.12. The surface factor models appear to provide a slight improvement in accuracy in comparison to the surface weighted Bradley-Terry model. However, the maximum scores achieved by all of the models in average log probability is approximately equal. Additionally, the surface factor models require larger regularisation constants to reach the maximum score in this category.

The surface factor models rely on random initialisation which is discussed in Section 5.5.6. A consequence of this is that the solution of any optimisation depends upon the initialisation which is used. This means that predictions and therefore performance scores may be change if the is model retrained. An important consideration is what level of uncertainty this creates in the performance scores shown in Table 6.7. In order to quantify this uncertainty, we re-run the 3 factor surface model 20 times and examine the standard deviation in each of the performance scores. We find that the scores are consistent with a standard deviation of ±0.04%, ±0.0001, ±0.0008, ±0.14% in each of the respective metrics shown in Table 6.7. This could suggest that: Although the parameter space in each optimisation has multiple maxima, each of these maxima are equivalently good and therefore the predictions differ little depending upon which maxima is reached. The uncertainty is large enough to account for the small differ-

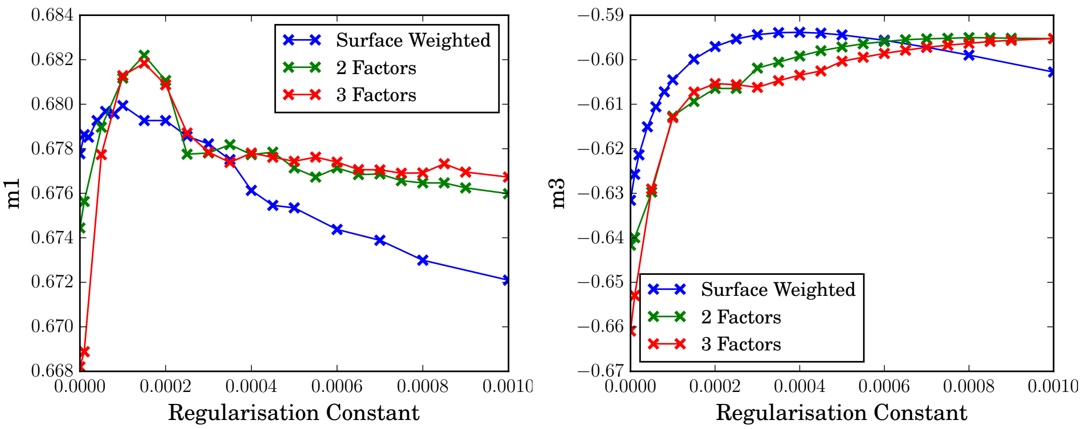

Figure 6.12: Performance of weighted Bradley-Terry model and surface factor models with 2 and 3 parameters for a range of regularisation constants. Accuracy left and Average log probability right. Predictions for the surface factor models are averaged from 10 repeated optimisations with different random initialisations.

ences in scores between the models with 2, 3 and 4 factors. It is therefore concluded that any performance difference between these models is not significant.

6.5 Joint Optimisation Time Series Model Results

The results in this section relate to the time series model described in Section 4.7, applied to a point level Bradley-Terry model using the joint optimisation approach described in Section 5.6.1. In this model, 3 years of previous historical data is broken into n time periods. Each player is then treated as having a distinct skills which are constant for all matches in a period. These skills are then jointly optimised using the variational procedure described in Section 5.5. Due to computational constraints, we only consider breaking the 3 year history into a maximum of 4 time periods. Table 6.8 provides a summary of the performance of models for different values of n, up to 4. A single time period means that each player is treated as having one constant skill for the full 3 years. This is equivalent to a standard Bradley-Terry model but without any recency weighting applied to matches. The performance of the weighted Bradley-Terry model is also shown in Table 6.8 for comparison. From Table 6.8 it can be seen that the performance of the time series model improves when the history of matches is broken into more time periods. This is expected since it means the variation in player skills is being modelled in greater resolution. Notability the performance of the time series

