Using Non-Standard Models for Football Predictions: Elo, TrueSkill, Neural Networks

by

Anton Bendrikov
URN: 6489183

A dissertation submitted in partial fulfilment of the requirements for the award of

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

May 2020

Department of Computing
University of Surrey
Guildford GU2 7XH

Supervised by: Paul Krause

I declare that this dissertation is my own work and that the work of others is acknowledged and
indicated by explicit references.
Anton Bendrikov
May 2020

© Copyright Anton Bendrikov, May 2020

Abstract

This paper aims to explore the application of non-standard models, such as Elo, TrueSkill and Neural Networks, to predict the outcome of football matches. While being widely used in other industries, Elo and TrueSkill have not been applied extensively in sports. Another method that has started gaining more attention, especially over the last decade, is the application of Neural Networks in sports forecasting. The main focus is to examine how these approaches perform with football by creating multiple systems that predict match outcomes. Specifically, the models aim to determine whether the match is going to finish in a Win, Draw or Lose.

The data for English Premiership, France Championnat, Germany Bundesliga and Italy Serie A competitions is used for tournaments from 2008 - 2009 until 2018 - 2019 to develop the models and determine how well a particular system can generalise across multiple competitions within the same sport. The effects that parameter tuning has on the models is recorded and modifications to the original systems are considered. Moreover, the developed Neural Network is applied to other sports to determine how well the models translate across a range of team-based sports. The produced systems are compared against each other as well as a well-known bookie using various metrics. Note, that the models are not targeted at making a profit from betting, but rather at maximising the prediction accuracy.

The Neural Network and Elo performed better on average than the bookie of choice, while TrueSkill displayed the worst levels of performance. Contrary to the common approach of feeding all available data to the model, it was discovered that adjusting system hyperparameters and extensions for individual competitions is necessary to maximise prediction accuracy. The Neural Network that was developed for football displayed promising results with other sports where amounts of training data were sufficient.

Acknowledgements

I want to thank my supervisor Paul Krause for helping me throughout this unusual project. It is because of his invaluable guidance and suggestions that I was able to produce this piece of work. I would also like to express gratitude towards my employer Gambit Research for introducing me to the world of gambling and forever spiking my interest in the industry. It is thanks to them that I have chosen this project. Special thanks goes to Jack Medley, who offered fruitful advice throughout this journey. Lastly, I want to say thank you to my family and friends, who have supported me greatly throughout this project and (sometimes) patiently listened to me talking about gambling for hours.

Contents

Key Terms

Term	Explanation
Competition	In this paper, competition is used to denote a series of recurring tournaments. For example, England Premiership, France Championnat, Germany Bundesliga and Italy Serie A are all examples of competitions (that are typically played every year).
Tournament	A series of matches with the goal of determining the best team/player. For example, England Premiership 2018 - 2019, England Premiership 2017 - 2018, France Championnat 2009 - 2010 are all separate instances of tournaments. Can be referred to as ‘season’.

Table 1: Definition of key terms

1. Introduction

1.1 Problem Background

Since the invention of online gambling, it has been gaining popularity. The rise of the market and the audience has been accompanied by an increase in the types of offerings for events. Before, customers were only able to bet on sports outcome, that is, win, draw (if applicable) or lose, now they were able to access a variety of different odds. For example, gamblers could now bet on the number of goals that a particular team will score (Moya F., 2012). In the UK alone, the gross gambling yield (GGY) has increased from £13.82bn from April 2016 to March 2017 to £14.58bn (April 2017 - March 2018) in just a year with remote casinos, betting and bingo contributing £4.72bn and £5.48bn for these periods respectively. The industry also accounted for 102,782 jobs in Great Britain in March 2019 (Gambling Commission, 2019).

Amongst the customers, some took it as a professional challenge to predict the outcome of a match, often with the incentive to make profits through gambling. Over the years, there have been many attempts at creating models capable of beating the bookies. Both bookmakers and professional gamblers are always looking for better ways to make profits. An obvious way to do this is to create a model that would predict the outcomes of events better than the current system does. One methodology, namely the Poisson model, became a standardised way of offering match predictions in the industry (Langseth H., 2013). However, a variety of other systems exist that are being used in other areas. Adopting models that have demonstrated successful applications in other industries could, perhaps, lead to a better way of predicting football outcomes. For example, one such system is Elo, which has been historically used in chess to rank players worldwide. Furthermore, the TrueSkill model, which was developed by Microsoft, has been successfully utilised in video games to calculate skill in team-based environments. Lastly, in this paper, application of Neural Networks in football is examined, which has received larger amounts of attention than the other two methods.

Betting Basics

Often, the odds are provided in a decimal form, such as 1.19, 4.5, etc. The numbers represent how much money the bookie will pay back for every £1 that is part of the bet. So if a bookie offers odds of 4.5 and a person wins that bet with a stake of £10, the person will get back £45 (with the original £10 being included in the winnings, resulting in £35 in profits). For the sake of simplicity, this excludes any additional fees the bookie may impose. Of course, if the person loses the bet, the entire stake of £10 is gone. The expected return in the case of a successful bet can be generalised using the following formula:

$$P = s * o \quad \textbf{(1)}$$

where $P$is profit, s is the bet amount (stake) and o is decimal odds for a particular outcome. The odds also represent how confident the bookie is in a particular outcome of the game, which is essentially the probability of that event occurring according to the bookie. For example, for the match Liverpool vs. Norwich on 9 August 2019, Bet365 offered 1.14 for a

win, 10 for a draw and 19 for away. By dividing 1 by the decimal odds and then multiplying by 100, decimal odds can be converted into probability. For the given example:

$$Bet365\ Win\ Probability = \frac{1}{1.14} * 100 = 87.72\%$$

$$Bet365\ Draw\ Probability = \frac{1}{1.10} * 100 = 10\%$$

$$Bet365\ Lose\ Probability = \frac{1}{1.19} * 100 = 5.26\%$$

When placing a bet, to secure a profit long-term, the gambler's confidence has to be more accurate than the bookie's. In this example, there is no point in placing a bet on a win if the gambler believes there is an 85% of winning as this will result in a loss long-term since the potential reward is less than the risk. Apart from trying to determine the actual probability of an event occurring, bookies also typically include a margin, which helps them to generate profit. The margin can be observed by summing the probabilities of every outcome. In the example:

$$Total\ Probability = 82.72\% + 10\% + 5.26\% = 102.98\%$$

The extra 2.98% is the margin that was included for this particular match, which can vary, generally somewhere between 2% to 6%. Generally, to gain profits, gamblers are looking for bets where a particular outcome is more likely than what the bookie claims it is, which includes the margin (Langseth H., 2013). It is also possible to make long-term profits by beating the bookie only on a particular outcome (for example, by only considering the likelihood of a draw in the first half of the match, which has been popular with specific types of bets), but this is out of the scope of this paper. In this project, the focus is only on predicting the results of the football matches and not on making a profit through gambling.

1.2 Project Description

The aim of this project is to explore multiple non-standard systems for football prediction. These include Elo, TrueSkill and Neural Networks. When developing the models, the effects of adjusting different parameters and introducing system modifications will be recorded. The similarities and differences in building the models will be documented and contrasted. To further examine how prediction models behave and generalise on football, multiple competitions will be considered. This will be used to determine how a change in competition will affect the systems' accuracy, offering insight into developing prediction models that generalise better. Moreover, the extent of Neural Network's generalisation will be examined by using data from distinct sports, such as rugby, cricket and others. Of course, the systems will be compared against each other as well as a well-known bookie for benchmarking.

To achieve this, thorough research will be conducted, first documenting common approaches in football prediction. Next, the existing applications of the proposed methodologies will be presented and the ideas behind them described in detail. Known

usages of TrueSkill, Elo and Neural Networks in other sports and industries will also be considered. Then, a variety of factors that may improve the systems will be considered, including the effects of tuning the hyperparameters as well as altering the original methods, for example, by introducing additional variables. The research will serve as a basis for model development, which will be done iteratively with the effect of each modification being recorded and explained.

To evaluate the models, a variety of metrics, which include prediction accuracy, Win/Lose F1, precision and recall, will be considered. These will be examined for each system (and a bookie) to determine if systems are better at predicting a particular subset of matches (for example, only win matches), which will also help set the course for future research. Additionally, the models are analysed cross-competition by using data from different leagues. This should help gain better understanding of the underlying nature of football matches. Note, that the systems only aim to maximise the prediction accuracy, that is, correctly classify the highest number of matches possible. The developed models do not aim to make a profit from gambling and should not be treated as such.

1.3 Project Aims

Aims of the paper are listed in this section, detailing what is set to be achieved as a result of this project. These goals are reflected upon in Evaluation.

Task 1: Research the existing ways of predicting match outcomes

Modern and historical approaches to producing predictions for football need to be researched and documented, any consistent assumptions and trends highlighted.

Task 2: Research existing implementations of Elo and TrueSkill to determine common applications and possible variations of the original systems

The findings from other papers that have successfully applied Elo and TrueSkill algorithms on various problems are to be researched and recorded. This would serve as a basis for creating Elo-based or TrueSkill-based models for football predictions, using other researchers' work as a guide for what has worked successfully and what system modifications are available.

Task 3: Find how other researchers applied Neural Networks in sports prediction

Research into Neural Network application in sports is needed to determine how they have been used to predict the outcomes of matches. This is an important aim that should help with creating a Neural Network model for football prediction.

Task 4: Ensure data quality

After loading the main dataset provided by Football-Data (nd), check that the data is not missing, malformed and meets all data quality criteria.

Task 5: Derive the relevant feature set from the original data

Using the original dataset and the literature review, choose relevant features for the selected algorithms. Where appropriate, derive necessary parameters using historical data for each competition.

Task 6: Develop Elo models

Using the conducted research and the final dataset, iteratively develop Elo models using. The change in the model behaviour across iterations should be recorded. Both the effects of the hyperparameters and any additions to the canonical implementations of the Elo system must be considered.

Task 7: Develop TrueSkill models

Similarly to Task 6, the research conducted in Task 2 should be used as a basis for developing a TrueSkill-based model, capable of predicting football matches. As with Elo, the models are built iteratively and a broad range of modifications is considered that includes the hyperparameters and potential extensions of the TrueSkill system.

Task 8: Create Neural Network-based system for football prediction

As in Tasks 6 and 7, the examined research is used as a platform for further model creation using Neural Networks. Based on other papers, relevant features are selected for the given model. Then, the Neural Network is adjusted to perform best for the given problem, namely football match prediction.

Task 9: Evaluate Neural Network generalisation across multiple sports

Neural Network, created as part of Task 8, is applied to other sports, such as rugby and cricket. This will help to determine how well the developed model for football generalises on other sports and if further tuning is required for the system to be usable in other sports.

Task 10: Evaluate the performance of the final models with each other as well as with one of the bookies

After finalising the models using the selected algorithms, evaluate their performance against each other. Moreover, check how models are performing in comparison to a bookie. This evaluation must go beyond a simple comparison of prediction accuracy and determine more detailed system characteristics. For example, comparing the models' prediction rate for wins and loses separately or considering more metrics, such as recall, accuracy and F1.

1.4 Structure of Report

Section 1: Introduction

This section introduces the topic of the paper, describing the basics of gambling and sets the objectives.

Section 2: Literature Review

In this section, work that is relevant to sports prediction is presented. Additionally, existing applications of TrueSkill, Elo and Neural Networks are examined.

Section 3: Analysis

In this section, the models that predict outcomes of football matches using TrueSkill, Elo and Neural Network approaches are developed. The whole process is thoroughly documented and the effects of calibrating the systems are recorded.

Section 4: Evaluation

This section presents a critical evaluation and comparison of the final models, not only determining the best model using the proposed algorithms but also comparing them to a bookie.

Section 5: Ethics

This section examines political, economic and social effects of gambling and betting, offering a balanced evaluation of positive and negative impacts.

Section 6: Conclusion

This section summarises the findings of the project and provides recommendations for future work.

2. Literature Review

2.1 Research Methodology

First of all, some papers on the topic have been found using general terms, such as ‘betting strategy’, ‘predicting football outcomes’ and other similar keywords. Then, references in these papers were followed and sources were explored. Getting more familiar with the topic produced more common keywords in the field, which were later used to identify additional papers.

The impact of research is another important factor that should be considered when selecting references. For this reason, the sources were then filtered, depending on how much influence they had within the area, which can be estimated by the number of citations. For example, Pollard R.’s 2008 paper titled ‘Home Advantage in Football: A Current Review of an Unsolved Puzzle’ has over 150 citations according to ResearchGate. Additionally, exploring the references that the paper includes is an important factor when assessing the quality of the given research. The final list of various research has been explored in detail.

2.2 Traditional Betting

Langseth H. (2013) introduces various statistical models commonly used amongst professional gamblers. The most popular system works under the assumption that the goals in football follow a Poisson distribution, an approach that was developed by Maher in 1982. The model would calculate attack and defence strength for each team based on normalised values for that league in such a way that the average for a particular competition is 1. Higher attack values mean that the team is likely to score more while lower defence suggests a team is expected to concede a relatively low number of goals. Detailed formulas, explaining how to derive probabilities using the Poisson distribution are presented in Appendix A.

From here, the model predicts the likelihood of one team scoring X goals and the other team conceding Y goals. This is done repeatedly for a reasonable number of goals (generally, the maximum is somewhere between 5 and 10) to compute a table of probabilities. An example can be seen in Figure 1, provided by Smarkets when evaluating a sample Tottenham vs Stoke match.

	Tottenham goals	0	1	2	3	4	5
Stoke goals	Poisson for number of goals per team	13.32%	26.85%	27.07%	18.19%	9.17%	3.70%
0	52.05%	6.93%	13.98%	14.09%	9.47%	4.77%	1.92%
1	33.99%	4.53%	9.13%	9.20%	6.18%	3.12%	1.26%
2	11.10%	1.48%	2.98%	3.00%	2.02%	1.02%	0.41%
3	2.42%	0.32%	0.65%	0.65%	0.44%	0.22%	0.09%
4	0.39%	0.05%	0.11%	0.11%	0.07%	0.04%	0.01%
5	0.05%	0.01%	0.01%	0.01%	0.01%	0.005%	0.002%

Figure 1: goal probabilities for home and away team for match Tottenham vs Stoke using Poisson distribution (Smarkets, nd)

Based on the figures provided in the table above, the most likely outcome is 2 - 0 with a probability of 14.09% and 1 - 0 being a close second with the probability of 13.98%.
This approach is successful to the extent that the goals follow a Poisson distribution. For example, Sheehan D. (2017) provides a graph that compares the predicted Poisson goals versus the actual goals of the English Premier League in the 2016 - 2017 season, which displays some Poisson-like properties (see Figure 2).

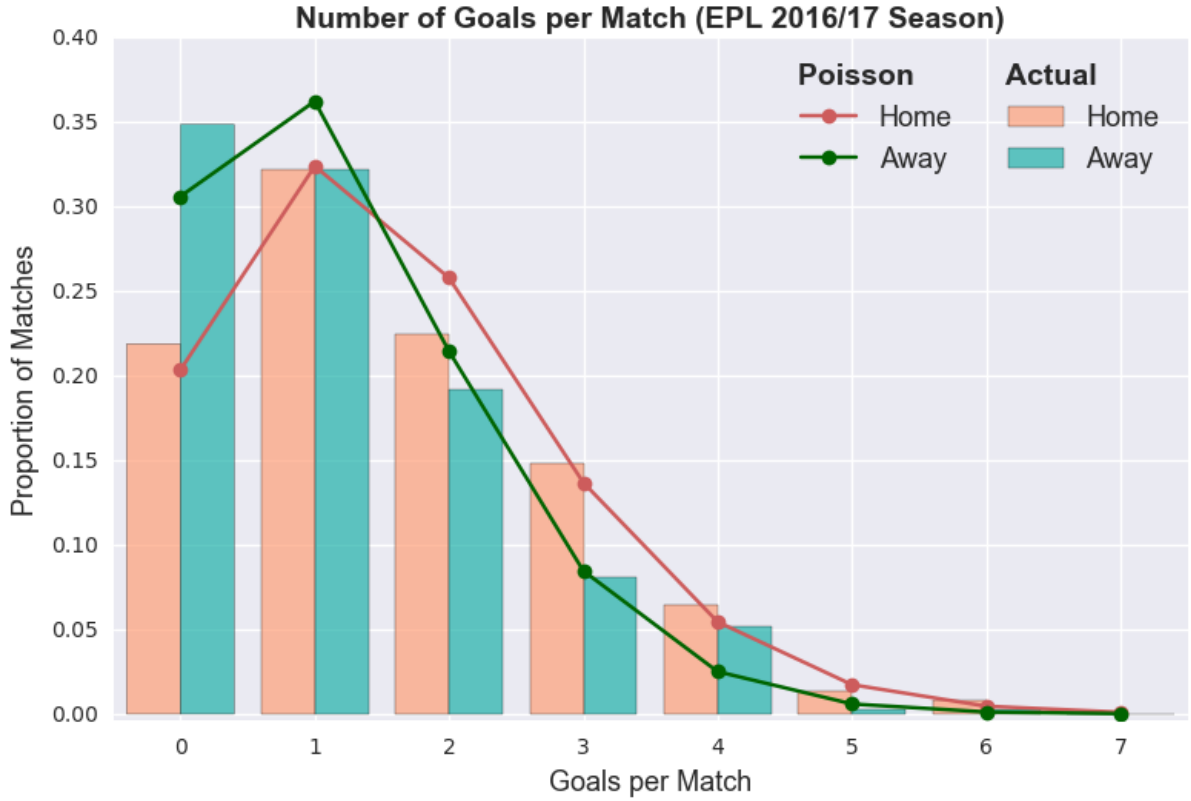

Figure 2: goal distribution as predicted by the Poisson model compared to the actual distribution (Sheehan D., 2017)

Additionally, Pollard R. (2008) argues that in football, team playing at home has an advantage, which is a well-known factor that is commonly taken into account. Thus, an updated Maher's model was introduced with an additional variable that would increase the likelihood of the home team winning.

Moreover, over the years, many other extensions have been created for Maher's model, such as that of Dixon M. and Coles S. (1997). In their paper, an extra variable was introduced that increased the probability of 0 - 0 and 1 - 1 draws as well as decreased the probabilities of 1 - 0 and 0 - 1 outcomes. Their studies were motivated by observing results from 6629 football competitions from between the years 1992 and 1995. One of the critiques of this approach was presented by Langseth H. (2013), who points out that this model does not take into account the lineup of the team and thus makes it oblivious to matches where a crucial member of a team is not participating (for example, because of an injury). There have also been attempts at modelling goal distribution in football using other methods, such as

Bivariate Weibull by Boshnakov et al. (2016), but Poisson remains one of the most popular and effective choices.

Karlis D. and Ntzoufras I. (1999) conducted research into goal correlation between opposing teams. In 15 out of 24 championships, the findings suggested that goals of opposing teams are independent. Where the correlation was found, it was of no importance for modelling match outcomes. Also, Karlis D. and Ntzoufras I. (2003) explored in detail the bivariate Poisson distribution. The main difference between his and Maher's model is that using the bivariate Poisson assumes that there is a correlation between the number of goals that opposing teams score in the same match. Using data for Serie A between 1991 and 1992, the researchers found that the bivariate Poisson with constant covariance performed better.

Azhari H. R. et al (2018) present a practical application of the Poisson Regression model using English Premier League 2017 - 2018 as their dataset. While using standard Poisson Regression, they were able to achieve a 61% overall accuracy for rounds from 29th through 38th of the tournament. Boldrin B. (2017) developed a Poisson model for the same tournament, which achieved 57.6% accuracy while targeting profits, not overall accuracy.

Furthermore, one factor that can bring the noise to the data is the notion of the importance of the match. There are cases where a match is not important for either or both teams playing. In such cases, it can be more difficult to predict the outcome of the match. Moreover, the results from such games cannot be fully trusted and, unless manually removed, they will remain in the training data. These can include friendly matches, matches that have no outcome for a team (for example, when the team has already secured the score for a promotion). Lastly, some portion of the football games is rigged, meaning one team is trying to lose on purpose. Sometimes bookmakers spot this and often have the right to void bets in such cases (Bet365, 2020), but it is difficult to tell which matches in the dataset must be excluded for the reasons described above. Usually, researchers keep all the matches in the dataset and focus on a specific league.

Another important factor that should be considered with any football prediction system is Home Advantage. It is commonly known that the home team has an advantage. Pollard R. (2008) gives some reasons why this is the case. In his paper 'Home Advantage in Football: A Current Review of an Unsolved Puzzle', he considered factors, such as the effect the crowd has on each team, how travelling impacts the away team, familiarity of the team with the surroundings (including the stadium), whether or not referees may be biased and others. Regardless of the true reason behind the phenomena, it is a fact that in football the home teams typically win the majority of the matches.

2.3 Elo

To begin deriving a proper application of the Elo system in football prediction, its historical development and known usages must be explored. Originally, the Elo Ranking system was developed in the late 1950s by Arpad Elo and it was based on the Harkness System. This model was designed to give an estimate of chess players' skills and has been used for that purpose from when it was created and remains the most popular ranking system in chess in the present day (Besterand D. W., & Maltitz M. J., 2013). In the Elo system, each player is

assigned a score, which is updated after every game he plays based on the expected score of that player against his opponent and the actual outcome of the match. The extent to which the player's rating is affected by the result of a single game is decreased with the total number of games that the player has participated in (Vecek N. et al, 2014). As such, Elo is able to offer an estimate of a performance rating by calculating a score for every player in the game. Next, how Elo has been applied in football and how it has been modified for this problem specifically is examined.

World Football Elo Ratings maintain Elo rankings for the World Cup as well as several other football tournaments and sports. The website also provides data with historical changes of each teams' Elo ratings. While it does not offer any predictions, this demonstrates some level of interest in the implementation of this algorithm for football predictions. More importantly, the website provides information about its implementation of the Elo. First, the value of K-Factor, which is a multiplier that determines how much a match result affects each players' ratings, is examined. Here, a K-Factor between 20 and 60 is used that depends on the type of competition. Generally, the more important the match, the higher the K-Factor that is used. It is also adjusted based on the goal difference of the match, increasing the K-Factor as the goal gap rises. This is one possible extension of the original Elo system that takes into account the actual game score rather than simply the binary outcome. This would make for a more sophisticated prediction system in football based on the assumption that the goal difference between winning and losing team is representative of teams' skill levels.

Sullivan C. and Cronin C. (2015) were able to train an Elo-based model for football prediction. In their study, they used data from England Premier League for the past 5 seasons. Apart from using the base Elo formulas, the researchers incorporated home advantage, momentum and draw predictions into the system. The latter was based on the assumption that if the system produces a chance of winning between 40% and 60%, the match is likely to end in a draw. In the same paper, Sullivan C. and Cronin C. found that a team that is on a winning streak was more likely to break it in the next match rather than continue it. This, perhaps, could be attributed at least in part to the regression to the mean. Moreover, the results showed that training an Elo model on more than a single season resulted in lower accuracy of the model. However, these results are obtained from testing the model on a single tournament after the training (+1.84% increase in accuracy when training on 1 season vs. 5 seasons). Thus, these figures may be subject to bias since the testing included only a single season of one tournament, which may have happened to have a specific set of games. Lastly, the researchers also attempted to apply the model to Champions League, League 2 and League 1, achieving the highest accuracy of 45.11%, 38.77% and 45.83% respectively. These figures are obtained by applying the same model that was derived from adjusting the hyperparameters for England Premier League and so the study notes that the model appears to overfit for England Premier League. However, it is possible that the real cause behind a change in performance is the difference in the nature of each competition.

There have also been attempts at using Elo for making bets in football. For example, Hvattum L. and Arntzen H. (2010) in their paper called 'Using ELO ratings for match result prediction in association football' developed 2 Elo-based models that were tested using simulated betting along with 5 other models. In this research, it was found that Elo-based

models performed better than some of the naive methods, such as assuming past outcome probabilities or a uniform distribution of results. However, these models were not the best, producing slightly worse results than odds-based systems.

An interesting addition to the original system is the concept of momentum, which has been explored by Besterand D. W. and Maltitz M. J. (2013) in their paper called ‘Introducing Momentum to the Elo rating System’. The idea behind momentum is that a team is more likely to win for each previous game they had won consequently. A deficit occurs when a player that is on a streak suddenly loses a game. The assumption is that a high-skilled player can lose accidentally to a low-skill player by chance and not as a reflection of his skill. In this case, the losing player’s skill will not be affected by the result of the match. Similarly, if a player wins randomly after a series of losses, he will not increase his rating. This method can help with the problem of the importance of a match. For example, when a team is participating in multiple tournaments simultaneously, they may purposely perform worse than they can in one to save the energy for the other one. However, this concept would be more difficult to implement in football for other reasons. One of the factors that can affect this is that the team lineup can affect whether or not a team performs well in the field. So an unexpected loss may indeed be a reflection of the team’s skill, or rather team’s lineup at the time of the lost match, rather than mere chance. Furthermore, a team may have performed well in their last tournament, but its skill may have changed dramatically if there has been a substantial time difference between the two. Elo has also been applied in other sports and industries. These cases are examined to motivate the implementation of Elo for football, determine other common usages of the system and additional extensions of the original model.

In a recent study, Leighton V. et al (2019) used an Elo-based rating system to predict tennis matches for the 2018 Wimbledon championship. The purpose of the study was to compare the Elo model to official world rankings and betting odds metrics to calculate how they perform in terms of several factors. Prediction accuracy, which reflects how often the system predicts the correct outcome of the match. Calibration, which describes how well-calibrated and biased the model is. And model discrimination, which indicates if the probabilities are more accurate for wins than upsets within the matches. The researchers found that betting odds outperformed other metrics in terms of predicting accuracy and calibration. However, model discrimination was better in the Elo system. Thus, official world rankings consistently performed worse than the other metrics that were considered.

Elo has also been applied to board games. As such, in 2018, Balduzzi D. et al applied traditional Elo and a Multidimensional Elo to Go, a popular board game. Multidimensional Elo (mElo2k) overcomes the short sight of Elo that relative skill is transitive. While Elo assumes that there is a single strategy that dominates in all cases, multidimensional Elo is meant to consider a wider range of factors, thus allowing for more sophisticated predictions. The regular Elo ranking performed poorly, incorrectly predicting who is more likely to win a game of Go, while the multidimensional Elo correctly predicted the likely winner in all cases that were considered.

Elo has also been applied more uncommonly. In their paper, Lehmann, R. & Wohlrabe, K. (2017) used the Elo model to rank scientific journals. The system intended to assess the quality of a scientific publication, which could then be used by publishers to determine

whether a specific paper is worth publishing. They assigned an Elo rating to each journal, which was determined based on the impact it made that year. Using data for over 20,000 journals from 1999 until 2015, they were able to produce rankings similar to the widely used Tournament Model and SNIP (source normalised impact per publication), which demonstrated that Elo ranking could be used as an alternative to the existing approaches. Yet another unusual application of Elo was developed by Compton R. (2014), who attempted to compare YouTube video quality by applying the Elo ranking system to them. Using data generated by users, who were presented with 2 videos and then vote for which one was better, the researcher was able to assign a ranking score for each video. As an outcome of the experiment, it was found that for that particular problem the order of comparisons did not matter. This makes sense because the quality of a video is constant and does not change once it is produced. However, this would not be true for football. Player's skill changes over time and more importantly, teams change their lineups (especially between seasons) and are constantly training to beat their competition. Thus, Liverpool 10 years ago can be drastically different in terms of skill to Liverpool today.

Another field where Elo can be applied was explored by Penalek R. (2016). In this study, the researcher compared different approaches of estimating student skills, arguing that Elo is potentially a better option than simply calculating the proportion of the correct answers or using joint maximum likelihood estimation (JMLE). It was found that estimating students' skills by deriving the proportion of correct answers was inadequate for adaptive item selection, where a student is presented with questions that are closest to his skill level. For this task, Elo was found to be optimal (from the approaches that were used within the research) since it both accomplished the required task and had a lower computational load than JMLE. It also helped that Elo formulas could be easily adjusted to accommodate different types of tests.

All in all, from the conducted research, several different Elo-based models can be identified for further examination. Firstly, varying how fast the model learns and, potentially, introducing some decay and momentum could improve the system. Additionally, exploring the effects of using a constant and dynamic draw gap, which is a chance that a match is going to result in a draw, could have a positive effect on the overall prediction accuracy. Lastly, since the home advantage is a well-known fact in football, adding a constant to account for that should also improve the model. Given that the data for each competition contains what teams participated in the match and the outcome, the Elo rating is assigned to every team that is involved in the competition. In this paper, the Elo system is considered on a per-team basis, that is, every team has an Elo ranking. However, it is also possible to assign an Elo ranking to each player, which, perhaps, would yield better results since it could take into account the team lineup when determining the result of a match. To achieve this, a data set that contains information about each player for matches is required.

2.4 TrueSkill

Another methodology that is explored in this paper is using the TrueSkill system in football prediction. TrueSkill was developed by Microsoft Research in 2005. The system was developed for Xbox Live to estimate a player's skill level to allow fair matchmaking in certain games or even specific game modes (for example, player's skill may be different in a 3

versus 3 match in comparison to his skill in a 5 versus 5 match). This system became widely adopted by Microsoft and has been used in popular video games, such as Halo 3 and Forza Motorsport 7 (Herbrich R. et al, 2007).

TrueSkill can be used to estimate and track player skill in a variety of different game scenarios. For example, it can calculate skill levels based on outcomes of various matchmaking set ups, such as 2 players Free-For-All or 2 Teams each with 4 players. The number of players can vary. It would only change the number of games required to give an accurate estimate of one's skill level (see Appendix B for the number of matches required to initialise a player's skill). For Free-For-All games, with an increasing number of players, the required number of games to assess a player's skill goes down. The reverse is true for an increasing number of players for team-based games. This is because when all players compete with each other, knowing how a player ranks in a larger scoreboard yields more information about a player's skill with respect to others. The opposite is true for team-based games because when one team wins, it is less clear how much one team member contributed as a whole to the overall success of the team (unless other factors are considered). This uncertainty increases with the number of players per team (Herbrich R. and Graepel T., 2006; Herbrich R. et al, 2007).

2.5 TrueSkill 2

An updated TrueSkill model, labelled TrueSkill 2 was introduced by Microsoft Research in 2018. This ranking system was successfully adopted by Gears of War and Halo. The main addition to the system is that TrueSkill 2 considers more factors about the match. For online shooters, the model takes into account player score, experience, number of kills, deaths, team members and other data about the game rather than simply looking at the result. In their research, Microsoft was able to predict 68% of historical matches. In comparison, original TrueSkill was able to predict only 52% (Minka T. et al, 2018). To our knowledge, apart from being widely used in the video game industry, TrueSkill does not have any other applications.

In this paper, the implementation of TrueSkill where each team gets a team rating assigned to it and recalculated after each match is explored. This method has an obvious flaw that is caused by the fact that team transfers, substitutions and other events can affect the outcome of the game. TrueSkill has several parameters that need to be tuned for football, including, but not limited to Beta, Tau, Mu and Sigma, which are described in detail in Section X. Additionally, since TrueSkill already provides a way of predicting draws, both constant and dynamic draw chance is explored. Similarly to Elo, Home Advantage is added to the TrueSkill to determine if it results in higher overall prediction accuracy.

Overall, it appears that there has not been much research into TrueSkill applications in industries, other than video games. Based on the general information gathered about football, it seems reasonable to attempt to utilise some of the same principles that are commonly used with static algorithms. For example, examining the effect of the home advantage seems like a reasonable addition to the system to cater for the given problem.

2.6 Neural Networks

While TrueSkill and Elo's approaches have not been applied extensively in football prediction, Neural Networks have received greater amounts of attention. Some work in using the NNs in this industry has been done as early as in the 1990s but the interest in this machine learning approach in football has been gaining popularity over the last decade. It is no wonder that researchers are exploring the possible practical operations of Neural Networks in football since this undoubtedly powerful technique has been successfully used across multiple other industries, including healthcare, finance, insurance and others. According to Purucker (1996), Neural Networks have a larger potential in sports predictions than traditional algorithms due to the fact that the latter typically has some restrictions associated with them because of the static nature of formulas. Given that Neural Networks can often overcome these limitations, they may offer a better way of predicting the matches.

One of the most important considerations for Neural Networks is selecting relevant features from the original dataset. In his paper, Buursma D. (2011) used averages for the team's past games to determine a team's feature vector. Using WEKA, provided by Machine Learning Group (nd), he determined that using 20 past matches as a measure of the team's performance is the optimal option when using past averages for predicting future outcomes. It is worth noting that the researcher did observe an improvement in performance as the number of past matches that were considered increased, but concluded that additional benefits were insignificant in comparison to the computational and memory requirements.

Apart from using averages for past N games, there are other parameters that deserve consideration. As such, average goals home/away, win, lose and draw percentages in current/previous season are all commonly selected to be part of the final feature set (Mccabe and Trevathan, 2008; Tax N. and Joustra Y., 2015). Furthermore, including bookmaker odds is something that has been explored by multiple researchers, demonstrating predictive potential (Tax N. and Joustra Y., 2015). In their paper, Odachowski K. and Grekow J. (2012) were able to achieve up to 70% accuracy on a sample of 2615 matches outcomes from various competitions by utilising a binary classifier that only received information about changes in match odds from a bookie up to 10 hours before the match started. Tax N. and Joustra Y. (2015) also found that Multi-Layer Perceptrons that included both the data about the match and bookie odds performed better than the ones that only included one or the other.

There is more than one way of predicting the outcome of a game using a Neural Network. For example, one approach would be to predict the number of goals scored by each team. This would also allow placing bets not only on match outcome (Home, Draw, Away) but also on Over/Under results (for example, Over/Under 1.5 Goals). Another way would be to create a NN that only predicts the outcome of the game. That is, create a system that would take as an input a feature vector data for a game and output one of the three possible classes, namely Win, Draw or Lose (Goddard, 2006; Tax N. and Joustra Y., 2015). In his paper, Goddard (2006) demonstrated that predicting categorical results yields higher prediction accuracy. While it only allows for one type of betting, accuracy is considered as the main metric in this paper. Hence, a NN developed in this paper follows the latter approach.

Several other factors significantly impact the creation of the model. An important consideration is whether to use cross-validation or provide the data to the model sequentially. A visualisation of how 10-fold cross-validation splits the dataset is presented in Figure 3.

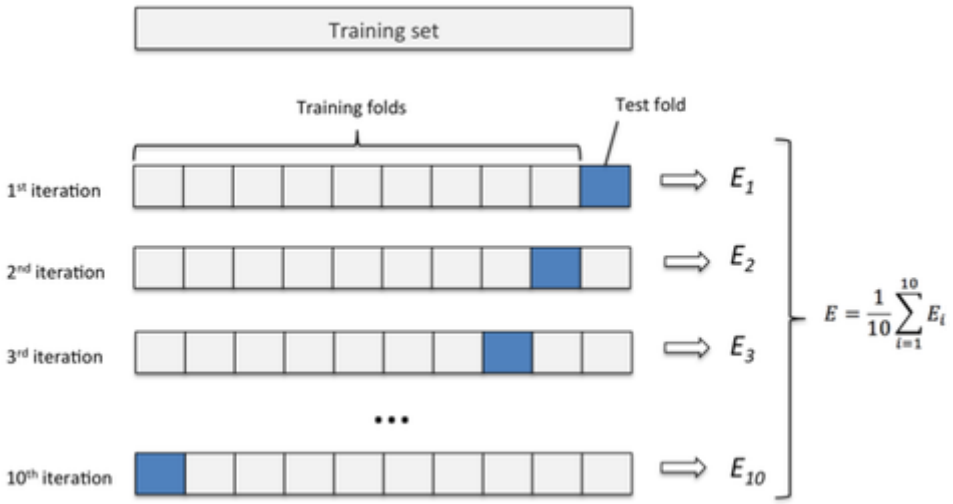

Figure 3: K-Fold cross-validation visualisation for K equal to 10 (Raschka, nd)

As can be seen in Figure 3, there are cases when K-fold Cross-Validation would use data from future matches to predict the past, which may block it from learning the most recent patterns and reduce the accuracy. However, that greatly depends on the type of artificial intelligence as well as on the aim. According to Tax N. and Joustra Y. (2015), using cross-validation for this type of classification is inappropriate because of the fact that the NN will learn future patterns that would not have been available to it before.

The architecture of the Neural Network also deserves some consideration. Notably, the majority of the researchers tend to use a neural network with 1 or 2 hidden layers, which seem to perform well for predicting football (Kou-Yuan and Kai-Ju, 2011). An example architecture of such a Neural Network is presented in Figure 4. Here, the Neural Network has N input neurons, 1 hidden layer and 3 neurons in the output layer, each representing a distinct outcome.

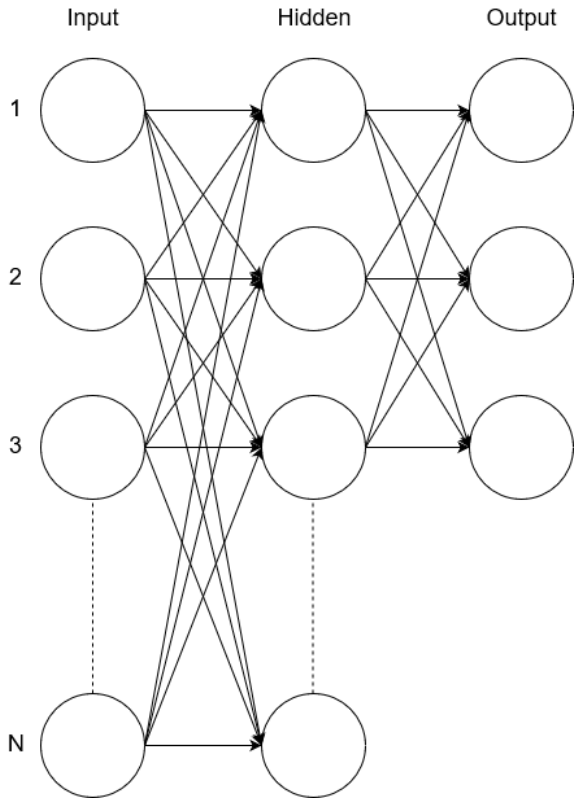

Figure 4: Example of a Neural Network with 1 hidden layer that can be used to classify football matches

However, Neural Networks have not been applied exclusively to football. As such, there are other sports that have been explored which could, perhaps, contain some useful lessons that could be applied in football. Miljkovic (2010) developed a model to predict NBA matches with similar features as described in the Neural Networks, such as averages of teams' performances. His system also incorporated sport-specific attributes, such as the number of blocks, fouls and free throws per game, which have been used with relative success to predict the outcomes of basketball matches.

In their paper, Kou-Yuan and Kai-Ju (2011) were able to train a Multi-Layer Perceptron to predict results of NFL matches. In addition to using data about the number of goals, the model also made use of information about shots, possession and number of fouls to determine the outcome of the match. Purucker (1996) also experimented with Adaptive Resonance Theory networks for predicting NFL results, which was able to achieve an overall accuracy of about 50%. Interestingly enough, the research also compares a Multi-Layer Perceptron with an architecture that only used input and an output layer. The described model achieved an accuracy of 64.3% for the same week of NFL games. Arguably, the number of games the models were tested on was potentially not representative since the test set only included 14 matches. Additionally, Baio and Blangiardo (2010) proposed an entire model based on the number of goals scored and conceded.

Some experimentation has been done when it comes to the type of Neural Networks that is used. Pettersson D. and Nyquist R. (2017) proposed a Recurrent Neural Network and Long Short-Term Memory Neural Network models for predicting football match outcomes at every

15th minute of the match. This means that the model would predict the outcome when the match starts, after 15, 30, 45, 60, 75, 90 minutes as well as overtime. This research utilised game events as inputs to the RNN that contained information about the lineup, player positions, goals, cards, substitutions and penalties. Of course, since the developed models were driven by events that had already occurred during the match, they did not produce high prediction accuracy before the start of the match (so predicting the outcome of the game at 0 minutes).

In their research, Constantinou et al (2012) explored psychological factors, such as team morale and fatigue, among other features for a Bayesian model that predicted outcomes of English Premier League. With their model, which was called pi-football, they were able to present an algorithm that could, according to the authors, even make a profit in long-term betting.

Overall, developments in applications of Neural Networks for football predictions have been explored. Parameters that have demonstrated to be successful, the explored architectures and other factors, that are necessary for creating a working model, have been examined. The attributes that are selected for the final feature set of the Neural Network are based on the findings of other researchers that have been presented in this section. Of course, the types of features that can be used are also constrained by their availability within the dataset. Thus, in this paper, a hybrid model that includes both the bookmaker odds and the statistical information about the match, such as the average number of goals, will be created. As with Elo and TrueSkill techniques, it is not clear whether the system can be tuned for a given sport or if it should be adjusted for individual competitions to cater for the innate differences between them. This is one of the aims that is examined by creating different Neural Network models (see Section 3.4). Using this hybrid model, the hyperparameters of the neural network are tuned for each competition specifically to determine if competitions have different optimal parameters. This is based on the proposition that competitions (and even individual tournaments) can have a different distribution of wins, draws, loses and differ in a variety of other ways, such as the number of transfers. All of these can have an effect on one or more hyperparameters in the Neural Network, such as learning rate, batch size and other.

3. Analysis

3.1 Dataset

In this paper, data for 4 competitions are used: England Premiership, France Championnat, Germany Bundesliga and Italy Serie A between 2008 and 2018. These are some of the most popular football competitions in the world so they should have plenty of data available online. Football-Data (nd) has information about all of these competitions. For each of them, there are over 3,000 matches available in the last decade.

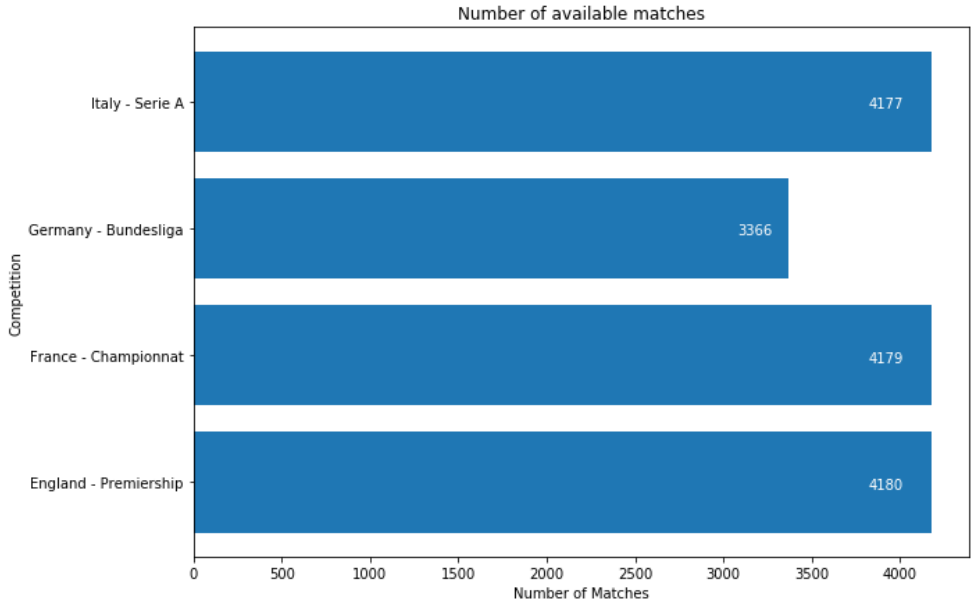

Figure 5: total number of matches for each competition in the dataset

Each match contains a multitude of information. Full description of this is provided in Appendix C. It is also available at Football-Data (nd). As can be seen, the majority of the data for a match is odds from different bookmakers or markets. This can certainly be useful as it represents bookies' own prediction in the match results. While these are not essential for Elo or TrueSkill, they may prove useful for predictions using Neural Networks by providing odds as part of the input. Historical information about match outcomes can be used for training and testing the models. Columns and sample values for a match are available in Appendix D.

Additionally, data quality checks are performed to ensure that matches that are used as part of the training and testing data set are complete and correct. This includes ensuring that both home and away teams are known. Furthermore, the score of the game is an essential piece of information that cannot be null for the match entry to be useful. Lastly, at the very minimum, the odds from the main bookie (for this dataset - Bet365) for Home, Draw and Away must be available. Exact number of matches post-quality check can be found in Appendix E.

3.2 Elo

First, the mathematics behind Elo are described in detail. Assume that *R(A)* is the skill of the first team and *R(B)* is the skill of the second team with a default of 1200. To recalculate the skill-based on the match outcome, the transformed rating has to be found first:

$$R`(A) = 10^{\frac{R(A)}{400}} \quad \textbf{(2)}$$

$$R`(B) = 10^{\frac{R(B)}{400}} \quad \textbf{(3)}$$

Where *R`(A)* is transformed rating of team A with rating *R(A)* and *R`(B)* is transformed rating of team B with rating *R(B)*.

Then, expected scores *E(A)* and *E(B)* for each team need to be calculated.

$$E(A) = \frac{R`(A)}{R`(A) + R`(B)} \quad \textbf{(4)}$$

$$E(B) = \frac{R`(B)}{R`(A) + R`(B)} \quad \textbf{(5)}$$

Where *E(A)* is the expected outcome for team A and *E(B)* is the expected outcome for team B between 0 and 1. These represent what the system thinks the outcome of the match should be. These values are also later used to determine how correct the system is. Lastly, using the K-score, teams' scores can be updated.

$$R(A) = K * (O(A) - E(A)) + R(A) \quad \textbf{(6)}$$

$$R(B) = K * (O(B) - E(B)) + R(B) \quad \textbf{(7)}$$

Where *R(A)* and *R(B)* are updated team scores and K is the K-Score. The K-score determines the extent to which the match outcome is reflective of the actual team skill. For example, in chess, K of 32 is typically used. *O(A)* and *O(B)* represent match outcomes and can be either 1, 0.5 or 0 in case of a win, a draw or a loss respectively.

Next, the Elo system is demonstrated on sample teams. For example, two new teams, Team A and Team B play each other. Since the teams are new and the system does not know anything about their skills, both of them start with an initial default rating of 1200. After Team A plays Team B, there are 3 possible outcomes. Assume ***R(A)*** is the skill of Team A and ***R(B)*** represents the skill level of Team B. Regardless of the outcome, the transformed rating needs to be calculated first using Formulas 2 and 3:

$$R`(A) = 10^{\frac{1200}{400}}$$

$$R`(A) = 10^{3}$$

$$R`(A) = 1000$$

$$R`(B) = 10^{\frac{1200}{400}}$$
$$R`(B) = 10^{3}$$
$$R`(B) = 1000$$

Followed by the expected score that can be calculated using Formulas 4 and 5:

$$E(A) = \frac{1000}{1000 + 1000}$$
$$E(A) = \frac{1000}{2000}$$
$$E(A) = 0.5$$

$$E(B) = \frac{1000}{1000 + 1000}$$
$$E(B) = \frac{1000}{2000}$$
$$E(B) = 0.5$$

The values of *E(A)* and *E(B)* suggest that Elo thinks the match should result in a draw. There are 3 possible outcomes as a result of the match. For the purpose of this example, it can be assumed that K-factor is 32. Lastly, the updated ratings can be found using Formulas 6 and 7, depending on the actual outcome. If Team A wins Team B:

$$S(A) = 1$$
$$S(B) = 0$$

$$R(A) = 1200 + 32 * (1 - 0.5)$$
$$R(A) = 1200 + 32 * 0.5$$
$$R(A) = 1200 + 16$$
$$R(A) = 1216$$

$$R(B) = 1200 + 32 * (0 - 0.5)$$
$$R(B) = 1200 + 32 * (-0.5)$$
$$R(B) = 1200 - 16$$
$$R(B) = 1184$$

Another outcome is possible if Team B wins Team A:

$$S(A) = 0$$

$$S(B) = 1$$

$$R(A) = 1200 + 32 * (0 - 0.5)$$
$$R(A) = 1200 + 32 * (-0.5)$$
$$R(A) = 1200 - 16$$
$$R(A) = 1184$$

$$R(B) = 1200 + 32 * (1 - 0.5)$$
$$R(B) = 1200 + 32 * 0.5$$
$$R(B) = 1200 + 16$$
$$R(B) = 1216$$

Or, if the match results in a draw:

$$S(A) = 0.5$$
$$S(B) = 0.5$$

$$R(A) = 1200 + 32 * (0.5 - 0.5)$$
$$R(A) = 1200 + 32 * 0$$
$$R(A) = 1200 + 0$$
$$R(A) = 1200$$

$$R(B) = 1200 + 32 * (0.5 - 0.5)$$
$$R(B) = 1200 + 32 * 0$$
$$R(B) = 1200 + 0$$
$$R(B) = 1200$$

As can be seen from the examples, when the match does not result in a draw, the winner's rating is increased by 16 while the loser's score is lowered by the same amount. However, in the case of a draw, both teams retain their rating. This is because their Elo is the same, which means the expected score is 0.5, so the match is expected to result in a draw.

3.2.1 Methodology

To calculate Elo ratings, Python 3 with open-source software Jupyter Notebook was used for the training process using all match data. Apart from comparing the performance of different models, it is important to determine if the systems are suitable for use by the bookies or professional gamblers that are trying to outperform the bookies. Hence, a bookie is selected

to be used when analysing the performance of models. Initially, metrics for Bet365 (a bookie of choice) are given within the context of the same tournaments. These are then used as a benchmark for the models.

First, the performance of the classic Elo model with K of 32 without any extensions is presented. Next, an optimal Elo model is developed using an iterative process. Constant K-Factor, Formula K-Factor, Constant Draw Gap, Dynamic Draw Gap, Home Advantage and time-dependent K-Factor are considered. After every iteration, the model is evaluated and findings are recorded.

Data for tournaments starting from season 2008 - 2009 and up to (including) 2017 - 2018 are used to train the model when finding the best hyperparameter values. The tournaments for 2018 - 2019 season are used to evaluate the performance of the system on unseen data. A more sophisticated method of testing the model, such as K-Fold Cross-Validation, is not possible with Elo since it requires data to be fed in historically. The training and testing process visualised in Figure 6. The same process is later applied to TrueSkill while Neural Networks follow a different approach.