Model	Accuracy	Average Probability	Average Log Probability	ROI
Recency Weighted	67.98%	0.609	−0.599	−6.05%
1 Period	67.76%	0.607	−0.604	−7.25%
2 Periods	68.06%	0.617	−0.608	−6.10%
3 Periods	68.15%	0.617	−0.604	−5.85%
4 Periods	68.25%	0.616	−0.603	−6.32%
4 Periods Refined	68.33%	0.617	−0.601	−5.79%

Table 6.8: Performance of the jointly optimised time series model for different n. The 'Recency Weighted' model is a simple Bradley-Terry model in which matches are weighted according to their recency. All models are based on point level information and are fitted using a Bayesian approach described in Section 5.5 with a prior variance of 1 and Diagonal covariance matrix. A drift parameter of 0.9 (α in equation 4.9) is used in all of the time series models. For all metrics higher is better.

model with only 2 time periods is similar to that of the recency weighted model. This shows that the recency weighting method described in Section 5.2.1 is not a particularly effective way to account for the variation of player skills since its performance can be matched by even a simple time series model.

The final entry in Table 6.8 relates to a time series model where the 3 year history is broken up into unequal chunks of time rather than equal chunks. These chunks consist of two 12 month periods followed by an 8 month period and then a 4 month period. The effect of this is that the player skills are modelled in greater resolution towards then end of the 3 years at the expense of lower resolution at the start. This is a favourable trade off as the predictions are always made based upon the player skills at the end of the 3 years. If computation time wasn't a problem then it wouldn't be necessary to consider this strategy at all since the full 3 years could be broken into 4 month (or finer) chunks. However, we employ the strategy since it allows us to achieve finer resolution where it matters but at the same cost of the standard 4 time period model. It can be observed that this approach results in improved performance.

An example of the fitted skills for 3 players through time based on the refined 4 time period model is provided in Figure 6.13. This confirms that the players skills do vary a meaningful amount within the 3 year span considered.

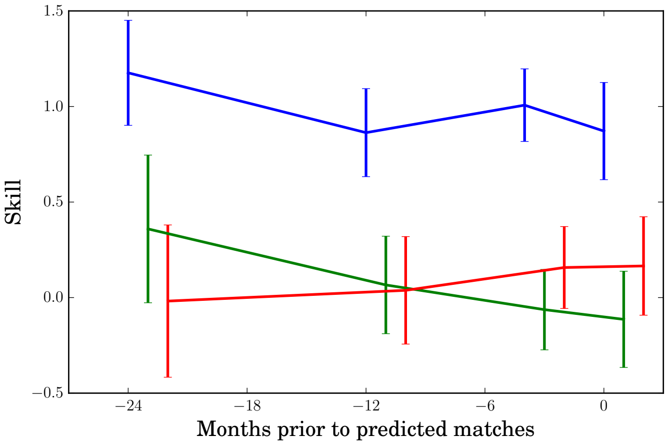

Figure 6.13: Plots of fitted skills for 3 players through time for the 4 step refined series model. Each player is one colour. The error bars are the standard deviation of the marginal skill posteriors at different points in time. The model treats all matches that happen in between two error bars as if they happen at the time of the error bar ahead.

6.5.0.1 Exploring Different Drift Values

We explore how changing the drift parameter (α) in equation 4.9 effects the performance of the time series model by testing a range of different values. The results are shown in Table 6.9. It can be seen that the model performance is reasonably insensitive to the exact value of drift which is used. Generally all values in the range of 0.75 - 0.95 provide good performance.

It should be noted that we consider α as standardised to the drift for a 12 month period of time. This value is then rescaled to suit the actual lengths of the different time periods. If we didn't consider drift in this manner then a parameter of 0.9 would have different overall effect depending upon length of the time period. This standardisation also applies to the results given in Table 6.8 which are all based on a α value of 0.9.

α	Accuracy	Average Probability	Average Log Probability	ROI
0.75	68.24%	0.615	−0.601	−6.41%
0.8	68.22%	0.615	−0.600	−6.22%
0.85	68.28%	0.616	−0.601	−5.93%
0.875	68.29%	0.617	−0.601	−5.71%
0.9	68.33%	0.617	−0.601	−5.79%
0.925	68.24%	0.617	−0.602	−5.93%
0.95	68.20%	0.618	−0.603	−5.84%

Table 6.9: Performance of the 4 step refined time series model for different drift parameters. For all metrics higher is better. Lower drift values allow for greater variation in skills whilst higher values encourage the skills to change more slowly.

6.6 Filtered Time Series Model Results

The results in this section relate to the time series model described in Section 4.7 which is applied using the filtered approach described in Section 5.6.2. In this model, the skills of all players are jointly updated after each round of every tournament using the variational procedure described in Section 5.5. The updated skills at each step are used as prior for the update at the next step. In between updates a drift is applied to the skills according to the length of time between the tournament rounds, measured from the start of one round to the start of the next. We explore applying the filtered time series model to point, game and match level data and also fitting posteriors with both full and diagonal covariance matrices at each step. Table 6.10 provides a summary of the results and Figure 6.14 shows an example plot of the progression of the skills over time for several players. From Figure 6.14, it can be seen that the skills of players do vary considerably over time. It can also be seen that the uncertainty in the skills in this model are smaller than those observed in the joint optimisation model (Figure 6.13). A possible reason for this is that in the joint optimisation model, the skills are only based on the previous 3 years of data, whilst in this model, they are based on the full history of matches.

It can be seen from Table 6.10 that the correlated covariance matrix models have superior performance to the diagonal covariance matrix models. This is in notable contrast

Model	Accuracy	Average Probability	Average Log Probability	ROI
Point Level Diagonal	68.21%	0.615	−0.598	−6.75%
Point Level Correlated	68.48%	0.618	−0.599	−6.12%
Game Level Diagonal	68.08%	0.600	−0.591	−10.07%
Game Level Correlated	68.26%	0.604	−0.591	−9.57%
Match Level Diagonal	67.87%	0.584	−0.594	−14.69%
Match Level Correlated	68.04%	0.589	−0.592	−14.97%

Table 6.10: Performance of point, game and match level filtered time series models. All models are based on a drift parameter of 0.9 and prior variance of 1. For all metrics higher is better.

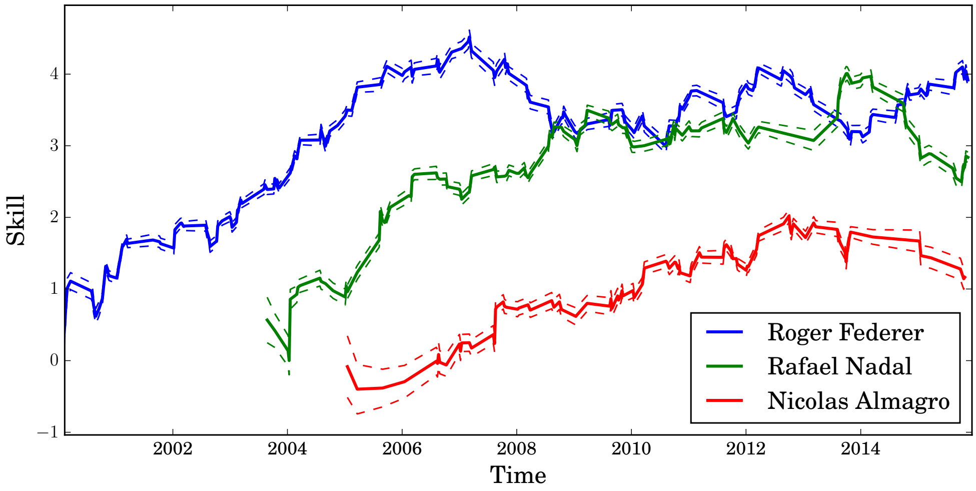

Figure 6.14: Example progression of player skills through time in Match level model. The solid lines are the means of the skills and the dotted lines are 1 standard deviation on either side of the mean.