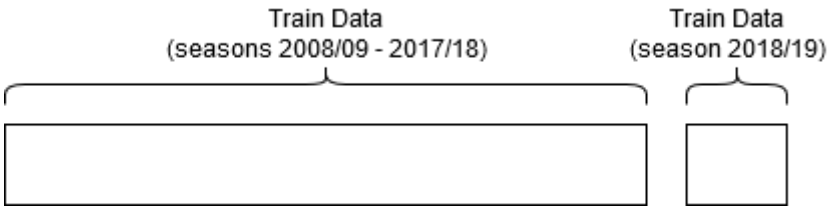

Figure 6: Elo training and evaluation splits visualised

3.2.2 Bookie

To set a baseline for the results, Bet365 accuracies for competitions 2018 - 2019 are presented in Figure 7.

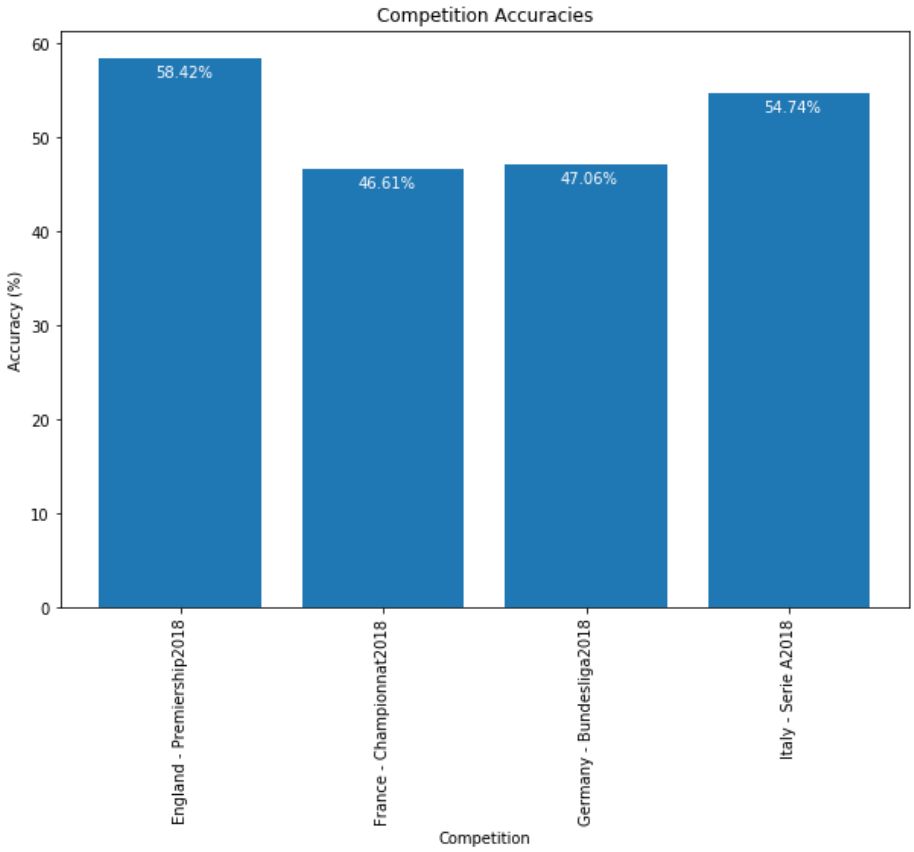

Figure 7: Bet365 prediction accuracy for selected tournaments 2018 - 2019

As can be seen in Figure 7, the bookie does particularly well with the England Premiership, followed by Italy Serie A. With France Championnat and Germany Bundesliga, the results are considerably lower. Full range of metrics, which includes Recall, Precision and F1 for wins, draws and loses, can be found in Appendix F. Additionally, confusion matrices are provided.

Generally, the bookie appears to predict wins better than loses with an average F1 of 0.6375 for wins and 0.485 for loses. In regards to draws, the bookie normally predicts the minority of draws, suggesting that predicting draws is significantly harder than wins or loses. This, in part, can be caused by the fact that there are significantly fewer draws than wins or loses. This makes sense, which is also why, typically, the decimal draw odds are higher than those of other outcomes.

3.2.3 Basic Elo

How Basic Elo performs on unseen data after running training a Basic Elo system using the formulas presented above is given on the diagram. Table 2 summarises competition accuracies of the Basic Elo model.

Competition	Accuracy
England Premiership	52.89%
France Championnat	42.28%
Germany Bundesliga	48.69%
Italy Serie A	52.37%

Table 2: Basic Elo Competition Accuracies

Confusion matrices and useful metrics for the Elo models that are developed in this section can be found in Appendix G. In Figures 8 and 9, results for the Initial Elo model are presented in a confusion matrix and summary of evaluation metrics for England Premiership.

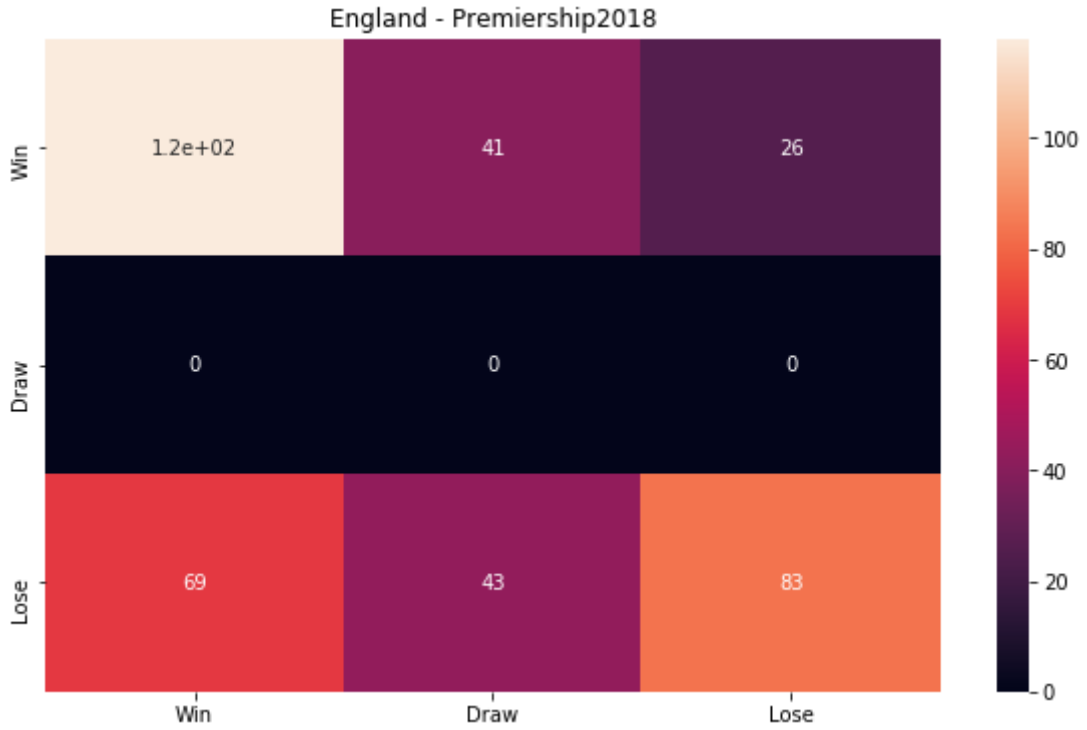

Figure 8: Initial Elo confusion matrix for England Premiership 2018 - 2019

```
Accuracy: 52.89%
MAE: 0.32
RMSE: 0.34
Precision:      Win 0.64        Draw 0.00       Lose 0.43
Recall:         Win 0.63        Draw 0.00       Lose 0.76
F1:     Win 0.63        Draw 0.00       Lose 0.55
```

Figure 9: Initial Elo metrics for England Premiership 2018 - 2019

As can be seen from the confusion matrix for England Premiership 2018 - 2019, the model did not predict the draw as a game outcome a single time. This is not surprising for the basic Elo model that was described as it only predicts which team would win. Draw prediction is something that will be added to the model later. Still, in the majority of cases, the model does predict the correct outcome. However, there seems to be a significant number of games where the system expected team B to win, but the opposite outcome occurred. This

potentially could be explained by the fact that the model does not account for home advantage. A similar trend can be observed for other competitions in the dataset (all figures can be found in Appendix G). Next, competition accuracies of the Basic Elo and Bet365 are compared against each other in Table 3.

Competition	Basic Elo Accuracy	Bet365 Accuracy	Prediction Accuracy Difference
England Premiership	52.89%	58.42%	-5.53%
France Championnat	42.28%	46.61%	-4.33%
Germany Bundesliga	48.69%	47.06%	+1.63%
Italy Serie A	52.37%	54.74%	-2.37%

Table 3: Basic Elo and Bet365 Accuracy comparison

As can be seen from Table 3, Bet365 is better at predicting match outcomes in 3 out of 4 competitions, the only exception being Germany's Bundesliga, where Elo outperformed the bookie in terms of predictive accuracy by 1.63%.

3.2.4 Constant K-Factor

To begin improving the basic Elo model, the hyperparameters of the model need to be adjusted. By varying the value of the K-Factor and recording the accuracy that the model achieves, it is possible to compare different models and examine what value of K-Factor produces the highest overall accuracy. To do this, the K-Factor is assigned and the average accuracy for all train seasons, starting from 2008 - 2009 and up to (including) 2017 - 2018, is calculated. The process is described in detail in Appendix H and Appendix I gives justifications for selecting this methodology of tuning. Moreover, for the initial analysis, it is assumed that hyperparameters are independent. This assumption is tested in Appendix N.

Competition	Basic Elo Accuracy	Constant K-Factor Elo Accuracy	Prediction Accuracy Improvement
England Premiership	52.89%	56.32%	+3.43%
France Championnat	42.28%	45.78%	+3.5%
Germany Bundesliga	48.69%	48.37%	-0.32%
Italy Serie A	52.37%	53.16%	+0.79%

Table 4: Accuracy comparison of Basic Elo vs. Constant K-Factor Elo

As can be seen in Table 4, adjusting the K-Factor appears to have improved the overall prediction accuracy. The most significant improvements of over +3% were observed in the England Premiership and France Championnat. Interestingly, the accuracy appears to have dropped by -0.32% for Germany Bundesliga.

3.2.5 Constant Draw Gap

Currently, the model completely ignores the possibility of a draw, which can be seen from confusion matrices. There are several ways of integrating draw predictions into the Elo system that have been explored by other researchers (Sullivan C. and Cronin C., 2015).

Firstly, a constant draw chance is introduced to account for the possibility of a draw. If the absolute difference between expected scores of each team is smaller or equal to a defined draw chance, then the match result will be considered to be a draw. The effect of using different constant draw gaps can be observed in Figure 10.

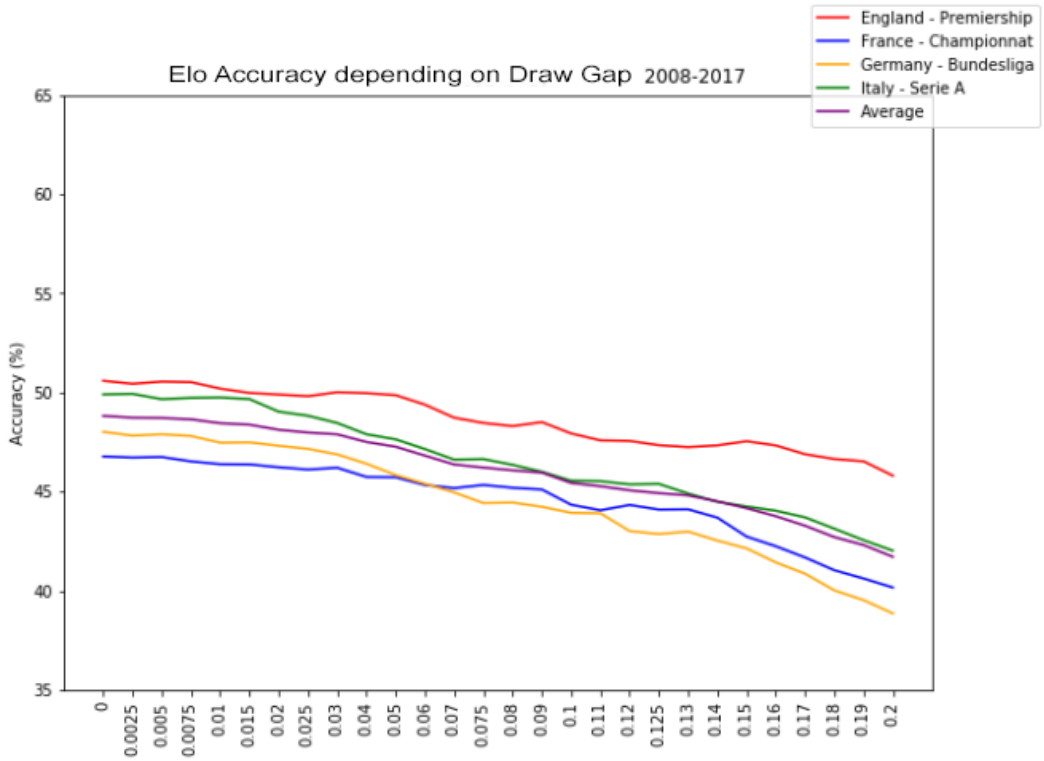

Figure 10: Overall accuracy for train set (y) varying depending on the constant draw gap (x)

As can be observed in Figure 10, introducing a constant draw gap results in lower overall accuracy (for more detailed figures, see Appendix J). This is not surprising since predicting a draw is more difficult than a win or lose as a difference between whether the match will result in a draw or not is always only 1 goal. As a result, it appears that adding some draw chance does not have a positive impact on the overall prediction accuracy. As the draw gap was increased, the overall prediction accuracy dropped. In their paper, Tax N. and Joustra Y. (2015) developed a Multi-Layer Perceptron that did not include draws as part of its predictions so excluding the possibility of a draw is something that has been observed in previous research.

Another approach of introducing draws prediction was attempted where the system kept track of the current draw chance of the competition (on a per-competition basis). The system would then use that percentage as a marker of future draws. However, this did not appear to improve the model (see Appendix K).

3.2.6 Home Advantage

To combat this issue, a constant is added to the expected home score and the same amount is subtracted from the expected away score. The value of the variable by which we perform these calculations is going to be varied from 0 to 0.3 (every 0.01). For every constant, we retrain the model and calculate the accuracy again. The results of these modifications are shown in Figure 11.

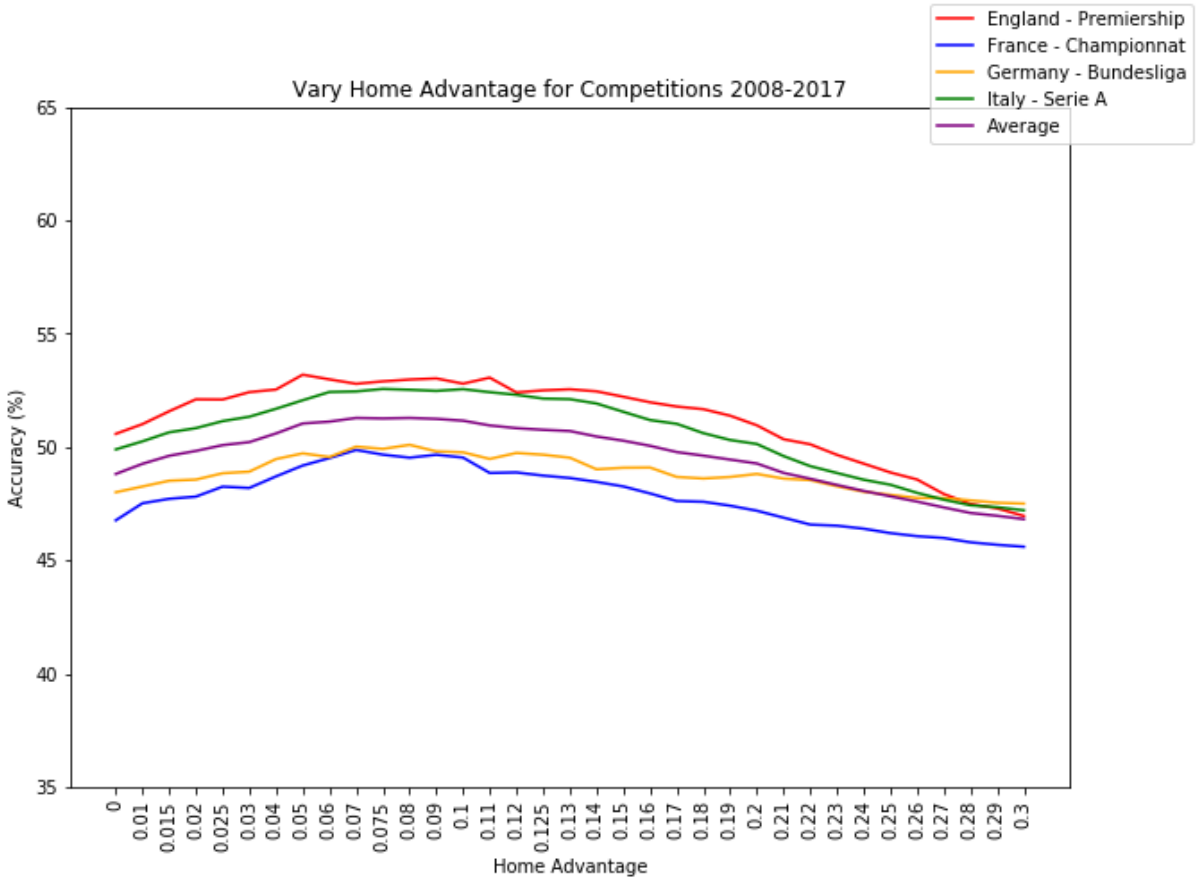

Figure 11: Overall accuracy for train set (y) varying depending on a home advantage (x)

As can be seen in Figure 11, adding home advantage certainly helps increase the performance of the model. It appears that 0.08 is the best home advantage constant overall, resulting in a 51.29% average accuracy. As with other variables, it is worth observing the changes for individual tournaments.

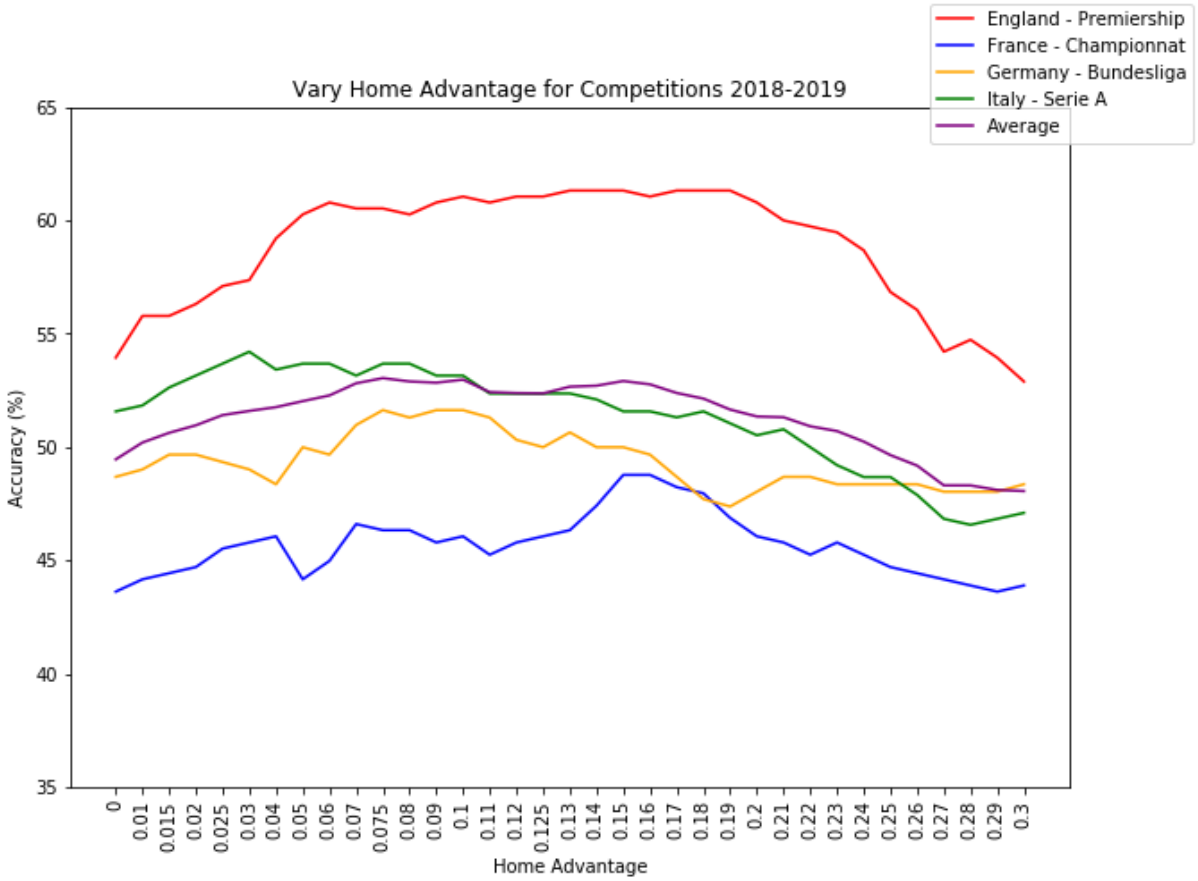

Figure 12: Overall accuracy 2018-2019 (y) varying depending on a home advantage (x)

Figure 12 presents the results of varying the home advantage for 2018-2019 tournaments. In the graph, it can be seen that the home advantage of 0.075 results in the best accuracy. For these tournaments, the home advantage is very close to the best overall value. It is worth noting that, based on Figure 12, the positive effects of adding a home advantage varied. For example, in England - Premiership, increasing a home advantage had a positive effect on accuracy up to about 0.2, while for Italy - Serie A, the accuracy started to gradually decline after the home advantage of 0.04.

Next, it is useful to compare the Elo system with Home Advantage integrated against the previous version of the model, which encompassed the constant K-Factor.

Competition	Constant K-Factor Elo Accuracy	Home Advantage Elo Accuracy	Prediction Accuracy Improvement
England Premiership	56.32%	60.26%	+3.94%
France Championnat	45.78%	46.34%	+0.56%
Germany Bundesliga	48.37%	51.31%	+2.94%
Italy Serie A	53.16%	53.68%	+0.52%

Table 5: Accuracy comparison of Constant K-Factor Elo and Home Advantage Elo

As presented in Table 5, the home advantage had a positive effect on the overall accuracy of all competitions. The largest impact was observed in England Premiership, where the introduction of a 0.08 home advantage increased the accuracy by 3.94%.

3.2.7 Time-based K-Factor

Another approach to defining the K-Factor would be to take into account the number of days that have passed between matches. The assumption is that after a significant amount of time has passed, the team's performance could have changed drastically for a number of reasons, such as team transfers. Assigning a higher K-Factor for recalculating the team's rating when it plays a match after a significant break could help improve the overall model performance by allowing it to adjust to changes faster. The results of using different Time-based K-Factors are presented in Appendix L. On average, K-Factor with the formula:

$$KFactor = 18 * d \quad \textbf{(8)}$$

Where *d* is the number of days since the last match (or 1, whichever is bigger) performed best overall.

3.2.8 Non-Binary Results for K-Factor

Another modification of the standard Elo system that is tested is the utilisation of non-binary results. This means that when ratings for each player are recalculated, as a result of a match between the parties, the model considers the actual score that the match ended with. This is different from the classical approach of simply rewarding the player that won a match, penalising the loser or adjusting the ratings after a draw. Using this type of K-Factor appears to improve the performance of the Elo model (see Appendix M for more detail). Overall, the best accuracy was achieved with the following formula:

$$KFactor = 10 * g \quad \textbf{(9)}$$

Where *g* is the goal difference of the match or 1, whichever is highest.

For individual competitions, the constant (which is 10 for overall accuracy) was different. For England Premiership, France Championnat and Italy Serie A, this constant was 9, 12 and 9 respectively. However, for Germany Bundesliga, Time-based K-Factor remained a better option.

3.2.9 Average Elo vs. Competition-based Elo

Throughout Section 3, hyperparameters that perform best when considering a single Elo model that is used to predict all selected competitions were discovered. However, another set of hyperparameters, which achieve the highest performance when creating a separate Elo model for each competition, were also documented. Now, these two approaches are compared to determine whether it is more effective to create a single model that maximises overall accuracy or to develop separate models that are tuned to individual competitions.

To compare the two approaches, the accuracies for 2008 - 2017 tournaments are used. Full figures are given in Appendix O and Table 6 presents a summary of the findings. As can be observed from Table 6, the Competition-based values for the Elo models perform better for every competition between 2008 - 2017.

Competition	Average Elo	Competition-based Elo	Difference
England Premiership	53.07%	53.41%	+0.34%
France Championnat	48.86%	49.89%	+1.03%
Germany Bundesliga	49.48%	50.71%	+1.23%
Italy Serie A	52.43%	52.79%	+0.36%

Table 6: Competition accuracy comparison of Average Elo vs. Competition-based Elo for 2008 - 2017 tournaments

Because of this, the competition-based hyperparameters are considered as the best choice for the Elo model. It appears that the competitions are indeed different and while they clearly share some similarities, best results can be achieved by tuning the model for each competition separately.

3.3 TrueSkill

TrueSkill is a ranking system that is based on Bayesian probabilities. This approach was originally developed by Microsoft Research for use with Xbox Live players. Similarly to Elo, TrueSkill aims to track the level of skill of the players for the purpose of matching players of similar levels with each other in a variety of games. It is used in several video games, such as Halo 3 and Forza Motorsport 7. While Elo has found applications in chess, football rankings and other areas, there was no specific algorithm for ranking players in video games. Often, players get matched with each other only for a single game and other algorithms do not take into account the fact that, while the team may have lost the match, some players could have performed better than others and, thus, deserve different skill adjustment. That was the idea behind creating the TrueSkill model (Herbrich R. & Graepel T., 2006).

Unlike Elo, TrueSkill uses 2 numbers to figure out the player's skill level. For every player, the model holds a number that represents the player's skill level and another number that represents the degree of uncertainty. The latter is what allows the system to change the skill value for players drastically when the uncertainty is high. On the other hand, when the system is fairly sure in a player's skill, his skill level will not be affected as much. TrueSkill only takes into account the end result of the match for recalculating the rating. When a new player joins, he is assigned a default rating and uncertainty. It takes several games for the system to calculate players' rating so typically uncertainty is relatively high at the start (Moser J., 2010).

Appendix B gives figures on how many games are required to calibrate a player's rating as described by Microsoft. As can be seen from the table, for 2 teams, each consisting of 8 players, the number of games required to calibrate the system is 91 per gamer. It is twice as small for teams consisting of 4 players each, suggesting a linear relationship. That would mean that for 2 teams of 11 players, the system requires about 125 games per player. It is a considerable amount per player. However, given the richness of historical data, the system may be able to get just enough data to do that, or at least come close.

As the system gets more and more training data, the uncertainty for each player gets lower. However, TrueSkill is designed in a way that increases the uncertainty between games because it assumes that something may have changed. This also does not allow the uncertainty to drop to 0. To calculate which player is the winner, TrueSkill uses Total Probability Theorem (Weisstein, E. W., 2000) to find expected performance for each player. The model then decides if the player performance is different enough for it to consider one team or the other a winner, or if it should settle the match as a draw.

TrueSkill is similar to the Elo system in that one player's rating is increased while the other one is decreased depending on the skill difference (and outcome) of the players. However, the major difference is that the degree to which skills are changed is at least somewhat proportional to the level of uncertainty. Given that each player has his own variable for uncertainty, a new player (with high uncertainty) can move up or down faster when playing against another player, for whom the system is confident about the skill level (Herbrich R. & Graepel T., 2006).

To implement the TrueSkill model, 'trueskill' package for Python (Heungsub L., 2012) was used. The library provides classes and functions required to create the teams with relevant ratings and run them through the training process by adjusting the rating after each match. It also contains methods for predicting game results.
The following hyper-parameters are set by default:

- Mu = 25.0
- Sigma = 25.0 / 3 = 8.34
- Beta = 4.167
- Tau = 0.0834

Where Mu represents the starting point for a rating and Sigma is the standard deviation of ratings (typically, one-third of Mu). Also, Beta is the value that represents a 76% winning chance and is normally half of the sigma. In this model, Beta is meant to represent the gap between different skill levels. For example, Player A has a rating of 50, while Player B has a rating of 45. What is the chance that Player A will win Player B? This depends on the value of Beta. In this example, if Beta is 5, then Player A will win Player B in about 76% matches. Moreover, TrueSkill uses Tau, which determines how easy it is for the model to become certain of a player's skill. In other words, it defines how fast a player can climb up the rank ladder. Before updating the player's skill, Tau squared is added to the player's standard deviation to ensure some volatility in player's rank positions. Lastly, draw probability is the chance that the outcome of a match is going to be a draw between the two teams (Herbrich R. & Graepel T., 2006; Moser J., 2010).

3.3.1 Basic TrueSkill

Using the default parameters that were described, the model is trained and evaluated. The initial accuracy is presented in Figure 13 and full figures are given in Appendix P.

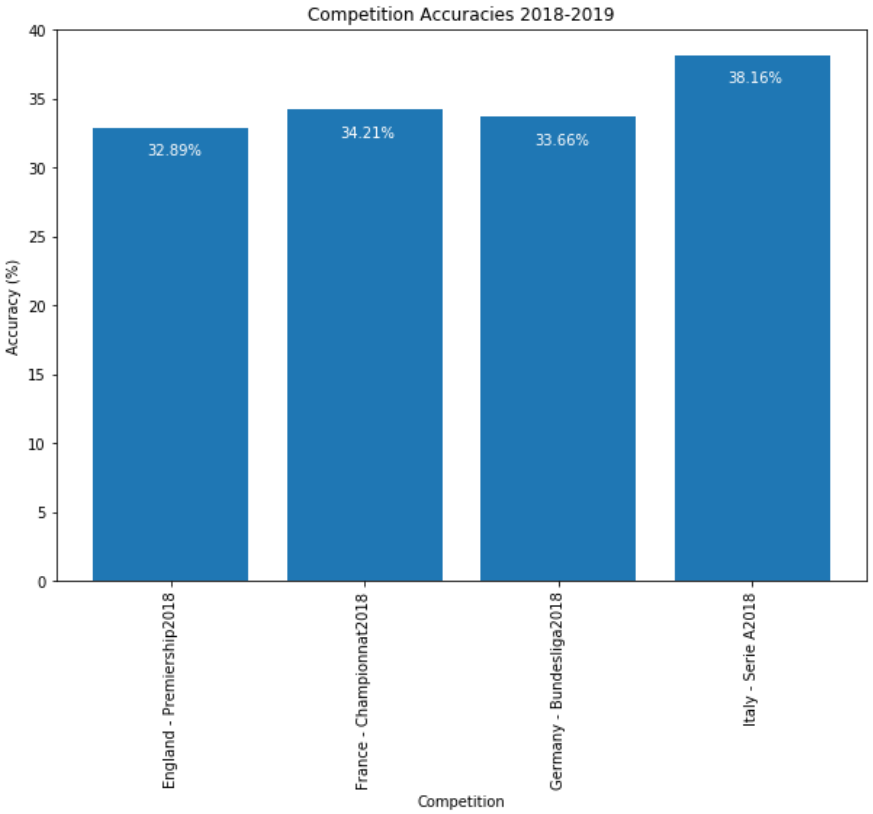

Figure 13: Basic TrueSkill competition overall prediction accuracies

The performance of the initial TrueSkill model appears to be about as effective as random guessing (given that there are 3 possible outcomes in football). To determine the cause of this, the confusion matrix and useful metrics are presented in Figures 14 and 15.

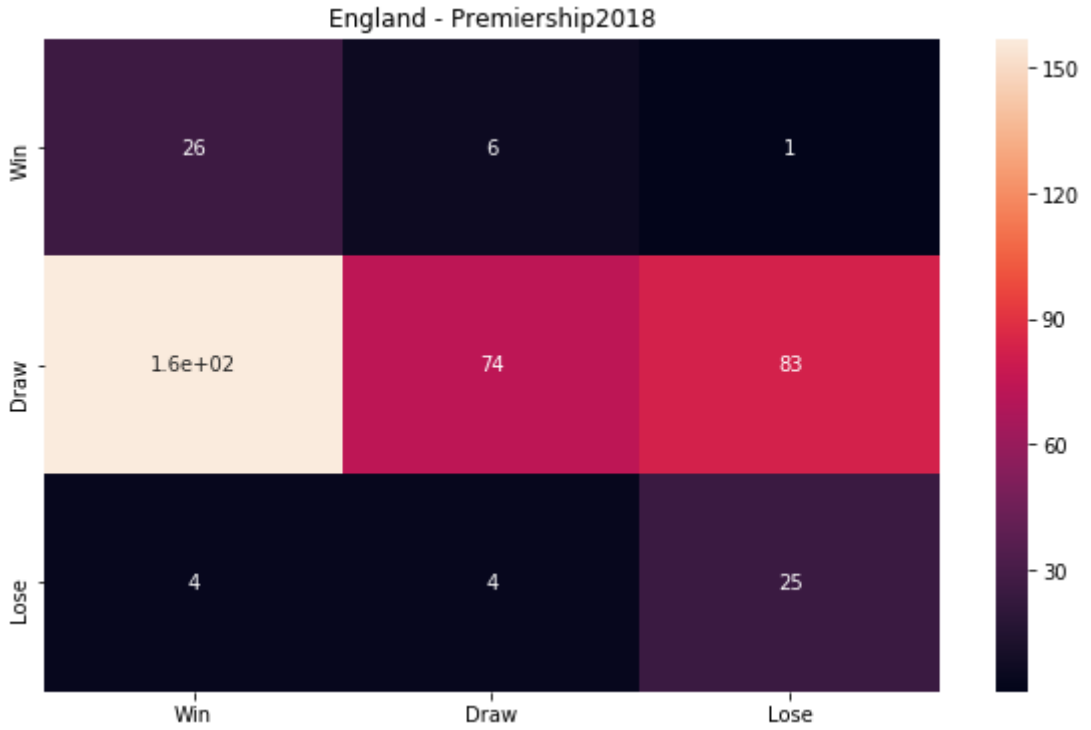

Figure 14: Initial TrueSkill confusion matrix for England Premiership 2018-2019

```
Accuracy: 32.89%
MAE: 0.38
RMSE: 0.40
Precision:      Win 0.79        Draw 0.24       Lose 0.76
Recall:         Win 0.14        Draw 0.88       Lose 0.23
F1:     Win 0.24        Draw 0.37       Lose 0.35
```

Figure 15: Initial TrueSkill metrics for England Premiership 2018-2019

As can be observed in Figures 14 and 15 (all additional figures for TrueSkill can be found in Appendix P), the reason behind why the initial TrueSkill model performed poorly, nearly random, is that it appears to only select draw as a possible outcome in the majority of the cases. Given this fact, it seems logical to tune the value of Beta.

3.3.2 Beta

To determine the optimal value of Beta, it is varied from 0.01 to 1.9. For each value of Beta, a model is trained and it's accuracy is obtained. Graphs, which are created for seasons 2008/09 - 2017/18, are created after obtaining team data for at least 10 games (per unique team). As evident in Figure 16, adjusting the value of Beta improves the overall accuracy by over 10%. The best value of beta for both the train set and the 2018 - 2019 tournaments appears to be 0.07, which resulted in an average accuracy of 45.03% (see Appendix Q for more information). Specific prediction accuracies for each competition can be seen in Figure 17.

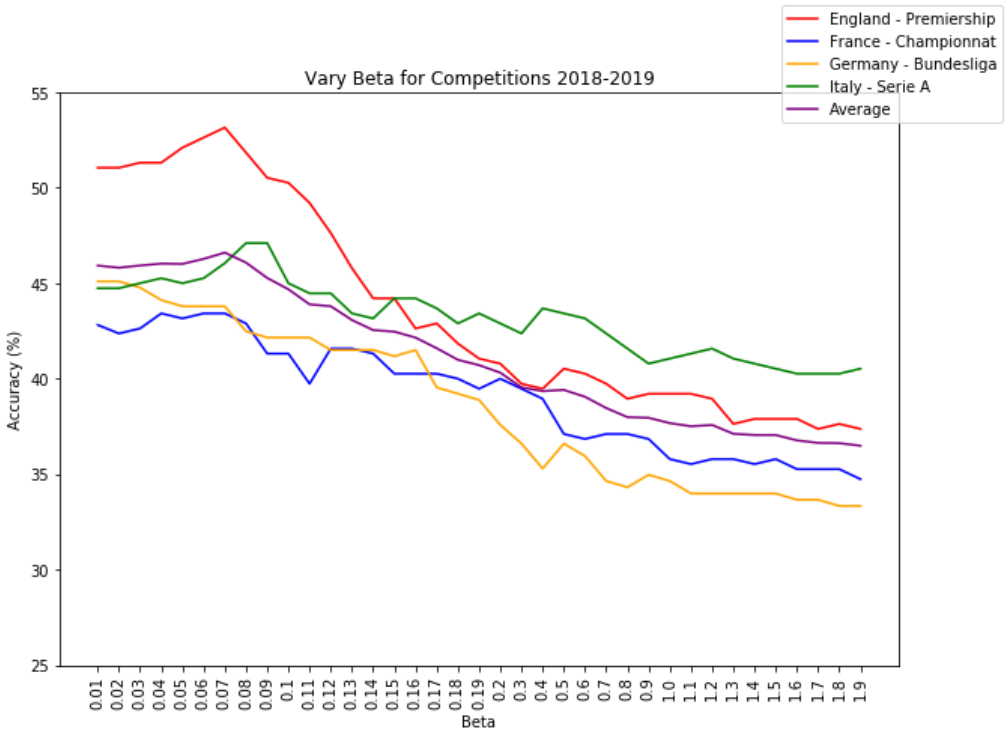

Figure 16: Overall accuracy 2018-2019 (y) varying depending on a beta (x)

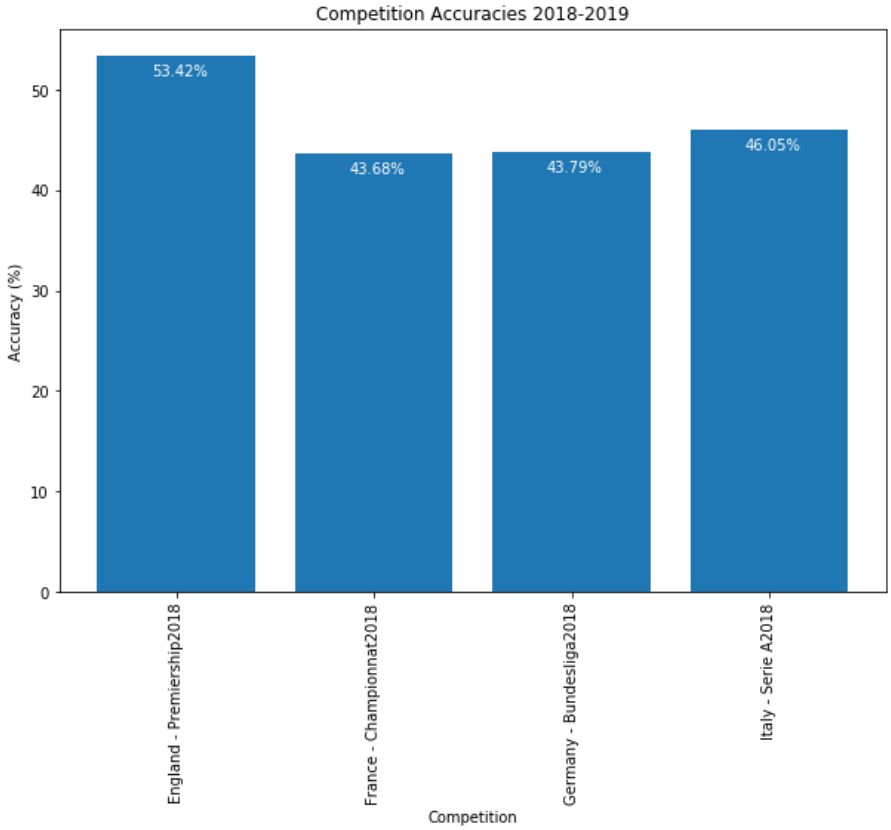

Figure 17: Beta TrueSkill competition 2018-2019 prediction accuracies

Next, the accuracy of the TrueSkill and Bet365 are compared against each other.

Competition	Beta TrueSkill Accuracy	Bet365 Accuracy	Prediction Accuracy Difference

England Premiership	53.42%	58.42%	-5%
France Championnat	43.68%	46.61%	-2.93%
Germany Bundesliga	43.79%	47.06%	-3.27%
Italy Serie A	46.05%	54.74%	-8.69%

Table 7: Beta TrueSkill and Bet365 accuracy comparison

As can be observed in Table 7, the updated TrueSkill model underperforms against the bookie consistently. An updated confusion matrix and accuracy metrics are presented in Figures 18 and 19.

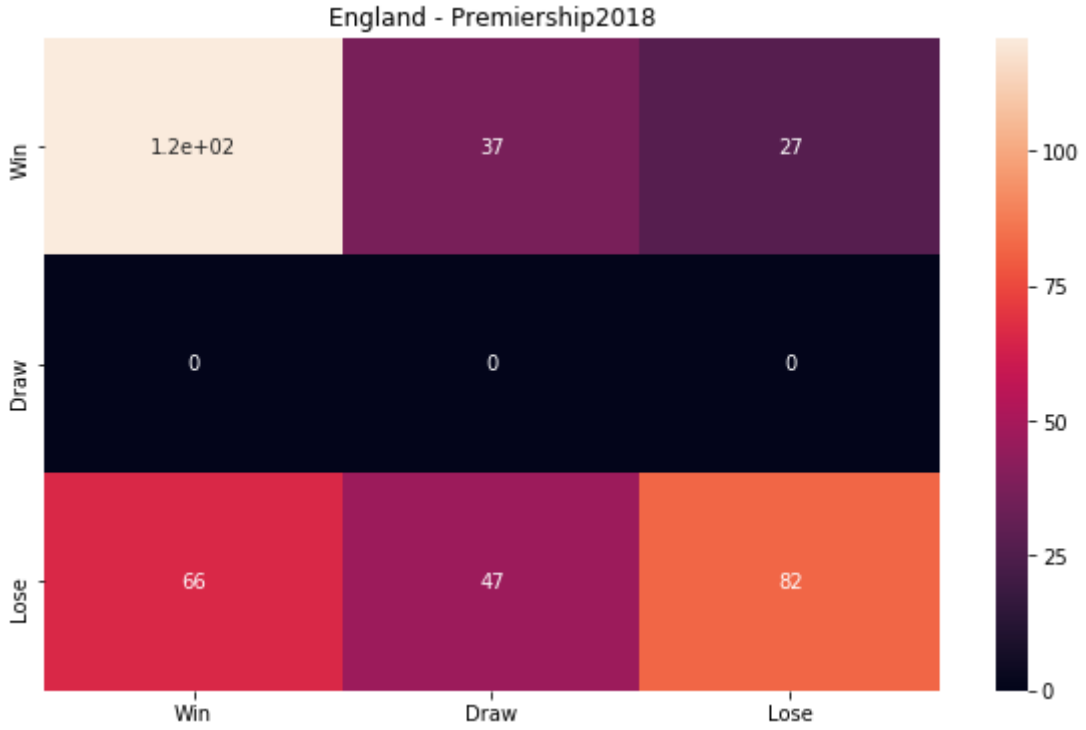

Figure 18: Beta TrueSkill confusion matrix for England Premiership 2018-2019

```
Accuracy: 53.42%
MAE: 0.25
RMSE: 0.29
Precision:      Win 0.65        Draw 0.00       Lose 0.42
Recall:         Win 0.65        Draw 0.00       Lose 0.75
F1:     Win 0.65        Draw 0.00       Lose 0.54
```

Figure 19: Beta TrueSkill metrics for England Premiership 2018-2019

It seems that with the adjusted value of Beta, the model does not predict the draw at all. This is similar to what was observed earlier with the Elo system. Perhaps, exploring other hyperparameters and additions to the system may improve the performance of TrueSkill further.

3.3.3 Tau

Another variable of TrueSkill that may benefit from some adjustment is Tau. As described earlier in the paper, Tau is responsible for the volatility of the system.
The results of varying the variable and its effects on the accuracy can be seen in Figure 20.

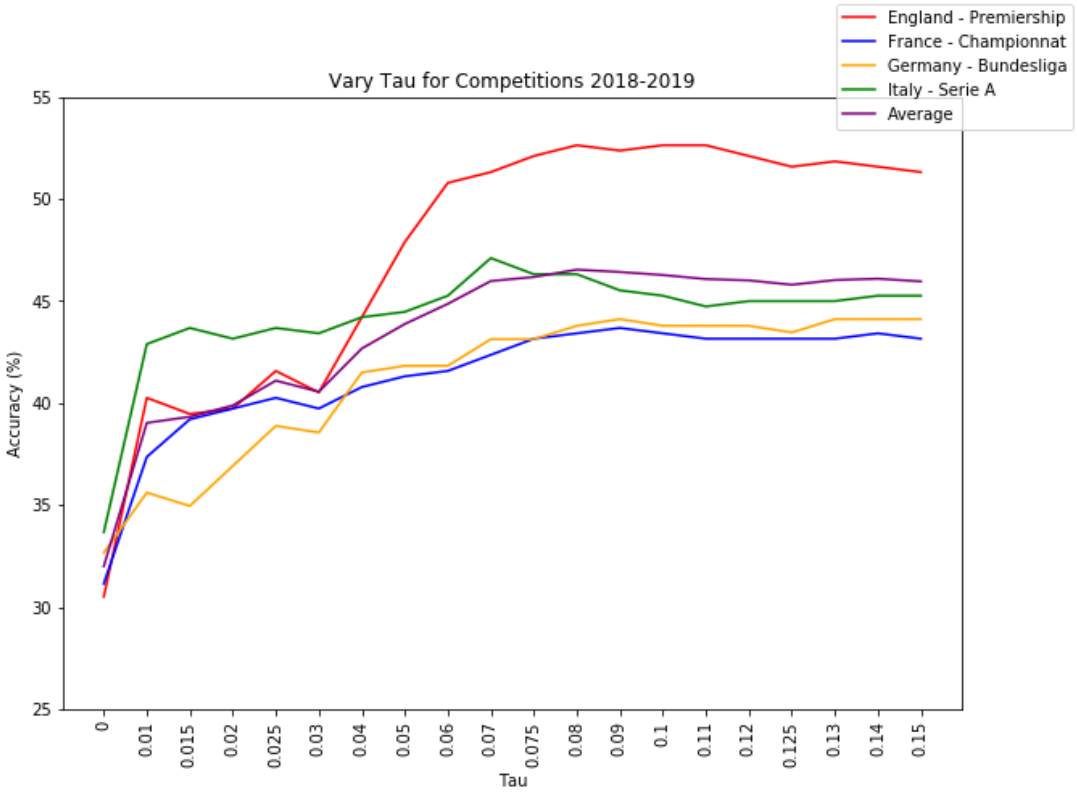

Figure 20: Overall accuracy 2018-2019 (y) varying depending on Tau (x)

As evident in Figure 20, the optimal value for Tau appears to be the default 0.0834. This value worked best for both the train set and the 2018 - 2019 tournaments. Thus, this variable is left unchanged from the original TrueSkill system.

3.3.4 Mu and Sigma

Another pair of variables that may be subject to further tuning is Mu and Sigma, which determine the starting rating of each team and the initial standard deviation. Surprisingly, changing Mu and Sigma appeared to have no effect on the overall system (see Appendix R).

3.3.5 Home Advantage

Next, the home advantage variable is added to the formula, adding a constant factor to the home team's chances of winning. The results are presented in Figure 21.

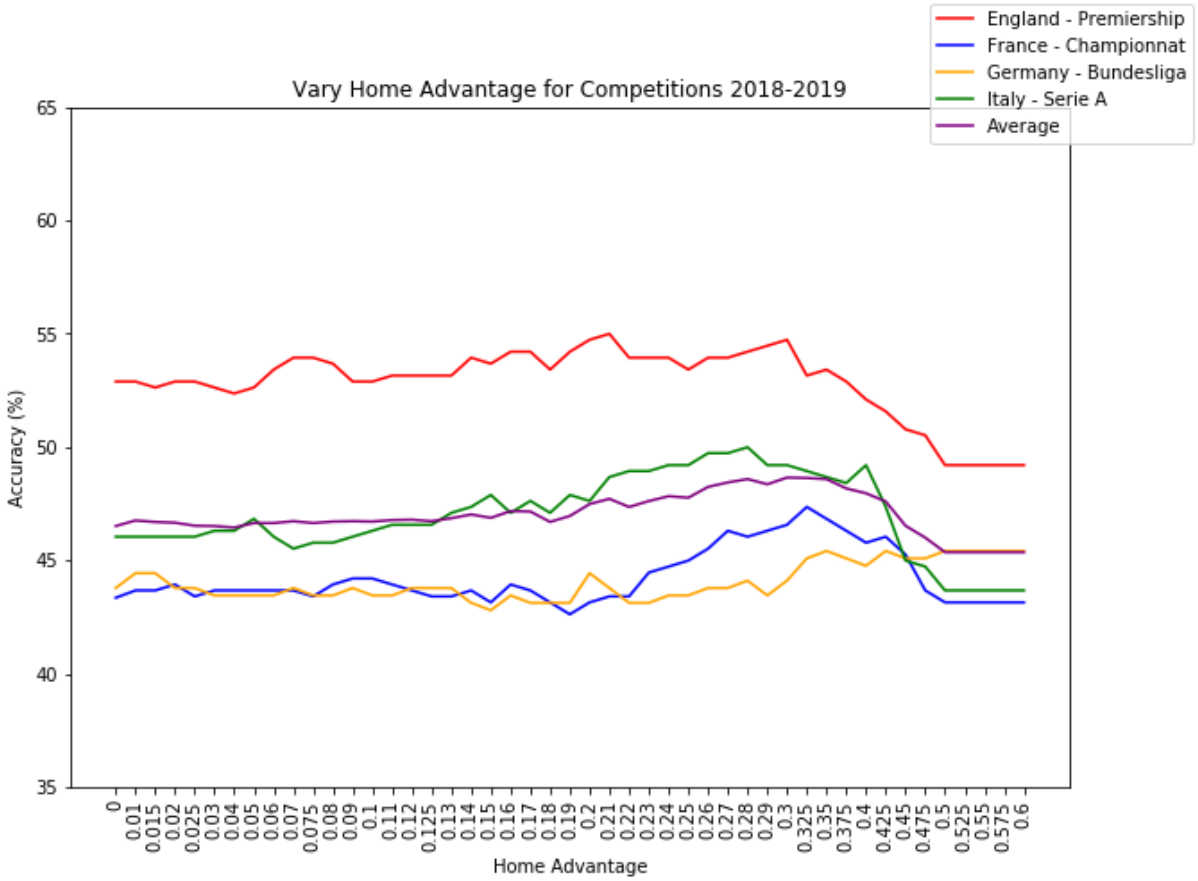

Figure 21: Overall accuracy 2018-2019 (y) varying depending on a home advantage (x)

As can be seen in Figure 21, a certain amount of home advantage improves the overall accuracy of the competitions. However, a large enough home advantage will decrease the accuracy instead of improving it. Based on Figure 21, the best-performing model had a home advantage of 0.3. The exact accuracy for each competition is presented in Figure 22.