to the results in Section 6.2.2 which showed virtually no difference in performance between using the correlated and diagonal covariance matrices. However, there are reasons why we would expect using a correlated covariance to be beneficial in this case which do not apply to the model in Section 6.2.2: This model uses filtering to sequentially update the skills of players as match outcomes are observed. In this process information is propagated forward in time. If the filtering updates were exact, then the estimates of all player skills at the current point in time would always be correct. However, in our model we only approximately update the skills at each step using the variational procedure in Section 5.5. Due to these approximations some information is lost during the forward propagation process, resulting in inaccurate estimation of the skills at the current point in time. This problem can be illustrated by considering a model where the skills of each player are approximately updated immediately after every match. If several matches are played on the same day then it's clear the ordering of the matches will effect the final estimates of the skills. We reduce this issue in our models by jointly updating all of the matches within one round of a tournament at the same time, so that the result is independent of the match ordering within that round.

By using a correlated covariance matrix instead of a diagonal covariance the problem described above is reduced further because less information is lost between updates. For example, consider a player who gets knocked out at some point in the first half of a tournament. If a diagonal covariance matrix was being used then their skill would not change in the updates for rounds in the second half of the tournament. However, if a full covariance matrix was used then their skill could change in the updates for later rounds even though they played no matches in those rounds. This is because correlations between them and other players would be present in the prior covariance at each update, so their skill would be adjusted as more information is revealed about players that they recently played against. A consequence of this is that at every update the optimisation contains the full set of players. This is considerably more expensive than the diagonal case where each optimisation need only contain the subset of players who played matches in that tournament round. It would be possible to also fit correlations based on updates with subsets of players. However, due to time constraints, this has not been explored in this project and it is unknown whether it would result in the same performance improvements shown in Table 6.10.

6.6.0.1 Comparison to Joint Optimisation Time Series Model

Both the filtered and joint optimisation time series models are based on the same underlying model (Section 4.7) where each player is treated as having distinct skills at different points in time. Comparatively the chunks of time considered in the joint optimisation model a far coarser than in the filtered model. However, despite this it still achieves almost the same performance as the correlated filtered model. It is possible that if the joint optimisation model was fitted at a finer resolution then its performance would surpass the filtered model. The reason we would expect the performance of the joint optimisation model to be better, is the information loss issue in the filtered model which is described previously. This issue is completely avoided in the joint optimisation case since the skills for all points in time are fitted together. It should be noted that only a diagonal covariance matrix was used in all of the results for the joint optimisation model in Section 6.8.

The cost associated with evaluating the joint optimisation model is much higher than the filtered model. However, the filtered model requires the updates to be performed sequentially. This means that evaluating the model can only be performed on a single core. In contrast, the evaluation of the joint optimisation model can be spread over multiple cores, making it more scalable. Additionally, the performance of the filtered model is only superior for the correlated covariance matrix case which is likely to scale poorly if the number of players in the data set was increased.

6.7 Final Model and Test Set Results

As a final model, we aggregate predictions from the point, game and match filtered times series models in the previous section. These are fitted with a correlated covariance matrix and with a drift parameter of 0.9. We evaluate the final models performance on both the training and test set of data. Tables 6.11 and 6.12 provide a summary of the scores achieved by the aggregated model, along with the point model baselines, on the training and test sets respectively. Also shown is the score for a maximum likelihood fitted (unregularised) game level Bradley-Terry model which is also a model from the literature (McHale and Morton, 2011). It can be seen that the aggregated time series model is superior to both models from the literature. Additionally, the improvement is consistent across both the training and test set. This confirms that the model has not been over fitted to the training set through hyper-parameter selection and model choices.

Model	Accuracy	Average Probability	Average Log Probability	ROI
Bookmakers Baseline	70.28%	0.607	−0.565	–
Point Model Baseline	67.15%	0.588	−0.605	−10.38%
ML Game Bradley-Terry	67.83%	0.612	−0.611	−7.02%
Aggregated Time Series	68.62%	0.606	−0.586	−5.24%

Table 6.11: Final model training set performance. Approximately 26,700 predictions for matches between 2005 to 2015. For all metrics higher is better.

Model	Accuracy	Average Probability	Average Log Probability	ROI
Bookmakers Baseline	70.87%	0.610	−0.566	–
Point Model Baseline	67.59%	0.587	−0.608	−10.43%
ML Game Bradley-Terry	67.98%	0.610	−0.605	−6.00%
Aggregated Time Series	69.27%	0.607	−0.585	−6.54%

Table 6.12: Final model test set performance. Approximately 3,000 predictions for matches from 2016 to 2017. For all metrics higher is better.

In order to show whether the performance difference between the final model and models from the literature is consistent over time, we plot performance as a 12 month moving average (Figure 6.15). This confirms that the difference is consistent over time and therefore we conclude that the aggregated time series model is overall a superior model. However, it should be noted that the size of the performance difference is only moderate (0.75-1% greater accuracy) whilst the additional complexity involved in implementing the aggregated time series model is relatively high. It can be seen that the performance of the final model is still considerably below the bookmaker baseline.

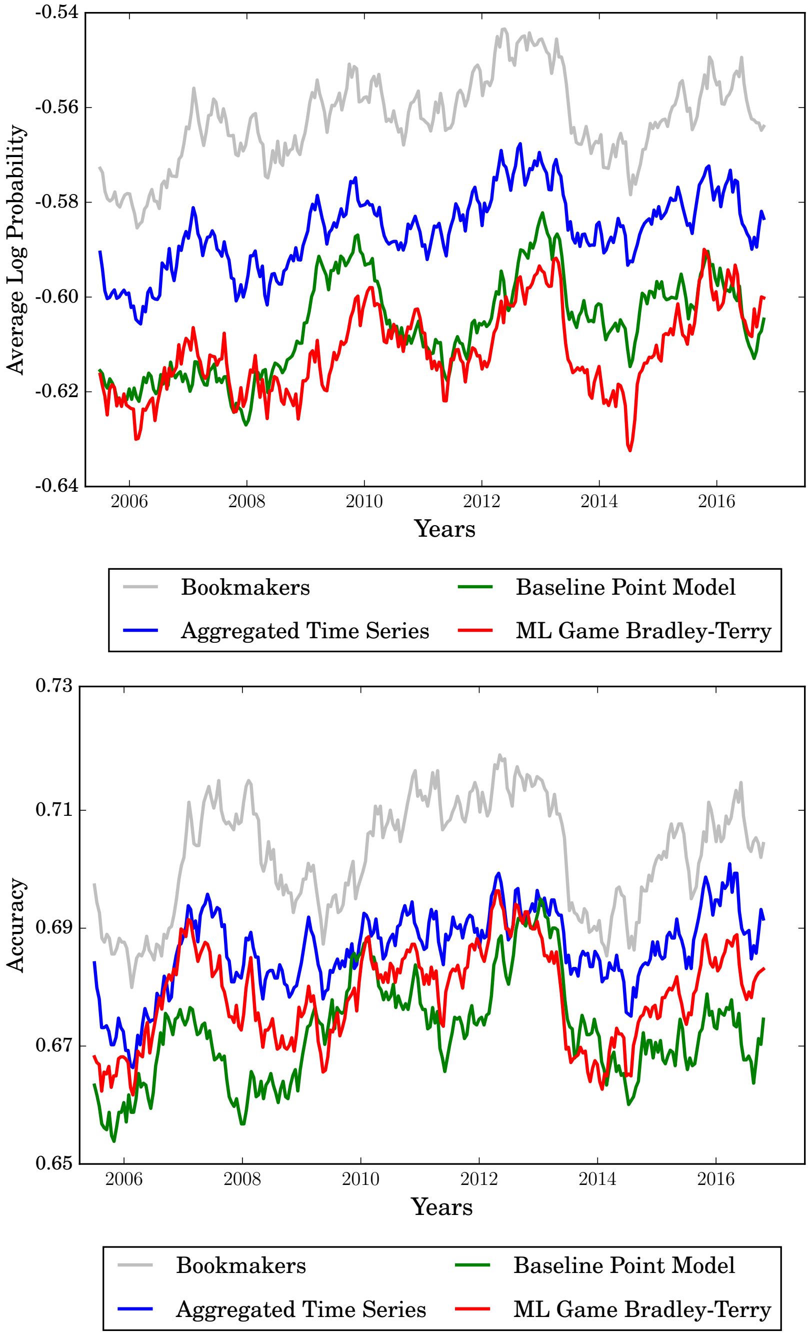

Figure 6.15: Model Accuracy (top) and Average Log Probability (bottom) over time based on a 12 month moving window. For both metrics higher is better.