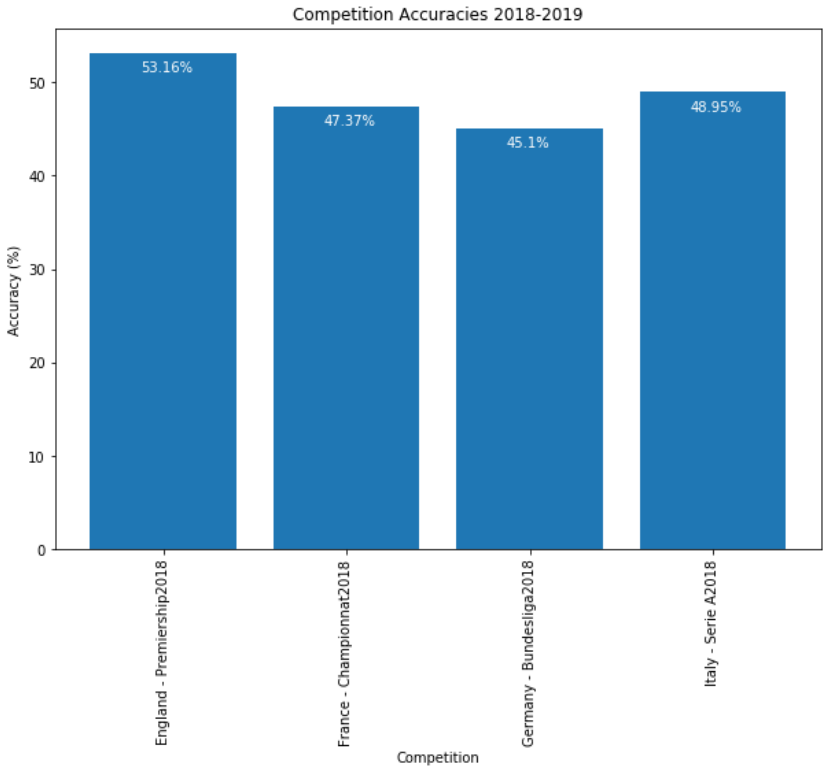

Figure 22: Home Advantage TrueSkill competition 2018-2019 prediction accuracies

The results of comparing the previous TrueSkill model and the one that has a home advantage accounted for is presented in Table 8.

Competition	Previous TrueSkill Model	Updated TrueSkill Model	Difference
England Premiership	53.42%	53.16%	-0.26%
France Championnat	43.68%	47.37%	+3.69%
Germany Bundesliga	43.79%	45.10%	+1.31%
Italy Serie A	46.05%	48.95%	+2.9%

Table 8: Comparison of the accuracy performance of the previous TrueSkill model with the one that utilises Home Advantage

As evidenced in Table 8, it appears that the introduction of the home advantage improved the overall accuracy for 3 out of 4 competitions, with the only exception being the England Premiership.

3.3.6 Overall vs. Competition-based TrueSkill

Similarly to Section 3.2.9 in Elo, the hyperparameters are obtained using 2 different methods. The Overall TrueSkill hyperparameters are obtained by selecting those that yield the highest accuracy overall. Using previous sections, the best overall hyperparameters can be selected. Similarly to the Elo model, grid search is used to examine whether there is a set

of parameters that were not discovered when exploring each variable independently. Indeed, for TrueSkill, grid search revealed a better combination for beta and home advantage (see Appendix S). Performance of the Overall TrueSkill on test and train sets is presented in Figures 23 and 24.

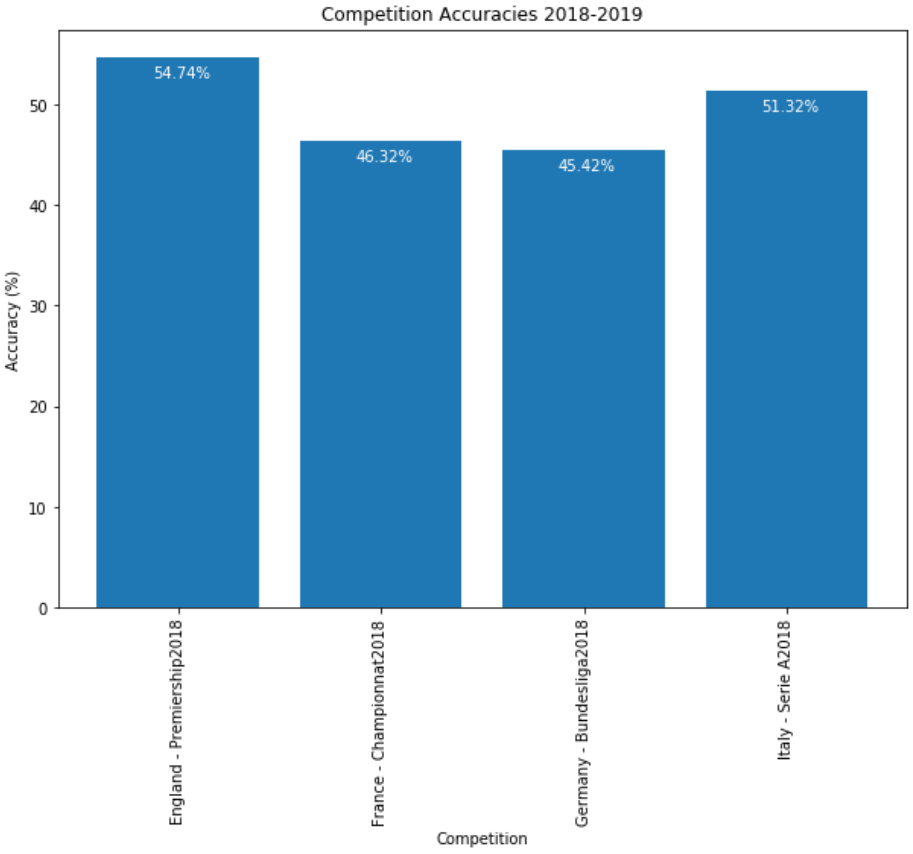

Figure 23: Overall TrueSkill test set accuracies (tournaments 2018 - 2019)

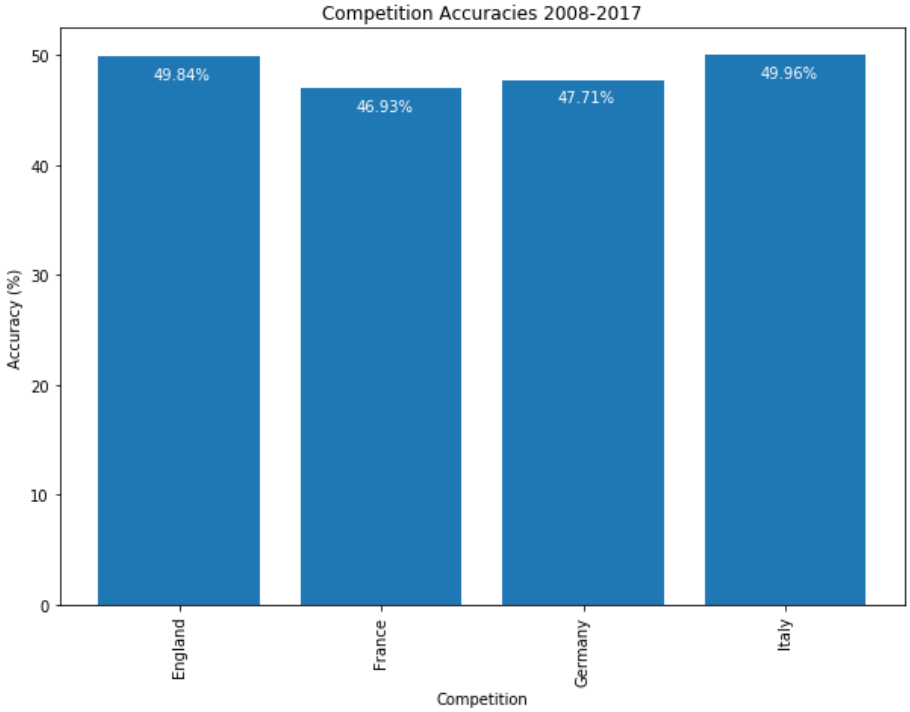

Figure 24: Overall TrueSkill train set accuracies (tournaments 2008 - 2017)

Next, TrueSkill hyperparameters are calculated for each competition individually. Similarly to the Overall TrueSkill, a grid search is used (see Appendix S).

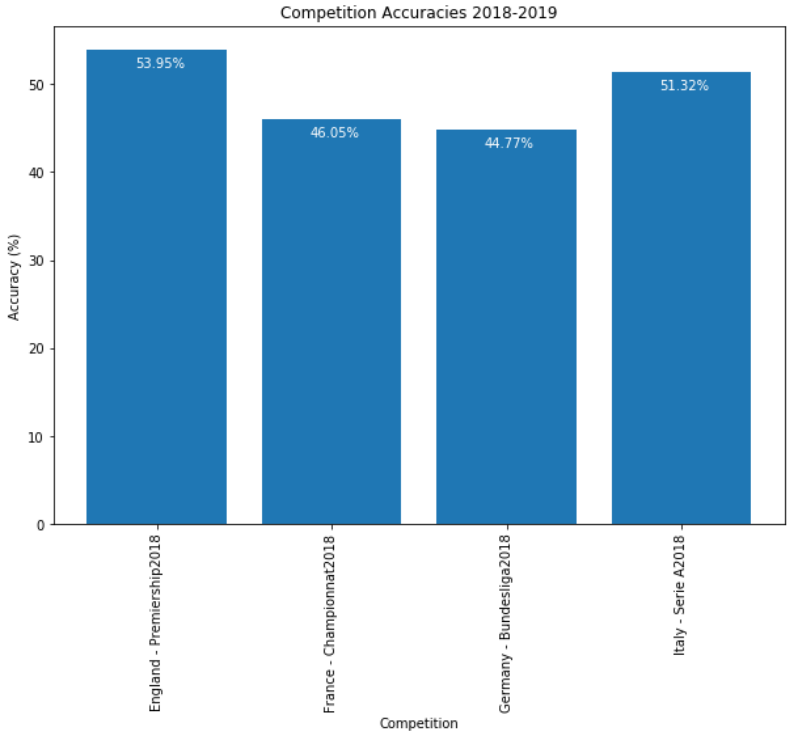

Figure 25: Per-competition TrueSkill test set accuracies (tournaments 2018 - 2019)

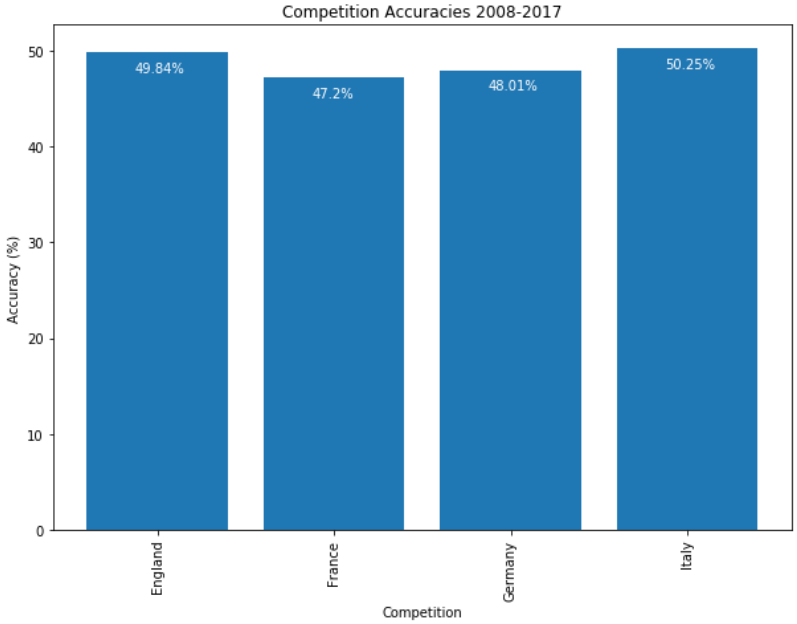

Figure 26: Per-competition TrueSkill train set accuracies (tournaments 2008 - 2017)

Evidently, Per-competition TrueSkill outperforms Overall TrueSkill in all competitions on the train set. Only for the England Premiership, the performance stayed at the same level for both systems (Appendix T has more detailed information about this). The difference between the Overall TrueSkill and Competition-based TrueSkill is highlighted in Table 9.

Competition	Overall TrueSkill	Per-competition TrueSkill	Accuracy Difference
England Premiership	49.84%	49.84%	0%
France Championnat	46.93%	47.2%	+0.27%
Germany Bundesliga	47.71%	48.01%	+0.3%
Italy Serie A	49.96%	50.25%	+0.29%

Table 9: Overall TrueSkill vs Per-competition TrueSkill performance

3.4 Neural Network

Using the Neural Network in professional betting is something that has been increasing in popularity in the 2010s. There are various challenges that come with using NNs for this purpose. First of all, Neural Networks require vast amounts of training data to generalise well on something so complex as football and it can be difficult to retrieve necessary volumes of reliable data. Moreover, this is complicated by the fact that data can often be malformed or missing.

Furthermore, if the dataset is not carefully examined, Neural Networks may learn things that are not true in nature. For example, it is possible for a Neural Network to see more examples of football matches that resulted in 8 - 0 rather than 7 - 0. This may lead it to believe that 8 - 0 is a more likely result than 7 - 0 (or some other result), which is not true for obvious reasons (scoring 8 goals is less probable than 7).

Using individual player's data as part of the input to the Neural Network rather than team averages could allow for better prediction accuracy. However, there simply may not be enough accurate data available for the NN to generalise. Consider, for example, a case where Neural Network relies on data about players' performance. Some players will have little to no data available for them, or the only data available will be outdated or irrelevant, such as one for a completely different tournament. This is further complicated by the fact that players can change multiple times during the course of a match.

In this paper, a Multilayer Perceptron is developed to make it learn the underlying patterns and distributions within the data. To train Elo and TrueSkill, only the historical results of the matches were required. However, to train an MLP, a variety of other features are needed, some examples of which have been discussed. Firstly, using the existing dataset, the bookie odds (Bet365) are used as part of the feature vector since these can provide some insight into what the bookie thinks the outcome of the game should be.

Following the example of the Poisson model, for each team, the average number of scored goals home, the average number of conceded goals home, the average number of score goals away and the average number of conceded goals away is calculated. Additionally, win, draw and lose chances at home and away are used. The detailed list of final feature vector fields can be found in Appendix U.

The first NN architecture that is created only has 1 hidden layer with the same number of neurons as the input layer. For predicting the outcome, the model uses 3 neurons in the output layer, each representing a distinct match outcome (win, draw, lose). Upon computing the outcome probabilities, a softmax function is applied to scale the results and retrieve the most likely one. The results from this model allow us to benchmark other architectures against to determine whether an updated model has better performance.

A variety of parameters are considered to improve the performance of the initial Neural Network. These are batch size, the number of epochs, type of optimizer, learning rate, momentum (where appropriate), weight initialisation mode, activation function, dropout and different architectures.

To find the optimal hyperparameters, a grid search is used to find best-performing values. When tuning a parameter in this manner, the data that the model is trained on is for tournaments starting from 2008 - 2009 up to 2017 - 2018 (inclusive). When applying the grid search, the GridSearchCV from the scikit-learn package is used with k-fold cross-validation of 10. Note, that while k-fold cross-validation is used during a general grid search, separate metrics that are relevant to the 2018 - 2019 tournaments are provided where appropriate. In such cases, the training process is exactly the same as it would be in a real scenario of creating a working model. The final evaluation is performed on a completely unseen set of data for tournaments 2018 - 2019. This process is visualised in Figure 27.

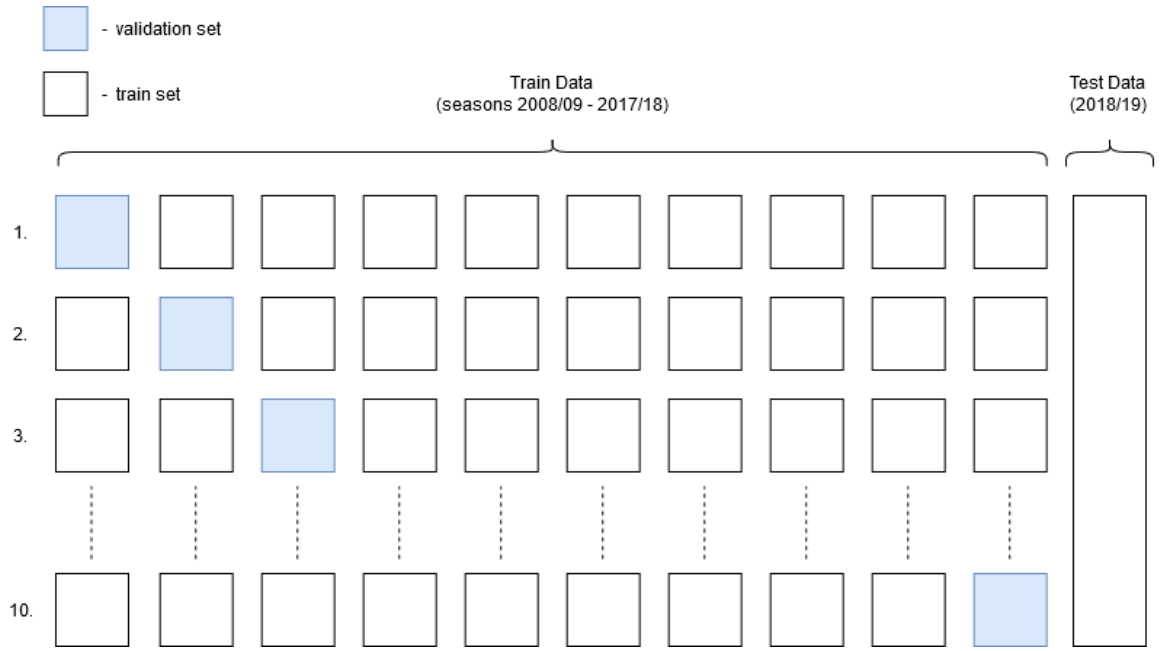

Figure 27: neural network training and evaluation splits visualised

3.4.1 Learning Rate

After getting familiar with the data, the batch size and number of epochs are set to 128 and 30 respectively. These are preliminary values that should help first determine the optimizer and an appropriate learning rate before tuning other parameters more carefully. The learning rate impacts how much the weights are adjusted after every epoch. A small learning rate may result in slow training or the neural network getting stuck. On the other hand, a learning

rate value that is too high can cause unstable learning. To tune this hyperparameter, the best optimizer for every competition is used and a grid search is utilized to find an optimal value for the learning rate (since this variable is something that all selected optimizers share in common).

Using a Grid Search, accuracies for different learning rates are calculated and compared against each other to find optimal parameters. Both average accuracy for all competitions and optimal parameter values for each competition in isolation are considered. Appendix V presents the results of the grid search. Using the overall accuracy, learning rate 0.001 appears to perform best both in terms of accuracy and standard deviation of the results. Generally, lower values of learning rate seem to perform better for football, yielding both higher accuracy and lower standard deviation. Considering the learning rate on a per-competition basis, a similar trend can be observed. For the England Premiership, France Championnat and Italy Serie A 0.001 is selected as the optimal learning rate. Only for Germany Bundesliga, the value of 0.005 is chosen. When other parameters, such as the architecture of the Neural Network, are adjusted, the learning rate may need to be revisited for further tuning.

3.4.2 Optimizer

To find an optimal optimizer for the Neural Network, several NNs are created, each with a different optimizer. The performance of these models is then compared in terms of prediction accuracy and the standard deviation to find one that is most suitable for the football problem. Based on the workings in Appendix W, Adamax optimizer is the best choice for overall accuracy, that is, when a single Neural Network is used to predict all of the competitions. It appears to offer the best accuracy to standard deviation trade-off. Considering the optimizer on a per-competition basis, meaning that a separate model is created and trained for a single competition, results in other values. For the England Premiership and France Championnat, a similar trend can be observed. Thus, Adamax is also used for both of those competitions. However, while Adamax has high accuracy for both Germany Bundesliga and Italy Serie A, it lacks the performance when it comes to the standard deviation. Adagrad optimizer offers a better combination of accuracy and standard deviation for these competitions.

3.4.3 Batch Size and Epochs

The batch size determines the number of examples from the overall training set that the Neural Network uses during a single epoch. This requires less training to train the network since it only uses a subset of the data. It can also result in faster training because the weights are updated at the end of the epoch. However, small batch size can result in the Neural Network being less accurate because it is not calculating the gradient correctly.

The number of epochs sets how many times an entire dataset is passed through the Neural Network during training. Choosing an optimal number of epochs is vital to create a model that is neither under fitted or overfit. Too few epochs can result in the Neural Network not generalising enough to learn the underlying function. On the other hand, a high number of epochs may cause the model to learn all the points in the data set perfectly, thus overfitting. See Appendix X workings and relevant figures.

To determine the optimal values for batch size and number of epochs, a grid search can be used. Separate searches are used to find values for the overall Neural Network model as well as per-competition values. Batch size of 128 and 40 epochs appear to be best overall. However, each competition has a different combination of these that result in the highest accuracy and standard deviation balance. For the England Premiership, a batch size of 64 with 30 epochs appears to perform best. France Championnat has 128 batch size and the same number of epochs. The German Bundesliga has 64 batch size and 50 epochs. Lastly, Italy Serie A has 128 batch size and 30 epochs.

3.4.4 Initialisation Mode

The initialisation mode determines how the weights of the Neural Network's layers are set at the beginning. There are numerous different initialisation modes available in Keras. These include uniform, lecun_uniform, normal, zero, glorot_normal, glorot_uniform, he_normal, he_uniform. By varying the kernel initialisation and building separate models with each one, it is possible to examine the differences in accuracy and their consistency for different initialisation modes.

The best kernel initialisation mode overall appears to be 'glorot_uniform' with 'glorot_normal' close second. Several other kernel initialisers have similar performance and may also be appropriate to use for this problem. Generally, zero-mode demonstrated the lowest standard deviation, but also one of the lowest accuracies. In regards to per-competition initialisers, England Premiership, France Championnat and Italy Serie A all had 'glorot_uniform' as the best-performing kernel. However, for the German Bundesliga, 'uniform' seems to work best. Similarly to the average performance, each competition has more than one kernel that appears to be appropriate for use (see Appendix Y for all workings).

The updated accuracy and standard deviation for the Neural Network with per-competition kernel initialisers are presented in Figures 28 and 29.

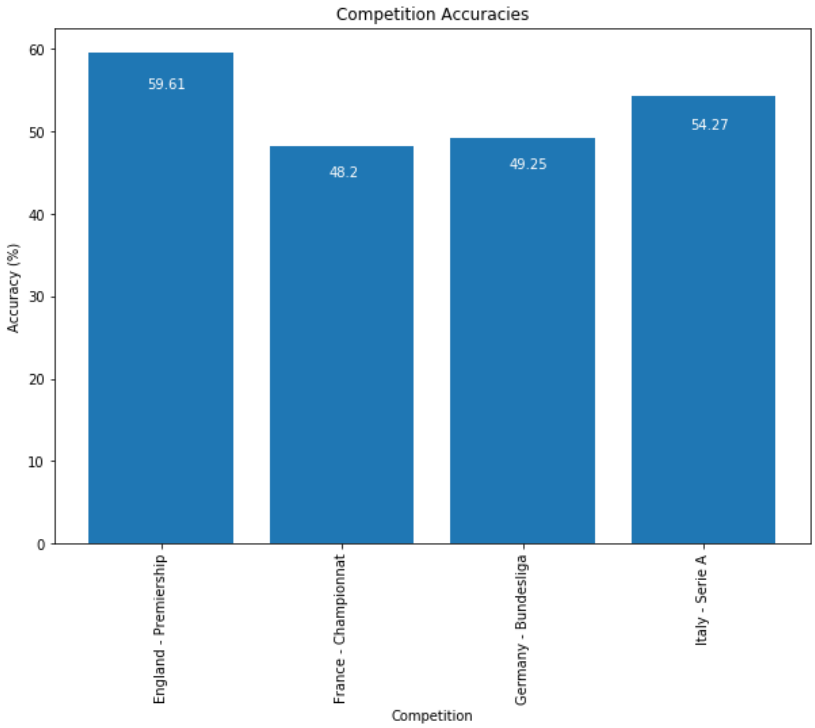

Figure 28: Overall competition accuracy with per-competition kernel initialisation for tournaments 2018 - 2019

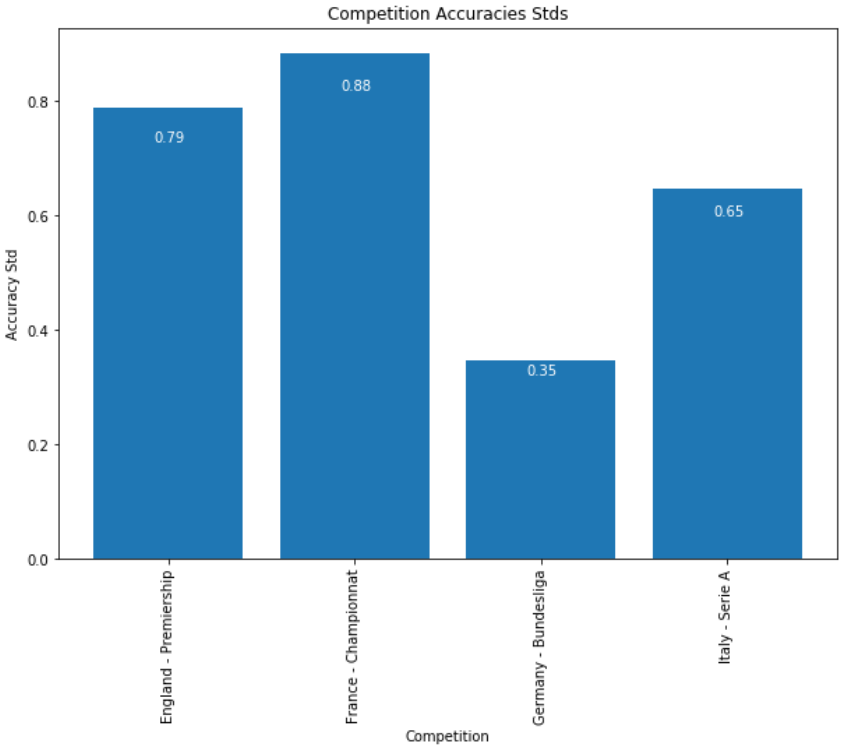

Figure 29: Overall competition accuracy std with per-competition kernel initialisation for tournaments 2018 - 2019

3.4.5 Architecture

Several different architectures were selected for further examination. To determine the optimal architecture, a grid search that encompassed 3 hidden layers was used. For the first

layer, 10, 20, 25, 30, 35, 40, 45, 50, 60 and 70 number of nodes were used. For the second layer, the figures were 0, 5, 10, 15, 20, 25, 30, 35 and 40. Lastly, for the 3rd layer, 0, 3, 5, 10, 15, 20 and 25 were attempted (full calculations are presented in Appendix Z). However, unlike with other parameters, it is impossible to compare the consistency of architectures from a single graph since there are a total of 2,520 different architectures (630 for each competition). Thus, each architecture is assigned a quality score that is calculated as:

$$Quality = accuracy - 1.5 * sd \quad \textbf{(10)}$$

The idea behind this is to provide an objective score measure for each architecture that consists of both the accuracy and standard deviation. Using Formula 10, the best overall architecture has 2 hidden layers. The first hidden layer has 10 neurons while the second one has 10. The average accuracy and standard deviation is 52.18% and 2.35% respectively.

Considering the architectures on a per-competition basis, the results are different. For the England Premiership, the best architecture (using the score formula) has 20 neurons in the first hidden layer and 20 in the second. This architecture results in 54.45% accuracy and 2.6% standard deviation. Next, for France Championnat, the best-performing architecture has 50.63% accuracy and 1.89% standard deviation. The architecture has 3 layers with 30, 5 and 5 neurons respectively.

Finding optimal architecture for Germany Bundesliga, the performance metrics are 49.45% and 0.96% for accuracy and standard deviation respectively with 25 and 25 neurons in the two hidden layers. Lastly, tuning Italy Serie A, the best accuracy of 53.42% and deviation of 2.53% can be achieved with 3 layers that consist of 20, 20 and 10 neurons.

The updated architectures have resulted in an increased standard deviation for all competitions except England Premiership. This appears to be caused by the complexity of the updated architectures, which in turn require adjustments to the learning rate, batch size and number of epochs. For France Championnat, the number of epochs was increased to 70 and the learning rate was decreased to 0.00025. Next, for Germany Bundesliga, only the number of epochs needed to be decreased to 10. Lastly, for Italy Serie A, the learning rate and the number of epochs required adjustment. The learning rate was reduced from 0.001 to 0.0005, while the number of epochs was raised to 40. This appears to decrease the standard deviation back to the original levels. The updated accuracy and standard deviation of the models for 2018 - 2019 tournaments are presented in Figures 30 and 31.

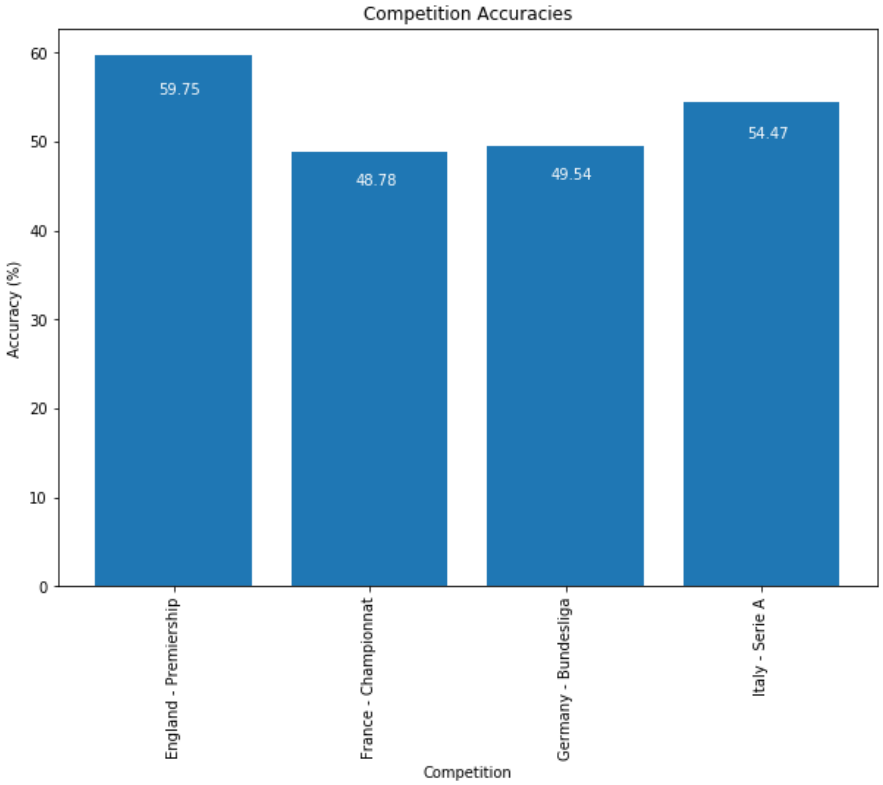

Figure 30: Overall competition accuracy with updated learning rate, batch size and number of epochs for tournaments 2018 - 2019

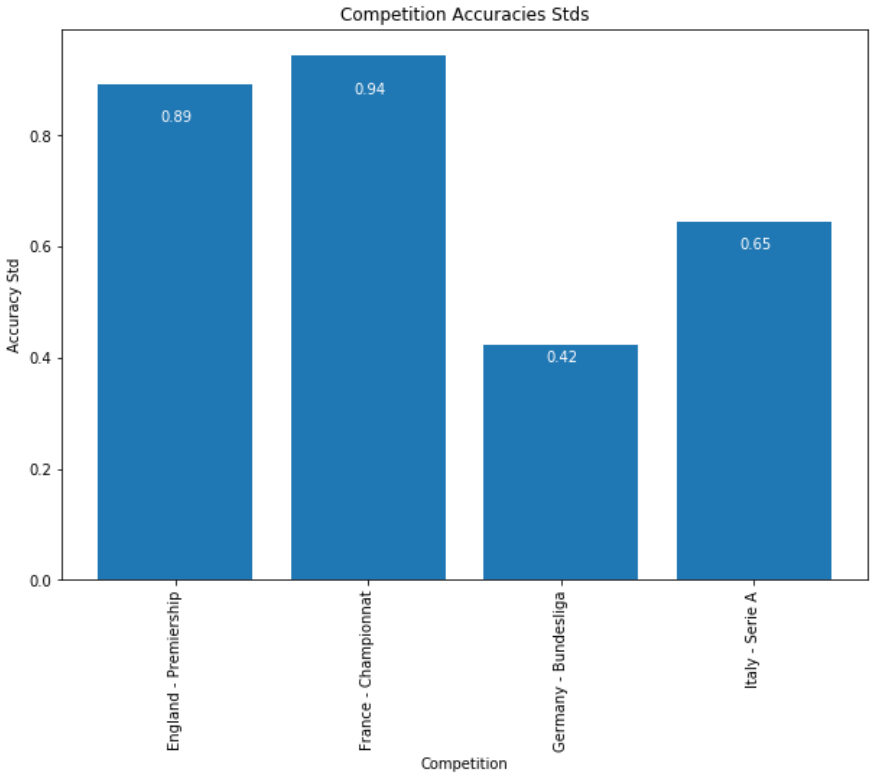

Figure 31: Overall competition accuracy std with updated learning rate, batch size and number of epochs for tournaments 2018 - 2019

3.4.6 Activation Function

Several activation functions were attempted for this problem. These include softmax, softplus, softsign, relu, tanh, sigmoid, hard_sigmoid and linear.

Overall, softsign was the best activation function for the Neural Network (see Appendix AA for figures). However, softplus, relu and tanh all had high accuracy metrics as well. For the England Premiership, tanh was selected as the optimal activation function. Interestingly, the linear function had a similarly good performance for the England Premiership. Next, for France Championnat, the default relu activation function had the best performance. This was also the case for Germany Bundesliga, where activation functions did not significantly impact the performance. Lastly, Italy Serie A worked best with tanh.

3.4.7 Dropout

For a large neural network, it is relatively simple to overfit when the data set is comparatively small. Different values of dropout are used to determine if it would improve the overall accuracy of the network (Ding et al, 2019).

Comparing the accuracy and loss graphs during the training, the models do not appear to be overfitting the data. So, since no dropout is needed to adjust for overfitting, the final accuracy and standard deviation metrics for neural network models for each competition are presented in Figures 32 and 33.

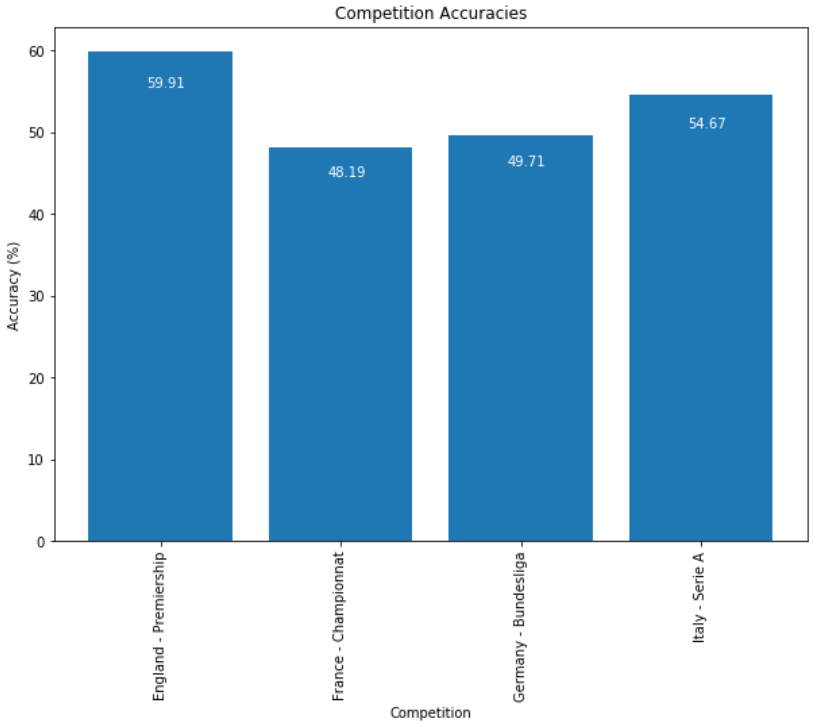

Figure 32: Neural Network competition accuracies for 2018 – 2019 tournaments

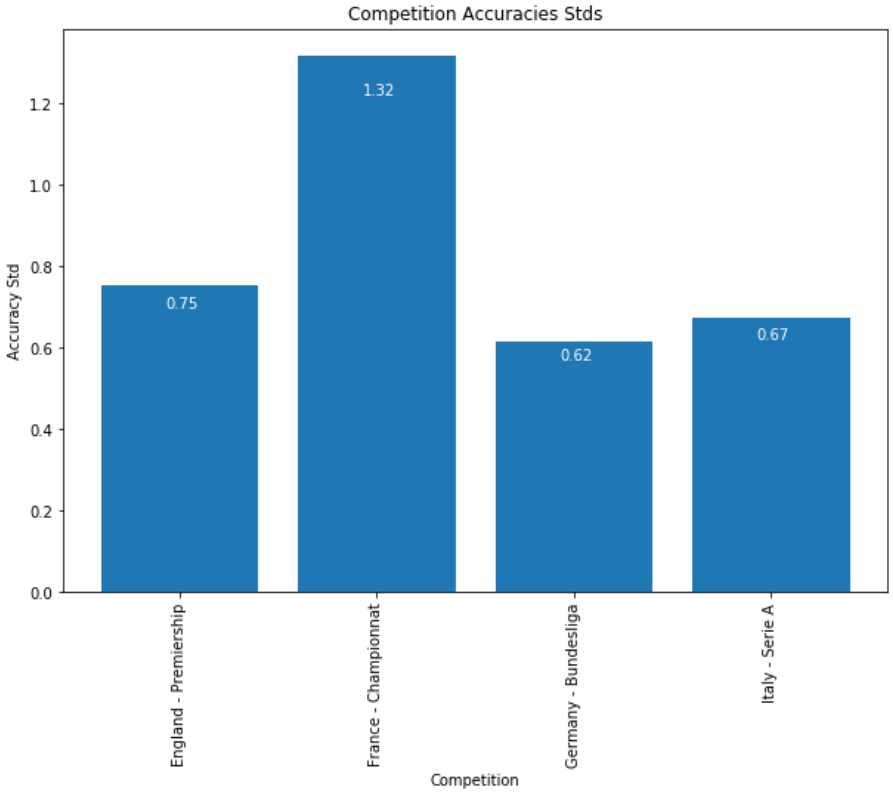

Figure 33: Neural Network competition accuracies standard deviations for 2018 – 2019 tournaments

3.4.8 Final Hyperparameters

Similarly to Section 3.2 about Elo and 3.3 about TrueSkill, a set of hyperparameters was obtained for both the Overall Neural Network as well as Competition-based ones (as described in Sections 3.4.1 to 3.4.7). Appendix AC contains a detailed comparison of Overall Neural Network versus Competition-based one. Clearly, Competition-based Neural Network approach outperforms the former in all competitions but Italy Serie A, where the Competition-based Neural Network performed -0.67% worse than its counterpart. Here, the final set of parameters that were obtained for the Neural Network are listed. For the overall NN, they are as follows:

- England Premiership:
    - 0.001 Learning Rate
    - 64 Batch Size
    - 30 Epochs
    - Adamax Optimizer
    - `glorot_uniform` Kernel Initialisation Mode
    - 20 Neurons in the first layer, 20 Neurons in the second layer
    - Tanh Activation Function
- France Championnat:
    - 0.00025 Learning Rate
    - 128 Batch Size
    - 70 Epochs
    - Adam Optimizer
    - `glorot_uniform` Kernel Initialisation Mode

        - 30 Neurons in the first hidden layer, 5 Neurons in the second hidden layer and 5 Neurons in the third hidden layer
        - RelU Activation Function
- Germany Bundesliga:
    - 0.005 Learning Rate
    - 64 Batch Size
    - 10 Epochs
    - Adagrad Optimizer
    - `normal` Kernel Initialisation Mode
    - 25 Neurons in the first hidden layer and 25 Neurons in the second hidden layer
    - RelU Activation Function
- Italy Serie A:
    - 0.0005 Learning Rate
    - 128 Batch Size
    - 40 Epochs
    - Adagrad Optimizer
    - `glorot_uniform` Kernel Initialisation Mode
    - 20 Neurons in the first hidden layer, 20 Neurons in the second hidden layer and 10 Neurons in the third hidden layer
    - Tanh Activation Function

3.4.9 Ensemble Approach

The standard deviation of the Neural Network still appears to be quite high. This problem has not been discussed often in the literature. For football prediction, the fact that the developed Neural Network can have a deviation in its accuracy by as much as 1% is catastrophic. As could be seen from the results in Sections 3.2 and 3.3, often, models achieve similar levels of accuracy and it is the few percentages that help gain the upper hand when considering hundreds and thousands of bets. Indeed, this is part of the reason why, historically, static, consistent approaches were preferred.

To help combat this issue and reduce the standard deviation of the developed system, an ensemble approach is used. This means that multiple identical models are created, all of which are trained using the exact same data. When making predictions, models classify the given match independently. Then, the final prediction is considered to be the one which received the majority of ‘votes’ from the Neural Networks. The positive effects this approach has on reducing the standard deviation are presented in Figures 34 through 37.

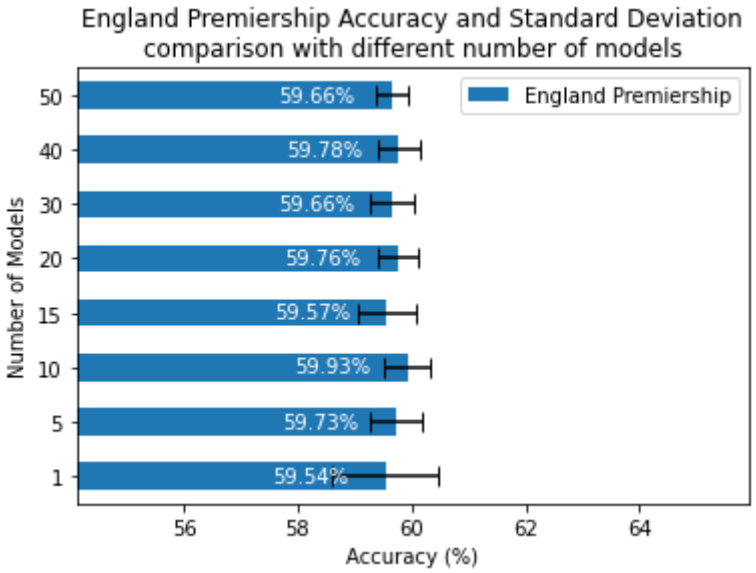

Figure 34: England Premiership accuracy and standard deviation variation when using a different number of models as part of the ensemble approach

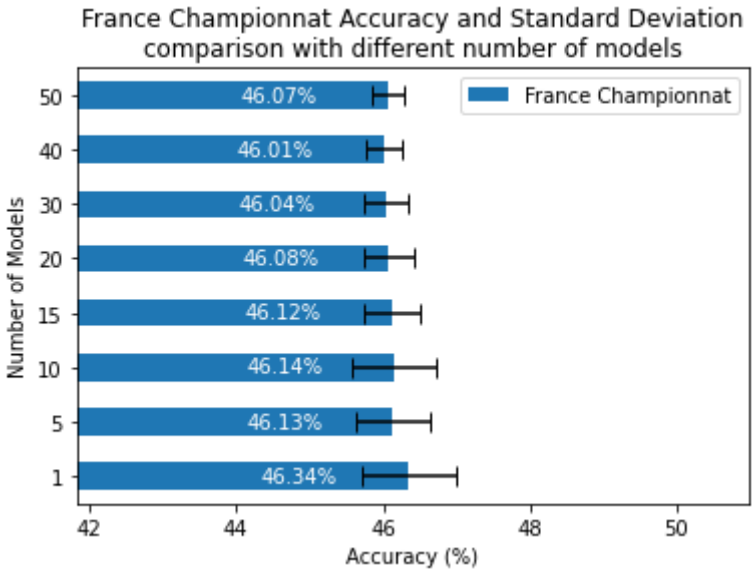

Figure 35: France Championnat accuracy and standard deviation variation when using a different number of models as part of the ensemble approach

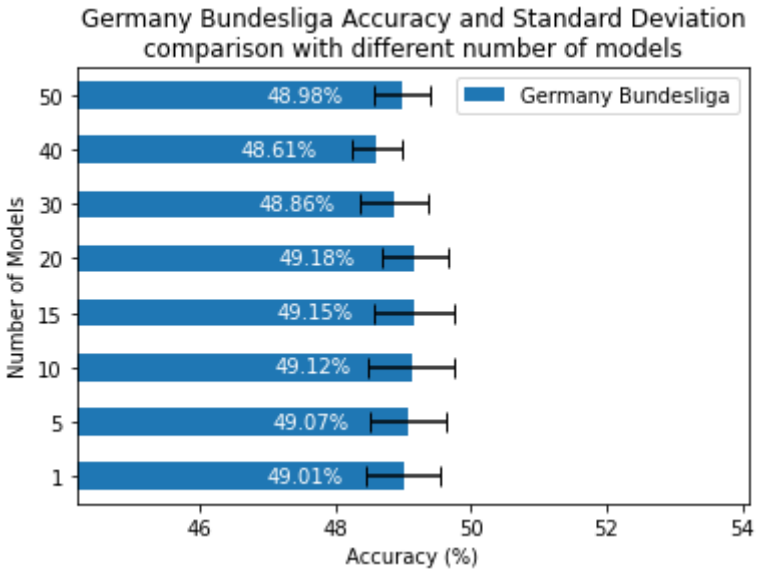

Figure 36: Germany Bundesliga accuracy and standard deviation variation when using a different number of models as part of the ensemble approach

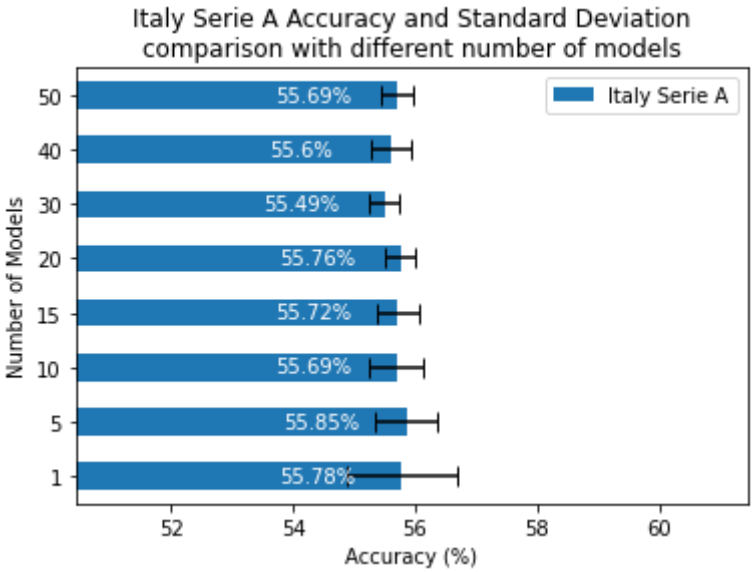

Figure 37: England Premiership accuracy and standard deviation variation when using a different number of models as part of the ensemble approach

The largest number of models that were attempted was 50 due to computational constraints and based on the evidence, the consistency of the system can be improved further by using a larger number of models. From Figures 34 to 37, it is evident that, as the number of models in the ensemble approach increases, the standard deviation of the results decreases significantly. Based on this, 50 Neural Networks are used together to make predictions. This approach reduced the standard deviation of results, leading to a significantly more stable system overall (see Appendix AB for workings). Figures 38 and 39 present updated accuracies and standard deviations for all competitions.