Finally, we examine the calibration of the aggregated time series model shown in Figure 6.16. It can be seen that the model is fairly well calibrated but is slightly over confident at higher probabilities.

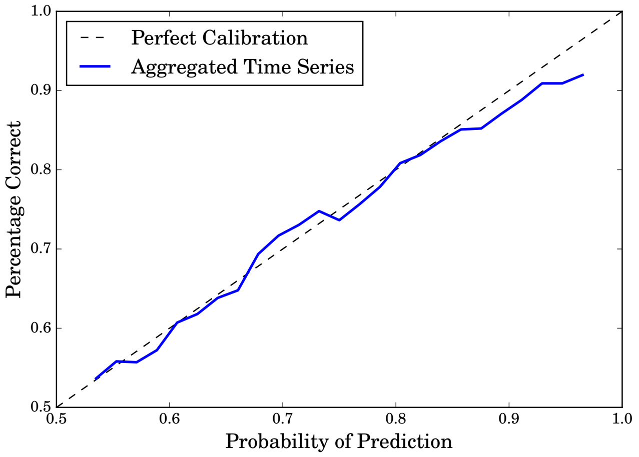

Figure 6.16: Calibration of the aggregated Time Series model based on test and training sets. Evaluated as a moving average in bins of 0.07. The x-axis relates to the models predictions and the y-axis to the percentage of time predictions of that probability were correct.

6.8 Improving Profit

In all of the results presented in the previous sections we have largely ignored profit. This section will access whether our final model described in the previous section can be made profitable. In order to do this, we consider creating a confidence threshold such that no bets are placed on any predictions with a confidence below this threshold. Confidence is defined as described in Section 5.7. Figure 6.17 shows a curve of the number of predictions against profit based on varying the confidence threshold. It is observed that profit improves gradually as more of the least confident matches are discarded. The model appears profitable if only around 2000 of the predictions are used. However, this is a very low percentage of the total predictions and the curve also appears unstable in that region. Therefore, it is considered unlikely that the model would actually be successful if used on future odds from the same distribution.

Throughout the project, we selected hyper-parameters and made model choices based primarily upon improving the score of the other performance metrics. It is possible that the model profitability would be improved if these choices were instead made with respect to maximising profit.

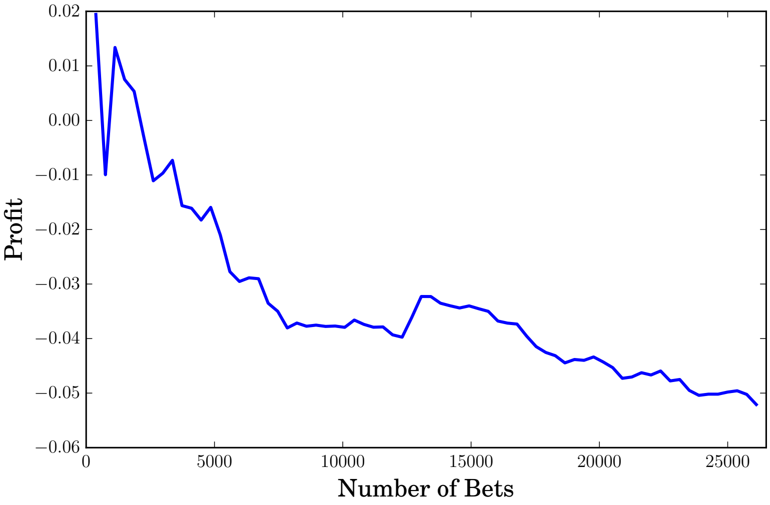

Figure 6.17: Profit vs number of predictions for the aggregated times series model. A lower number of predictions relates to stricter confidence threshold. Results are based on the training set of data consisting of matches from 2005 to 2015.

Chapter 7

Conclusion

The goal of this project was to improve upon the performance of leading tennis prediction models from the literature by exploring new ways to model the effects of surface type and the variation of player skills through time. Previous work in tennis modelling had predominately addressed these two areas by applying weightings to historical matches.

We implemented a time series model which assumes a Gaussian drift on player skills over time. Models of the nature have been explored in other fields but had previously not been applied to tennis prediction. We apply updates in our time series model by using a variational method which we derived based on stochastic variational inference. We also use this method in an additional time series model where we jointly fit the skills of players at multiple points in time, rather than applying filtering updates. We show that the performance of both time series models is superior to an equivalent recency weighted model. Furthermore, we find that the performance of filtering based time series models can be improved by tracking correlations between players through time. Both the joint optimisation and filtering time series models give similar overall performance. The joint optimisation model may offer better performance if implemented in higher resolution, but the additional computational expense makes this challenging.

We develop a new surface factor model where both players and surfaces are modelled using multiple free parameters and show that this produces better accuracy than an equivalent surface weighted mode. We find that using more than 2 factors gives no meaningful improvement. It is likely more complex interactions between the model parameters are required in order to better capture surface effects.

We explored fitting a range of standard Bradley-Terry models based on point, game and match level information using both penalised maximum likelihood and our derived variational procedure. We find that fitting the models with a Bayesian approach results in performance which is only marginally better than the same models fitted using regularised maximum likelihood. Additionally, using a correlated covariance matrix in these models makes almost no difference to the predictions. We develop an extended Bradley-Terry model with an additional free parameter per player in order to model distinct service probabilities of each player. However, this model only gives a very slight improvement in performance compared to a standard point level Bradley-Terry model.

Finally, we show that aggregating predictions from separately trained point, game and match models can provide a small performance improvement. We train a final aggregated times series model and demonstrate its superior predictive performance to two prediction models from the literature. Our final model is not profitable when compared against historical bookmaker odds. However, we do show that it can become close to profitable by taking into account the confidence of predictions by considering the variances of the player skills used in the predictions.

7.1 Future Work

The following key areas are identified as meriting further research:

7.1.1 Expanding the Data Set

In this project, we trained models using only data from ATP tour level matches. However, for some players this meant that there was only a small amount of data available because they play the majority of their matches at a lower level. This is highlighted by the fact that the uncertainty present in the many of the skills in the Bayesian fitted models is high (Figure 6.6). We carried out a preliminary experiment where we evaluated a standard maximum likelihood game level Bradley-Terry model but included data from both ATP and Challenger level matches. Note that predictions were only made for the same subset of matches as the equivalent ATP only model. Our results suggest that there is improvements to be made from including the additional data. We therefore recommend that this is explored in future work.

7.1.2 Combing the Surface Factor and Time Series Models

In separate models, we demonstrated that both surface effects and the variation of player skills through time could be successfully addressed using alternative approaches to match weighting. Future work could implement a time series version of the surface factor model. The variational procedure that we derive in Section 5.5 would be suitable for performing the updates in this model if it was assumed that the surface weighting parameters were constant scalar values. This should be a reasonable assumption since the physical properties of the surfaces do not change over time. Additionally, we would expect the posteriors of the surface parameters to be extremely peaked anyway, due to the large amount of data for them.

7.1.3 Additional Factors

Although our surface factor model represents players with multiple parameters, it still does not allow for non-transitive relationships between players on a single surface. This can be achieved by including non-linear interactions between the parameters of players (Stanescu, 2011). Expanding the models in this project to include additional player parameters with non-linear interactions presents a further opportunity for future work.