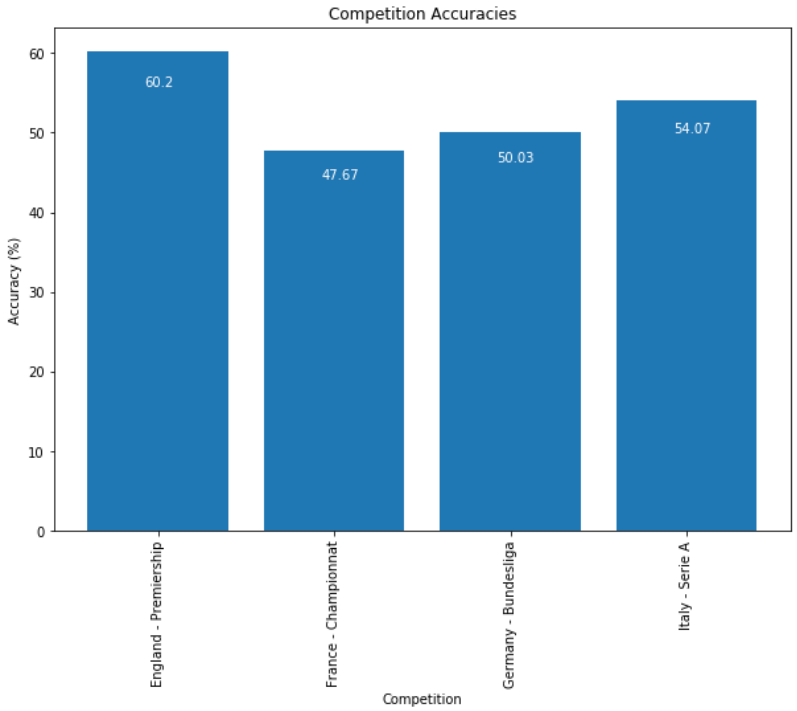

Figure 38: Final Neural Network competition accuracies for 2018 – 2019 tournaments

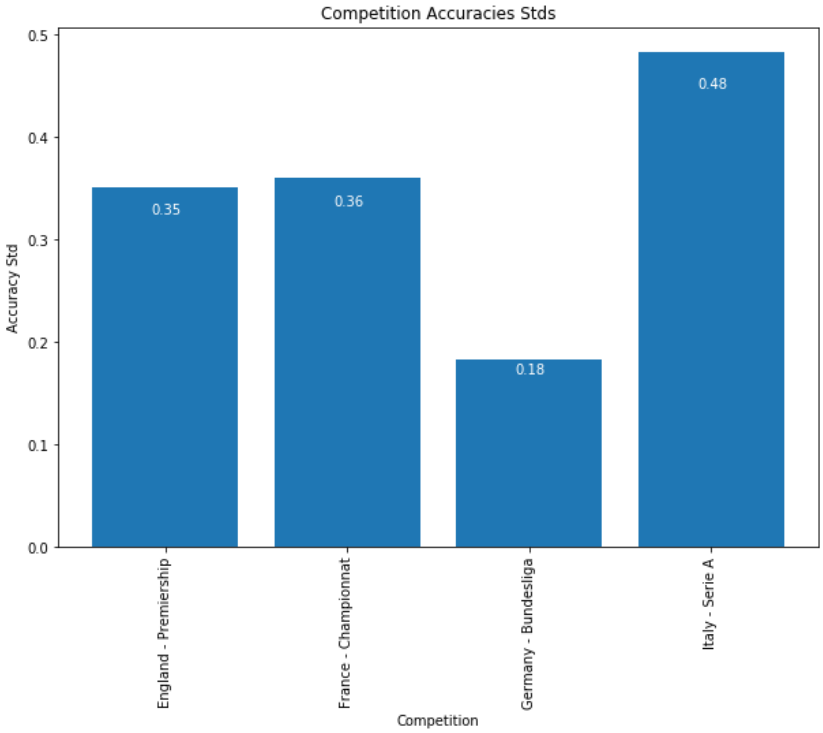

Figure 39: Final Neural Network competition accuracies standard deviations for 2018 – 2019 tournaments

3.4.10 Model Generalisation

To further test how general is the developed Neural Network, it is tested on different sports. Performance of the Neural Network with hyperparameters that were overall best for football is tested against bookies, similarly to what was done for football leagues.

Neural Network with the following parameters is used:

- Batch size: 128
- Epochs: 40
- Optimizer: Adam
- Learning rate: 0.001
- Kernel Initialisation Mode: glorot_uniform
- Architecture: 2 hidden layers, 10 neurons each
- Activation: RelU

The sports competitions that the model's performance is tested on are National Football League, Super Rugby, Twenty20 Big Bash and the National Baseball League. The data is provided by Australia Sports Betting (nd).

For full figures that contain performance metrics for Neural Networks and different sports, see Appendix AD. A comparison of Bet365 performance against Neural Network is given in Table 10. For the National Football League, the Neural Network had 64.59% accuracy on average with 0.58% standard deviation, while Bet365 had 65.08% for that season. The difference in performance, however small one, comes from the fact that the bookie is better at predicting loses. A major difference in predictions is the fact that Neural Network often misclassified loses as wins, while the bookie misclassified wins as loses.

Performance Metric	Neural Network	Bet365
Accuracy	64.59%	65.08%
F1 Win	0.74	0.73
Precision Win	0.83	0.68
Recall Win	0.66	0.78
F1 Lose	0.46	0.52
Precision Lose	0.38	0.58
Recall Lose	0.58	0.47

Table 10: comparison of metrics for NFL using a Neural Network and Bookie predictions

Next, the Neural Network is compared against the NBL season 2018 - 2019. For this sport, it appears that the Neural Network is able to generalise to some extent, but the model cannot clearly distinguish between Wins and Loses, resulting in a significantly worse performance than the bookie. For NBL, the NN had 59.53% accuracy with 0.41% standard deviation while Pinnacle had 63.33% accuracy. This may be explained by the fact that the Neural Network was created with football predictions in mind so the difference in the nature of these sports results in a good performance for NFL, but worse performance for NBL. This would be true, given the assumption that American football is closer to football than baseball, by some measure. The poor performance also could be, in part, as a result of the insufficient size of

the data, which results in the model underfitting and thus, selecting the win class for almost all of the predictions. This assumption is examined later. Both the Neural Network and bookie's performance metrics are given in Appendix AD and a comparison summary is presented in Table 11.

Performance Metric	Neural Network	Pinnacle
Accuracy	59.53%	63.33%
F1 Win	0.73	0.67
Precision Win	0.93	0.70
Recall Win	0.60	0.64
F1 Lose	0.25	0.59
Precision Lose	0.16	0.56
Recall Lose	0.62	0.63

Table 11: comparison of metrics for NBL using a Neural Network and Bookie predictions

Another sport that Neural Network is tested on is rugby. Specifically, the model is trained and evaluated on Super Rugby, results of which are given in Table 12. Intuitively, rugby should be relatively similar to American football, so results comparable to those observed with NFL are expected. For the Neural Network, the accuracy of predictions was 67.38% with 0.97% standard deviation. On the other hand, the bookie only had 62.7% accuracy. This appears to be mainly as a result of the bookie's poor performance with predicting wins. Notably, the Neural Network still predicts loses worse than a bookie.

Performance Metric	Neural Network	Bet365
Accuracy	67.38%	62.7%
F1 Win	0.78	0.69
Precision Win	0.91	0.71
Recall Win	0.68	0.67
F1 Lose	0.48	0.57
Precision Lose	0.38	0.52
Recall Lose	0.65	0.64

Table 12: comparison of metrics for Super Rugby using a Neural Network and Bookie predictions

Lastly, the Neural Network is used to predict the Twenty20 Big Bash. One of the challenges with this sport is that there is not that much data available and thus, the model may not have

enough to generalise and learn the trends appropriately. Nonetheless, the results of testing the model on a sport that is drastically different from football could provide useful insight. Similarly, as with baseball, the Neural Network was unable to produce higher prediction accuracy than the bookie for cricket. With the bookie achieving 45.88% accuracy, the model was able to only reach 44.7% accuracy with 0% standard deviation. Upon inspecting the NN predictions closer, it seems that the model predicted every game as a lose. Thus, it seems that the hyperparameters that were obtained from football are not appropriate for this sport. Moreover, the amount of data available for this competition is significantly lower than what was used for other sports. This could also explain why the model was unable to generalise well and invalidate the conclusions in regards to this sport. The performance metrics for both the NN and the bookie are given in Table 13.

Performance Metric	Neural Network	Bet365
Accuracy	44.7%	45.88%
F1 Win	0	0.53
Precision Win	0	0.51
Recall Win	0	0.55
F1 Lose	0.62	0.36
Precision Lose	1	0.38
Recall Lose	0.45	0.34

Table 13: comparison of metrics for Twenty20 Big Bash using a Neural Network and Bookie predictions

Next, the average number of matches in each competition is examined to determine if some models were inadequate as a result of the lack of data. The total number of matches per tournament for each competition are given in Figure 40. Only the matches with valid data that could be used for the Neural Network were considered.

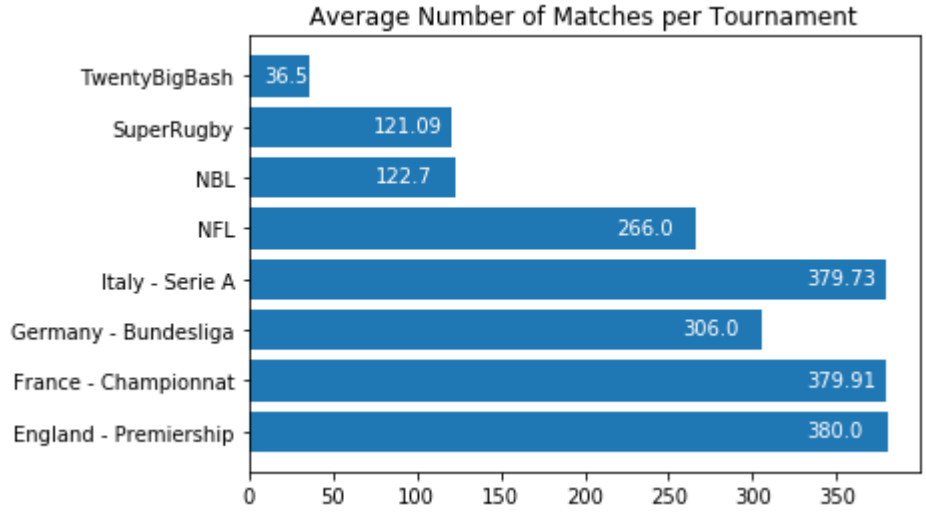

Figure 40: Average number of matches per tournament in different competitions

It is clear that Twenty 20 Big Bash, Super Rugby and NBL have significantly fewer games per season than other competitions that were considered. It is evident that, as the number of matches per tournament drops (and with it the quantity of the data that the Neural Network is trained on), the model loses the ability to distinguish well between different types of outcomes. With a lower number of matches, which can be observed in Super Rugby and NBL, the model almost always predicted wins and was poorly able to classify loses. However, when the number of games is extremely low (Twenty20 Big Bash), the neural network was unable to distinguish between different classes. This is true for the NN architecture that was originally derived for football and may change should the model be tuned specifically to each sport. Even within a single sport, the model performs significantly better when tuned to a specific competition rather than the sport overall so it is likely that the Neural Network could achieve higher accuracy than the bookie in National Football League and Twenty20 Big Bash, where the performance difference is already low.

3.4.11 Neural Network Feature Set Considerations

The predictive power of Neural Networks is in large dependent on the quantity and quality of the data that it observes. As described in Section 2.6, there is a large variety of features that have been researched and the Neural Networks developed in Section 3 utilised the most popular combinations of features. These include odds from a bookie (in this case, Bet365) and statistics for both teams that include home and away performance.

Several other Neural Networks are created, using a different set of features to compare effects that data has on the prediction accuracy of the model. First, a Neural Network with extra fields is created. In addition to all previous features, this model's inputs include information about past team performance against each other for the last 5 matches. This data is one-hot-encoded and appended to the feature set. For example, if in the last 5 matches between team A and B the results were Win, Draw, Draw, Win, Lose, it would be represented as:
[<*other features*>... 1, 0, 0 (first win), 0, 1, 0 (first draw), 0, 1, 0 (second draw), 1, 0, 0 (second win), 0, 0, 1 (first lose).

This model is trained using the same set of hyperparameters as specified in Section 3.4.8. As with the original Neural Network, the same train/test process is applied. 50 models are created that 'vote' for the majority class to make a prediction. To understand how consistent the model is, the training and testing is conducted 25 times (independently of each other) and results are averaged. Using this methodology, prediction accuracy and standard deviations can be obtained, which are presented in Appendix AE. Performance comparison of the original Neural Network and the one that was developed with extra features is presented in Tables 15 and 16.

Tournament 2018 - 2019	Original NN	Past Performance NN	Accuracy Difference
England Premiership	60.20%	59.67%	-0.53%

France Championnat	47.67%	45.13%	-2.54%
Germany Bundesliga	50.03%	49.07%	-0.96%
Italy Serie A	54.07%	54.49%	+0.42%

Table 15: Accuracy of different Neural Networks with different tournaments

Tournament 2018 - 2019	Original NN Std	Past Performance NN Std	Std Difference
England Premiership	0.35%	0.41%	-0.06%
France Championnat	0.36%	0.38%	-0.02%
Germany Bundesliga	0.18%	0.39%	-0.21%
Italy Serie A	0.48%	0.31%	+0.17%

Table 16: Accuracy Standard Deviation of different Neural Networks with different tournaments

Comparing the two Neural Networks, it appears that the original (competition-based) NN outperforms the Past Performance NN slightly. However, this is likely due to the fact that the architecture was specifically grid-searched for Competition-based Neural Network and, with some tuning, accuracies would likely be more similar. This is something that was already observed when comparing Overall NN with the Competition-based Neural Network in Section 3.4.8. Nonetheless, it appears that the extra features, namely one-hot-encoded teams' history against each other, did not improve the overall predictive accuracy in any meaningful way.

4. Evaluation

In Table 17, accuracies for each competition by different systems are presented. Table 18 also gives average accuracy for every system. These are used to compare the performance of the developed models against each other and a bookie.

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	**61.32%**	53.95%	60.20%	58.42%
France Championnat	45.53%	46.05%	**47.67%**	46.61%
Germany Bundesliga	**50.65%**	44.77%	50.03%	47.06%
Italy Serie A	52.89%	51.32%	54.07%	**54.74%**

Table 17: Performance Comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Accuracy	52.60%	49.02%	**52.99%**	51.71%

Table 18: Performance Comparison of different models overall

Using the average accuracy as a key metric, the Neural Network has the highest performance, followed by Elo, Bet365 and, lastly, TrueSkill. TrueSkill consistently performed worse in 3 out of 4 competitions. The only competition where TrueSkill was not the worst model is France Championnat, where it has higher accuracy than Elo. Interestingly, for some competitions, both Elo and Neural Network have considerably higher accuracy than Bet365. Elo has higher accuracy in England Premiership by 2.9% and in Germany Bundesliga by 3.59%. However, for France Championnat the model performed worse (-1.08%) than Bet365. In comparison to the bookie, Elo had the lowest performance for Italy Serie A, -1.85%.

In regard to the Neural Network, the model has higher performance in 3 out of 4 tournaments with the only exception being Italy Serie A, where the model has 0.67% worse performance than Bet365. For England Premiership, France Championnat and Germany Bundesliga, Neural Network has 1.78%, 1.06% and 2.97% higher accuracy respectively than Bet365.

To understand better what one system or the other was better at and where it lacked, F1 metrics for Wins and Loses separately are considered. These are presented in Tables 19 through 22.

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	**0.73**	0.69	0.71	0.7
France Championnat	0.59	0.61	**0.62**	0.59
Germany Bundesliga	0.63	0.60	**0.64**	0.6
Italy Serie A	0.65	0.64	**0.67**	0.66

Table 19: Win F1 comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Win F1	0.65	0.635	**0.66**	0.6375

Table 20: Win F1 comparison of different models overall

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	**0.6**	0.35	0.59	0.59
France Championnat	**0.42**	0.30	0.41	0.4
Germany Bundesliga	**0.47**	0.30	0.39	0.39
Italy Serie A	0.56	0.47	**0.58**	0.56

Table 21: Lose F1 comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Lose F1	**0.5125**	0.355	0.4925	0.485

Table 22: Lose F1 comparison of different models overall

Comparing F1 performance of different models, it is more clear where one model is better than another (the exact figures for recall and precision are given in Appendix AF). For example, using these metrics, it is evident that while Neural Network has lower overall

accuracy with Germany Bundesliga matches, the Elo model performs better specifically when it comes to loses (+0.08). On the other hand, in terms of matches that result in a win, both are relatively similar in performance (-0.01). Using F1, precision and recall also highlight the types of matches that the models are better at than the bookie. Interestingly, it seems that overall, Elo is particularly good at predicting loses while Neural Network - wins. When considering only Wins, Neural Network has the best F1-measure in all competitions except England Premiership. As for Elo, it has the best Lose F1-measure for all competitions but Italy Serie A. This explains why Neural Network has the highest overall accuracy, given that the majority of the matches in football result in a win.

Various extensions of the original systems were considered and their effects on the models thoroughly documented. Based on these, it is evident that some features clearly work better for football, while others not so much. For example, accounting for home advantage appears to be one of the most important factors for a system that aims to classify match outcomes. However, introducing draw predictions is not as crucial for the overall performance of the model and can even be detrimental to the system's accuracy.

Additionally, it was found that adjusting the developed system for each competition individually yields better results than when creating a model using multiple competitions simultaneously. For every methodology, namely Elo, TrueSkill and Neural Network (MLP), the overall accuracy obtained from creating multiple models that were tuned for a given competition was higher than when tuning a single model that maximises the overall accuracy.

For example, when comparing a single Elo model that maximises the overall accuracy (called Average Elo) against multiple Elo models that were tuned to individual competitions (called Competition-based Elo), the latter had 0.74% better performance overall. Moreover, Competition-based Elo consistently had better accuracy than the Average Elo counterpart. Next, for TrueSkill, a similar trend was observed. For every examined competition, Competition-based TrueSkill outperformed Average TrueSkill with an average improvement of accuracy of 0.38% across all seasons.

Lastly, for 3 out of 4 competitions, Competition-based Neural Network produced significantly higher accuracy. The only competition where Competition-based Neural Network performed worse was Italy Serie A, with a 0.67% difference in performance. Average Neural Network on average achieved 1.04% less accuracy, which is not insignificant.

The fact that better performance can be achieved by tuning individual models for different competitions can be explained by a multitude of factors. For example, one possible reason why the same algorithm may require different hyperparameters for optimal results for the given competition is because even within the same sport, competitions are significantly different in their nature. For example, the distribution of wins, draws and loses varies greatly between competitions. This is also reflected in the fact that the same static algorithmic approach yields dramatically different accuracies for competitions. As such, most approaches, as was described in Table 17, will perform significantly better with England Premiership (generally, somewhere around 59%) in comparison to Germany Bundesliga, where most methodologies achieve about 10% lower accuracy. This idea contradicts what Sullivan C. and Cronin C. (2015) described when applying their Elo-based model to

competitions other than the one it was trained on. In their work, Sullivan C. and Cronin C. attributed the difference in accuracy to overfitting. However, it seems the case that it was rather caused by the fact that there are innate differences in competitions and, thus, in the top accuracies that can be achieved, as evidenced both by the bookie and presented models' performances.

5. Ethics

The ethics of gambling have been an ongoing debate for more than one decade. Various perspectives can be considered in this question and many critiques of gambling have been posed. Firstly, gambling in a broad sense is defined. It is a process of deciding the owner of an asset with the usage of chance. Generally, gambling activities have risk associated with them. However, there is often some amount of skill associated with them, such as having knowledge about horse racing or calculating probabilities in card games (Hobson, 1905). Early opposers of such activities (including sports betting) mainly based their argument around the claim that gambling is immortal and does the society wrong, some even going as far as branding it a sin (Bernhard et al, 2010). But seeing the potential financial benefits, governments of the world turned to legalising gambling to obtain an additional source of income. For example, the UK, Australia, Canada, New Zealand and several other countries have adopted gambling in the 1990s to increase the tax revenue. Thus, introducing gambling has had positive effects on the economy. However, a portion of the profits then need to be spent on research about problem gambling, but the financial benefits to some areas are undeniable (Hancock et al., 2017). Indeed, some cities rely heavily on the tourists that are generated by gambling attractions, such as Las Vegas and Macao (Chabra D. et al., 2015).

According to GDBC (2018), the revenue from the gambling sector in the US accounted for a total of $435 billion in 2017, a 4% growth from the previous year. There are other positive factors associated with gambling. It is also important to consider the fact that legalising the casinos and online betting reduces the need for the gambling black market (Chabra D. et al., 2015).

However, many studies suggest that gambling can be harmful. Without any safeguards in place, gambling can quickly get out of control and result in addiction. For example, it is estimated that in electronic gaming machines, VLT, slots and similar types of gambling, problem gamblers account for somewhere between 30% to 50% of net losses, which is a disproportionately large amount (Hancock et al., 2017). Cameron (2007) argues that identifying harmful gambling can be more difficult than identifying other addictions, such as harmful drinking, because of social awareness. In his article, he refers to the fact that there are many well-known signs about the negative effects of drinking and the recommended limits. This is not the case for gambling. Yet, one critique of this opinion could be that the measures that can help combat the issue of gambling addiction are young by historical standards. This means that they often do not have any evidence that would support their efficiency or even that the specific policies designed to prevent gambling addictions will have the desired effects (Blaszczynski et al., 2004).

Studies about the effects of gambling have been surrounded by controversy. As such, Chapman et al. (2018) suggests that rather than being the cause of the problems, addiction to gambling is often a symptom of an individual’s existing disorder. Another researcher, Choliz M (2018), argues that governments should promote the concepts of responsible gambling, demonstrating acceptable levels of gambling that would be appropriate for entertainment but would not result in the financial destruction of the individuals. So while legalizing gambling does appear to improve the overall financial situation through taxes and tourism, the social implications of it are not known to their full extent. One potential

conclusion from this is that caution should be exercised whenever gambling is allowed due to inconclusiveness of the studies regarding its effects on the society. However, this by no means suggests that the governments should avoid it completely, given that extra funds through taxation are used for good causes (Shani et al, 2015). Instead, one approach would be to attempt to follow some of the more reasonable guidelines that were developed by researchers regarding responsible gambling policies, which include measures, such as monitoring of false advertisement and informing gamblers about the potential dangers that they may be susceptible to (Blaszczynski et al, 2004).

All in all, gambling has some positive effects on the economy which can propagate from large scales down to the level of local communities. However, based on several researches, gambling clearly can go too far and potentially cause certain levels of harm to individuals. As such, it is up to the governments to enforce limits through relevant laws that would regulate gambling accordingly and minimise the negative effects of it while maintaining the positive ones.

6. Conclusion

All in all, the existing approaches to betting in football, specifically, using the Poisson model, have been researched and findings summarised. Then, literature review for Elo, TrueSkill and Neural Networks in football and other sports has been described. This included standard implementations of the models, such as using Elo in chess games. Moreover, work produced by researchers using data for different sports was explored. Through this research, extensions of the original models were identified. This included different implementations, such as incorporating home advantage, draw gap, momentum, decay and others.
Also, several models were created for the selected non-standard approaches in football betting, as motivated by the existing research. All of the models predict the outcome of the match, that is, Win, Draw or Lose. The results of Elo, TrueSkill and Neural Networks have been compared against each other as well as against Bet365, which was used as an example of a bookie. Moreover, a practical approach to significantly improve the consistency of non-deterministic models was proposed and implemented. Lastly, the results were critically evaluated and modifications for further research have been proposed.

Overall, both Elo and Neural Network models performed better than Bet365 using overall accuracy and F1 for Win/Lose prediction. No single model had better accuracy in every competition than Bet365. For example, Neural Network performed better than the bookie in England Premiership, France Championnat and Germany Bundesliga, but underperformed in Italy Serie A. On the other hand, Elo only had higher accuracy than Bet365 in England Premiership and Germany Bundesliga. TrueSkill was consistently worse than other systems.

Another important discovery was that each model, whether it was Elo, TrueSkill or Neural Network, achieved higher accuracy when being tuned to each competition specifically, rather than using data from a variety of tournaments across different leagues as part of the training data. This appears to be caused by the fact that there are innate differences between competitions even within the same sport as even the same bookie will have varying prediction performance for different competitions.

Several other implementations of the proposed models can be explored to potentially improve the performance. While TrueSkill has been explored in this paper, TrueSkill 2 has the potential to improve the presented system by considering more factors about each team. Additionally, developing a model where Elo and TrueSkill ratings are assigned to each player rather than each team may have a positive impact on the overall prediction accuracy.

Another interesting extension of the proposed models would be to utilise an ensemble approach that would effectively combine various systems to produce better results. As evidenced by the difference in F1, precision and recall metrics, certain models do better with wins, while others are better at predicting loses. This means that there is a potential for combining multiple models to produce a system that would be able to produce even better predictions overall.

7. References

Aldous D. (2017) Elo Ratings and the Sports Model: A Neglected Topic in Applied Probability? [Accessed 11 November 2019: https://www.stat.berkeley.edu/~aldous/Papers/me-Elo-SS.pdf]

Australia Sports Betting (nd) “Historical NBL Results and Odds Data” [Accessed 28 May 2020: http://www.aussportsbetting.com/data/historical-nbl-results-and-odds-data/]

Australia Sports Betting (nd) “Historical NFL Results and Odds Data” [Accessed 28 May 2020: http://www.aussportsbetting.com/data/historical-nfl-results-and-odds-data/]

Australia Sports Betting (nd) “Historical Super Rugby Results and Odds Data” [Accessed 28 May 2020: http://www.aussportsbetting.com/data/historical-super-rugby-results-and-odds-data/]

Australia Sports Betting (nd) “Historical Twenty20 Big Bash Results and Odds Data” [Accessed 28 May 2020: http://www.aussportsbetting.com/data/historical-twenty20-big-bash-results-and-odds-data/]

Azhari H. R., Widyaningsih Y. and Lestari D. (2018) Predicting Final Result of Football Match Using Poisson Regression Model. J. Phys.: Conf. Ser.1108 012066 [Accessed 16 December 2019: https://iopscience.iop.org/article/10.1088/1742-6596/1108/1/012066/pdf]

Baio G. and Blangiardo M. (2010). Bayesian hierarchical model for the prediction of football results. Journal of Applied Statistics. 37. 253-264. 10.1080/02664760802684177. [Accessed 26 October 2019: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.182.8659&rep=rep1&type=pdf]

Balduzzi D., Tuyls K., Perolat J. and Graepel T. (2018) Re-evaluating Evaluation. [Accessed 10 December 2019: https://arxiv.org/pdf/1806.02643.pdf]

Bernhard B. J., Futrell J. and Harper A. (2010) "Shots from the Pulpit:" An Ethnographic Content Analysis of United States Anti-Gambling Social Movement Documents. UNLV Gaming Research & Review Journal, Volume 14, issue 2, pp. 15-32.

Besterand D. W. and Maltitz M. J. (2013) ‘Introducing Momentum to the Elo rating System’. University of the Free State. [Accessed 14 December 2019: https://www.ufs.ac.za/docs/librariesprovider22/mathematical-statistics-and-actuarial-science-documents/technical-reports-documents/teg418-2069-eng.pdf?sfvrsn=243cf921_0]

Bet365 (2020) “Terms and Conditions”. [Accessed 28 May 2020: https://help.bet365.com/terms-and-conditions]

Blaszczynski A., Ladouceur R. & Shaffer H. (2004). A Science-Based Framework for Responsible Gambling: The Reno Model. Journal of gambling studies / co-sponsored by the National Council on Problem Gambling and Institute for the Study of Gambling and Commercial Gaming. 20. 301-17. 10.1023/B:JOGS.0000040281.49444.e2.

Boldrin B. (2017) Predicting The Result Of English Premier League Soccer Games With The Use Of Poisson Models. Stetson University. [Accessed 18 February 2020: https://www2.stetson.edu/~efriedma/research/boldrin.pdf]

Boshnakov G., Kharrat T. & McHale I. G. (2016) A Bivariate Weibull Count Model for Forecasting Association Football Scores. School of Mathematics, University of Manchester, UK and Centre for Sports Business, Salford Business School, University of Salford, UK.

Bunker R. P. and Thabtah F. (2019) A machine learning framework for sport result prediction. Volume 15, Issue 1, Pages 27-33, ISSN 2210-8327 (http://www.sciencedirect.com/science/article/pii/S2210832717301485)

Buursma D. (2011) Predicting sports events from past results Towards effective betting on football matches [Accessed 16 November 2019: https://www.semanticscholar.org/paper/Predicting-sports-events-from-past-results-Towards-Buursma/5e22c4362df3b0accbe04517c41848a2b229efd1#paper-header]

Cameron J. (2007) Problem gamblers and the duty of care: A response to Sasso and Kalajdzic. Gaming Law Review. 11(5): 2007; 554–571.

Chóliz M. (2018). Ethical Gambling: A Necessary New Point of View of Gambling in Public Health Policies. Frontiers in public health, 6, 12. doi:10.3389/fpubh.2018.00012 https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5797763/

Compton R. (2014) Elo Outside of the Competitive Gaming Realm. [Accessed 21 November 2019: https://users.soe.ucsc.edu/~rcompton/Papers/Essays/ELO%20experiments.pdf]

Constantinou A., Fenton N. and Neil M. (2012) Pi-football: A Bayesian network model for forecasting Association Football match outcomes. Knowledge-Based Systems, 36, 322-339. Knowledge-Based Systems. 36. 332-339. 10.1016/j.knosys.2012.07.008. [Accessed 2 December 2019: https://www.researchgate.net/publication/236944355_pi-football_A_Bayesian_network_model_for_forecasting_Association_Football_match_outcomes_Knowledge-Based_Systems_36_322-339]

Ding X., Huang C., Huang Y., Lin H., Paisley J. and Zeng W. (2019) Learning Rate Dropout. Xiamen University, China. Columbia University, USA.

Dixon M. and Coles S. (1997). Modelling Association Football Scores and Inefficiencies in the Football Betting Market. Journal of the Royal Statistical Society. Series C (Applied Statistics), 46(2), 265-280.

Football-data (nd) “Historical Football Results and Betting Odds Data” [Accessed 28 May 2020: https://www.football-data.co.uk/data.php]

Gambling Commission (2019) Industry statistics. [Accessed 02 February 2020: https://www.gamblingcommission.gov.uk/news-action-and-statistics/Statistics-and-research/Statistics/Industry-statistics.aspx]

GBDC (2019) Global Gambling Report 2019. [Accessed 17 December 2019 https://www.gbgc.com/2018/05/16/global-gambling-report-2018/]

Goddard J. (2006) Who wins the football? Significance. 3. 16 - 19. 10.1111/j.1740-9713.2006.00145.x. [Accessed 15 November 2019: https://www.researchgate.net/publication/229445055_Who_wins_the_football]

Grubbs J. B., Chapman H., Milner L., Gutierrez I. A. and Bradley D. F. (2018) Examining links between posttraumatic stress and gambling motives: The role of positive gambling expectancies. Psychology of Addictive Behaviors, 32(7), 821–831. https://doi.org/10.1037/adb0000399

Hancock L., Schellinck T. and Schrans T. (2008) Gambling and corporate social responsibility (CSR): Re-defining industry and state roles on duty of care, host responsibility and risk management, Policy and Society, 27:1, 55-68, DOI: 10.1016/j.polsoc.2008.07.005

Herbrich R., Minka T. and Graepel T. (2007) TrueSkill(TM): A Bayesian Skill Rating System.

Heungsub L. (2012) TrueSkill. [Accessed 16 October 2019: https://trueskill.org/]

Hobson J. A. (1905) "The Ethics of Gambling". International Journal of Ethics Vol. 15, No. 2, pp. 135-148

Huang K. and Chen K. (2011) Multilayer Perceptron for Prediction of 2006 World Cup Football Game. 10.1155/2011/374816. [Accessed 16 December 2019: https://www.researchgate.net/publication/258379249_Multilayer_Perceptron_for_Prediction_of_2006_World_Cup_Football_Game]

Hvattum L. & Arntzen H. (2010) Using ELO ratings for match result prediction in association football [Accessed 10 November 2019: http://www.collective-behavior.com/publ/ELO.pdf]

Ignatin G. (1984) "Sports Betting." The Annals of the American Academy of Political and Social Science 474

Karlis D. and Ntzoufras I. (1999). Statistical Modelling For Soccer Games: The Greek League. [Accessed 18 February 2020: https://www.researchgate.net/publication/2657570_Statistical_Modelling_For_Soccer_Games_The_Greek_League]

Karlis D. and Ntzoufras I. (2003) Analysis of sports data by using bivariate Poisson models

Koning R. H. and Albert J. (2007) Statistical Thinking in Sports. Boca Raton, Fla: Chapman & Hall/CRC, 2007. Print.

Langholf F. (2018) "The Self-Justifying Elo Rating System." arXiv.org, n. pag. Web [Accessed 19 November 2019: https://search.proquest.com/docview/2071248670?rfr_id=info%3Axri%2Fsid%3Aprimo]

Langseth H. (2013) 'Beating the bookie: A look at statistical models for prediction of football matches'. Frontiers in Artificial Intelligence and Applications. 257. 165-174. 10.3233/978-1-61499-330-8-165. [Accessed 20 October 2019: https://www.researchgate.net/publication/279122210_Beating_the_bookie_A_look_at_statistical_models_for_prediction_of_football_matches]

Lasek J., Szlavik Z. and Bhulai S. (2009) The predictive power of ranking systems in association football. [Accessed 2 December 2019: https://www.few.vu.nl/~zszlavik/papers/IJAPR.pdf]

Lehmann R. and Wohlrabe K. (2017) Who is the 'Journal Grand Master'? A new ranking based on the Elo rating system. [Accessed 4 November 2019: https://mpra.ub.uni-muenchen.de/77363/1/MPRA_paper_77363.pdf]

Leighton V., Chumping L. and Gerrard H. (2019) 'How well do Elo-based Ratings Predict Professional Tennis Matches?' No. 2019/3. ISSN 1478-9396

Machine Learning Group (nd) "Data Mining. Practical Machine Learning Tools and Techniques". University of Waikato. [Accessed 28 May 2020: https://www.cs.waikato.ac.nz/ml/weka/book.html]

Maher M. J. (1982) 'Modelling association football scores'. Statistica Neerlandica, 36: 109-118. doi:10.1111/j.1467-9574.1982.tb00782.x. [Accessed 14 December 2019: http://www.90minut.pl/misc/maher.pdf]

Mallios W. S. (2010) Forecasting in Financial and Sports Gambling Markets Adaptive Drift Modeling. Hoboken, N.J: Wiley, 2011. Print.

Mccabe A. and Trevathan J. (2008) Artificial Intelligence in Sports Prediction. 1194-1197. 10.1109/ITNG.2008.203. [Accessed 23 November 2019 https://www.researchgate.net/publication/220841301_Artificial_Intelligence_in_Sports_Prediction]

Miljković D., GajićL., Kovačević A. and Konjović Z. (2010) "The use of data mining for basketball matches outcomes prediction," IEEE 8th International Symposium on Intelligent Systems and Informatics, Subotica, pp. 309-312. [Accessed 25 November 2019 https://ieeexplore.ieee.org/document/5647440]

Minka T., Cleven R., and Zaykov Y. (2018) TrueSkill 2: An improved Bayesian skill rating system

Moser J. (2010) The Math Behind TrueSkill. [Accessed 20 October 2019: https://www.moserware.com/assets/computing-your-skill/The%20Math%20Behind%20TrueSkill.pdf]

Moya F. (2012) Statistical Methodology for Profitable Sports Gambling. [Accessed 27 October 2019: https://www.stat.sfu.ca/content/dam/sfu/stat/alumnitheses/2012/FabianMoyaFinalVersion.pdf]

Odachowski K. and Grekow J. (2012) Using Bookmaker Odds to Predict the Final Result of Football Matches. 7828. 196-205. 10.1007/978-3-642-37343-5_20. [Accessed 11 November 2019: https://www.researchgate.net/publication/262395354_Using_Bookmaker_Odds_to_Predict_the_Final_Result_of_Football_Matches/citation/download]

Pelanek R. (2016) Applications of the Elo Rating System in Adaptive EducationalSystems, Computers & Education , doi: 10.1016/j.compedu.2016.03.017. [Accessed 17 November 2019: https://www.fi.muni.cz/~xpelanek/publications/CAE-elo.pdf]

Pettersson D. and Nyquist R. (2017) “Football Match Prediction using Deep Learning” [Accessed 9 December 2019: https://pdfs.semanticscholar.org/e556/af01e86c3414042aa69831ea5fb398e66f94.pdf]

Pollard R. (2008) Home Advantage in Football: A Current Review of an Unsolved Puzzle. The Open Sports Sciences Journal. 1. 10.2174/1875399X00801010012. [Accessed 22 November 2019: https://www.researchgate.net/publication/228632270_Home_Advantage_in_Football_A_Current_Review_of_an_Unsolved_Puzzle]

Purucker M. C. (1996) Neural network quarterbacking. IEEE Potentials, vol. 15, no. 3, pp. 9-15. [Accessed 4 December 2019: https://ieeexplore.ieee.org/document/535226]

Quispe L. C. and Luna J. E. O. (2015) “A Content-Based Recommendation System Using TrueSkill.” Fourteenth Mexican International Conference on Artificial Intelligence (MICAI). IEEE, 2015. 203–207. Web. [Accessed 2 December 2019: https://ieeexplore.ieee.org/document/7429436]

Raschka (nd) Machine Learning FAQ. [Accessed 19 November 2019: https://sebastianraschka.com/faq/docs/evaluate-a-model.html]

Shani A., Fong L., Leung D., Law R., Gavriel-Fried B. and Chhabra D. (2015) Ethics of Gambling? Tourism Recreation Research. 39. 453-486. 10.1080/02508281.2014.11087011.

Sheehan D. (2017) Predicting Football Results With Statistical Modelling [https://dashee87.github.io/football/python/predicting-football-results-with-statistical-modelling/ Accessed 5 November 2019]

Skiena S. (2001) Calculated Bets Computers, Gambling, and Mathematical Modeling to Win. New York: Cambridge University Press, 2001. Print.

Smarkets (nd) How to calculate Poisson distribution for football betting [https://help.smarkets.com/hc/en-gb/articles/115001457989-How-to-calculate-Poisson-distribution-for-football-betting Accessed 5 November 2019]

Statista Research Department (2018) Revenue of selected sports betting companies in 2015. [Accessed 02 February 2020: https://www.statista.com/statistics/270757/revenue-sports-betting-companies/]

Sullivan C. and Cronin C. (2015) Improving Elo Rankings For Sports Experimenting on the English Premier League. [Accessed 3 November 2019: https://pdfs.semanticscholar.org/e28c/c07fc8153f745725133afc0f9c2ab7634a08.pdf]

Tax N. and Joustra Y. (2015) Predicting The Dutch Football Competition Using Public Data: A Machine Learning Approach. 10.13140/RG.2.1.1383.4729. [Accessed 13 December 2019

https://www.researchgate.net/publication/282026611_Predicting_The_Dutch_Football_Competition_Using_Public_Data_A_Machine_Learning_Approach]

Vecek N., Crepinsek M., Mernik M. and Hrncic D. (2014) A Comparison between Different Chess RatingSystems for Ranking Evolutionary Algorithms. [Accessed 19 November 2019: https://annals-csis.org/Volume_2/pliks/33.pdf]

Weisstein E. W. (2000) "Total Probability Theorem." [Accessed 18 February 2020: http://mathworld.wolfram.com/TotalProbabilityTheorem.html]

Williams L.V., Liu C., and Gerrard H. (2019) 'How well do Elo-based ratings predict professional tennis matches?' [Accessed 4 December 2019: https://www.ntu.ac.uk/__data/assets/pdf_file/0024/830562/how-well-do-elo-based-ratings-predict-professional-tennis-matches.pdf]

World Football Elo Ratings [Accessed 09 March 2020: https://www.eloratings.net/]

Appendix A: Using Poisson distribution for football predictions

To calculate the average number of goals for home and away, a simple formula can be used:

$$\underline{G}(H) = \frac{\sum_{i=1}^{N} h(i)}{N} \quad \textbf{(A1)}$$

where $\underline{G}(H)$ is the average number of goals home, $h(i)$ is the number of goals scored by home team in match *i*, $\sum_{i=1}^{N} h(i)$ is the total number of goals scored by the home team and $N$is the total number of games. For average number of goals away:

$$\underline{G}(A) = \frac{\sum_{i=1}^{N} a(i)}{N} \quad \textbf{(A2)}$$

In Formula A2, the $\underline{G}(A)$stands for the average number of goals the away team scores while playing away, $a(i)$ is the number of goals scored by home team in match *i*,$\sum_{i=1}^{N} a(i)$ is the total number of goals scored by the away team and N is the total number of games. Next, the average number of goals conceded by home and away teams is calculated as follows:

$$\underline{C}(H) = \frac{\sum_{i=1}^{N} a(i)}{N} \quad \textbf{(A3)}$$

$$\underline{C}(A) = \frac{\sum_{i=1}^{N} h(i)}{N} \quad \textbf{(A4)}$$

Similarly to Formulas A1 and A2, in Formula A3, $\underline{C}(H)$ is the average number of goals conceded by the home team, $a(i)$ is the number of goals conceded by home team (which is the same as the number of goals that away team scored) in match *i*,$\sum_{i=1}^{N} a(i)$ is the total number of goals that the away team typically scores against the home team and N is the total number of games. In Formula A4, $\underline{C}(A)$ stands for the average number of goals that the away team concedes with $\sum_{i=1}^{N} h(i)$ representing the total number of goals that the home team scores against the away team (when the away team plays away) and N is the total number of games.

Essentially, the home team's average goals per game can be calculated by dividing the total number of goals scored by the home team last season (while being the home team in the past matches) by the total number of home games played. The home team's strength can then be calculated by dividing the resulting value with the average number of goals home:

$$O(H) = \frac{\underline{G}(H)}{\underline{g}(h)} \quad \textbf{(A5)}$$

Where $O(H)$is the home team's offence strength, $\underline{G}(H)$ is the average number of goals scored by the home team and $\underline{g}(h)$ is the average number of goals scored by all home teams (usually, calculated for the past season). To get the away team's strength, a similar formula is used:

$$O(A) = \frac{\underline{G}(A)}{\underline{g}(a)} \quad \textbf{(A6)}$$

In Formula A6, the away team's strength ($O(A)$) is calculated by using $\underline{A}$(average number of goals scored by a particular away team) divided by the average number of goals typically scored by an away team ($\underline{g}(a)$) in that particular competition (often, as average of the previous season).

For calculating the values of defence, the reverse process is done. Formulas 8 and 9 show how to estimate defence strength for home and away teams:

$$D(H) = \frac{\underline{G}(A)}{\underline{g}(a)} \quad \textbf{(A7)}$$

$$D(A) = \frac{\underline{G}(H)}{\underline{g}(h)} \quad \textbf{(A8)}$$

Where $D(H)$ is home team defence, $\underline{G}(A)$ is the average number of goals conceded by home team (typically, in the past season). $\underline{g}(a)$ is the average number of goals conceded by a home team in the tournament. In regards to Formula A7, $D(A)$ is the away team defence, which is calculated by using $\underline{G}(H)$, the away team average number of conceded goals. Similarly to Formula A8, $\underline{g}(h)$ is the average number of goals that away team concedes in a tournament.

With these values, it is possible to calculate the expected number of goals the home team will make by multiplying the home team attack strength by the away team defence strength and then by the average number of home goals. Similarly, the expected number of goals for the away team can be calculated. With the average values computed, the probability of each outcome can be predicted using the Poisson formula:

$$P(k\ goals) = \frac{(\lambda^{k} * e^{-\lambda})}{k!}$$

(A9)

Where $P$is the probability that a team is going to score k number of goals, λ is the mean of the goal distribution (the expected number of goals) and e is a mathematical constant (equal to approximately 2.71828).

Appendix B: TrueSkill Number of Games to initialise

Game Mode	Number of Games per Gamer
16 Players Free-For-All	3
8 Players Free-For-All	3
4 Players Free-For-All	5
2 Players Free-For-All	12
4 Teams/2 Players Per Team	10
4 Teams/4 Players Per Team	20
2 Teams/4 Players Per Team	46
2 Teams/8 Players Per Team	91

Table B1: Number of games required to obtain accurate TrueSkill rating for a gamer in different modes

Appendix C: Football Data

Data publically available on Football-data (nd):
Div = League Division
Date = Match Date (dd/mm/yy)
Time = Time of match kick off
HomeTeam = Home Team
AwayTeam = Away Team
FTHG and HG = Full Time Home Team Goals
FTAG and AG = Full Time Away Team Goals
FTR and Res = Full Time Result (H=Home Win, D=Draw, A=Away Win)
HTHG = Half Time Home Team Goals
HTAG = Half Time Away Team Goals
HTR = Half Time Result (H=Home Win, D=Draw, A=Away Win)

Match Statistics (where available)
Attendance = Crowd Attendance
Referee = Match Referee
HS = Home Team Shots
AS = Away Team Shots
HST = Home Team Shots on Target
AST = Away Team Shots on Target
HHW = Home Team Hit Woodwork
AHW = Away Team Hit Woodwork
HC = Home Team Corners
AC = Away Team Corners
HF = Home Team Fouls Committed
AF = Away Team Fouls Committed
HFKC = Home Team Free Kicks Conceded
AFKC = Away Team Free Kicks Conceded
HO = Home Team Offsides
AO = Away Team Offsides
HY = Home Team Yellow Cards
AY = Away Team Yellow Cards
HR = Home Team Red Cards
AR = Away Team Red Cards
HBP = Home Team Bookings Points (10 = yellow, 25 = red)
ABP = Away Team Bookings Points (10 = yellow, 25 = red)

Note that Free Kicks Conceded includes fouls, offsides and any other offense committed and will always be equal to or higher than the number of fouls. Fouls make up the vast majority of Free Kicks Conceded. Free Kicks Conceded are shown when specific data on Fouls are not available (France 2nd, Belgium 1st and Greece 1st divisions).

Note also that English and Scottish yellow cards do not include the initial yellow card when a second is shown to a player converting it into a red, but this is included as a yellow (plus red) for European games.

Key to 1X2 (match) betting odds data:

B365H = Bet365 home win odds
B365D = Bet365 draw odds
B365A = Bet365 away win odds
BSH = Blue Square home win odds
BSD = Blue Square draw odds
BSA = Blue Square away win odds
BWH = Bet&Win home win odds
BWD = Bet&Win draw odds
BWA = Bet&Win away win odds
GBH = Gamebookers home win odds
GBD = Gamebookers draw odds
GBA = Gamebookers away win odds
IWH = Interwetten home win odds
IWD = Interwetten draw odds
IWA = Interwetten away win odds
LBH = Ladbrokes home win odds
LBD = Ladbrokes draw odds
LBA = Ladbrokes away win odds
PSH and PH = Pinnacle home win odds
PSD and PD = Pinnacle draw odds
PSA and PA = Pinnacle away win odds
SOH = Sporting Odds home win odds
SOD = Sporting Odds draw odds
SOA = Sporting Odds away win odds
SBH = Sportingbet home win odds
SBD = Sportingbet draw odds
SBA = Sportingbet away win odds
SJH = Stan James home win odds
SJD = Stan James draw odds
SJA = Stan James away win odds
SYH = Stanleybet home win odds
SYD = Stanleybet draw odds
SYA = Stanleybet away win odds
VCH = VC Bet home win odds
VCD = VC Bet draw odds
VCA = VC Bet away win odds
WHH = William Hill home win odds
WHD = William Hill draw odds
WHA = William Hill away win odds

Bb1X2 = Number of BetBrain bookmakers used to calculate match odds averages and maximums
BbMxH = Betbrain maximum home win odds
BbAvH = Betbrain average home win odds
BbMxD = Betbrain maximum draw odds

BbAvD = Betbrain average draw win odds
BbMxA = Betbrain maximum away win odds
BbAvA = Betbrain average away win odds

MaxH = Oddsportal maximum home win odds
MaxD = Oddsportal maximum draw win odds
MaxA = Oddsportal maximum away win odds
AvgH = Oddsportal average home win odds
AvgD = Oddsportal average draw win odds
AvgA = Oddsportal average away win odds

Key to total goals betting odds:

BbOU = Number of BetBrain bookmakers used to calculate over/under 2.5 goals (total goals) averages and maximums
BbMx>2.5 = Betbrain maximum over 2.5 goals
BbAv>2.5 = Betbrain average over 2.5 goals
BbMx<2.5 = Betbrain maximum under 2.5 goals
BbAv<2.5 = Betbrain average under 2.5 goals

GB>2.5 = Gamebookers over 2.5 goals
GB<2.5 = Gamebookers under 2.5 goals
B365>2.5 = Bet365 over 2.5 goals
B365<2.5 = Bet365 under 2.5 goals
P>2.5 = Pinnacle over 2.5 goals
P<2.5 = Pinnacle under 2.5 goals
Max>2.5 = Oddsportal maximum over 2.5 goals
Max<2.5 = Oddsportal maximum under 2.5 goals
Avg>2.5 = Oddsportal average over 2.5 goals
Avg<2.5 = Oddsportal average under 2.5 goals

Key to Asian handicap betting odds:

BbAH = Number of BetBrain bookmakers used to Asian handicap averages and maximums
BbAHh = Betbrain size of handicap (home team)
AHh = Oddsportal size of handicap (home team) (since 2019/2020)
BbMxAHH = Betbrain maximum Asian handicap home team odds
BbAvAHH = Betbrain average Asian handicap home team odds
BbMxAHA = Betbrain maximum Asian handicap away team odds
BbAvAHA = Betbrain average Asian handicap away team odds

GBAHH = Gamebookers Asian handicap home team odds
GBAHA = Gamebookers Asian handicap away team odds
GBAH = Gamebookers size of handicap (home team)
LBAHH = Ladbrokes Asian handicap home team odds
LBAHA = Ladbrokes Asian handicap away team odds
LBAH = Ladbrokes size of handicap (home team)
B365AHH = Bet365 Asian handicap home team odds

B365AHA = Bet365 Asian handicap away team odds
B365AH = Bet365 size of handicap (home team)
PAHH = Pinnacle Asian handicap home team odds
PAHA = Pinnacle Asian handicap away team odds
MaxAHH = Oddsportal maximum Asian handicap home team odds
MaxAHA = Oddsportal maximum Asian handicap away team odds
AvgAHH = Oddsportal average Asian handicap home team odds
AvgAHA = Oddsportal average Asian handicap away team odds

Appendix D: Example Match Entry

Div	E0
Date	09/08/2019
Time	20:00
HomeTeam	Liverpool
AwayTeam	Norwich
FTHG	4
FTAG	1
FTR	H
HTHG	4
HTAG	0
HTR	H
Referee	M Oliver
HS	15
AS	12
HST	7
AST	5
HF	9
AF	9
HC	11
AC	2
HY	0
AY	2
HR	0
AR	0

B365H	1.14
B365D	10
B365A	19
BWH	1.14
BWD	8.25
BWA	18.5
IWH	1.15
IWD	8
IWA	18
PSH	1.15
PSD	9.59
PSA	18.05
WHH	1.12
WHD	8.5
WHA	21
VCH	1.14
VCD	9.5
VCA	23
MaxH	1.16
MaxD	10
MaxA	23
AvgH	1.14
AvgD	8.75
AvgA	19.83
B365>2.5	1.4

B365<2.5	3
P>2.5	1.4
P>2.5	3.11
Max>2.5	1.45
Max>2.5	3.11
Avg>2.5	1.41
Avg>2.5	2.92
AHh	-2.25
B365AHH	1.96
B365AHA	1.94
PAHH	1.97
PAHA	1.95
MaxAHH	1.97
MaxAHA	2
AvgAHH	1.94
AvgAHA	1.94
B365CH	1.14
B365CD	9.5
B365CA	21
BWCH	1.14
BWCD	9
BWCA	20
IWCH	1.15
IWCD	8

IWCA	18
PSCH	1.14
PSCD	10.43
PSCA	19.63
WHCH	1.11
WHCD	9.5
WHCA	21
VCCH	1.14
VCCD	9.5
VCCA	23
MaxCH	1.16
MaxCD	10.5
MaxCA	23
AvgCH	1.14
AvgCD	9.52
AvgCA	19.18
B365C>2.5	1.3
B365C<2.5	3.5
PC>2.5	1.34
PC<2.5	3.44
MaxC>2.5	1.36
MaxC>2.5	3.76
AvgC>2.5	1.32
AvgC>2.5	3.43
AHCh	-2.25
B365CAHH	1.91
B365CAHA	1.99

PCAHH	1.94
PCAHA	1.98
MaxCAHH	1.99
MaxCAHA	2.07
AvgCAHH	1.9
AvgCAHA	1.99

Table D1: Football record with sample data that is available for England Premiership, France Championnat, Germany Bundesliga and Italy Serie A

Appendix E: Number of Matches in Competitions

Competition Name	Number of Matches
England Premiership	4180
France Championnat	4180
Germany Bundesliga	3366
Italy Serie A	4180

Table E1: Initial number of games per competition

After the dataset has been examined for quality, any match entries with missing odds and/or goals have been removed. Additionally, if the bookie odds or the goal figures have been incorrectly formatted (e.g. a string instead of a number), these match rows were excluded.

Competition Name	Number of Matches
England Premiership	4180
France Championnat	4179
Germany Bundesliga	3366
Italy Serie A	4177

Table E2: Number of games per competition after data quality check

Appendix F: Bet365 Football Competition Metrics

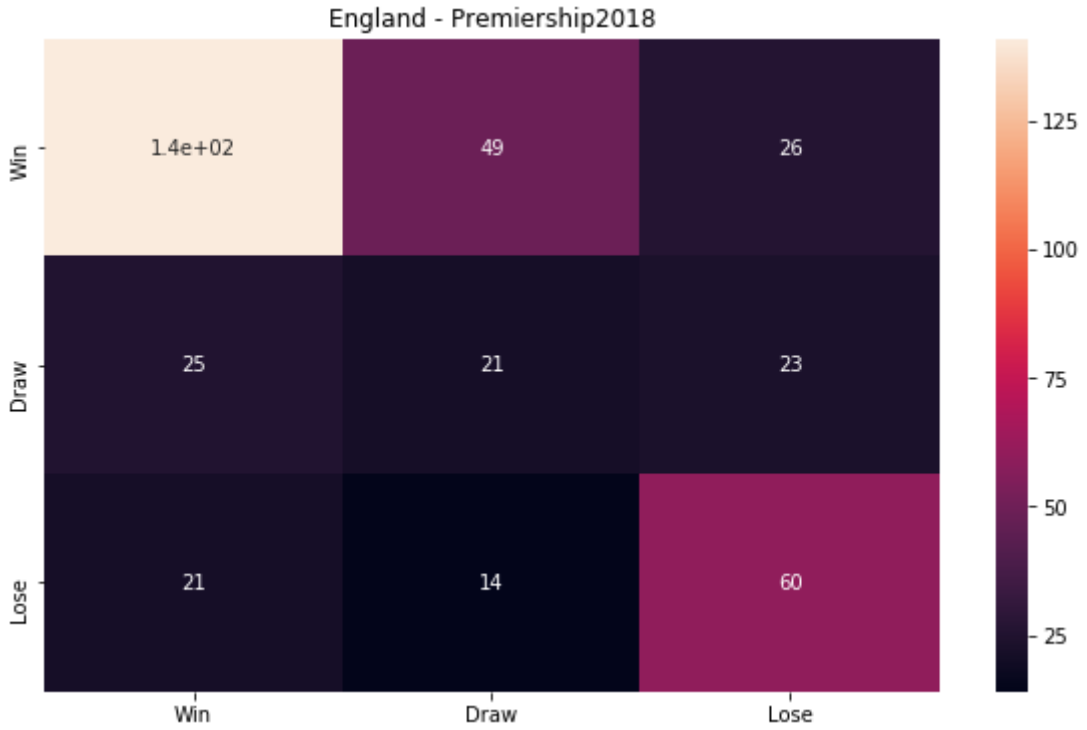

Figure F1: Bet365 confusion matrix for England Premiership 2018-2019

```
Accuracy: 54.74%
MAE: 4.94
RMSE: 5.96
Precision:      Win 0.60        Draw 0.36       Lose 0.57
Recall:         Win 0.74        Draw 0.24       Lose 0.56
F1:     Win 0.66        Draw 0.29       Lose 0.56
```

Figure F2: Bet365 metrics for England Premiership 2018-2019

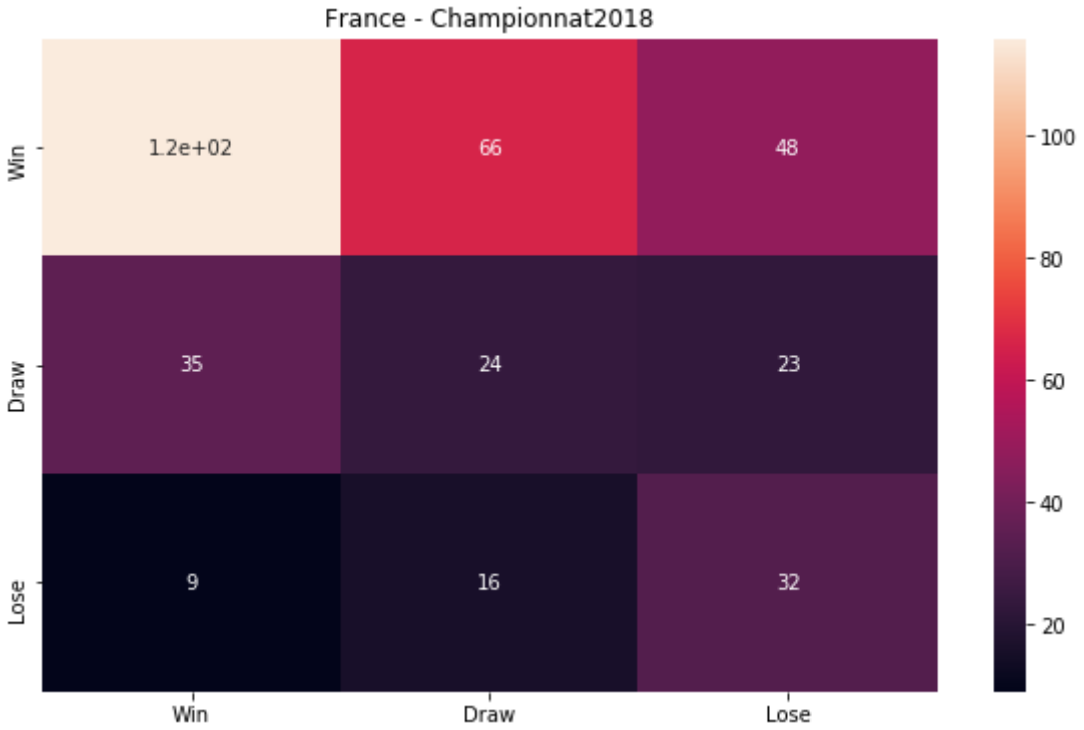

Figure F3: Bet365 confusion matrix for France Championnat 2018-2019

```
Accuracy: 46.61%
MAE: 4.27
RMSE: 5.43
Precision:      Win 0.50        Draw 0.29       Lose 0.56
Recall:         Win 0.72        Draw 0.23       Lose 0.31
F1:     Win 0.59        Draw 0.26       Lose 0.40
```

Figure F4: Bet365 metrics for France Championnat 2018-2019

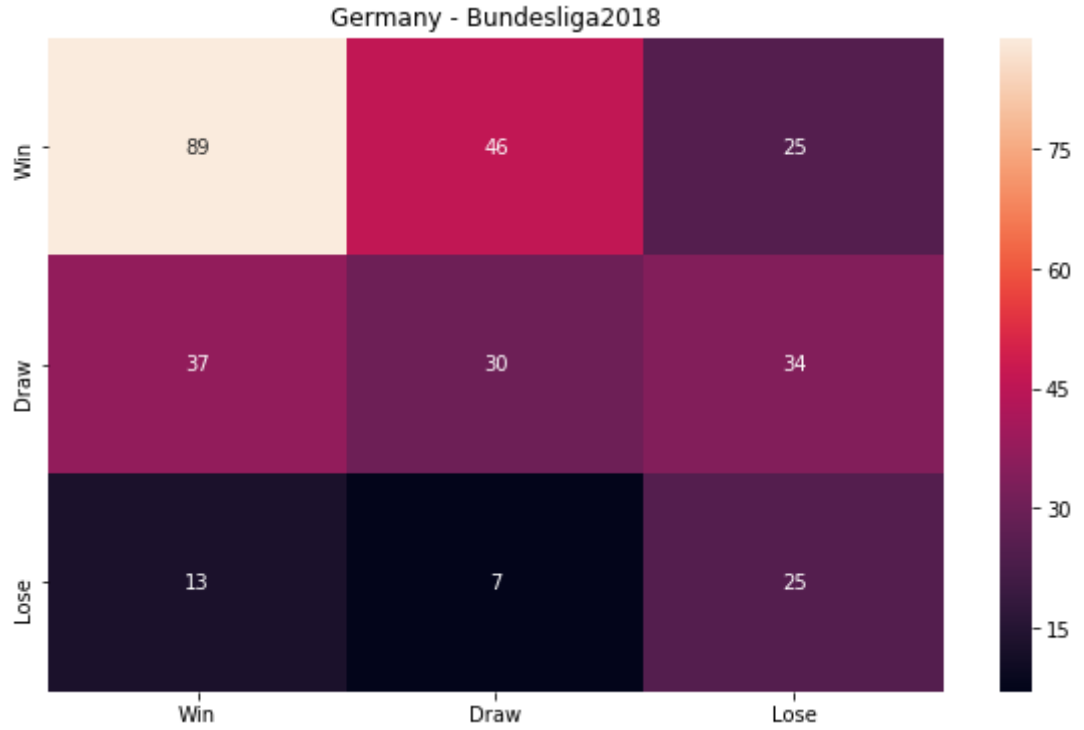

Figure F5: Bet365 confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 47.06%
MAE: 4.35
RMSE: 5.40
Precision:      Win 0.56        Draw 0.30       Lose 0.56
Recall:         Win 0.64        Draw 0.36       Lose 0.30
F1:     Win 0.60        Draw 0.33       Lose 0.39
```

Figure F6: Bet365 metrics for Germany Bundesliga 2018-2019

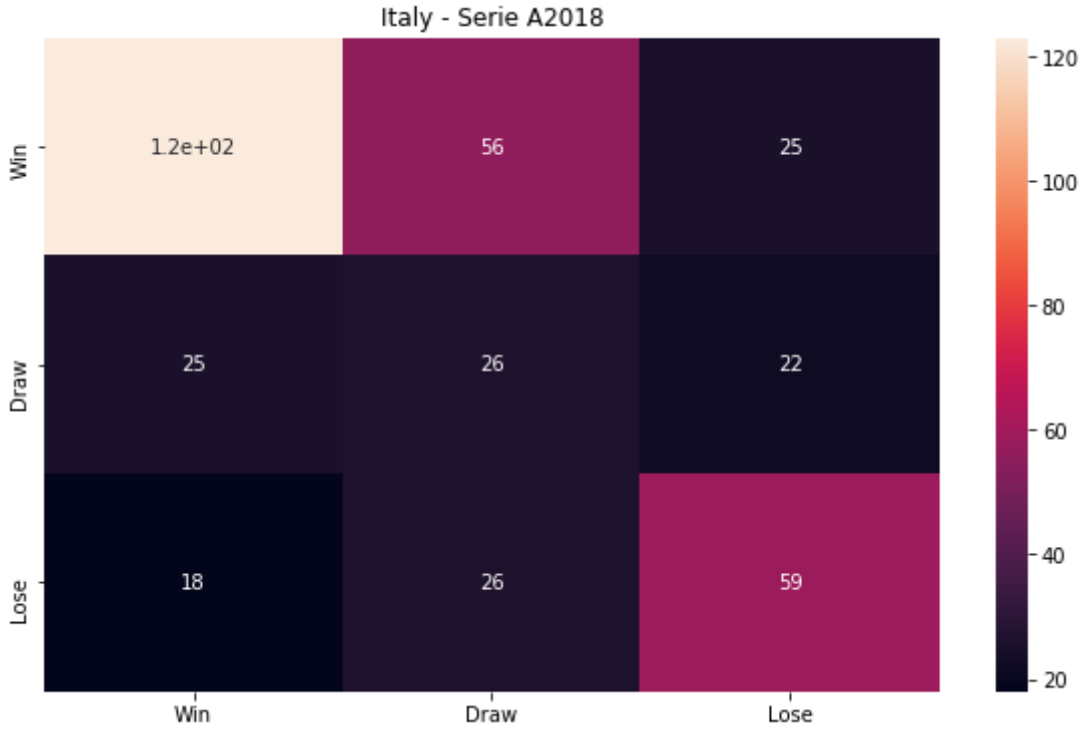

Figure F7: Bet365 confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 54.74%
MAE: 4.94
RMSE: 5.96
Precision:      Win 0.60        Draw 0.36       Lose 0.57
Recall:         Win 0.74        Draw 0.24       Lose 0.56
F1:     Win 0.66        Draw 0.29       Lose 0.56
```

Figure F8: Bet365 metrics for Italy Serie A 2018-2019

Appendix G: Basic Elo

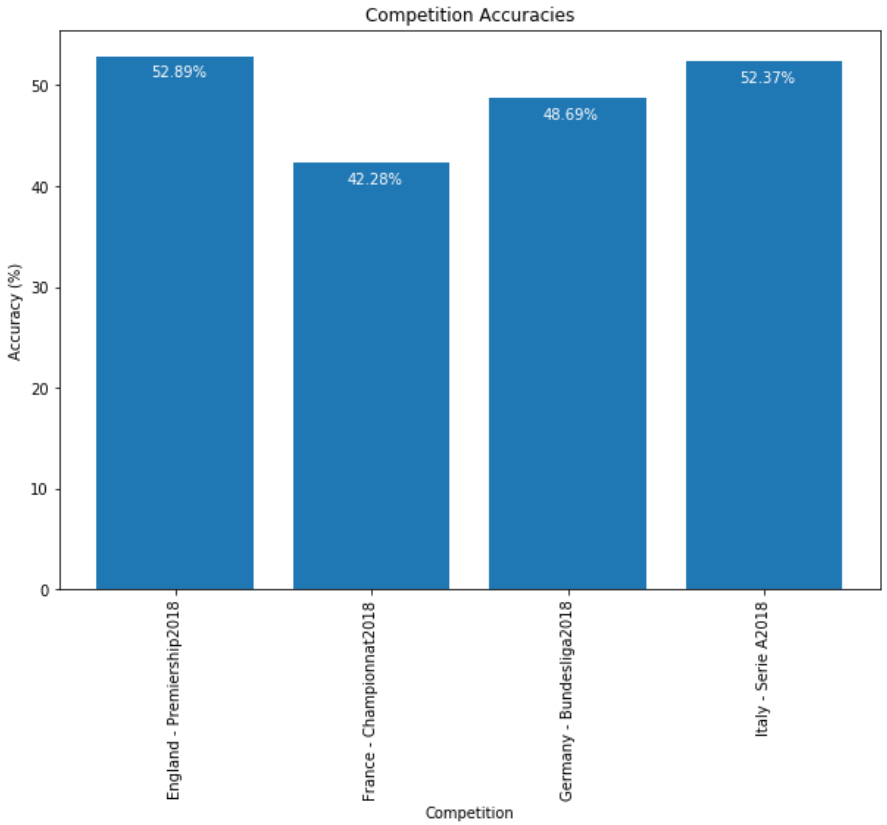

Figure G1: Basic Elo competition overall prediction accuracies

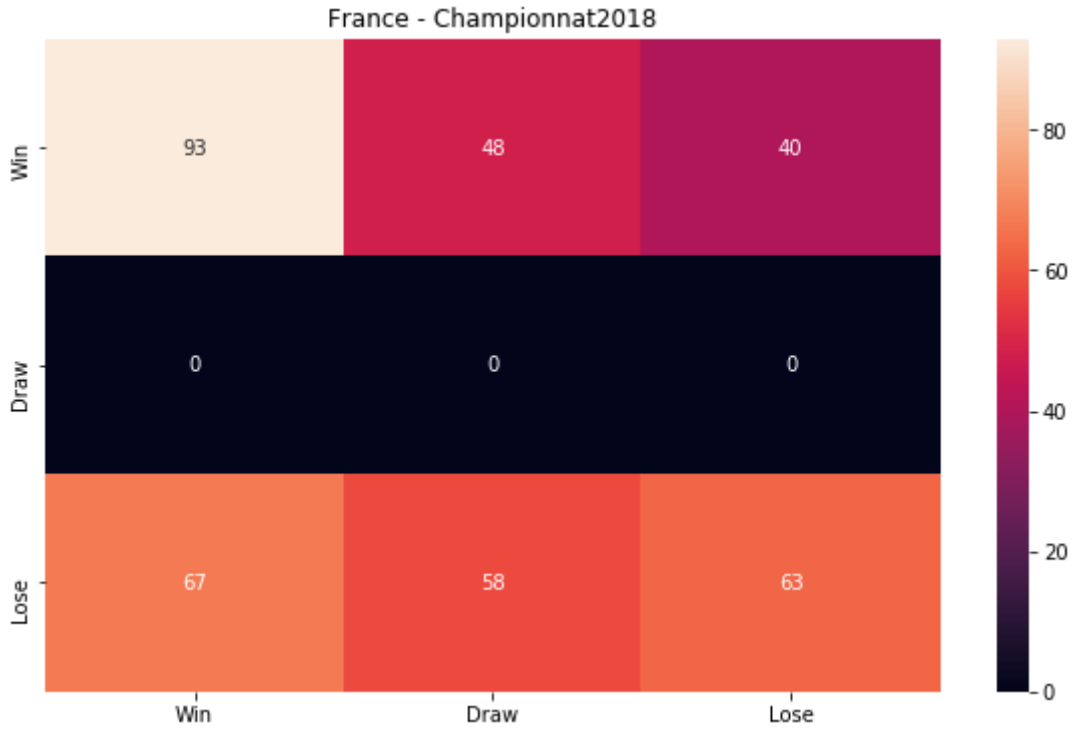

Figure G2: Initial Elo confusion matrix for France Championnat 2018-2019

```
Accuracy: 42.28%
MAE: 0.36
RMSE: 0.37
Precision:      Win 0.51        Draw 0.00       Lose 0.34
Recall:         Win 0.58        Draw 0.00       Lose 0.61
F1:     Win 0.55        Draw 0.00       Lose 0.43
```

Figure G3: Initial Elo metrics for France Championnat 2018-2019

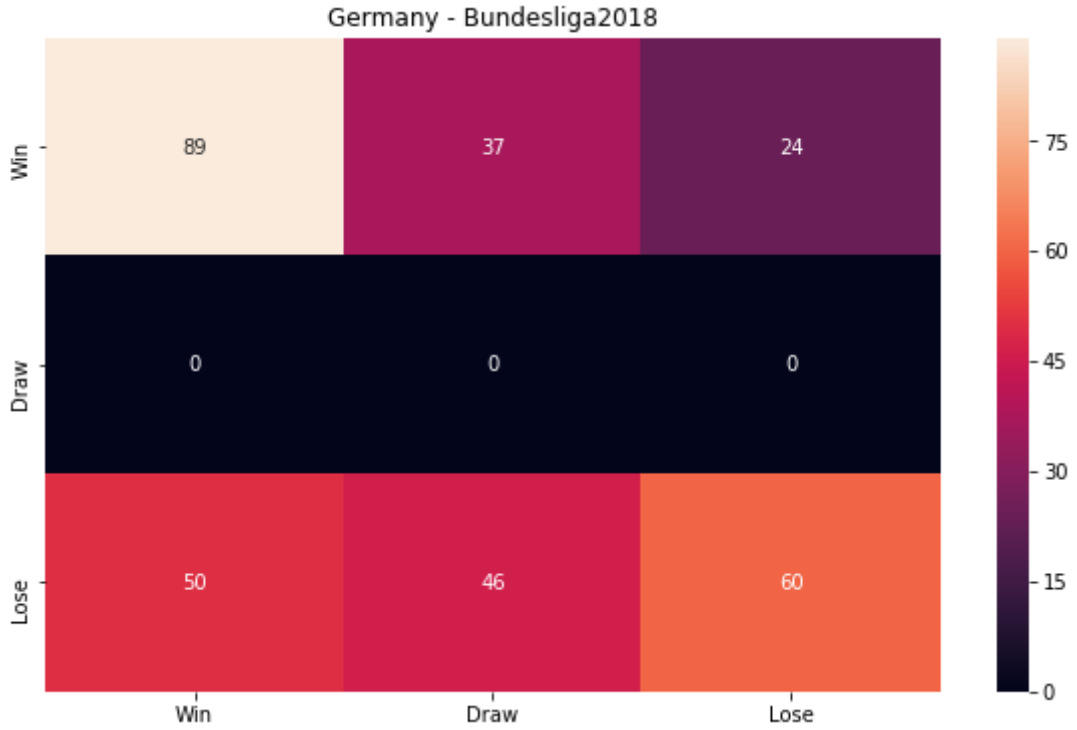

Figure G4: Initial Elo confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 48.69%
MAE: 0.36
RMSE: 0.37
Precision:      Win 0.59        Draw 0.00       Lose 0.38
Recall:         Win 0.64        Draw 0.00       Lose 0.71
F1:     Win 0.62        Draw 0.00       Lose 0.50
```

Figure G5: Initial Elo metrics for Germany Bundesliga 2018-2019

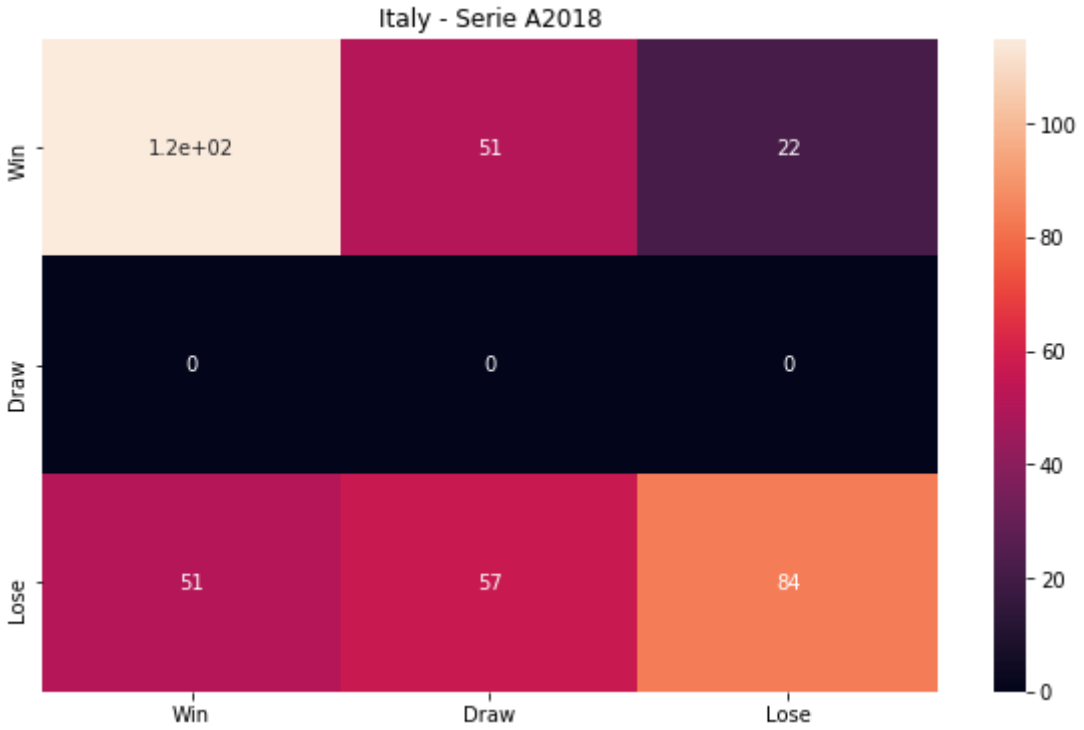

Figure G6: Initial Elo confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 52.37%
MAE: 0.31
RMSE: 0.33
Precision:      Win 0.61        Draw 0.00       Lose 0.44
Recall:         Win 0.69        Draw 0.00       Lose 0.79
F1:     Win 0.65        Draw 0.00       Lose 0.56
```

Figure G7: Initial Elo metrics for Italy Serie A 2018-2019

Appendix H: K-Factor Elo

First, the K-Factor is varied from 1 to 50 using increments of 1. The effects of applying different K-Factors during training are presented in Figure H1. As can be observed in Figure H1, varying the K-Factor does not appear to have a significant effect on the overall competition accuracies.

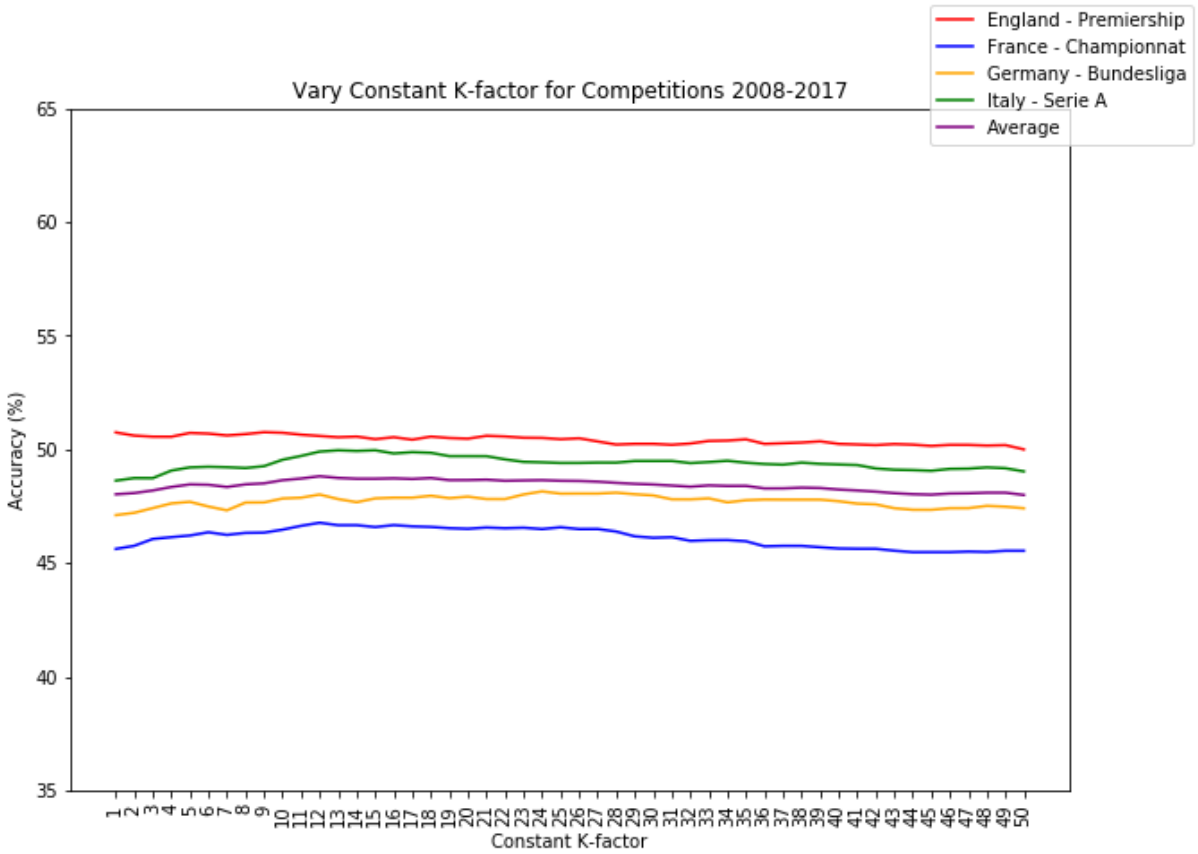

Figure H1: Overall accuracy of train set (y) varying depending on a constant K-Factor (x)

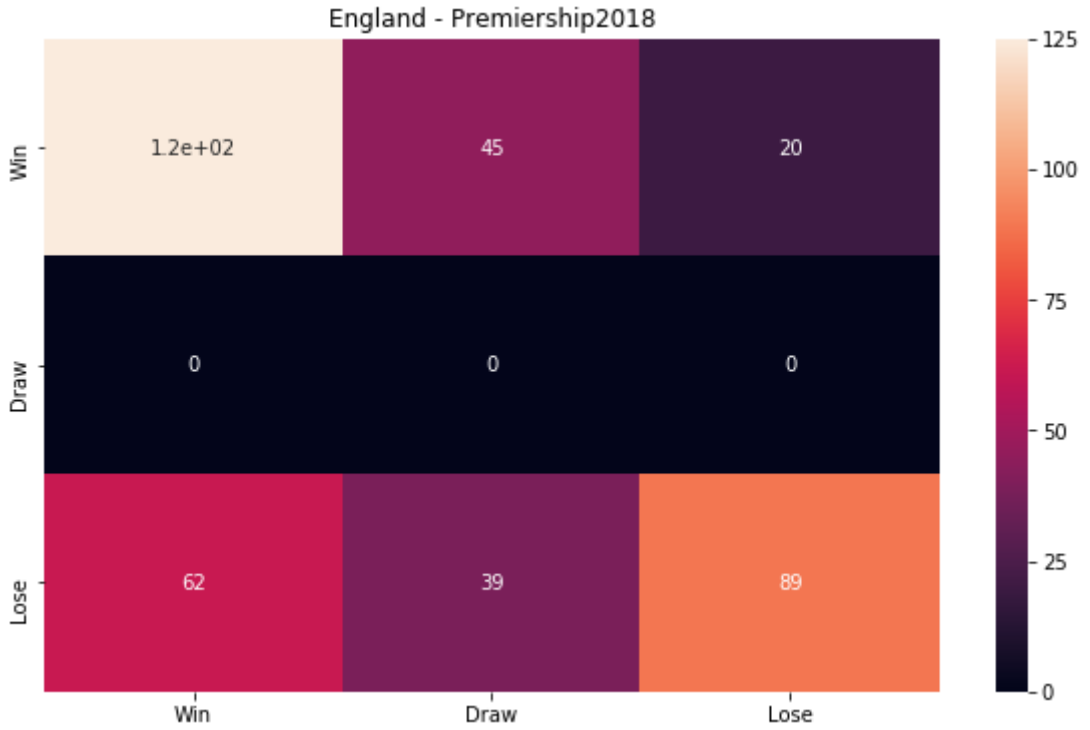

Figure H2: Constant K-Factor Elo confusion matrix for England Premiership 2018-2019

```
Accuracy: 56.32%
MAE: 0.40
RMSE: 0.41
Precision:      Win 0.66        Draw 0.00       Lose 0.47
Recall:         Win 0.67        Draw 0.00       Lose 0.82
F1:     Win 0.66        Draw 0.00       Lose 0.60
```

Figure H3: Constant K-Factor Elo metrics for England Premiership 2018-2019

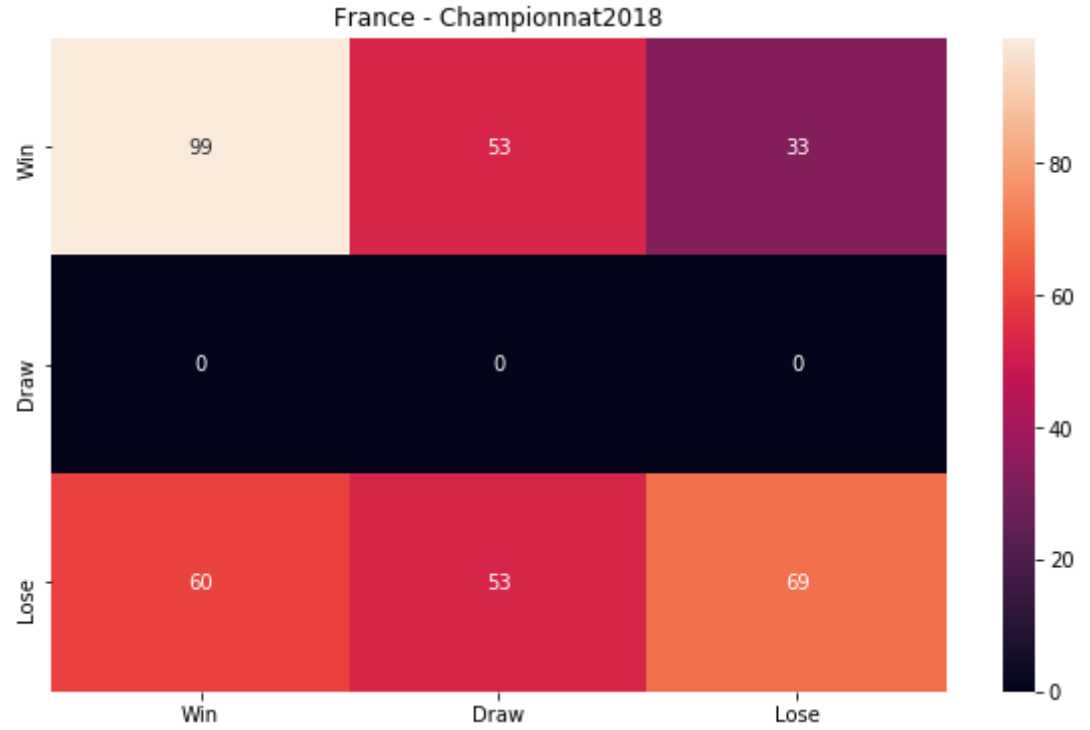

Figure H4: Constant K-Factor Elo confusion matrix for France Championnat 2018-2019

```
Accuracy: 45.78%
MAE: 0.42
RMSE: 0.43
Precision:      Win 0.54        Draw 0.00       Lose 0.38
Recall:         Win 0.62        Draw 0.00       Lose 0.68
F1:     Win 0.58        Draw 0.00       Lose 0.49
```

Figure H5: Constant K-Factor Elo metrics for France Championnat 2018-2019

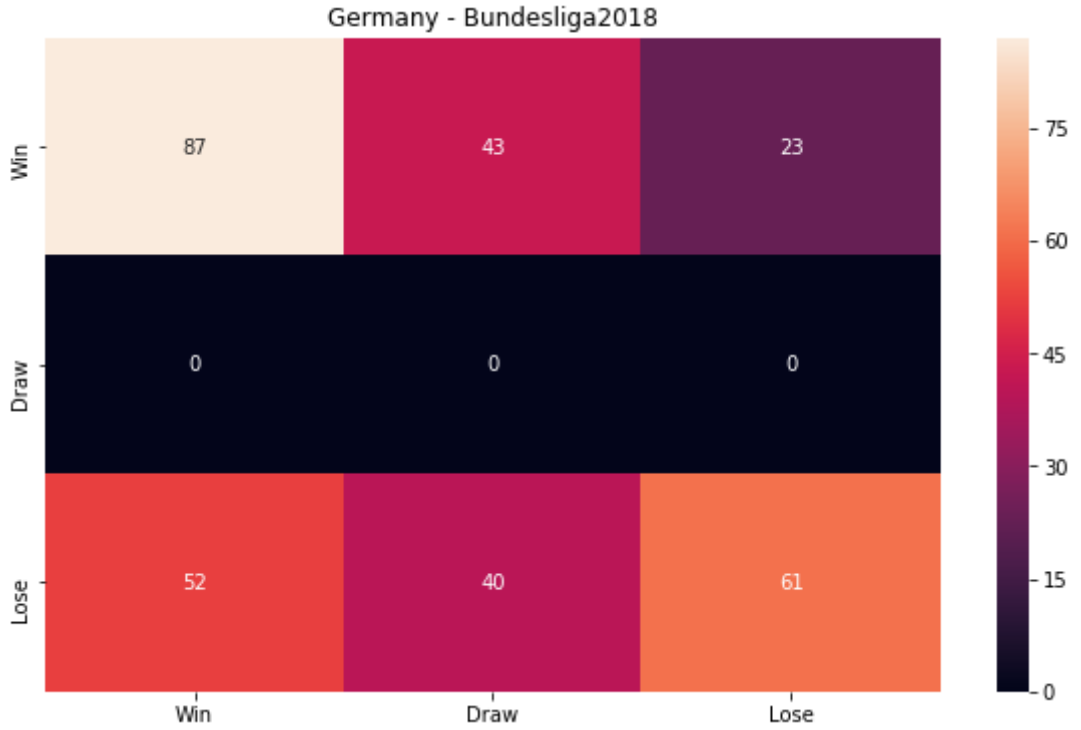

Figure H6: Constant K-Factor Elo confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 48.37%
MAE: 0.42
RMSE: 0.43
Precision:      Win 0.57        Draw 0.00       Lose 0.40
Recall:         Win 0.63        Draw 0.00       Lose 0.73
F1:     Win 0.60        Draw 0.00       Lose 0.51
```

Figure H7: Constant K-Factor Elo metrics for Germany Bundesliga 2018-2019

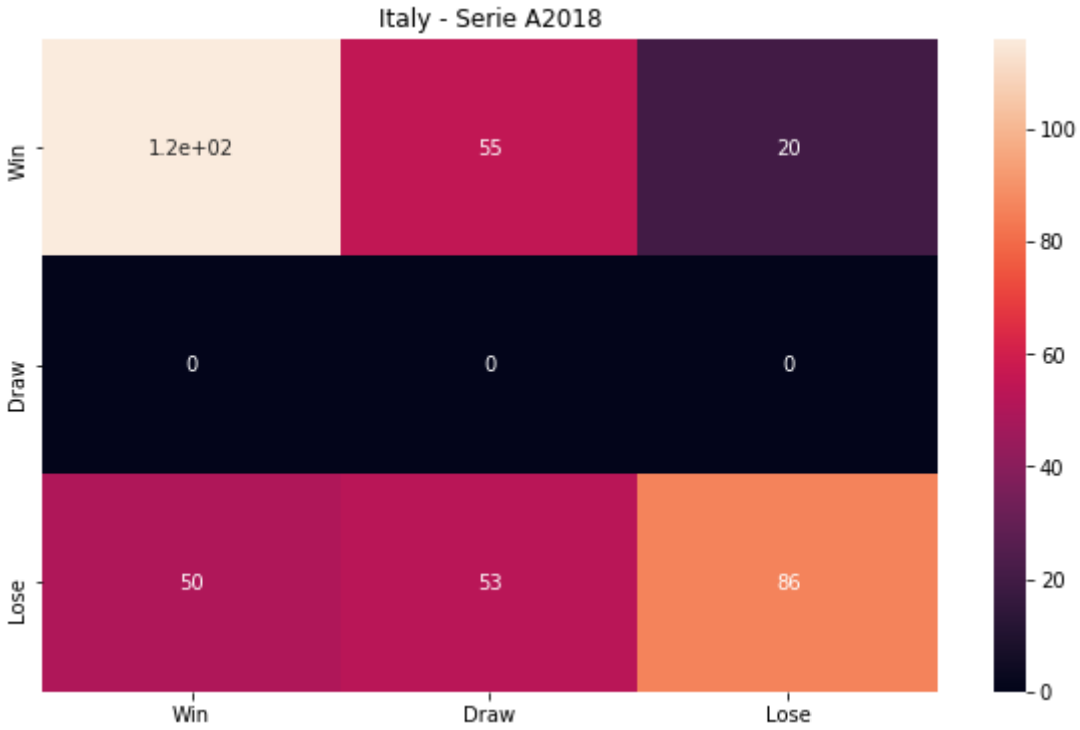

Figure H8: Constant K-Factor Elo confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 53.16%
MAE: 0.40
RMSE: 0.41
Precision:      Win 0.61        Draw 0.00       Lose 0.46
Recall:         Win 0.70        Draw 0.00       Lose 0.81
F1:     Win 0.65        Draw 0.00       Lose 0.58
```

Figure H9: Constant K-Factor Elo metrics for Italy Serie A 2018-2019

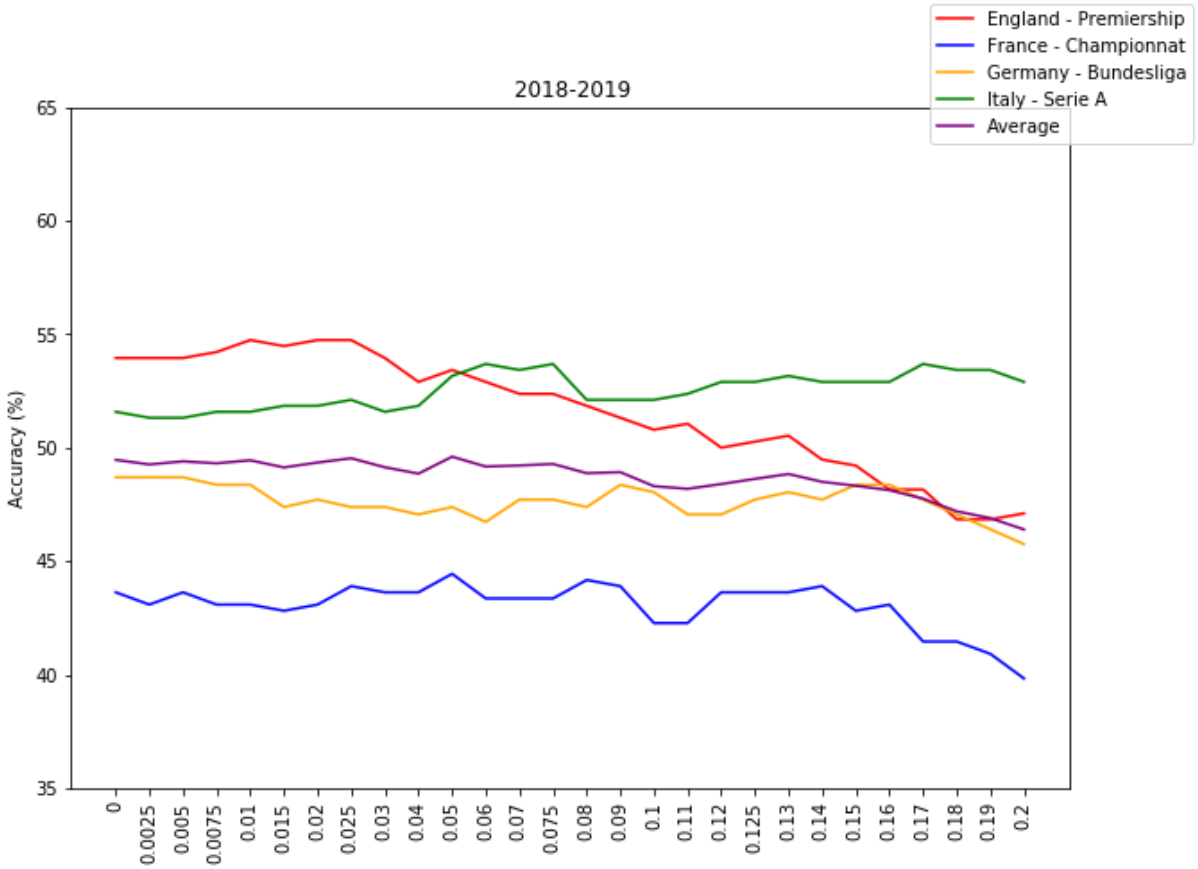

Figure H10: Overall accuracy 2018-2019 (y) varying depending on constant draw gap (x)

Appendix I: Variation of K-Factor within a single tournament

The constant value of K that achieved highest overall accuracy is 12, with the accuracy being 48.82%. Varying a constant K-Factor within a single tournament may give insight into whether the trend in Figure I1 is representative of all tournaments, or rather a result of averaging the accuracies.

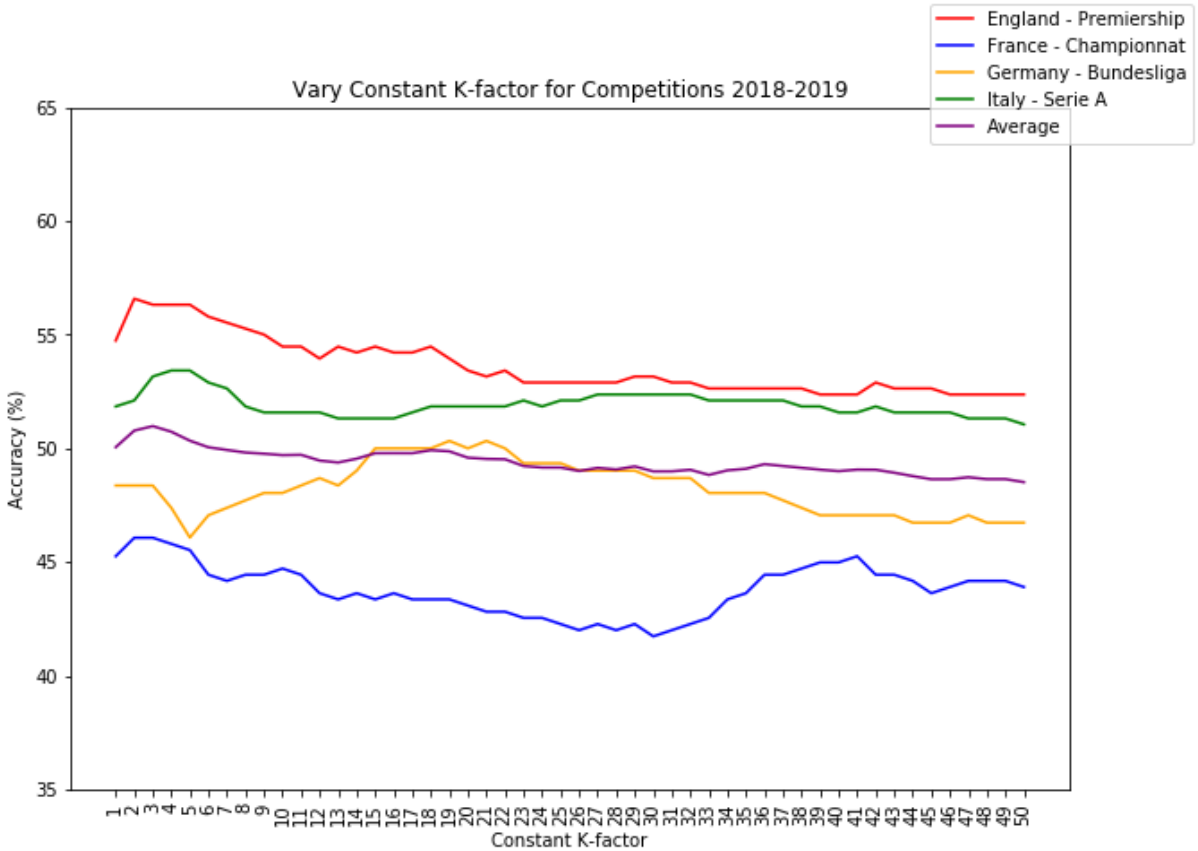

Figure I1: Overall accuracy 2018 - 2019 (y) varying depending on a constant K-Factor (x)

As can be seen in Figure I1, varying constant K-Factor across 1 tournament produces a lot more variability than can be observed as a result of varying the variable within the frame of competitions. While originally, 12 was found to be the best K-Factor to produce the highest overall accuracy, it is evident that for 2018-2019 tournaments, a K-Score of 3 was the optimal one. It appears that the best value for K-Factor varies from tournament to tournament. Thus, it is most reasonable to choose the optimal constant K-Factor based on the overall accuracy rather than a single tournament, since it would not be possible to compute the optimal K-Factor for a given year without the benefit of hindsight.

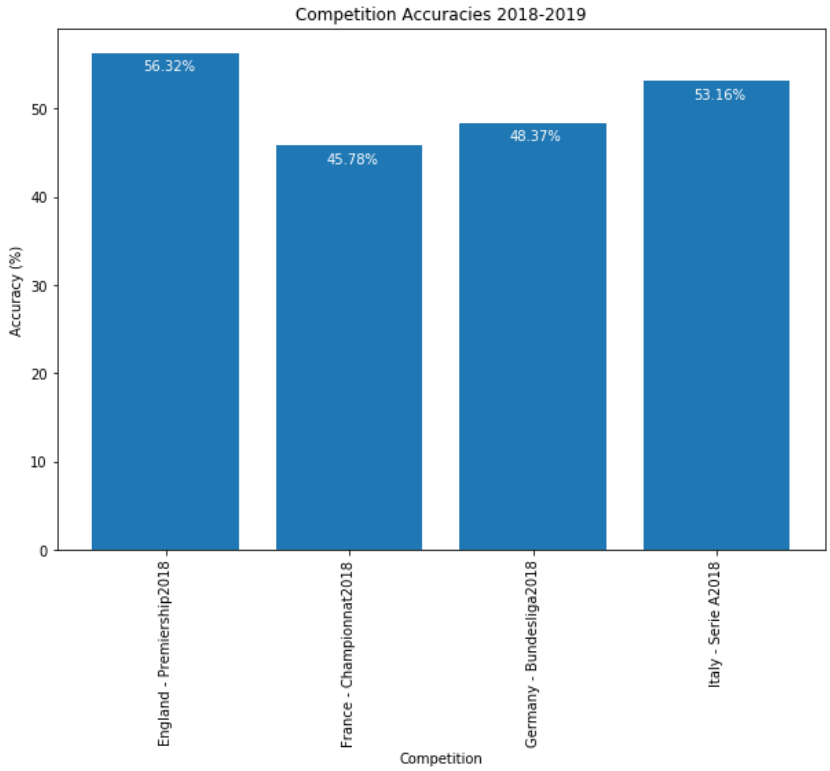

Figure I2: Constant K-Factor prediction accuracy for selected tournaments 2018 – 2019

Appendix J: Formula-Based K-Factor

A constant K-Factor is not always optimal when using the Elo system. For example, in chess, as a team's score gets higher, the K-Factor drops (Moser J., 2010). To further test the optimality of the constant K-Factor, it was compared against a variety of function-based K-Factors. A concept of decay is introduced, where K-Factor starts at a value of X and then decreases to Y linearly. The gradual drop of K-Factor can be caused by the number of games played, player rating or both.