Appendix A

Appendix

A.1 Variational Inference Likelihood Approximation

For following expectation:

$$\Big\langle \log(\sigma(x)) \Big\rangle_{N(v;\,0,1)} \approx \Big\langle -a\exp(-\frac{1}{2}bx^2) - 0.5x\,\mathrm{erf}(cx) + 0.5x \Big\rangle_{N(v;\,0,1)}$$

Performing the expectations on the right hand side per term results in:

$$\begin{aligned}
\Big\langle -a\exp(-\frac{1}{2}b(\mu+\tau v)^2) \Big\rangle_{N(v;\,0,1)} &= -\frac{a}{\sqrt{b\tau^2+1}}\exp(-\frac{b\mu^2}{2(b\tau^2+1)}) \\
\Big\langle -0.5(\mu+\tau v)\,\mathrm{erf}(c(\mu+\tau v)) \Big\rangle_{N(v;\,0,1)} &= -0.5\mu\,\mathrm{erf}\Big(\frac{c\mu}{\sqrt{1+2c^2\tau^2}}\Big) \\
&\quad -\frac{1}{\sqrt{2\pi}}\exp\Big(-\frac{c^2\mu^2}{2c^2\tau^2+1}\Big)\frac{c\tau^2}{\sqrt{c^2\tau^2+0.5}} \\
\Big\langle 0.5(\mu+\tau v) \Big\rangle_{N(v;\,0,1)} &= 0.5\mu
\end{aligned}$$

Bibliography

Barnett, T. and Clarke, S. R. (2005). Combining player statistics to predict outcomes of tennis matches. *IMA Journal of Management Mathematics*, 16(2):113–120.

Birlutiu, A. and Heskes, T. (2007). Expectation propagation for rating players in sports competitions. In *PKDD*, pages 374–381. Springer.

Boulier, B. L. and Stekler, H. O. (1999). Are sports seedings good predictors?: an evaluation. *International Journal of Forecasting*, 15(1):83–91.

Bradley, R. A. and Terry, M. E. (1952). Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Clarke, S. R. and Dyte, D. (2000). Using official ratings to simulate major tennis tournaments. *International Transactions in Operational Research*, 7(6):585–594.

Crooks, G. E. (2009). Logistic approximation to the logistic-normal integral. *Technical Report Lawrence Berkeley National Laboratory*.

Glickman, M. E. (2001). Dynamic paired comparison models with stochastic variances. *Journal of Applied Statistics*, 28(6):673–689.

Gneiting, T. and Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378.

Herbrich, Dangauthier, R., Minka, T., Graepel, T., et al. (2007). Trueskill through time: Revisiting the history of chess. In *NIPS*, pages 337–344.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. W. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14(1):1303–1347.

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

Klaassen, F. J. and Magnus, J. R. (2001). Are points in tennis independent and identically distributed? Evidence from a dynamic binary panel data model. *Journal of the American Statistical Association*, 96(454):500–509.

Knottenbelt, W. J., Spanias, D., and Madurska, A. M. (2012). A common-opponent stochastic model for predicting the outcome of professional tennis matches. *Computers & Mathematics with Applications*, 64(12):3820–3827.

McHale, I. and Morton, A. (2011). A Bradley-Terry type model for forecasting tennis match results. *International Journal of Forecasting*, 27(2):619–630.

Newton, P. K. and Keller, J. B. (2005). Probability of winning at tennis i. theory and data. *Studies in applied Mathematics*, 114(3):241–269.

Ng, E. W. and Geller, M. (1969). A table of integrals of the error functions. *Journal of Research of the National Bureau of Standards B*, 73(1):1–20.

Petersen, K. B., Pedersen, M. S., et al. (2008). The matrix cookbook. *Technical University of Denmark*, 7:15.

Sipko, M. and Knottenbelt, W. (2015). Machine learning for the prediction of professional tennis matches. *Master's thesis, Imperial College London, London, UK.*

Spanias, D. (2014). Professional tennis: Quantitative models and ranking algorithms. *Phd, Imperial College London, London, UK.*

Stanescu, M. (2011). Rating systems with multiple factors. *Master's thesis, School of Informatics, Univ. of Edinburgh, Edinburgh, UK.*

Stern, D. H., Herbrich, R., and Graepel, T. (2009). Matchbox: large scale online bayesian recommendations. In *Proceedings of the 18th international conference on World wide web*, pages 111–120. ACM.