Here, a variety of different functions were used. `games_played` is the number of games that the system has seen for a particular team while `rating` is the team's current elo score. As K-Score of 12 appears to perform best amongst constant K-Factors, it is used as a baseline to compare against.

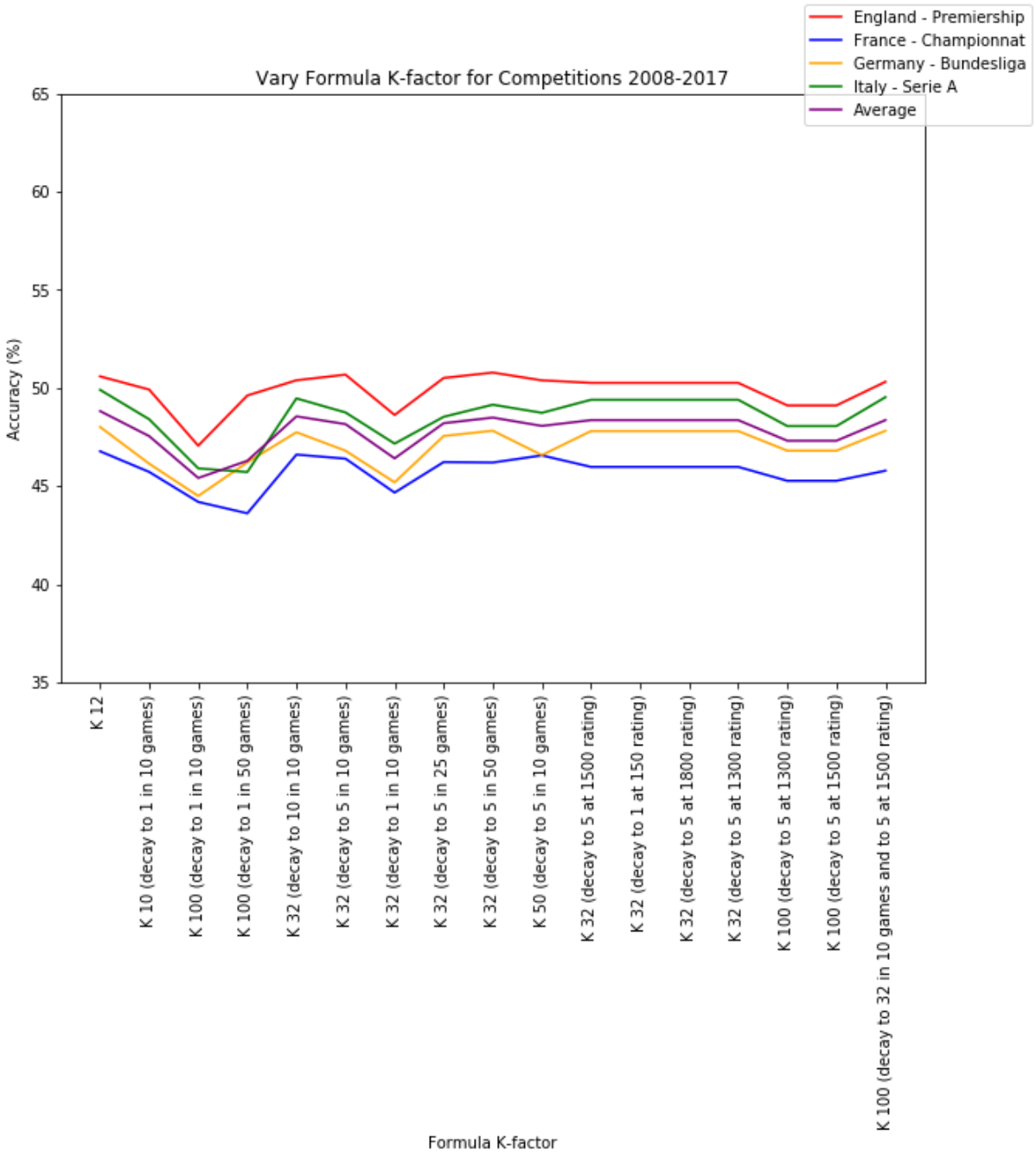

Figure J1: Overall accuracy for train set (y) varying depending on a formula K-Factor (x)

As can be seen in Figure J1, formula-based K-Factor resulted in worse performance than a constant K-Factor of 12 for all attempted formulas. Some formula-based K-Factors produced similar levels of accuracy as a constant one, falling short by about 0.5%. It is worth exploring an optimal K-Factor for the 2018-2019 tournaments.

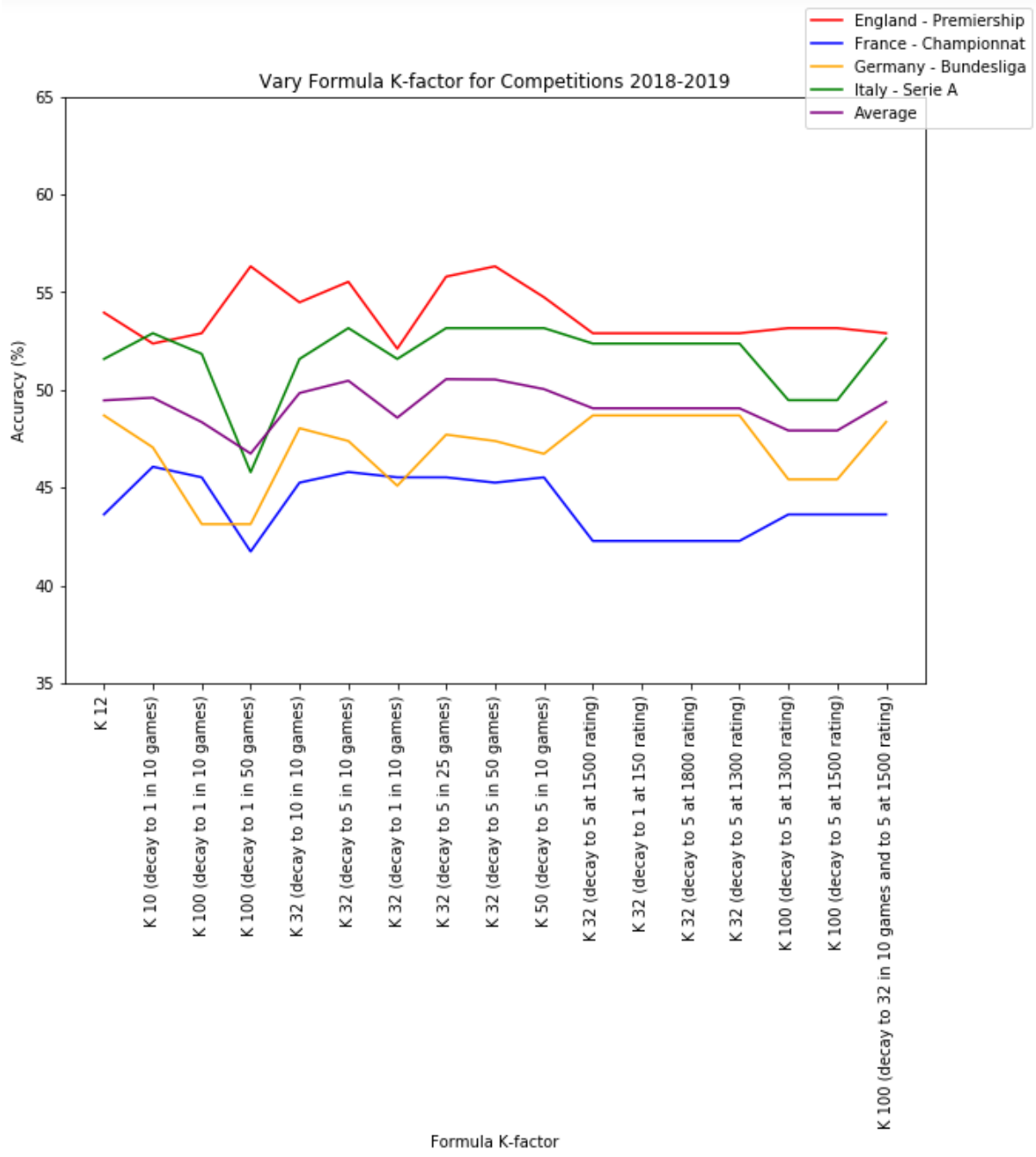

Figure J2: Overall accuracy 2018-2019 (y) varying depending on a formula K-Factor (x)

As can be seen in Figure J2, for 2018-2019 tournaments, the best-performing K-Score was a starting K-Factor of 32, which linearly decayed to 5 during the first 25 games for each team. This resulted in an overall accuracy of 50.55% for 2018-2019 tournaments, which is about 1% higher, than that of a constant K-Score of 12. This demonstrates the accuracy drop that the tournament may experience as a result of choosing a K-Score that works best overall,

rather than for a specific tournament. While a drop of 1% in accuracy is not a trivial one in betting, it is not critical and it does demonstrate that the process by which the K-Score value was chosen does work. And so, the final chosen K-Factor is a constant - 12, as it appears to perform best with the target competitions.

Appendix K: Elo Dynamic Draw Chance

Another way to predict draws is to keep track of the draw chance for a particular competition, given the historical results the model has seen so far. The draw chance is then dynamically updated after every match. As evident from Figures K1 through K3, it appears that merely calculating a draw chance of the tournament and using that as a marker for future draws does not result in an increase in the overall prediction accuracy.

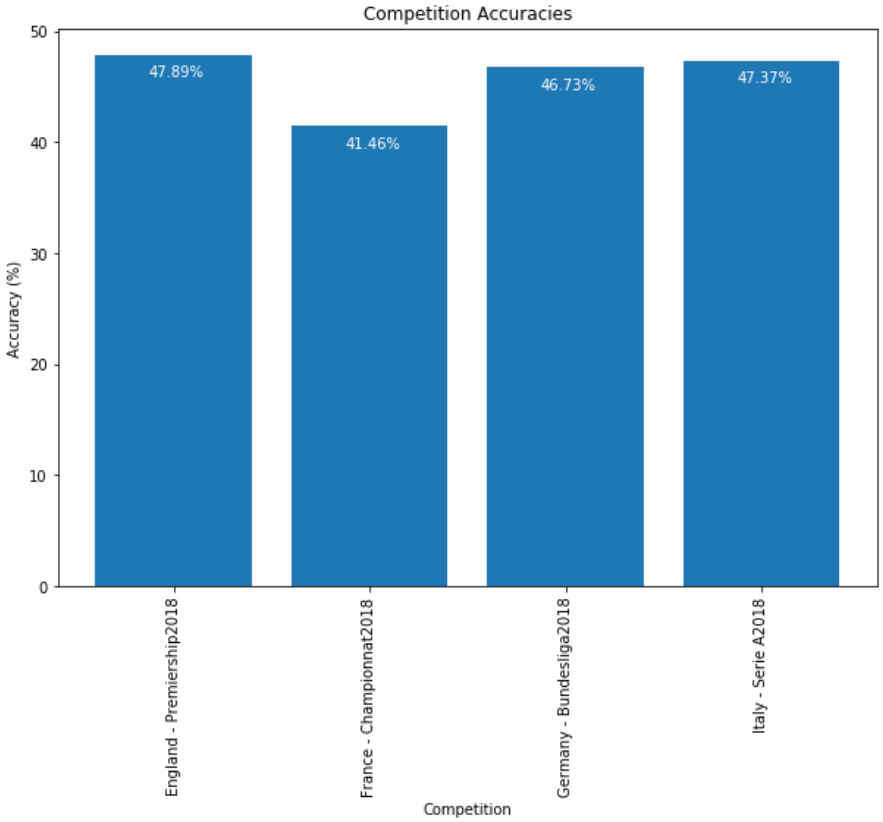

Figure K1: Dynamic Draw prediction accuracy for selected tournaments 2018 - 2019

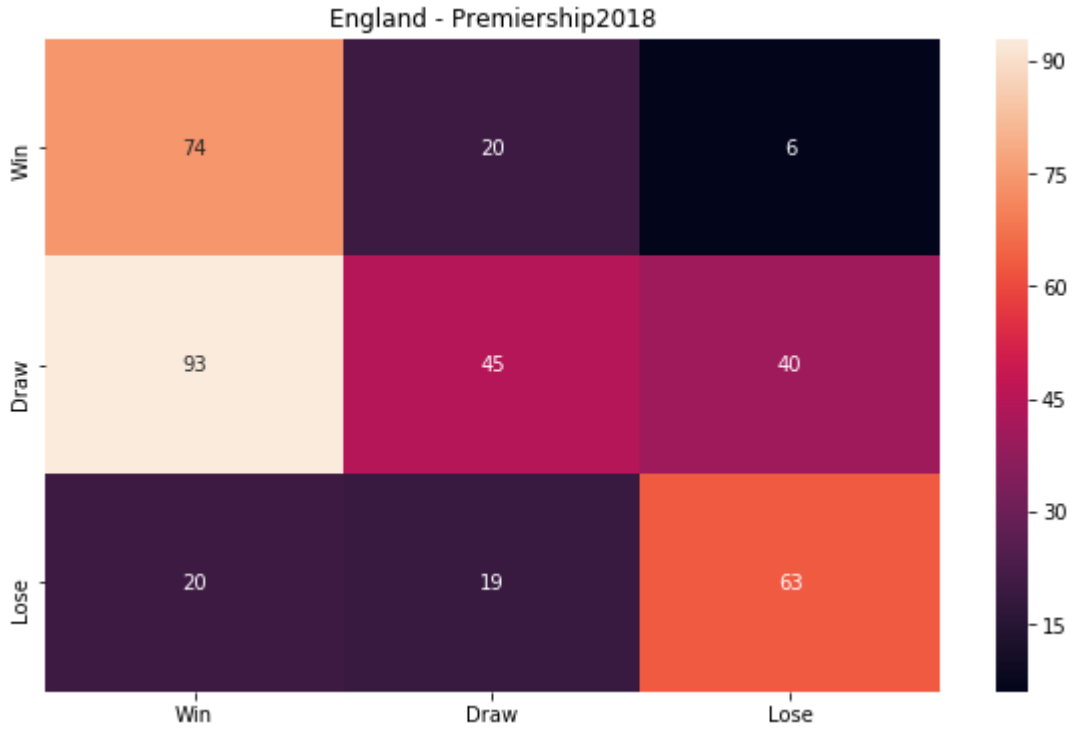

Figure K2: Dynamic draw confusion matrix for England Premiership 2018-2019

```
Accuracy: 47.89%
MAE: 0.41
RMSE: 0.42
Precision:      Win 0.74        Draw 0.25       Lose 0.62
Recall:         Win 0.40        Draw 0.54       Lose 0.58
F1:     Win 0.52        Draw 0.34       Lose 0.60
```

Figure K3: Dynamic draw metrics for England Premiership 2018-2019

Perhaps by tuning how much effect the past draws have on the future judgements by the system it is possible to find an optimal balance between the two where those tournaments, like England Premiership, that have a higher number of draws would benefit from the draw chance, while others, like Germany Bundesliga, would not be affected by it as much. However, as was discovered when using constant draw gap, introduction of any amount of draw gap is detrimental to the model's performance. Thus, no matter what percentage of draws the tournament may actually result in (which is greater than 0), using that value to predict future draws will not improve the model's overall predictions. This is all assuming that the penalty cost for misclassifying a result is the same for all outcomes.

Appendix L: Elo Time-based K-Factor

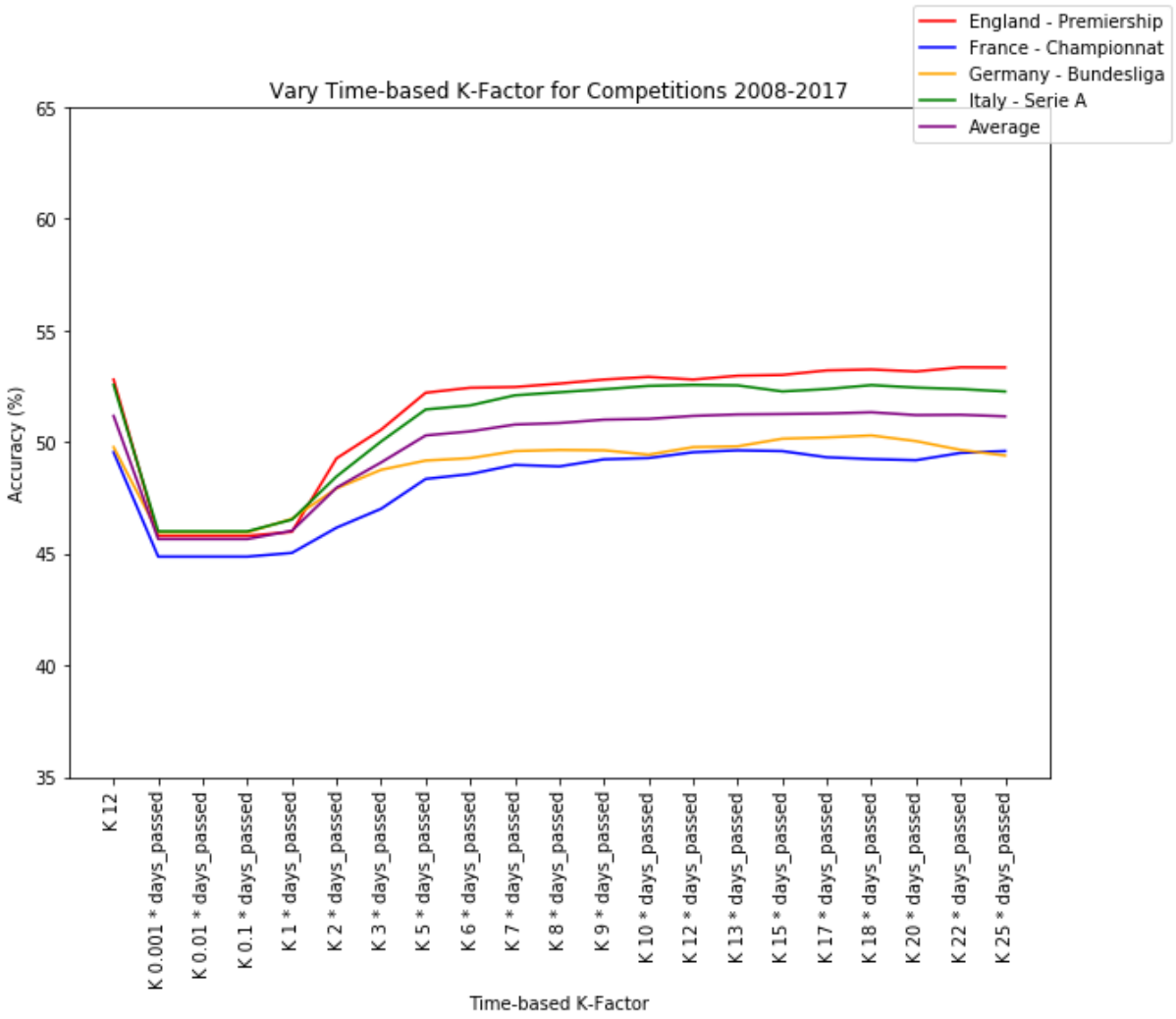

Figure L1: Accuracy for 2008 - 2017 tournaments using different Time-based K-Factors

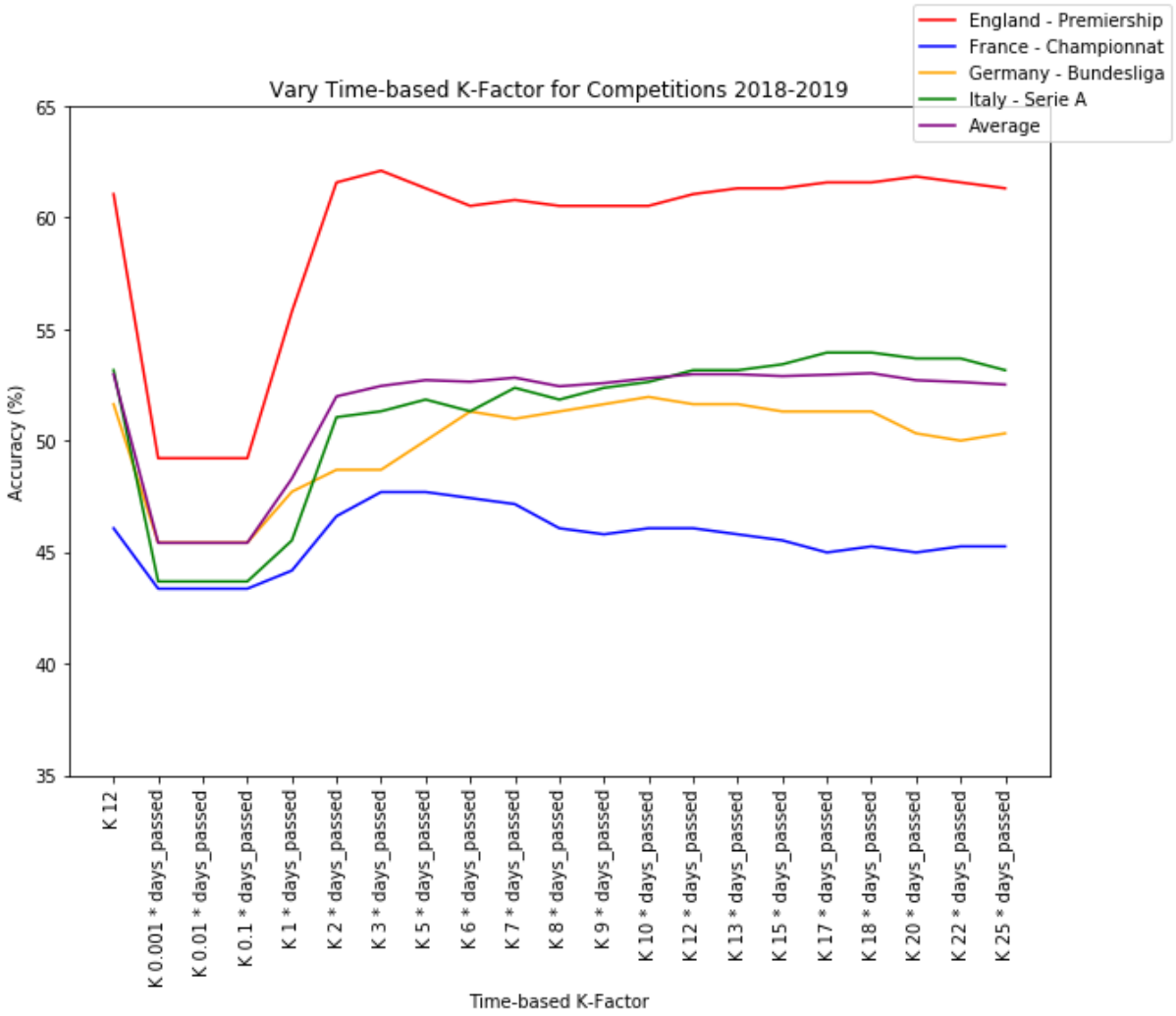

Figure L2: Accuracy for 2018 - 2019 tournaments using different Time-based K-Factors

Appendix M: Non-binary K-Factor for Elo

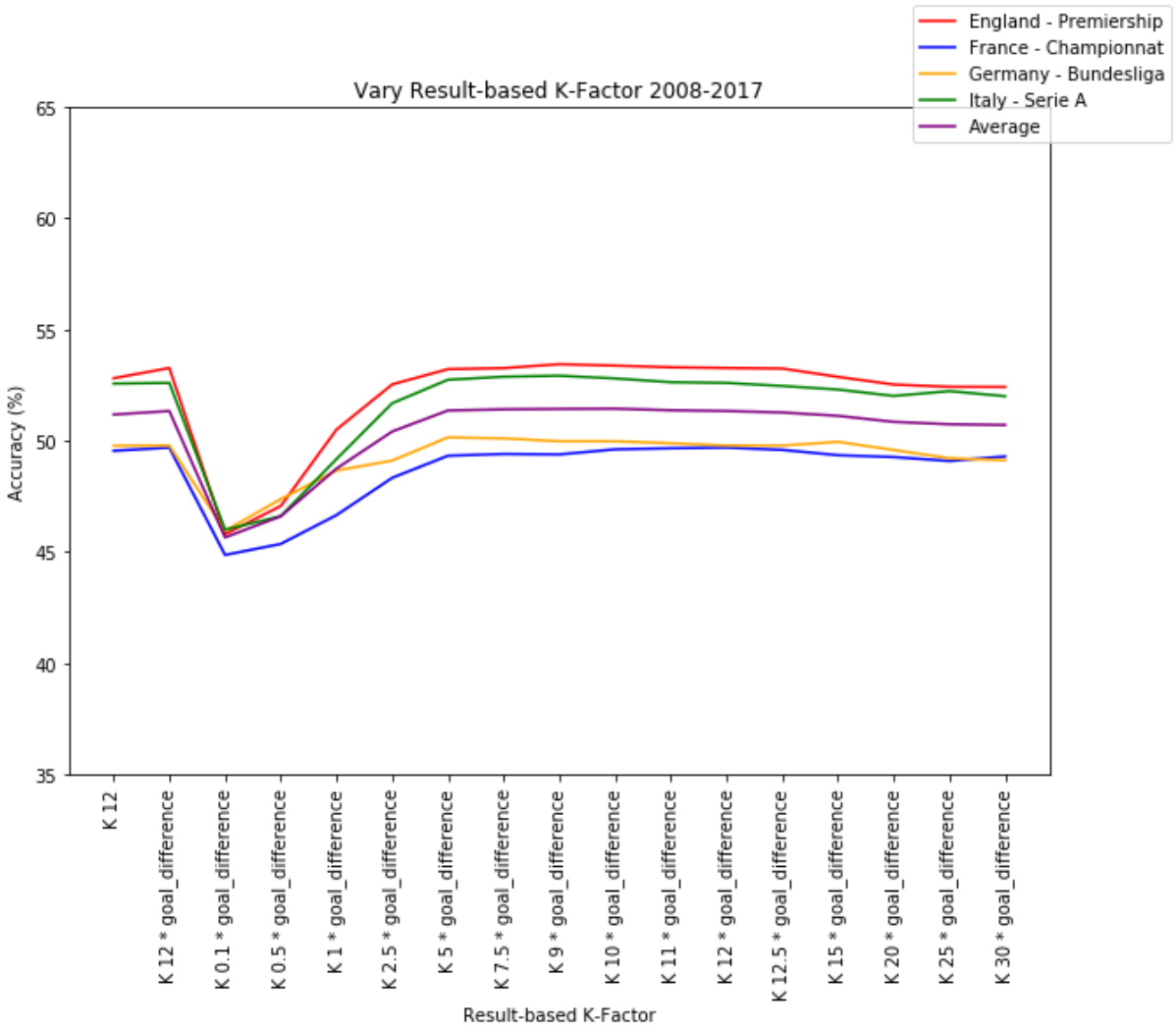

Figure M1: Accuracy for 2008 - 2017 tournaments using different Non-binary K-Factors

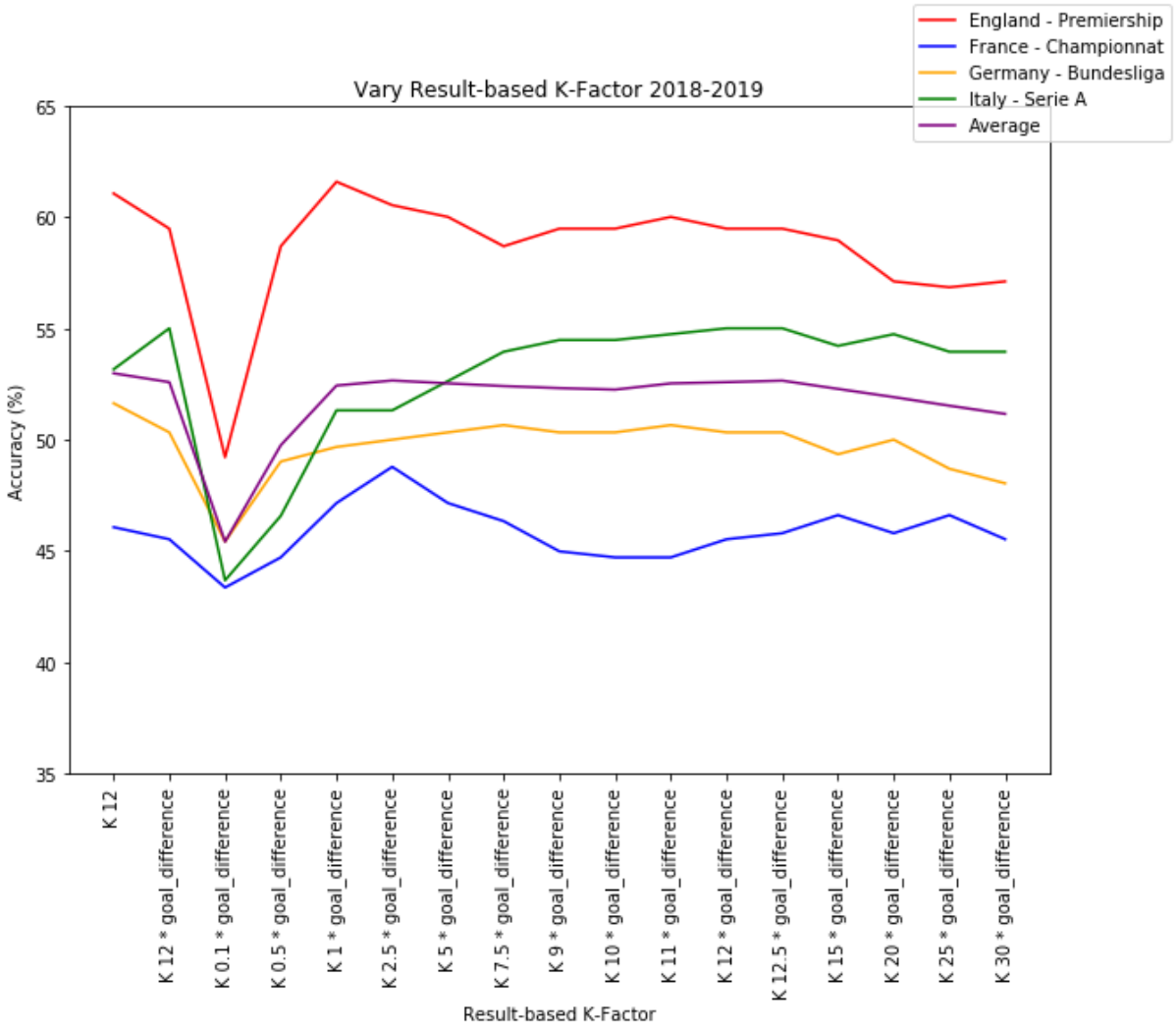

Figure M2: Accuracy for 2018 - 2019 tournaments using different Non-binary K-Factors

Appendix N: Grid Searching Multiple Hyperparameters for Elo

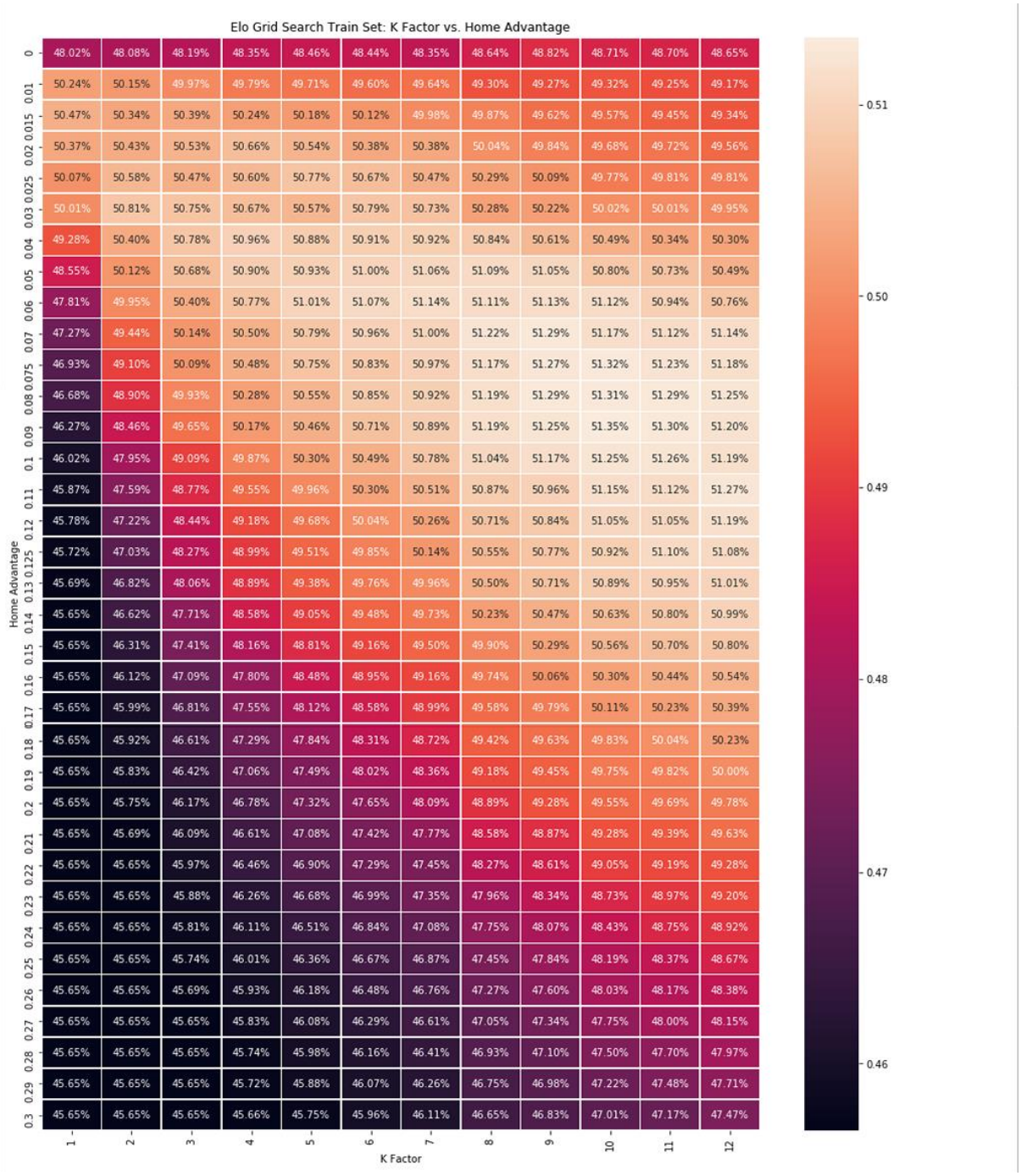

Figure N1: Using Grid Search to determine optimal K Factor and Home Advantage values based on the highest accuracy

The results of the Grid Search, which can be seen in Figure N1, show similar results to what is observed by varying individual variables. This further verifies the validity of the parameters that were discovered when tuning the Elo model.

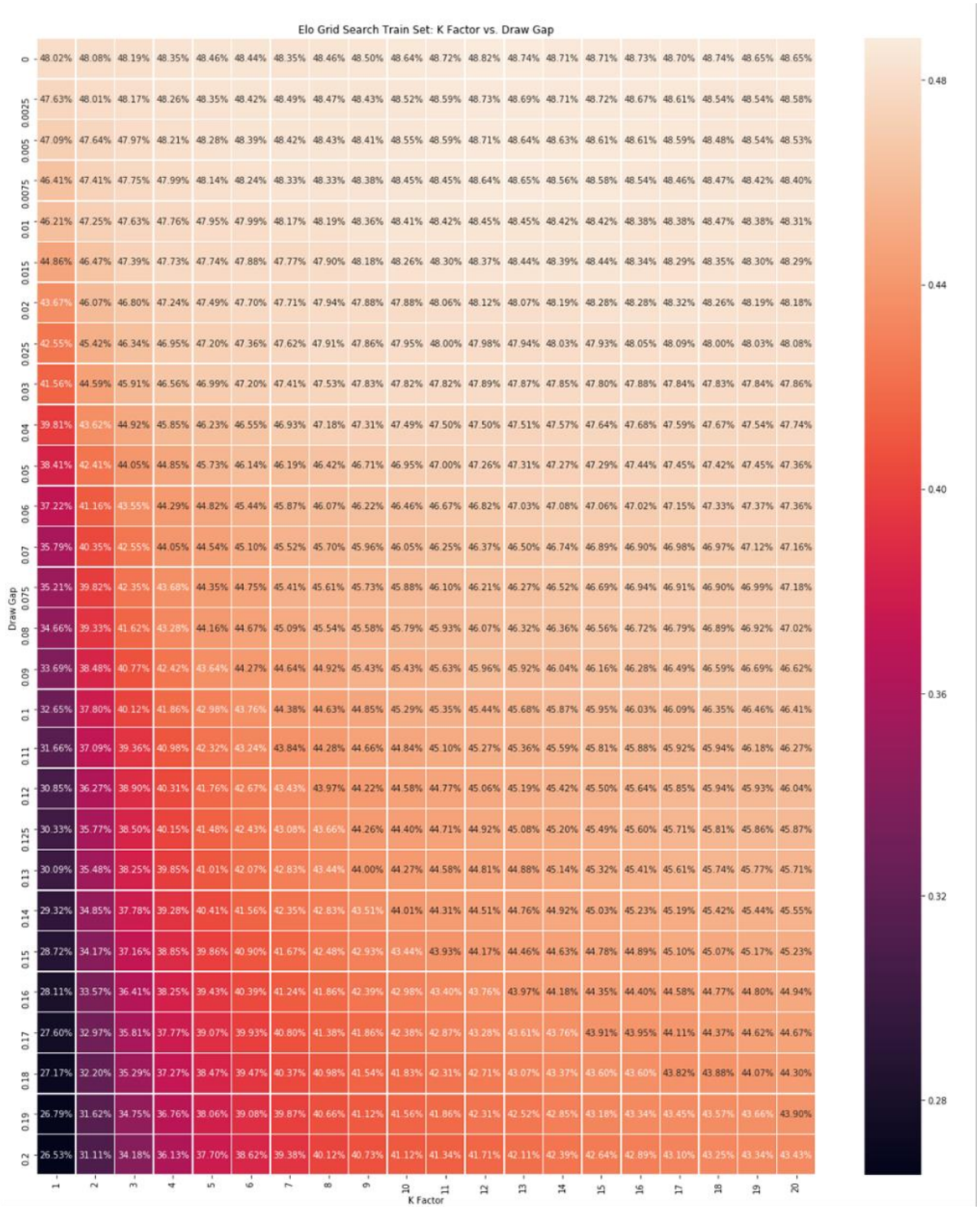

Figure N2: Using Grid Search to determine optimal K Factor and Draw Gap values based on the highest accuracy

As can be seen in Figure N2, introducing Draw Gap does reduce the prediction accuracy of the Elo model. The results on Figure N2 confirm that the optimal K Factor and Draw Gap are 12 and 0 respectively.

Appendix O: Average Elo vs. Competition-based Elo Comparison

Figures X through Y demonstrate the performance metrics for the Elo model that uses best parameters on average. The list of parameters:

- K-Factor: 12 * *d*
- Home Advantage: 0.011
- Draw Gap: 0

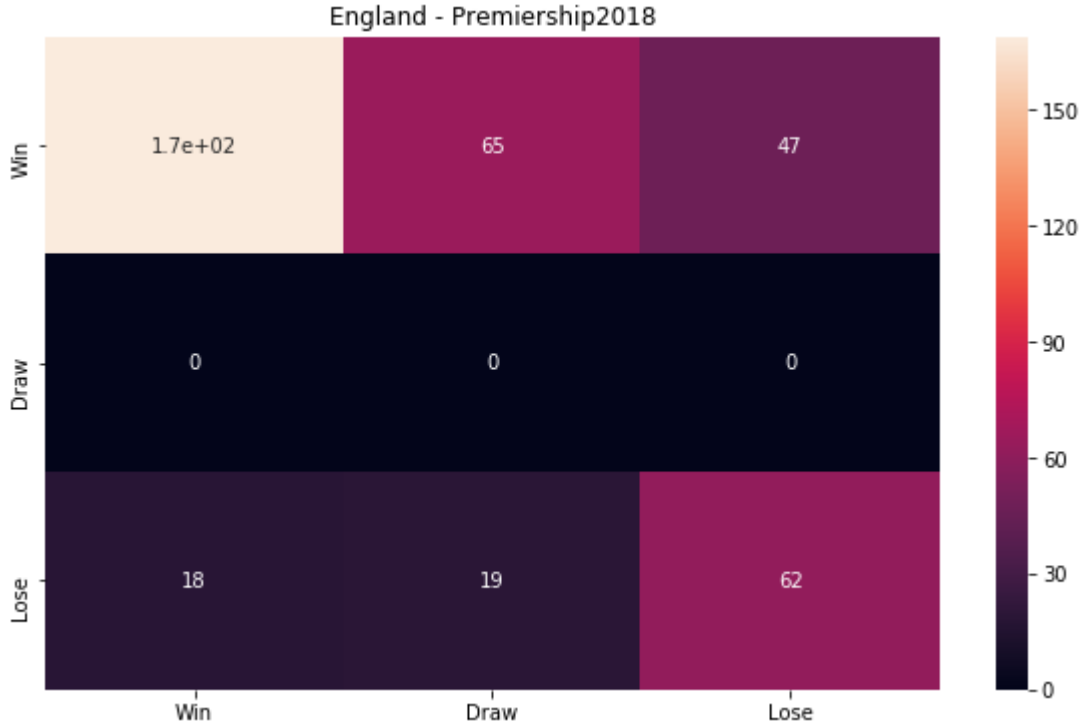

Figure O1: Best Average Elo confusion matrix for England Premiership 2018-2019

```
Accuracy: 60.79%
MAE: 0.32
RMSE: 0.34
Precision:      Win 0.90        Draw 0.00       Lose 0.57
Recall:         Win 0.60        Draw 0.00       Lose 0.63
F1:     Win 0.72        Draw 0.00       Lose 0.60
```

Figure O2: Best Average Elo metrics for England Premiership 2018-2019

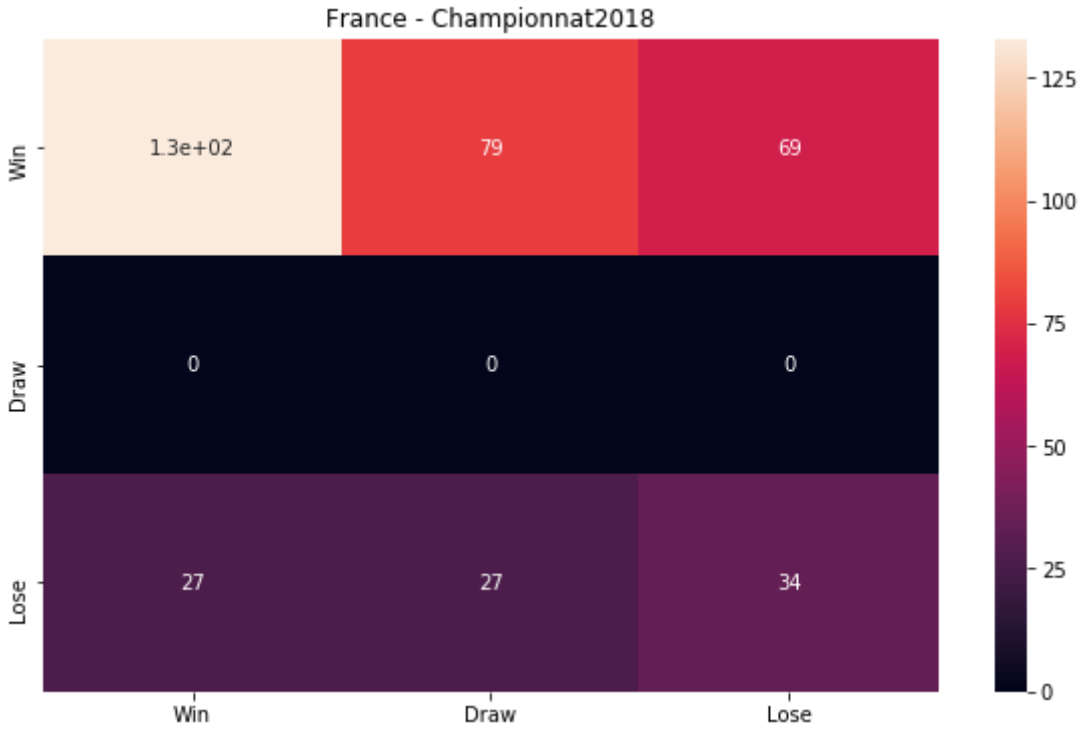

Figure O3: Best Average Elo confusion matrix for France Championnat 2018-2019

```
Accuracy: 45.26%
MAE: 0.35
RMSE: 0.37
Precision:      Win 0.83        Draw 0.00       Lose 0.33
Recall:         Win 0.47        Draw 0.00       Lose 0.39
F1:     Win 0.60        Draw 0.00       Lose 0.36
```

Figure O4: Best Average Elo metrics for France Championnat 2018-2019

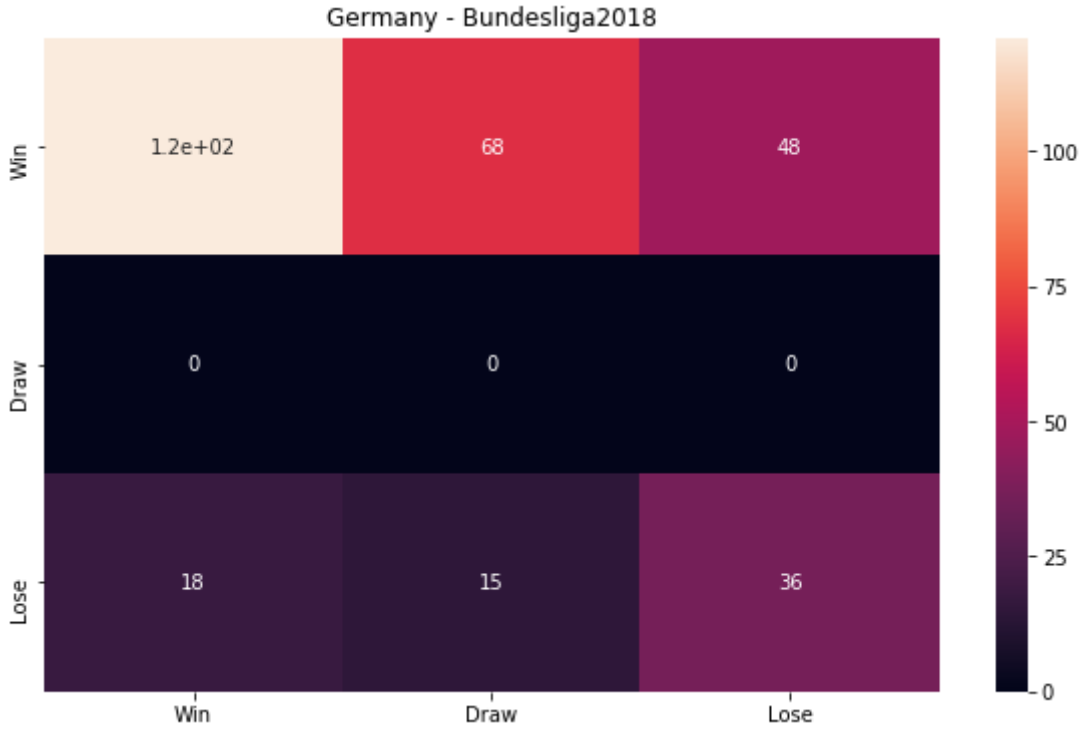

Figure O5: Best Average Elo confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 51.31%
MAE: 0.35
RMSE: 0.37
Precision:      Win 0.87        Draw 0.00       Lose 0.43
Recall:         Win 0.51        Draw 0.00       Lose 0.52
F1:     Win 0.64        Draw 0.00       Lose 0.47
```

Figure O6: Best Average Elo metrics for Germany Bundesliga 2018-2019

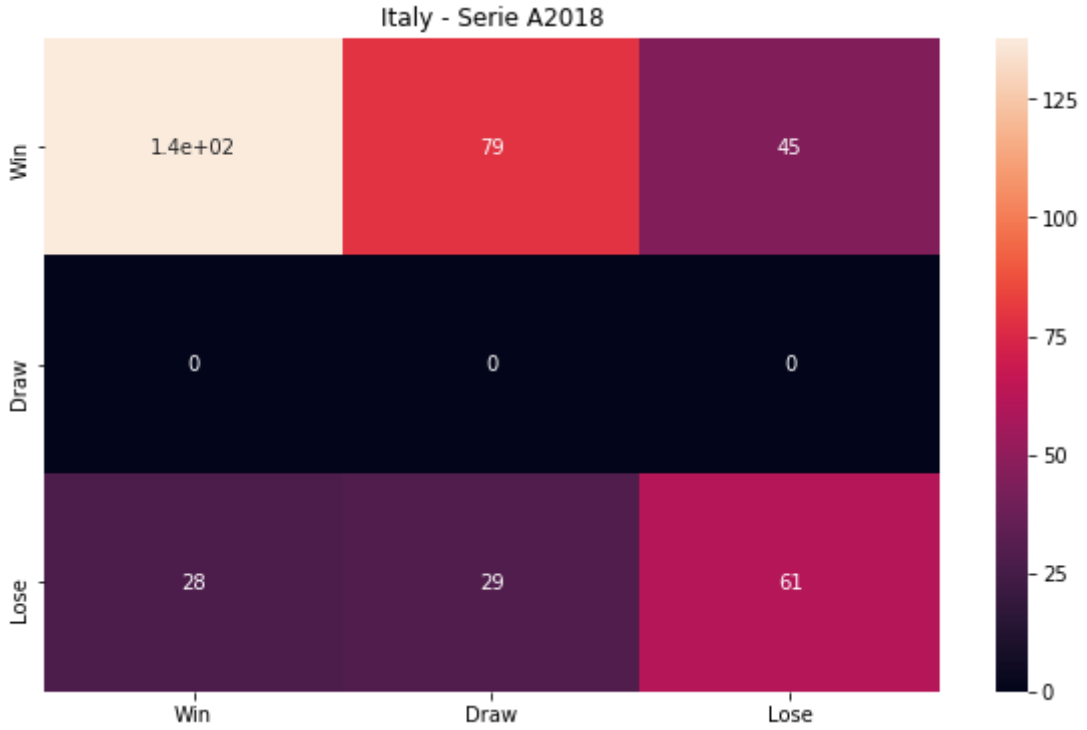

Figure O7: Best Average Elo confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 52.37%
MAE: 0.32
RMSE: 0.34
Precision:      Win 0.83        Draw 0.00       Lose 0.58
Recall:         Win 0.53        Draw 0.00       Lose 0.52
F1:     Win 0.64        Draw 0.00       Lose 0.54
```

Figure O8: Best Average Elo metrics for Italy Serie A 2018-2019

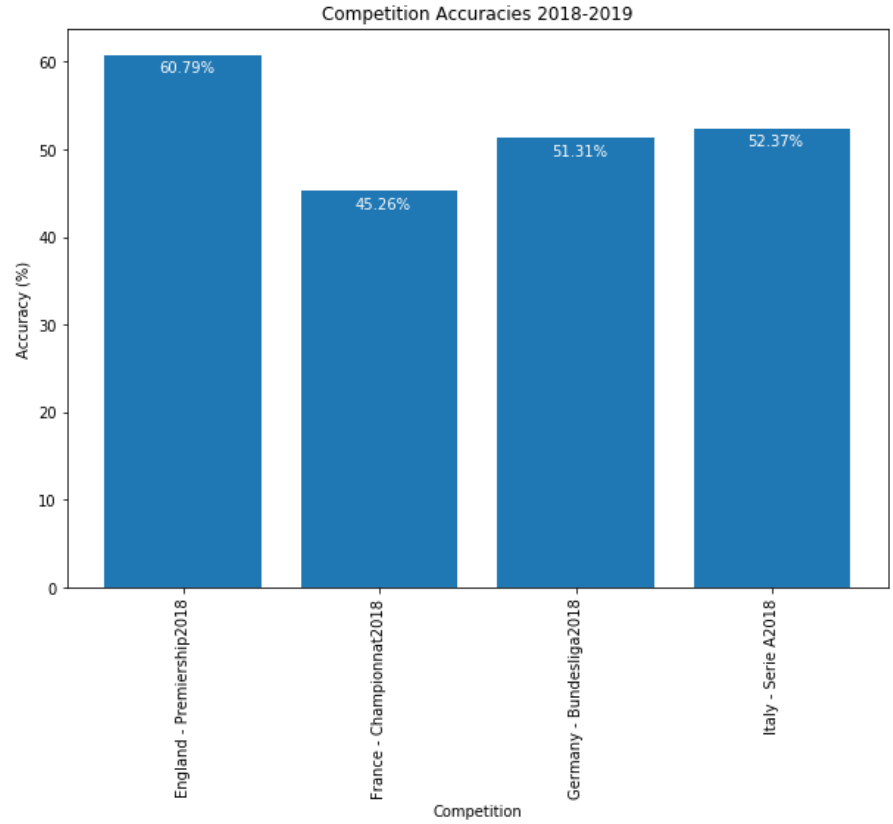

Figure O9: Best Average Elo accuracies for tournaments 2018 - 2019

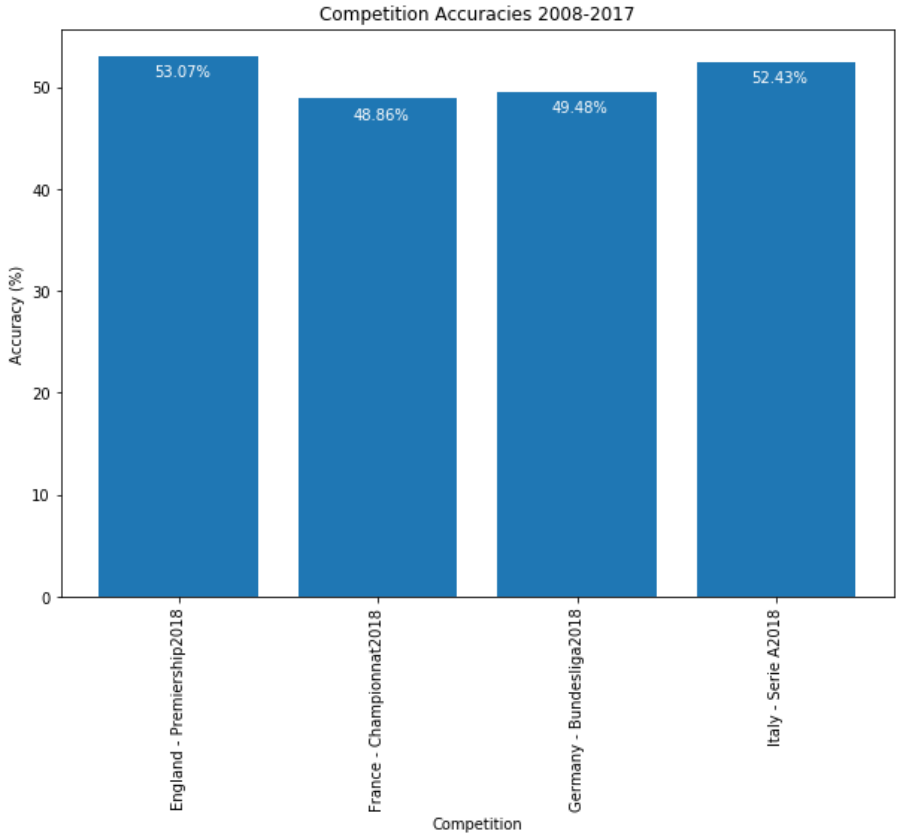

Figure O10: Best Average Elo accuracies for competitions 2008 - 2017

The Elo performance that uses the best hyperparameters, which were calculated using average competition metrics, is compared against the performance of Elo models that use hyperparameters that are best for the given competition.
List of hyperparameters for each competition:

- England Premiership
  - K-Factor: 12 * *d*
  - Home Advantage: 0.011
  - Draw Gap: 0
- France Championnat
  - K-Factor: 12 * *d*
  - Home Advantage: 0.011
  - Draw Gap: 0
- Germany Bundesliga
  - K-Factor: 12 * *d*
  - Home Advantage: 0.011
  - Draw Gap: 0
- Italy Serie A
  - K-Factor: 12 * *d*
  - Home Advantage: 0.011
  - Draw Gap: 0

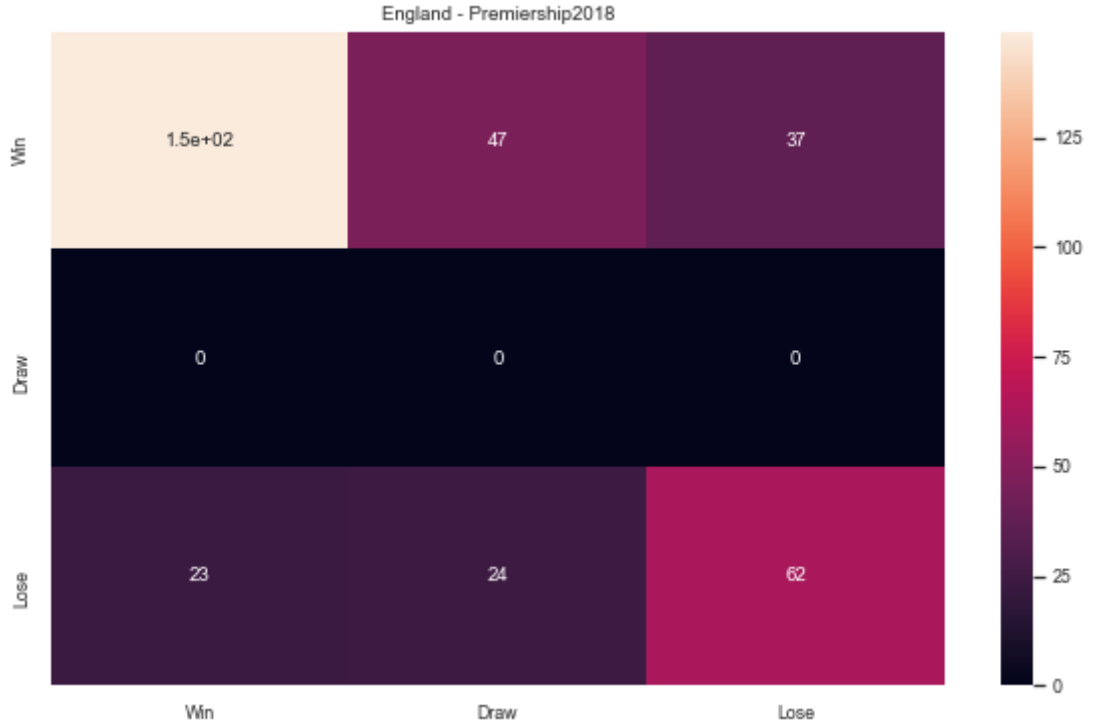

Figure O11: Competition-best Elo confusion matrix for England Premiership 2018-2019

```
Accuracy: 61.70%
MAE: 0.32
RMSE: 0.34
Precision:      Win 0.87        Draw 0.00       Lose 0.63
Recall:         Win 0.64        Draw 0.00       Lose 0.57
F1:     Win 0.74        Draw 0.00       Lose 0.60
```

Figure O12: Competition-best Elo metrics for England Premiership 2018-2019

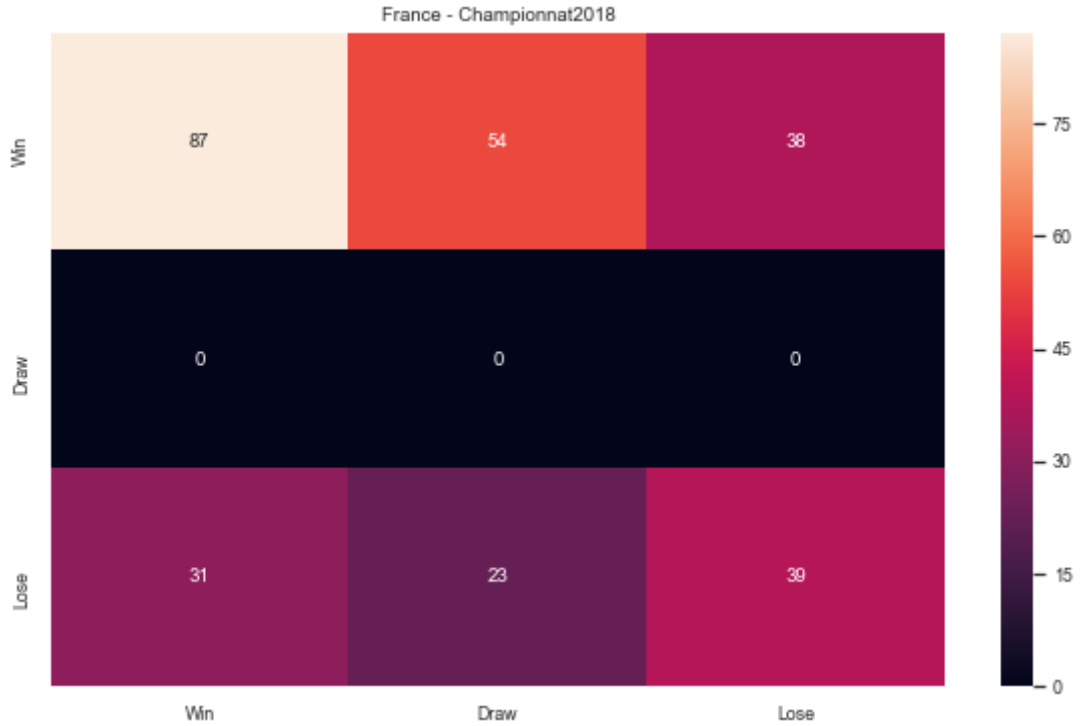

Figure O13: Competition-best Elo confusion matrix for France Championnat 2018-2019

```
Accuracy: 46.32%
MAE: 0.35
RMSE: 0.36
Precision:      Win 0.74        Draw 0.00       Lose 0.51
Recall:         Win 0.49        Draw 0.00       Lose 0.42
F1:     Win 0.59        Draw 0.00       Lose 0.46
```

Figure O14: Competition-best Elo metrics for France Championnat 2018-2019

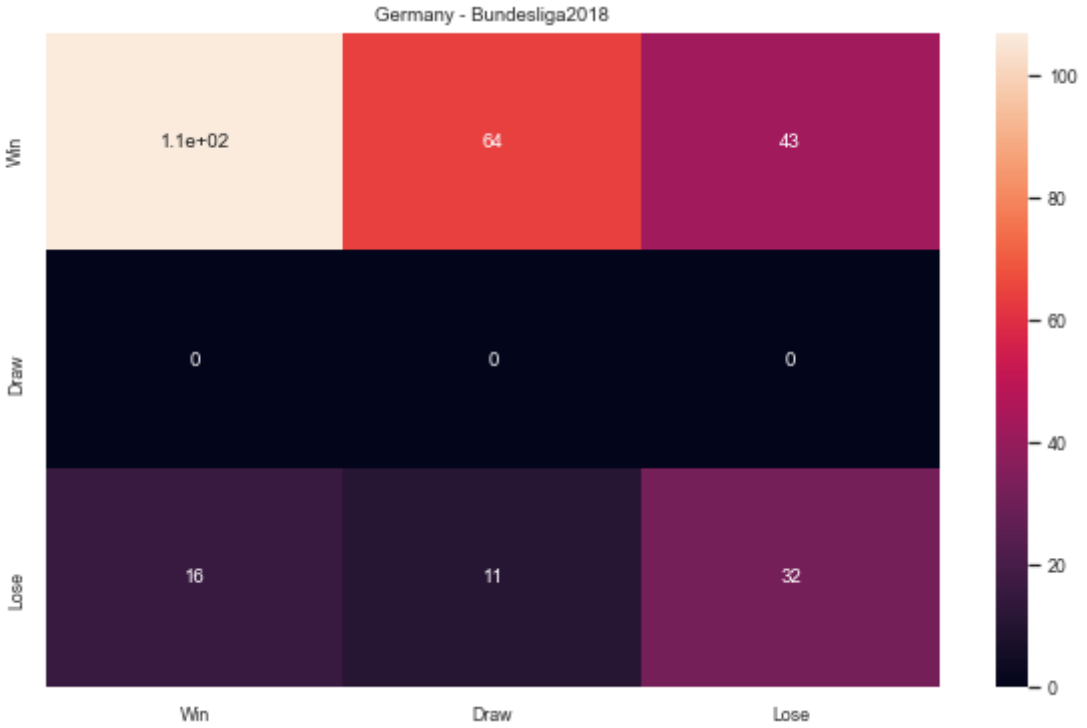

Figure O15: Competition-best Elo confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 50.92%
MAE: 0.32
RMSE: 0.35
Precision:      Win 0.87        Draw 0.00       Lose 0.43
Recall:         Win 0.50        Draw 0.00       Lose 0.54
F1:     Win 0.64        Draw 0.00       Lose 0.48
```

Figure O16: Competition-best Elo metrics for Germany Bundesliga 2018-2019

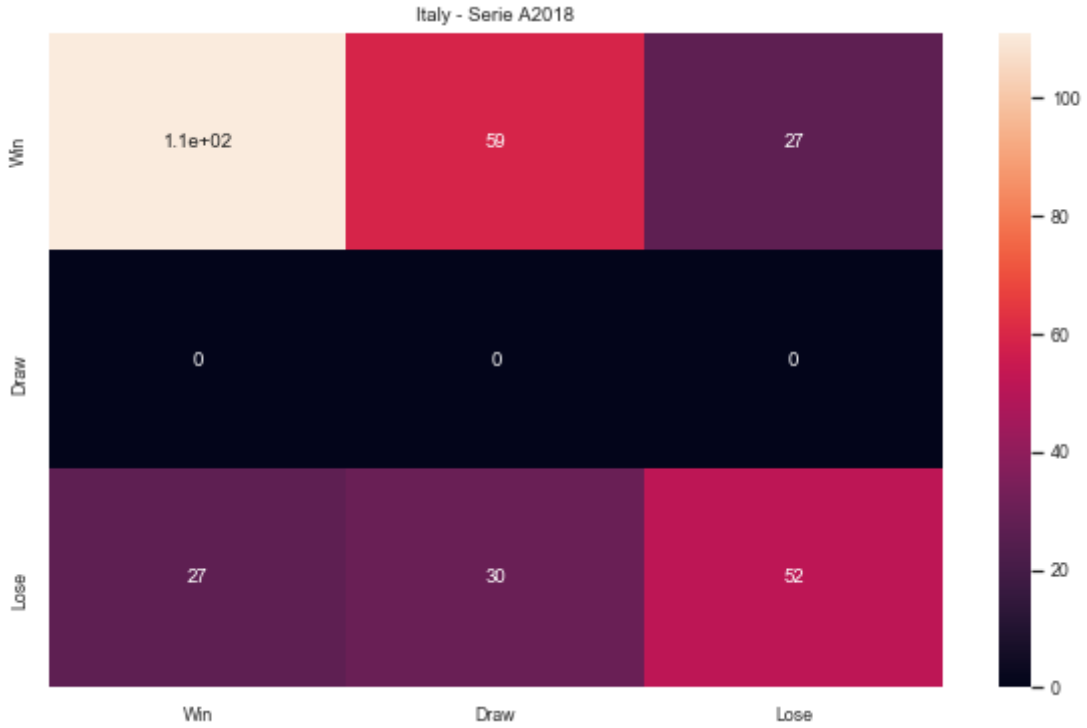

Figure O17: Competition-best Elo confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 53.27%
MAE: 0.33
RMSE: 0.35
Precision:      Win 0.80        Draw 0.00       Lose 0.66
Recall:         Win 0.56        Draw 0.00       Lose 0.48
F1:     Win 0.66        Draw 0.00       Lose 0.55
```

Figure O18: Competition-best Elo metrics for Italy Serie A 2018-2019

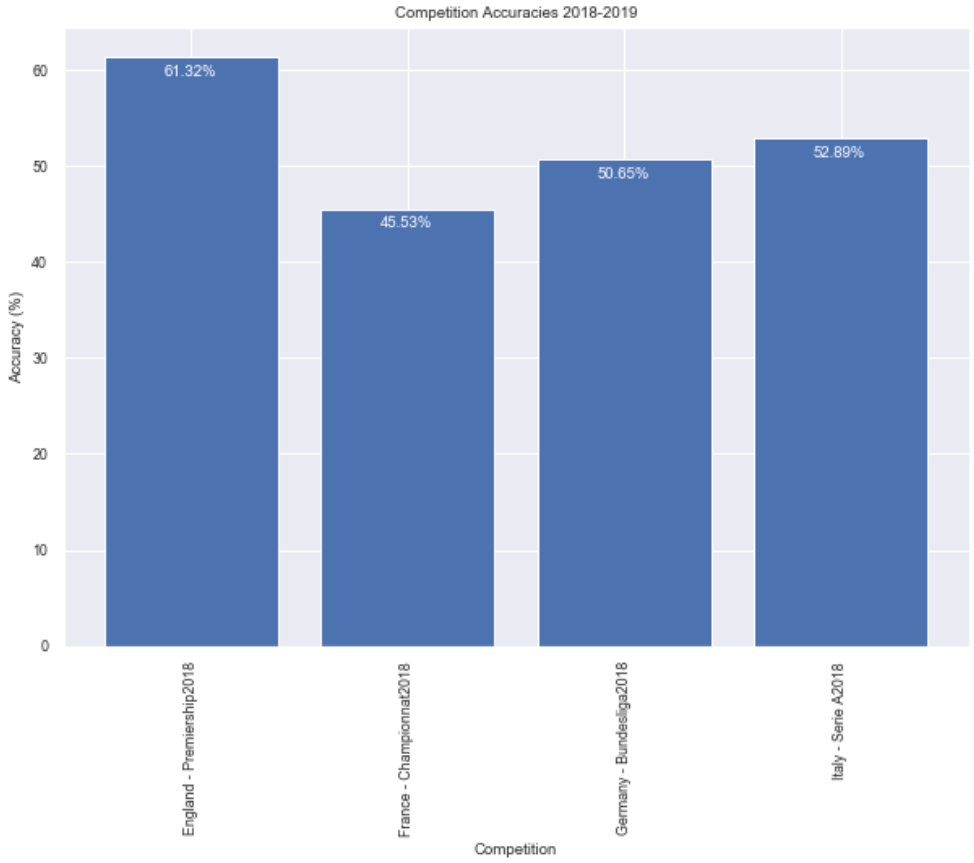

Figure O19: Competition-best Elo accuracies for tournaments 2018 - 2019

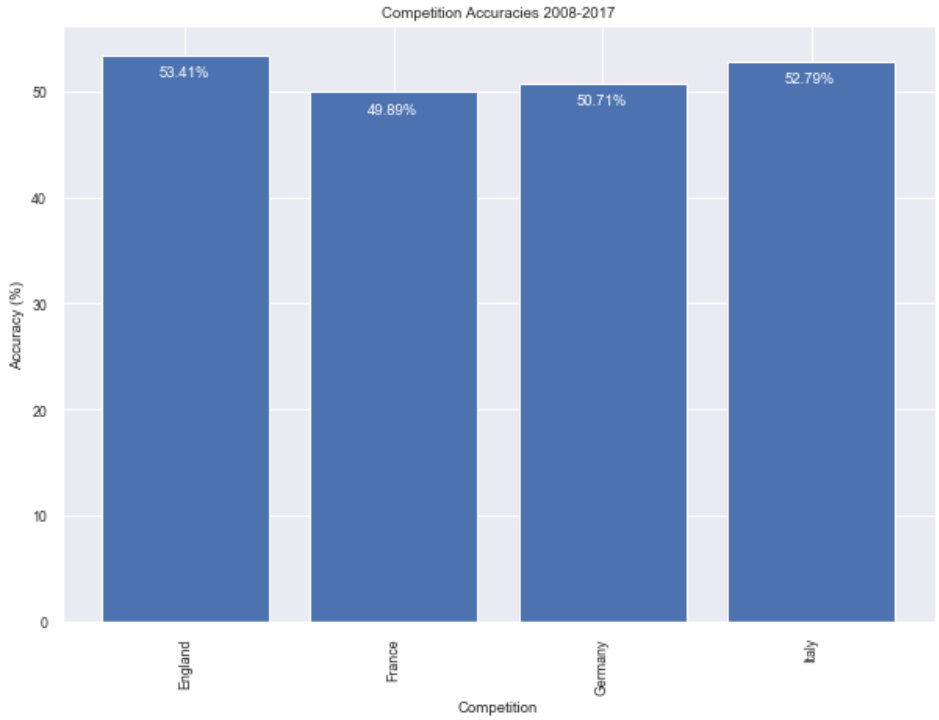

Figure O20: Competition-best Elo accuracies for competitions 2008 - 2017

Appendix P: Basic TrueSkill

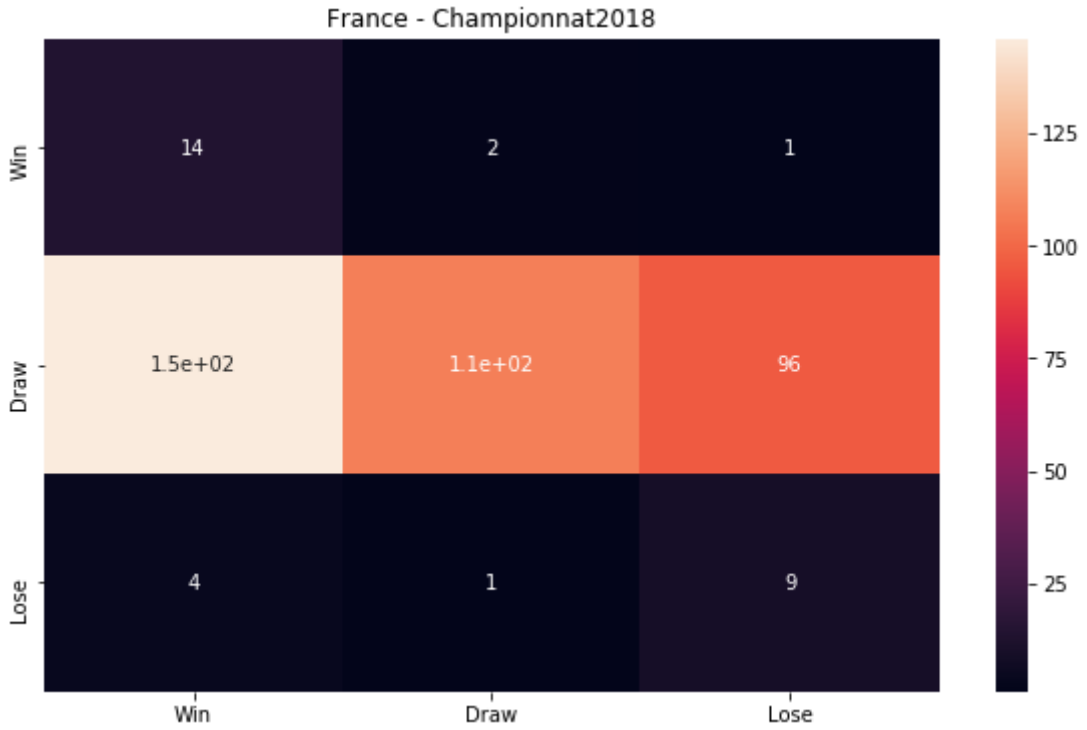

Figure P1: Initial TrueSkill confusion matrix for France Championnat 2018-2019

```
Accuracy: 34.21%
MAE: 0.42
RMSE: 0.43
Precision:      Win 0.82        Draw 0.31       Lose 0.64
Recall:         Win 0.09        Draw 0.97       Lose 0.08
F1:     Win 0.15        Draw 0.47       Lose 0.15
```

Figure P2: Initial TrueSkill metrics for France Championnat 2018-2019

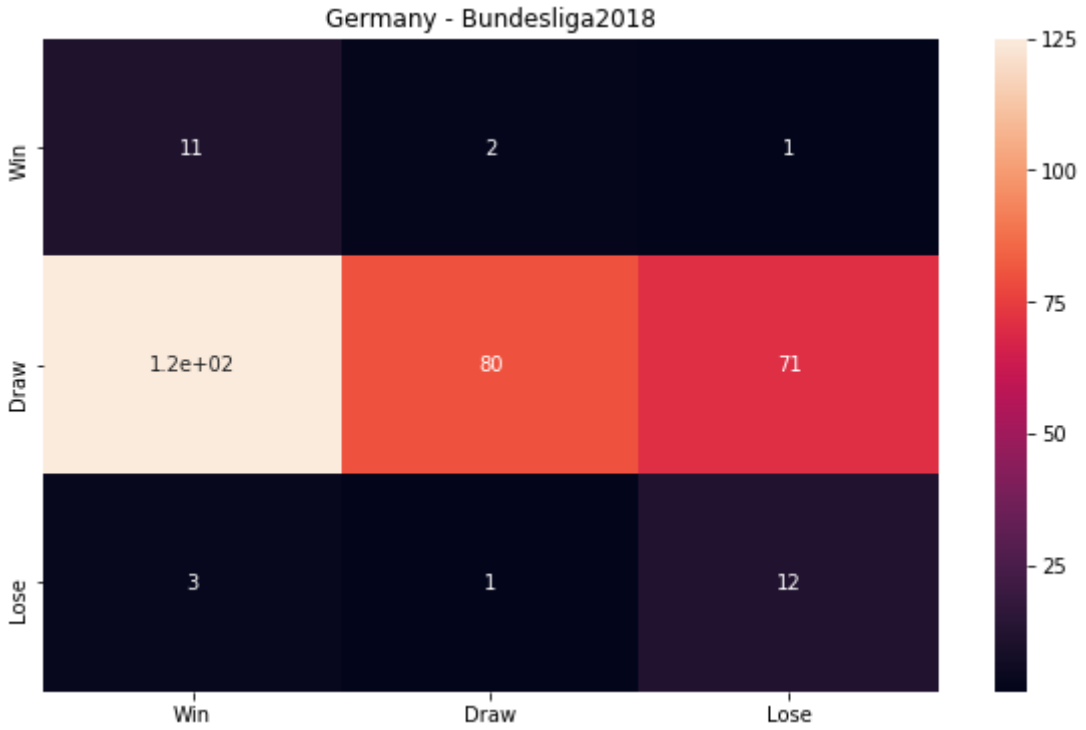

Figure P3: Initial TrueSkill confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 33.66%
MAE: 0.42
RMSE: 0.43
Precision:      Win 0.79        Draw 0.29       Lose 0.75
Recall:         Win 0.08        Draw 0.96       Lose 0.14
F1:     Win 0.14        Draw 0.45       Lose 0.24
```

Figure P4: Initial TrueSkill metrics for Germany Bundesliga 2018-2019

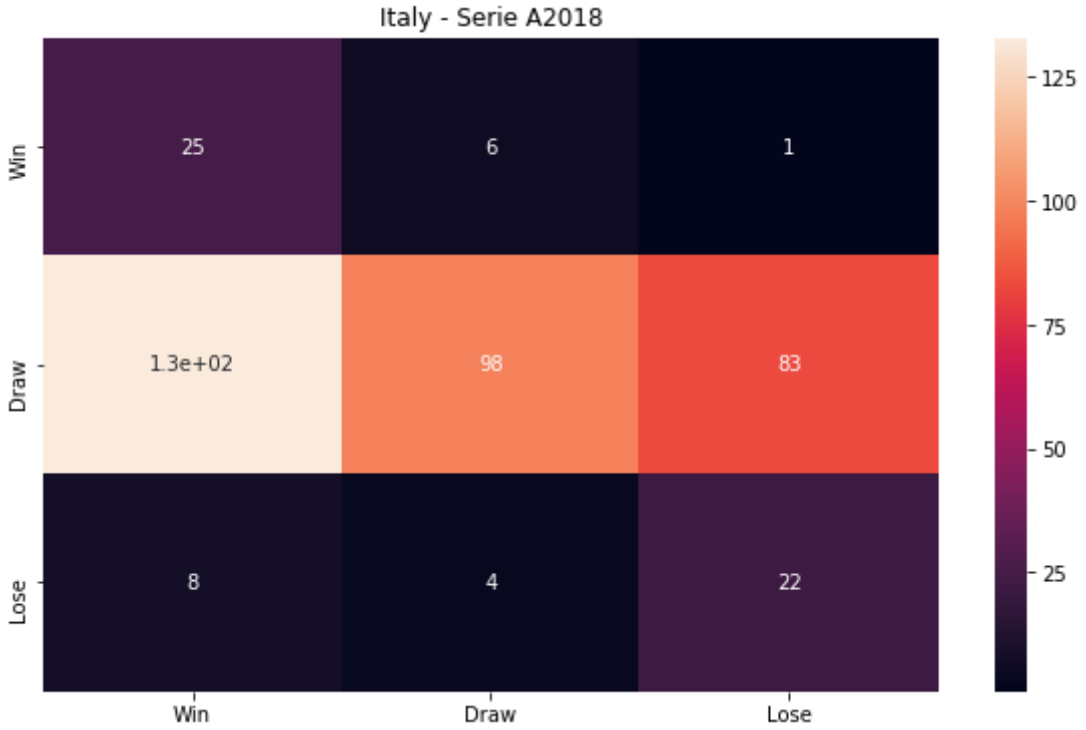

Figure P5: Initial TrueSkill confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 38.16%
MAE: 0.38
RMSE: 0.40
Precision:      Win 0.78        Draw 0.31       Lose 0.65
Recall:         Win 0.15        Draw 0.91       Lose 0.21
F1:     Win 0.25        Draw 0.46       Lose 0.31
```

Figure P6: Initial TrueSkill metrics for Italy Serie A 2018-2019

Appendix Q: Beta TrueSkill

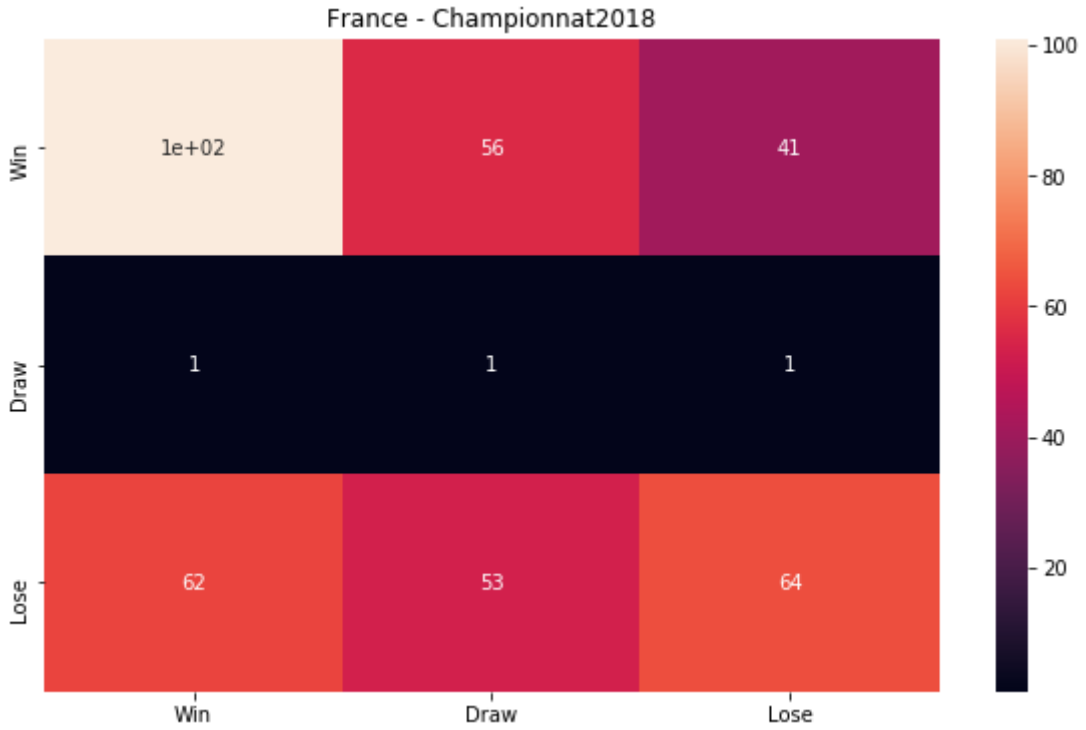

Figure Q1: Beta TrueSkill confusion matrix for France Championnat 2018-2019

```
Accuracy: 43.68%
MAE: 0.27
RMSE: 0.30
Precision:      Win 0.51        Draw 0.33       Lose 0.36
Recall:         Win 0.62        Draw 0.01       Lose 0.60
F1:     Win 0.56        Draw 0.02       Lose 0.45
```

Figure Q2: Beta TrueSkill metrics for France Championnat 2018-2019

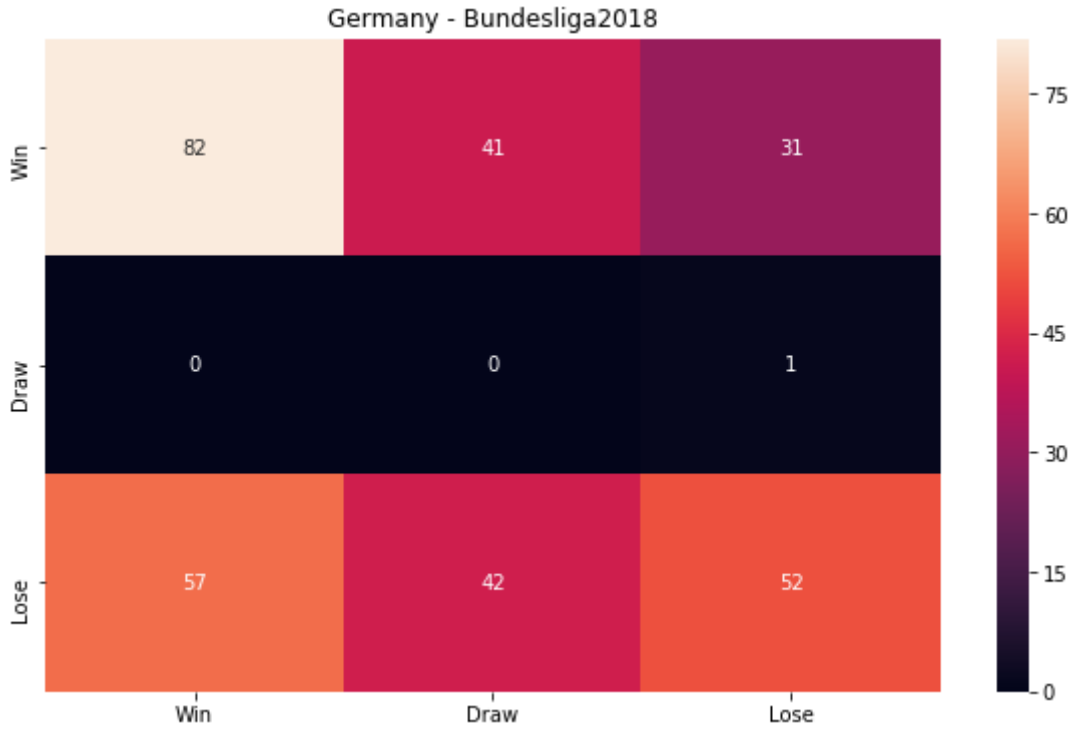

Figure Q3: Beta TrueSkill confusion matrix for Germany Bundesliga 2018-2019

```
Accuracy: 43.79%
MAE: 0.27
RMSE: 0.30
Precision:      Win 0.53        Draw 0.00       Lose 0.34
Recall:         Win 0.59        Draw 0.00       Lose 0.62
F1:     Win 0.56        Draw 0.00       Lose 0.44
```

Figure Q4: Beta TrueSkill metrics for Germany Bundesliga 2018-2019

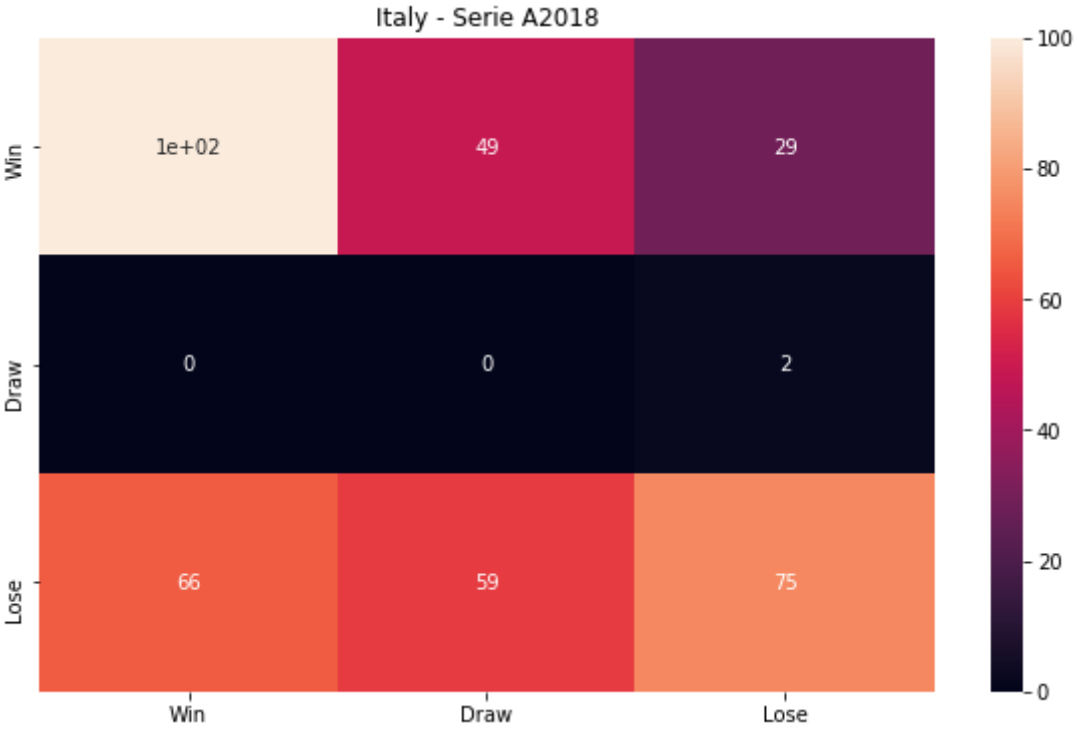

Figure Q5: Beta TrueSkill confusion matrix for Italy Serie A 2018-2019

```
Accuracy: 46.05%
MAE: 0.25
RMSE: 0.29
Precision:      Win 0.56        Draw 0.00       Lose 0.38
Recall:         Win 0.60        Draw 0.00       Lose 0.71
F1:     Win 0.58        Draw 0.00       Lose 0.49
```

Figure Q6: Beta TrueSkill metrics for Italy Serie A 2018-2019

Appendix R: TrueSkill Mu and Sigma

Using different combinations for Mu and Sigma produced a graph that can be seen in Figure R1.

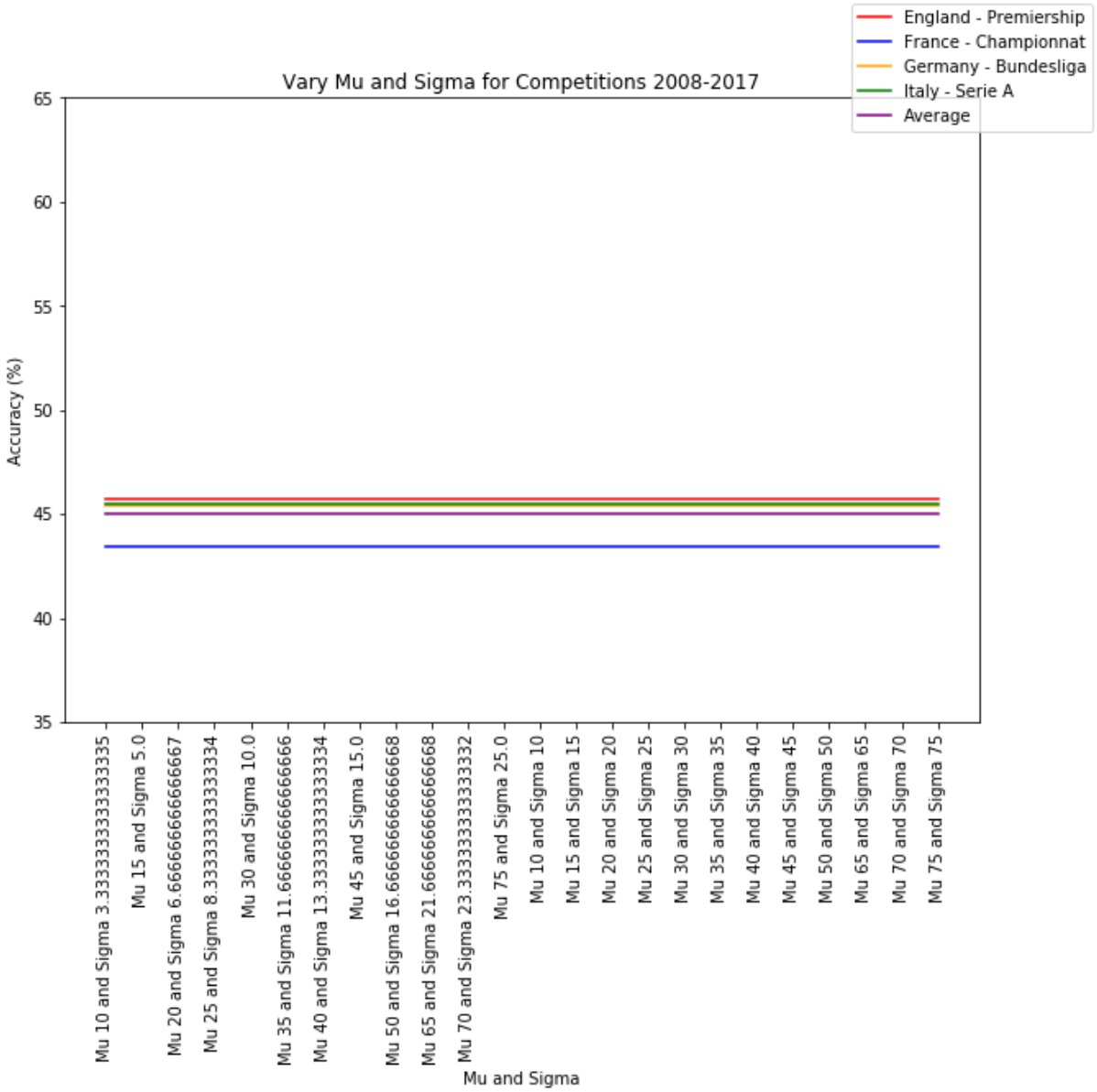

Figure R1: Overall accuracy for train set (y) varying depending on Mu and Sigma (x)

As evident from Figure R1, changing Mu and Sigma does not impact the overall prediction accuracy of football matches. This suggests that regardless of the rating with which the teams start or the initial standard deviation, after training the TrueSkill model on 10 years' worth of data, the system recalculated every teams' rating with a high confidence level.

Appendix S: TrueSkill Grid Search Multiple Parameters

From Figure S1, it appears that when a large enough BETA is used, the Home Advantage is increased, so is the Accuracy. However, for reasonable values of BETA for the given problem, the Home Advantage always results in the same best value. This confirms that the values that were chosen in this section were correct.

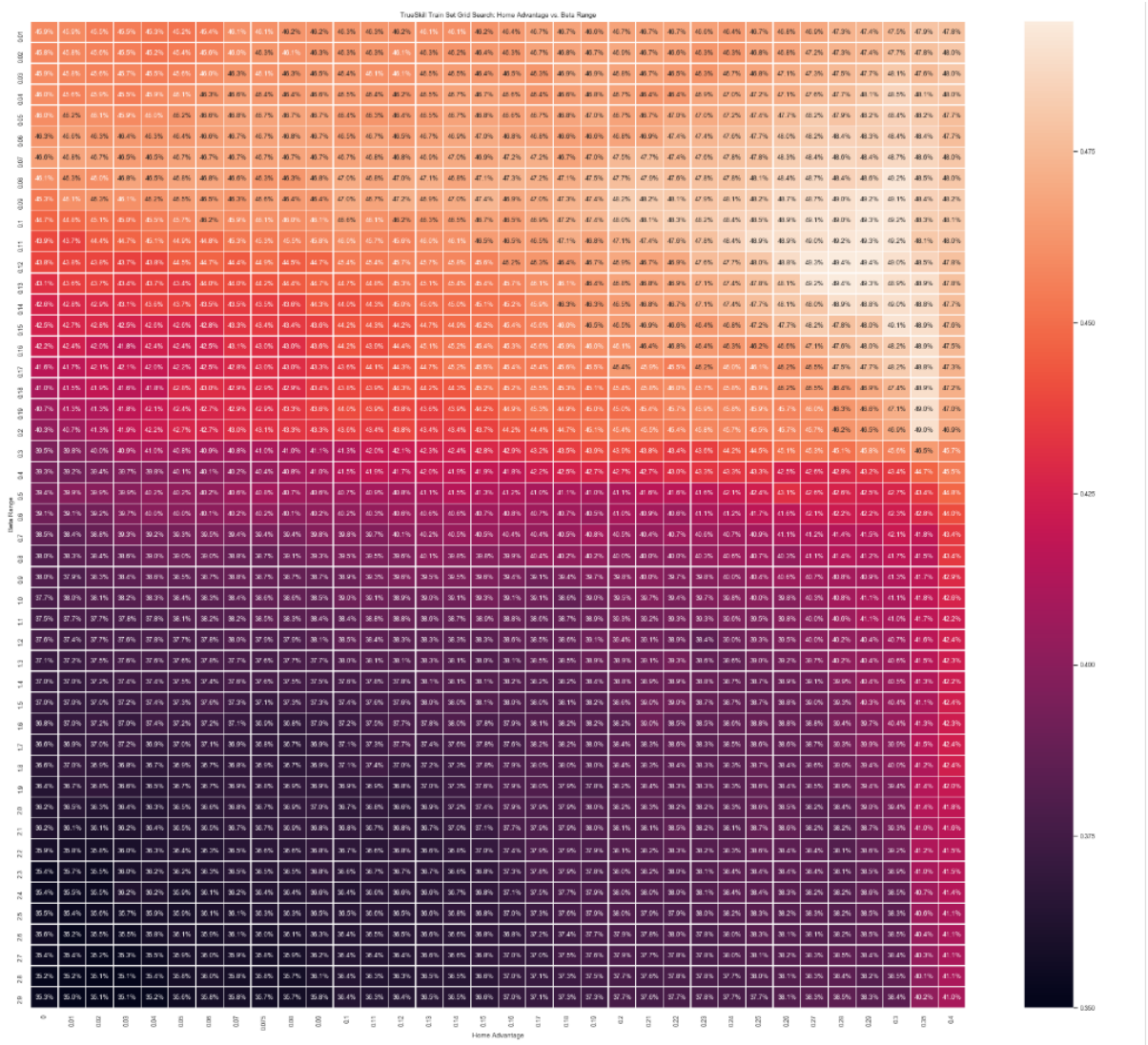

Figure S1: TrueSkill accuracy grid search of BETA and Home Advantage using the test set

The gridsearch using only the test set (so 2018 - 2019 tournaments) is presented in Figure S1. As can be seen from comparing the two grid searches, when taking into account a larger number of seasons, the interpolation between adjacent parameter values is not as smooth. However, for a single season, it seems that there is a clear (and obvious) pattern to the changes in accuracy based on parameter variation.

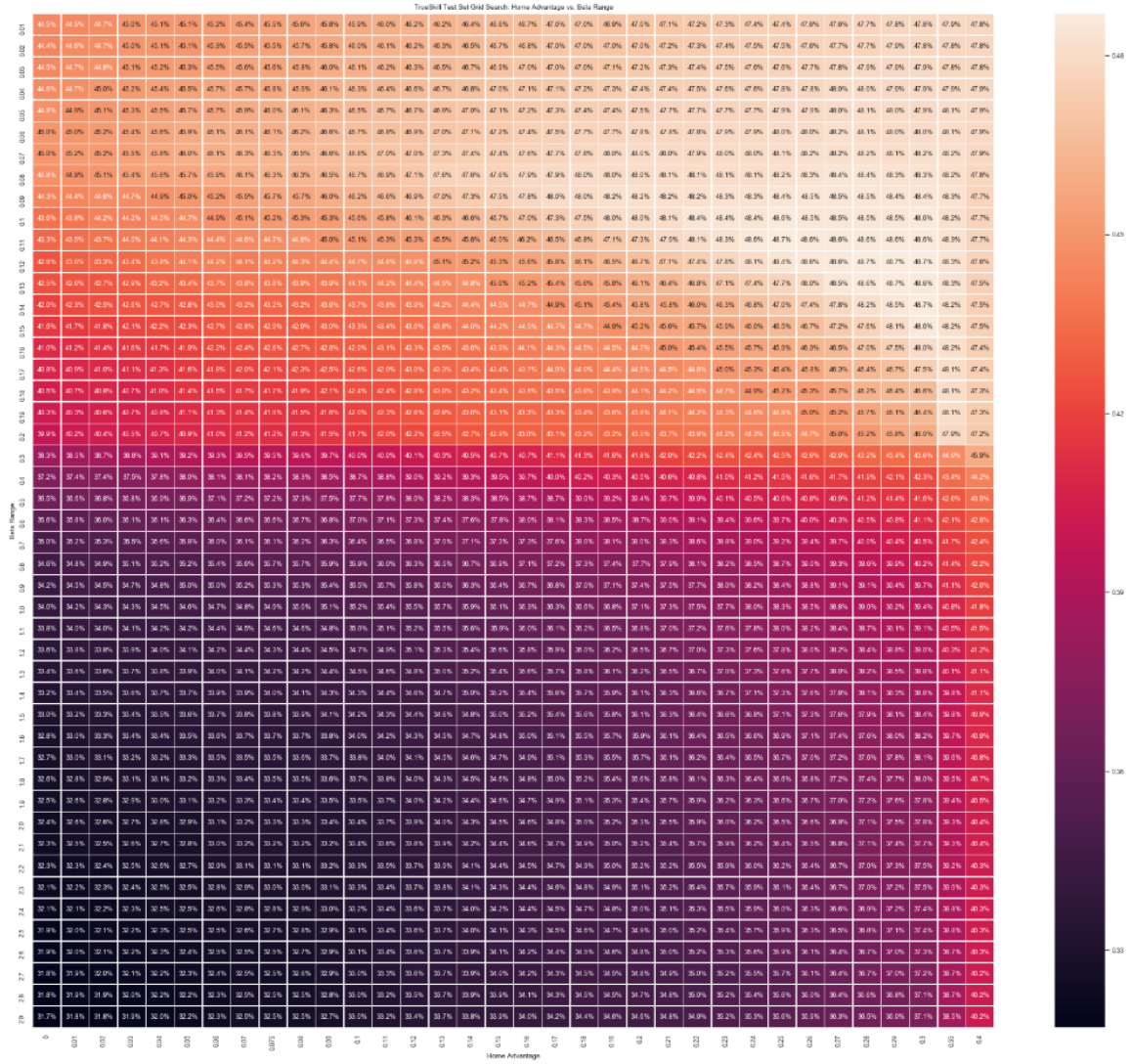

Figure S2: TrueSkill accuracy grid search of BETA and Home Advantage using the train set

Hyperparameters that are optimal for individual competitions are also selected using a grid search. The results of these are presented in Figures S3 through S6. From these, it is clear that for the England Premiership, the best values for Home Advantage and Beta are 0.14 and 0.29 respectively. For France Championnat, this would be 0.15 and 0.3. Next, for Germany Bundesliga, optimal Home Advantage is 0.12, while for Beta it is 0.26. Lastly, for Italy Serie A these values are 0.11 and 0.26.

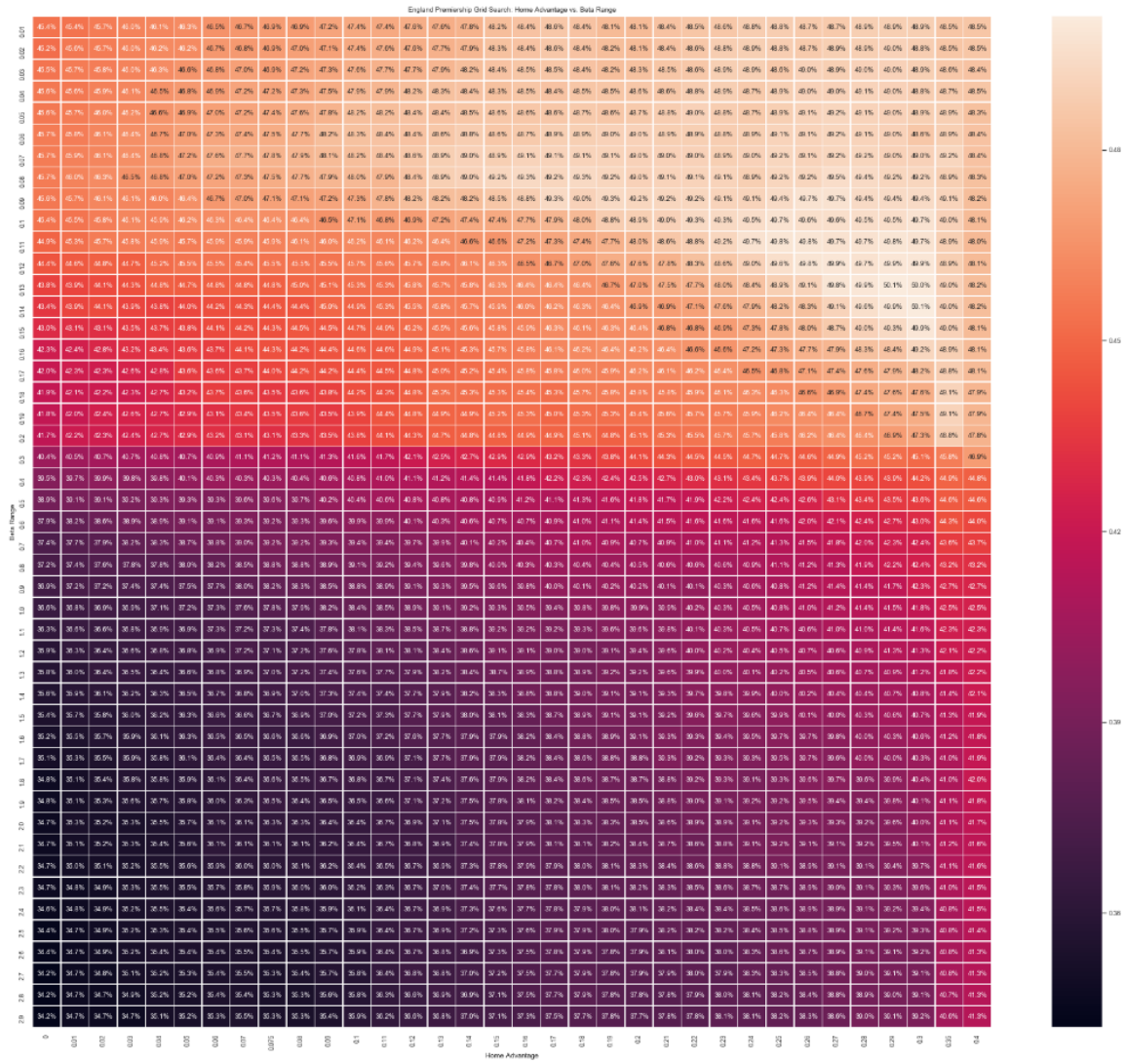

Figure S3: England Premiership grid search for TrueSkill Home Advantage and Beta

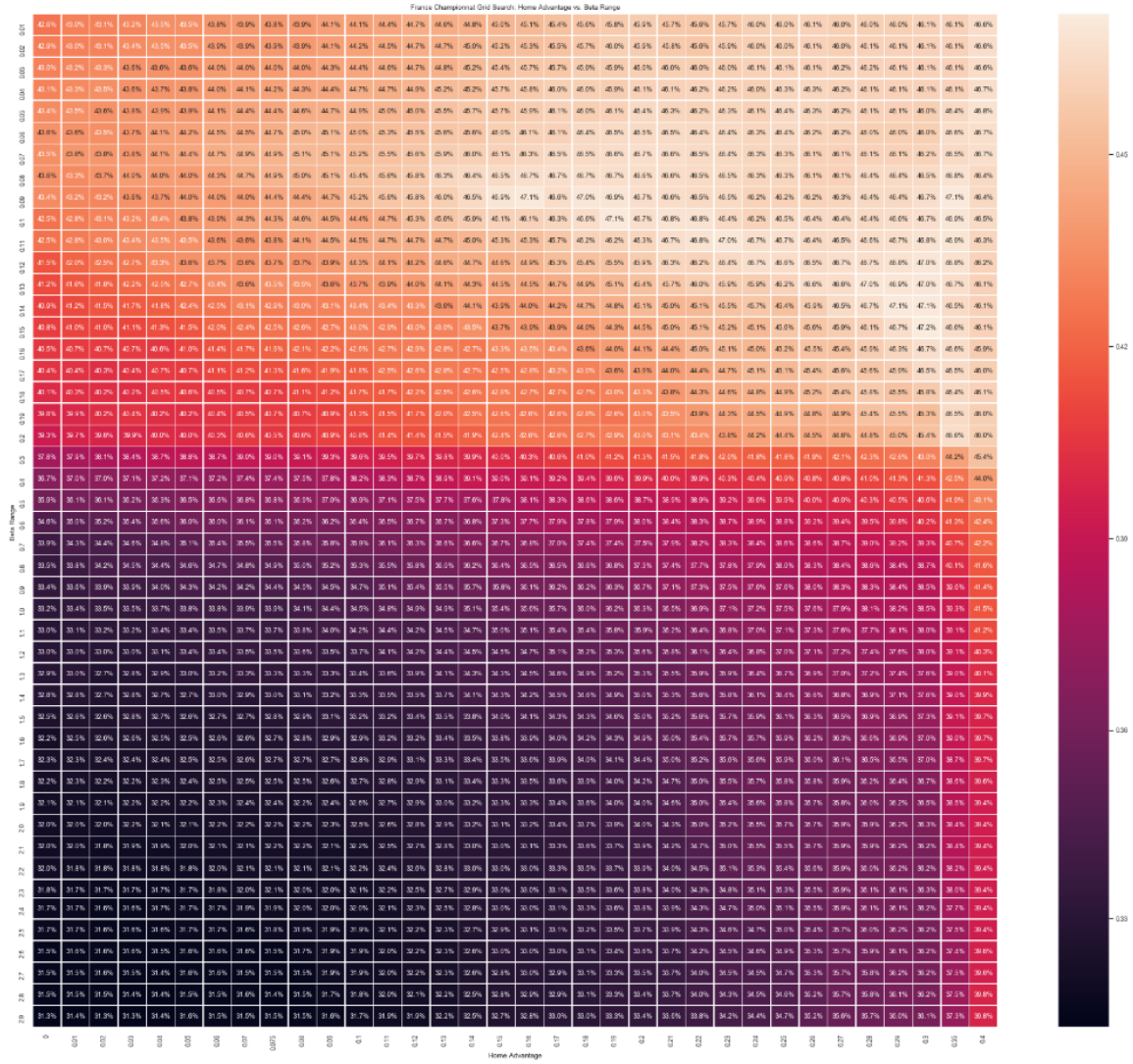

Figure S4: France Championnat grid search for TrueSkill Home Advantage and Beta

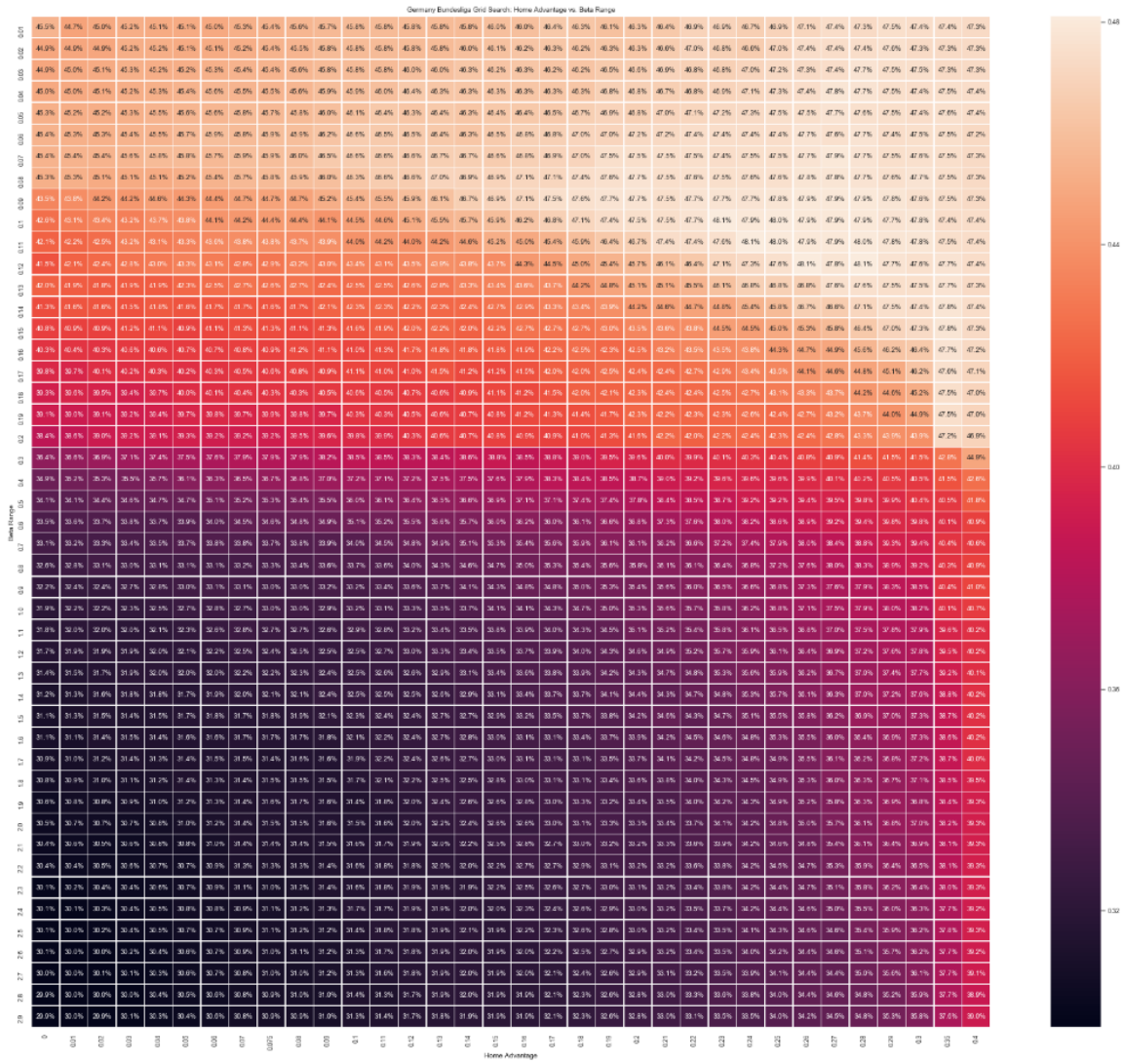

Figure S5: Germany Bundesliga grid search for TrueSkill Home Advantage and Beta

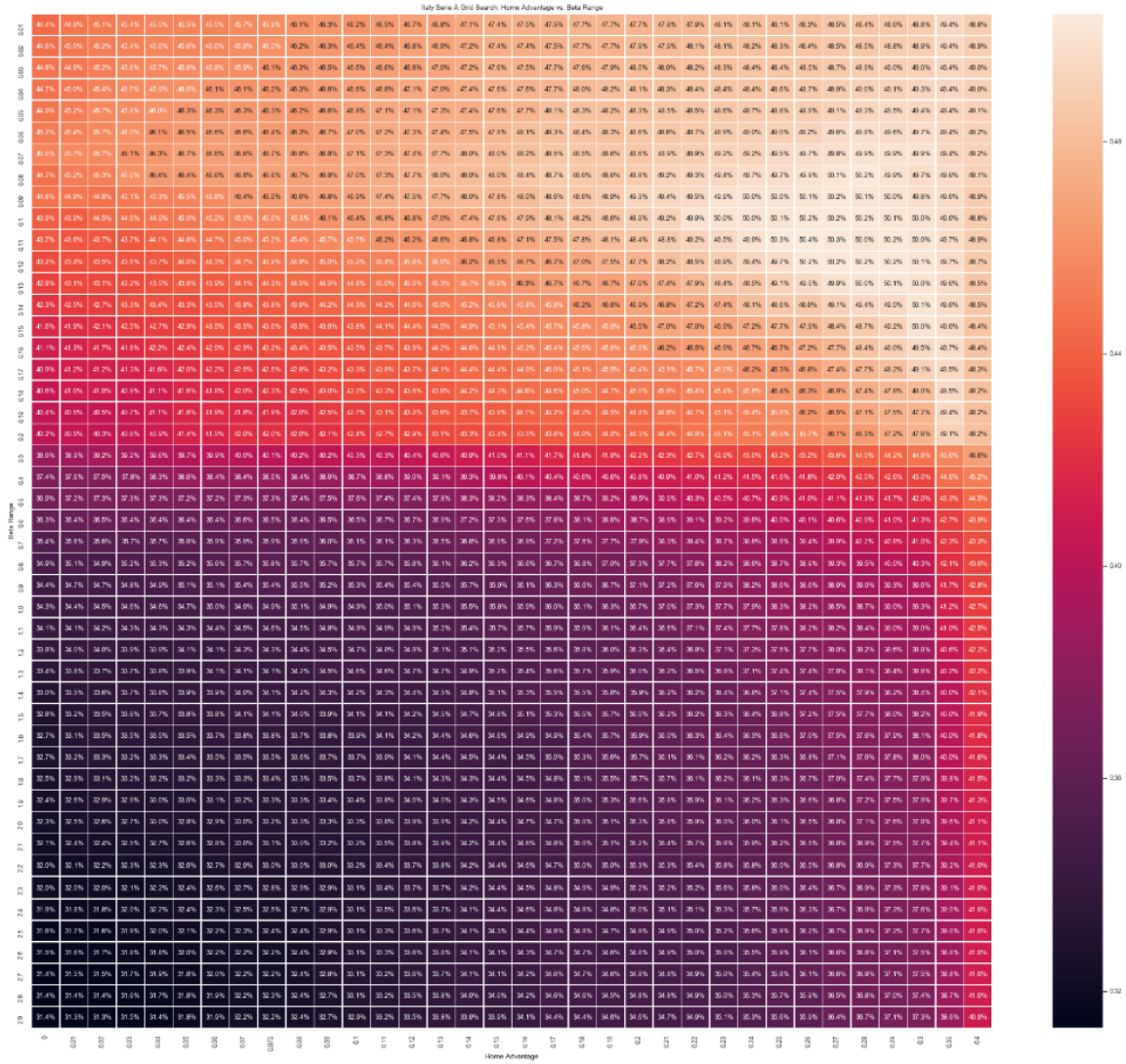

Figure S6: Italy Serie A grid search for TrueSkill Home Advantage and Beta

Appendix T: Average TrueSkill vs. Competition-based TrueSkill

Figures T1 through T8 present the metrics for the Average TrueSkill (TrueSkill model that uses parameters that yield the highest average accuracy when training and using a single model for all 4 competitions). It uses the following parameters:

- Beta: 0.07
- Home Advantage: 0.325
- Draw Probability: 0.1
- Tau 0.08334
- Mu 25.0
- Sigma 8.3334

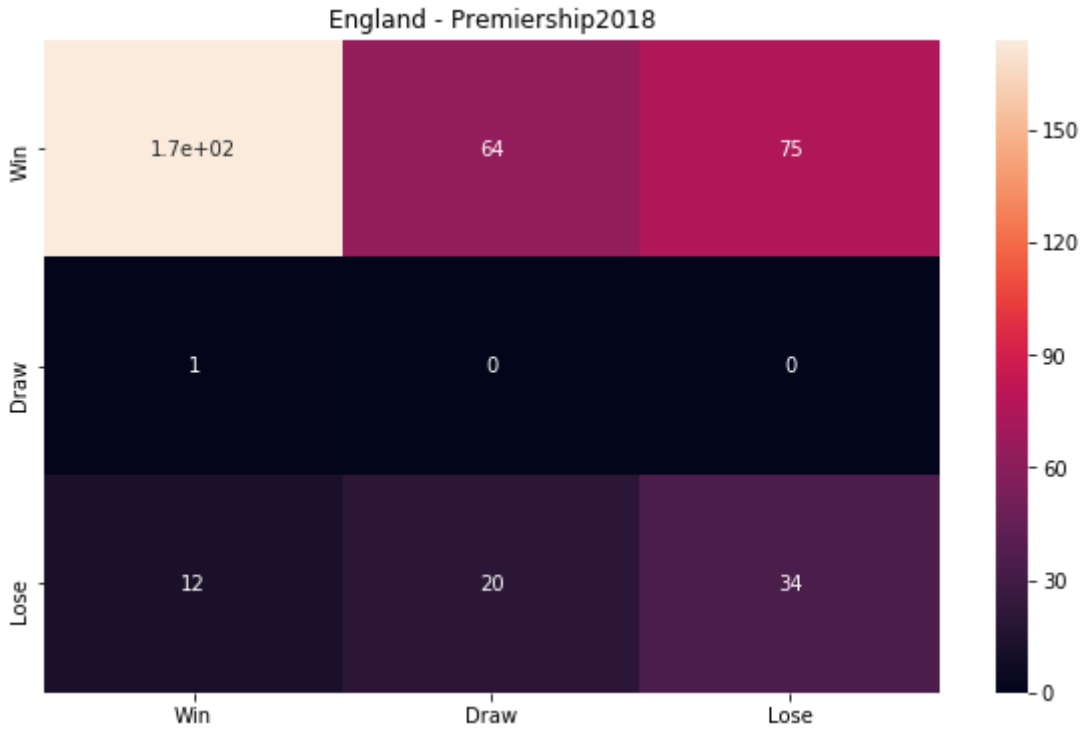

Figure T1: Overall TrueSkill confusion matrix for England Premiership 2018 - 2019

```
Accuracy: 54.74%
MAE: 0.52
RMSE: 0.63
Precision:      Win 0.93        Draw 0.00       Lose 0.31
Recall:         Win 0.56        Draw 0.00       Lose 0.52
F1:     Win 0.70        Draw 0.00       Lose 0.39
```

Figure T2: Overall TrueSkill metrics for England Premiership 2018 - 2019

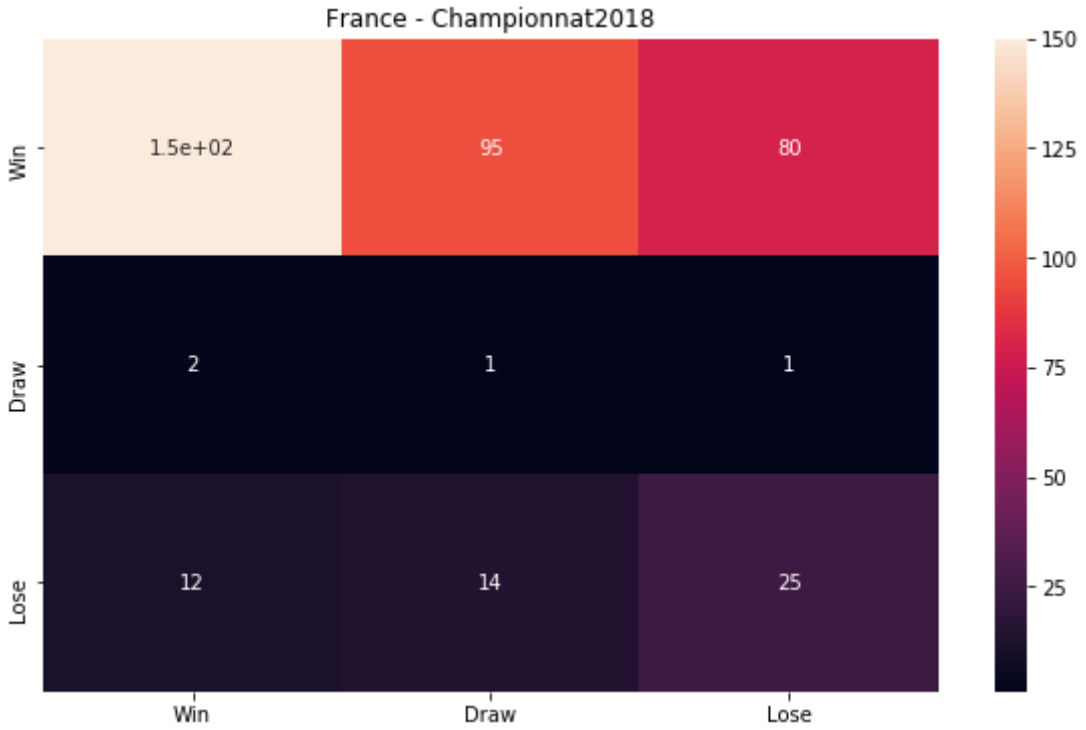

Figure T3: Overall TrueSkill confusion matrix for France Championnat 2018 - 2019

```
Accuracy: 46.32%
MAE: 0.58
RMSE: 0.69
Precision:      Win 0.91        Draw 0.01       Lose 0.24
Recall:         Win 0.46        Draw 0.25       Lose 0.49
F1:     Win 0.61        Draw 0.02       Lose 0.32
```

Figure T4: Overall TrueSkill metrics for France Championnat 2018 - 2019

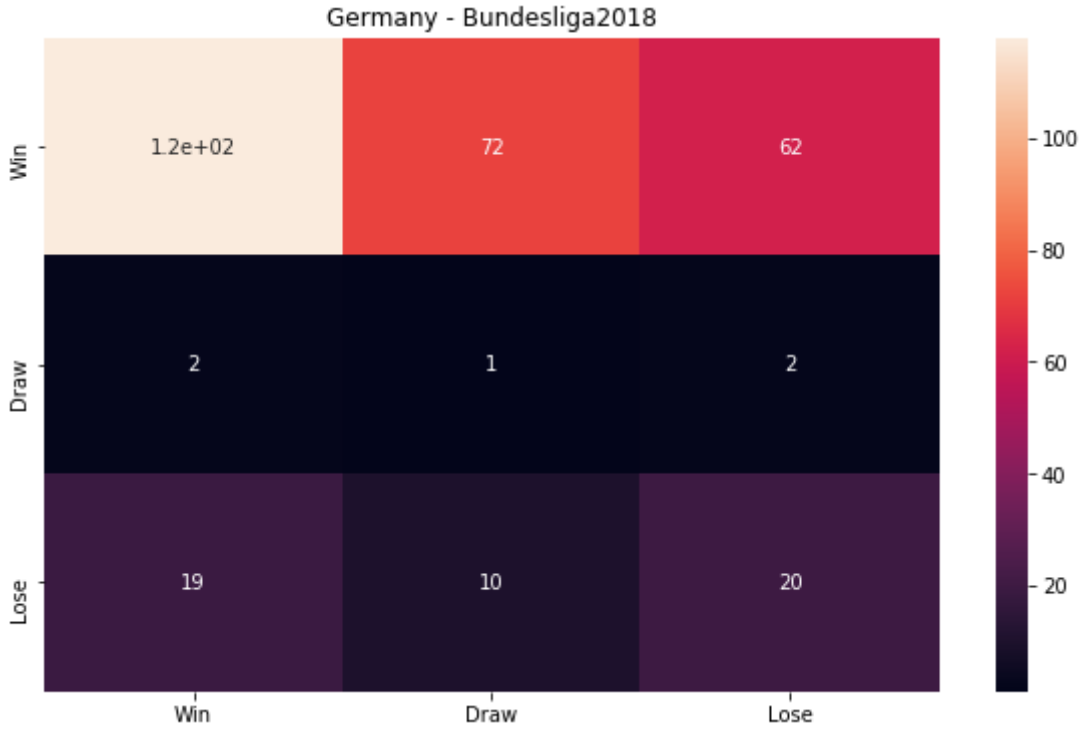

Figure T5: Overall TrueSkill confusion matrix for Germany Bundesliga 2018 - 2019

```
Accuracy: 45.42%
MAE: 0.57
RMSE: 0.67
Precision:      Win 0.85        Draw 0.01       Lose 0.24
Recall:         Win 0.47        Draw 0.20       Lose 0.41
F1:     Win 0.60        Draw 0.02       Lose 0.30
```

Figure T6: Overall TrueSkill metrics for Germany Bundesliga 2018 - 2019

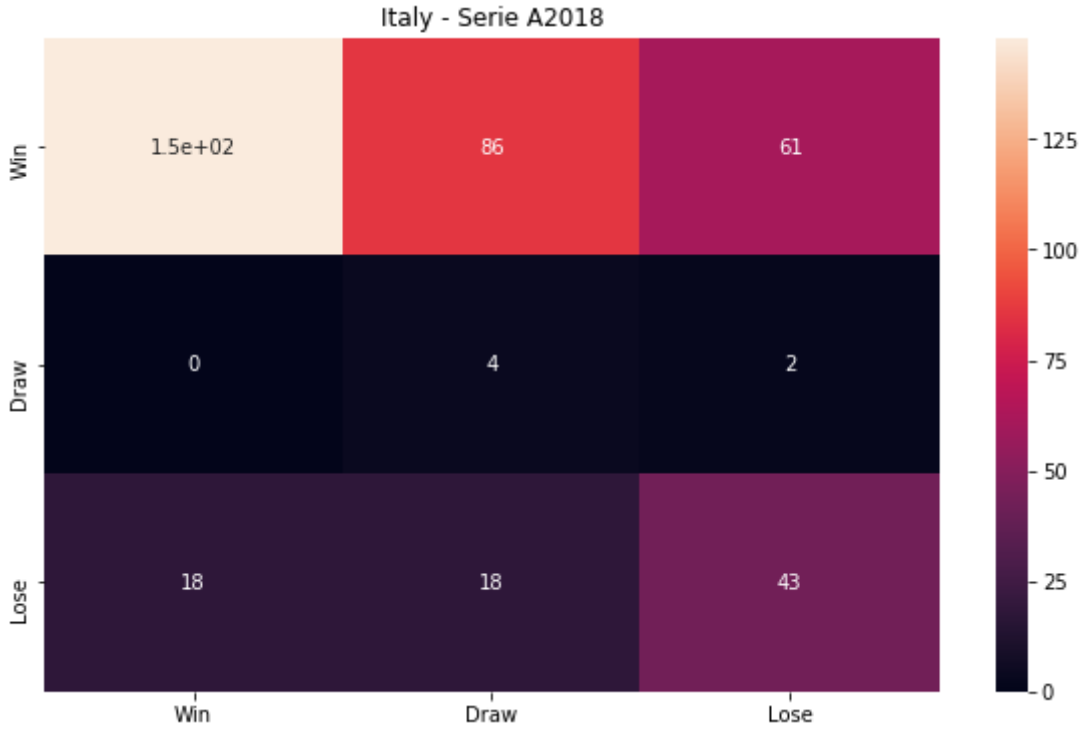

Figure T7: Overall TrueSkill confusion matrix for Italy Serie A 2018 - 2019

```
Accuracy: 51.32%
MAE: 0.56
RMSE: 0.66
Precision:      Win 0.89        Draw 0.04       Lose 0.41
Recall:         Win 0.50        Draw 0.67       Lose 0.54
F1:     Win 0.64        Draw 0.07       Lose 0.46
```

Figure T8: Average TrueSkill metrics for Italy Serie A 2018 - 2019

Next, the performance of the TrueSkill models with per-competition hyperparameters is presented. These models have the following parameters:

- England Premiership
    - Beta: 0.29
    - Home Advantage: 0.14
    - Draw Probability: 0.1
    - Tau: 0.08334
    - Mu: 25.0
    - Sigma: 8.3334
- France Championnat
    - Beta: 0.3
    - Home Advantage: 0.15
    - Draw Probability: 0.1

    - Tau: 0.08334
    - Mu: 25.0
    - Sigma: 8.3334
- Germany Bundesliga
    - Beta: 0.26
    - Home Advantage: 0.12
    - Draw Probability: 0.1
    - Tau: 0.08334
    - Mu: 25.0
    - Sigma: 8.3334
- Italy Serie A
    - Beta: 0.26
    - Home Advantage: 0.11
    - Draw Probability: 0.1
    - Tau: 0.08334
    - Mu: 25.0
    - Sigma: 8.3334

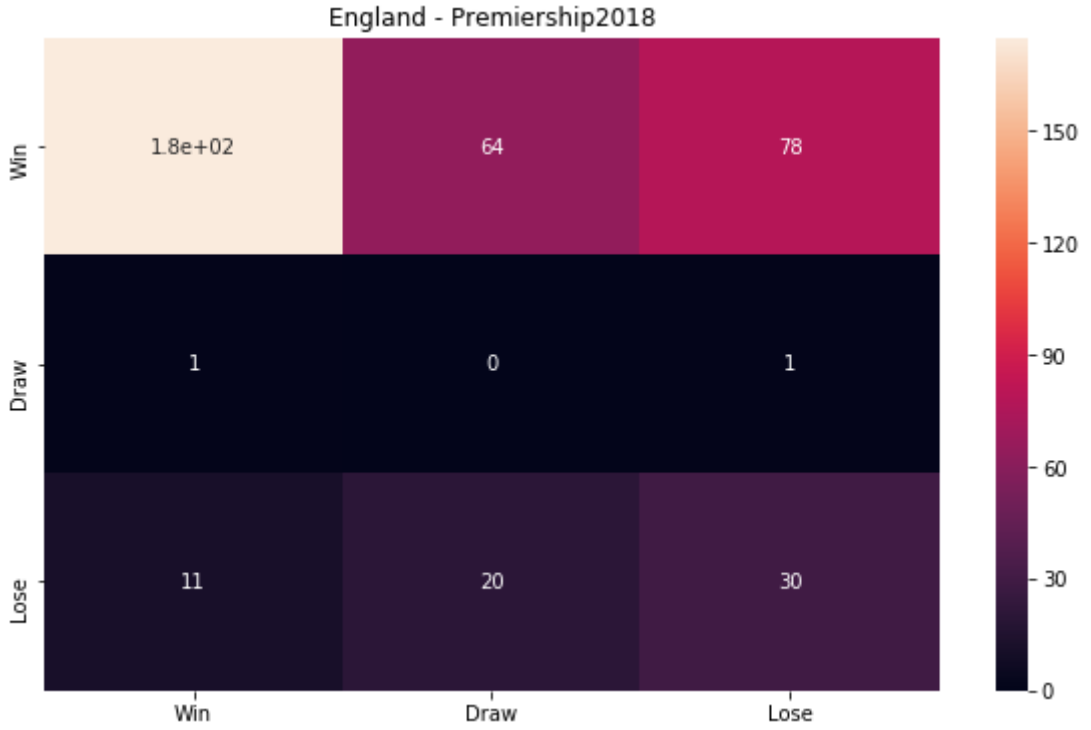

Figure T9: Per-competition TrueSkill confusion matrix for England Premiership 2018 - 2019

```
Accuracy: 53.95%
MAE: 0.52
RMSE: 0.63
Precision:      Win 0.94        Draw 0.00       Lose 0.28
Recall:         Win 0.55        Draw 0.00       Lose 0.49
F1:     Win 0.69        Draw 0.00       Lose 0.35
```

Figure T10: Per-competition TrueSkill metrics for England Premiership 2018 - 2019

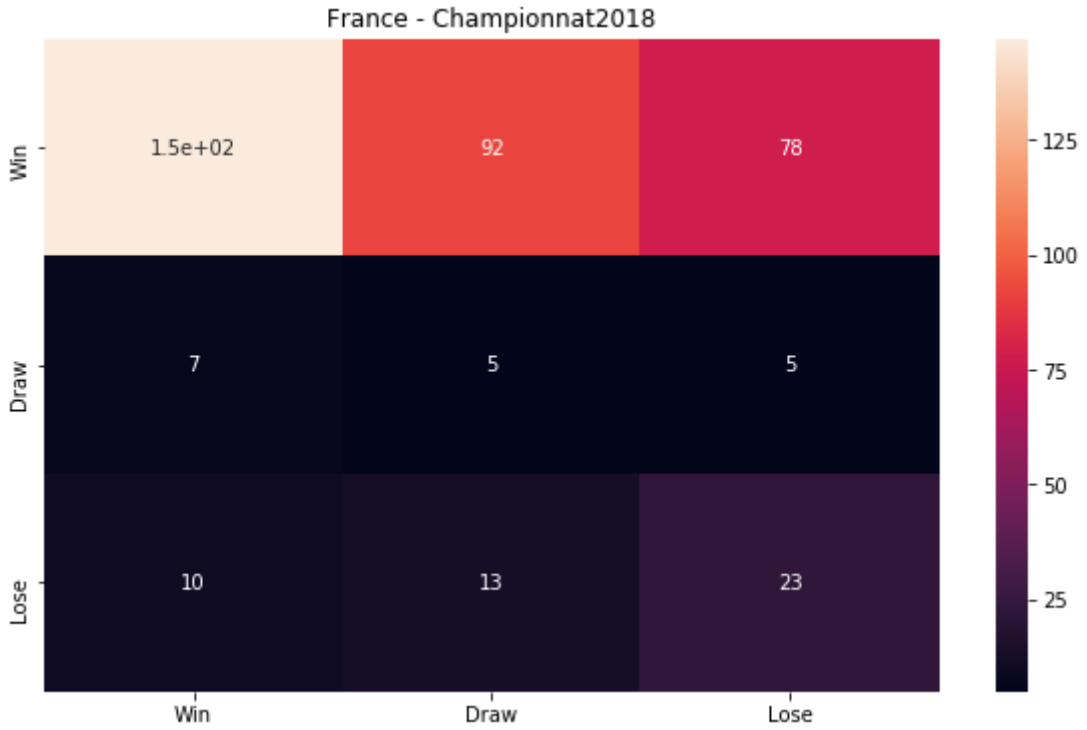

Figure T11: Per-competition TrueSkill confusion matrix for France Championnat 2018 - 2019

```
Accuracy: 46.05%
MAE: 0.58
RMSE: 0.69
Precision:      Win 0.90        Draw 0.05       Lose 0.22
Recall:         Win 0.46        Draw 0.29       Lose 0.50
F1:     Win 0.61        Draw 0.08       Lose 0.30
```

Figure T12: Per-competition TrueSkill metrics for France Championnat 2018 - 2019

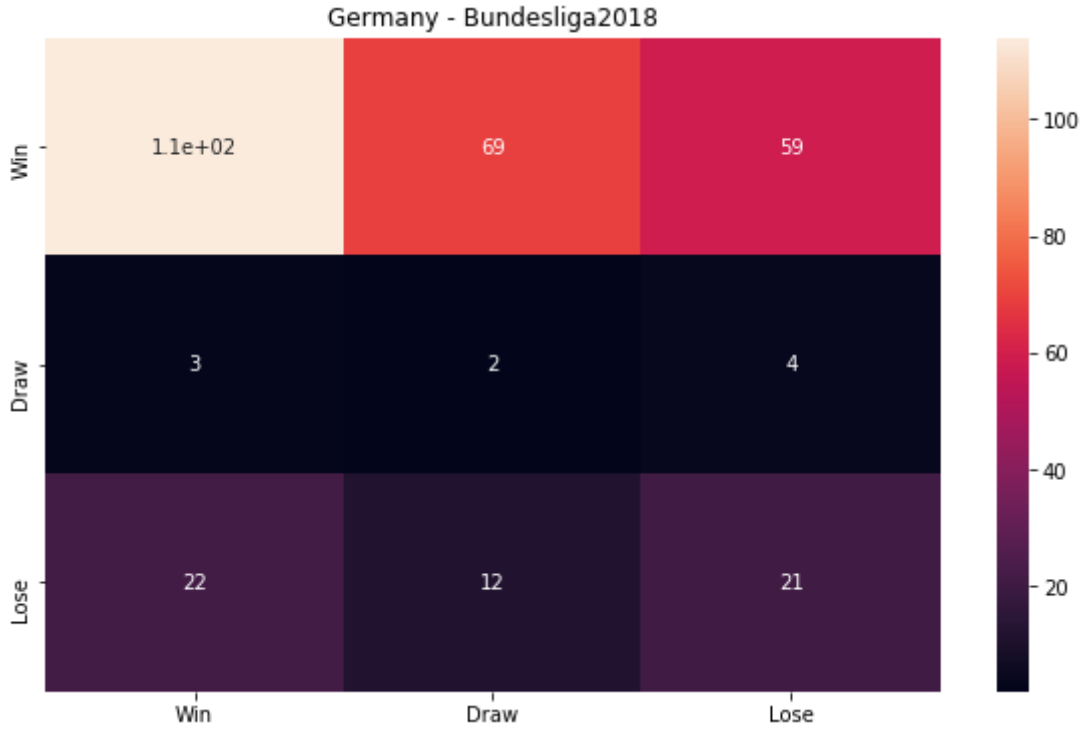

Figure T13: Per-competition TrueSkill confusion matrix for Germany Bundesliga 2018 - 2019

```
Accuracy: 44.77%
MAE: 0.57
RMSE: 0.67
Precision:      Win 0.82        Draw 0.02       Lose 0.25
Recall:         Win 0.47        Draw 0.22       Lose 0.38
F1:     Win 0.60        Draw 0.04       Lose 0.30
```

Figure T14: Per-competition TrueSkill metrics for Germany Bundesliga 2018 - 2019

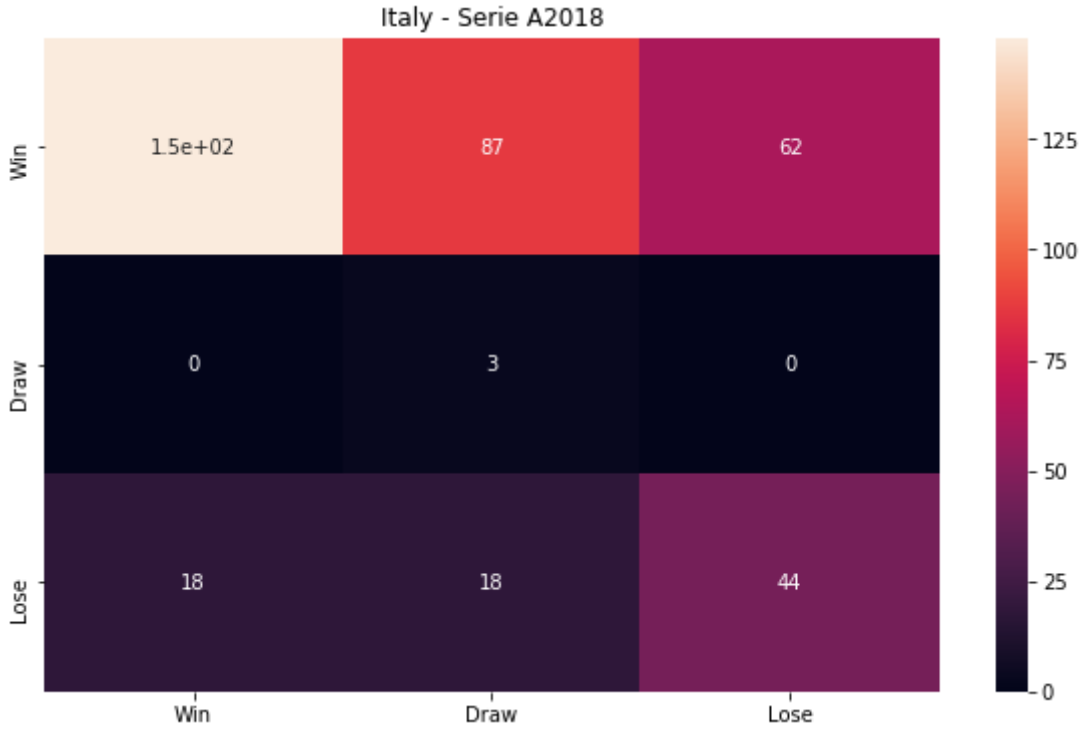

Figure T15: Per-competition TrueSkill confusion matrix for Italy Serie A 2018 - 2019

```
Accuracy: 51.32%
MAE: 0.56
RMSE: 0.66
Precision:      Win 0.89        Draw 0.03       Lose 0.42
Recall:         Win 0.50        Draw 1.00       Lose 0.55
F1:     Win 0.64        Draw 0.05       Lose 0.47
```

Figure T16: Per-competition TrueSkill metrics for Italy Serie A 2018 - 2019

The differences in performance between Average TrueSkill and Competition-based TrueSkill are summarised in Table T1. Clearly, Competition-based TrueSkill produces better results than Average TrueSkill across all examined competitions, yielding +0.38% higher accuracy on average.

Competition	Average TrueSkill	Competition-based TrueSkill	Difference
England Premiership	50.16%	50.29%	+0.13%
France Championnat	46.51%	47.04%	+0.53%
Germany Bundesliga	47.66%	47.74%	+0.08%

Italy Serie A	49.56%	50.35%	+0.79%

Table T1: Competition accuracy comparison of Average TrueSkill vs. Competition-based TrueSkill for 2008 - 2017 tournaments

Appendix U: Neural Network Final Feature Set

1) Bet365 home odds,
2) bet365 draw odds,
3) bet365 away odds,
4) home team win rate home,
5) home team win rate away,
6) home team draw rate home,
7) home team draw rate away,
8) home team lose rate home,
9) home team lose rate away,
10) home team average goals scored home,
11) home team average goals conceded home,
12) home team average goals scored away,
13) home team average goals conceded away,
14) away team win rate home,
15) away team win rate away,
16) away team draw rate home,
17) away team draw rate away,
18) away team lose rate home,
19) away team lose rate away,
20) away team average goals scored home,
21) away team average goals conceded home,
22) away team average goals scored away,
23) away team average goals conceded away,
24) home team number of games won at home in the last 20 games,
25) home team number of games drawn at home in the last 20 games,
26) home team number of games lost at home in the last 20 games,
27) home team number of games won away in the last 20 games,
28) home team number of games drawn away in the last 20 games,
29) home team number of games lost away in the last 20 games,
30) away team number of games won at home in the last 20 games,
31) away team number of games drawn at home in the last 20 games,
32) away team number of games lost at home in the last 20 games,
33) away team number of games won away in the last 20 games,
34) away team number of games drawn away in the last 20 games,
35) away team number of games lost away in the last 20 games

Appendix V: Learning Rate Optimization for Neural Network

In Figure V1, the results of running such grid search with learning rates of 0.001, 0.005, 0.01, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35 and 0.4 are presented.

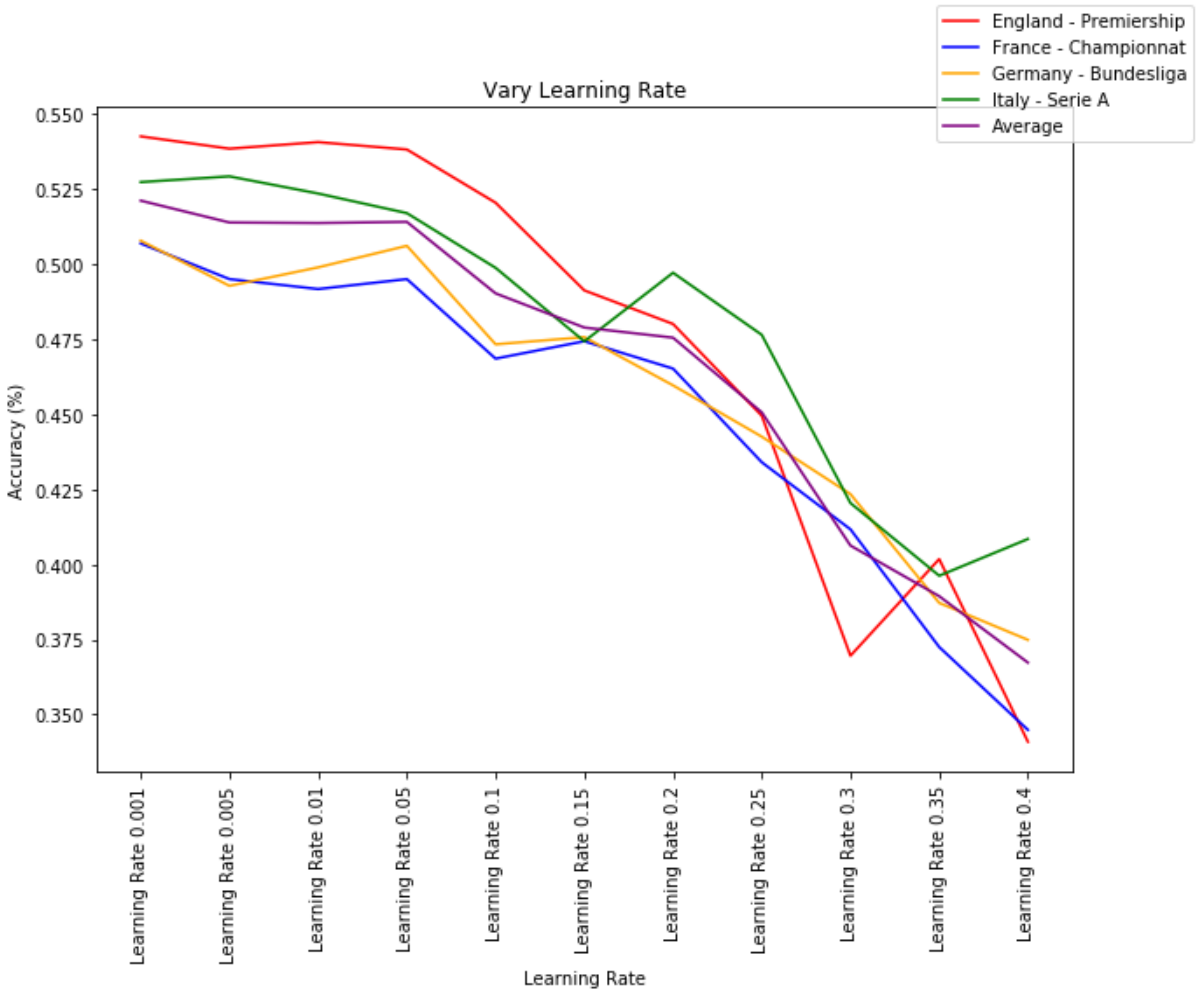

Figure V1: Competition accuracy for train set (y) varying depending on learning rate (x)

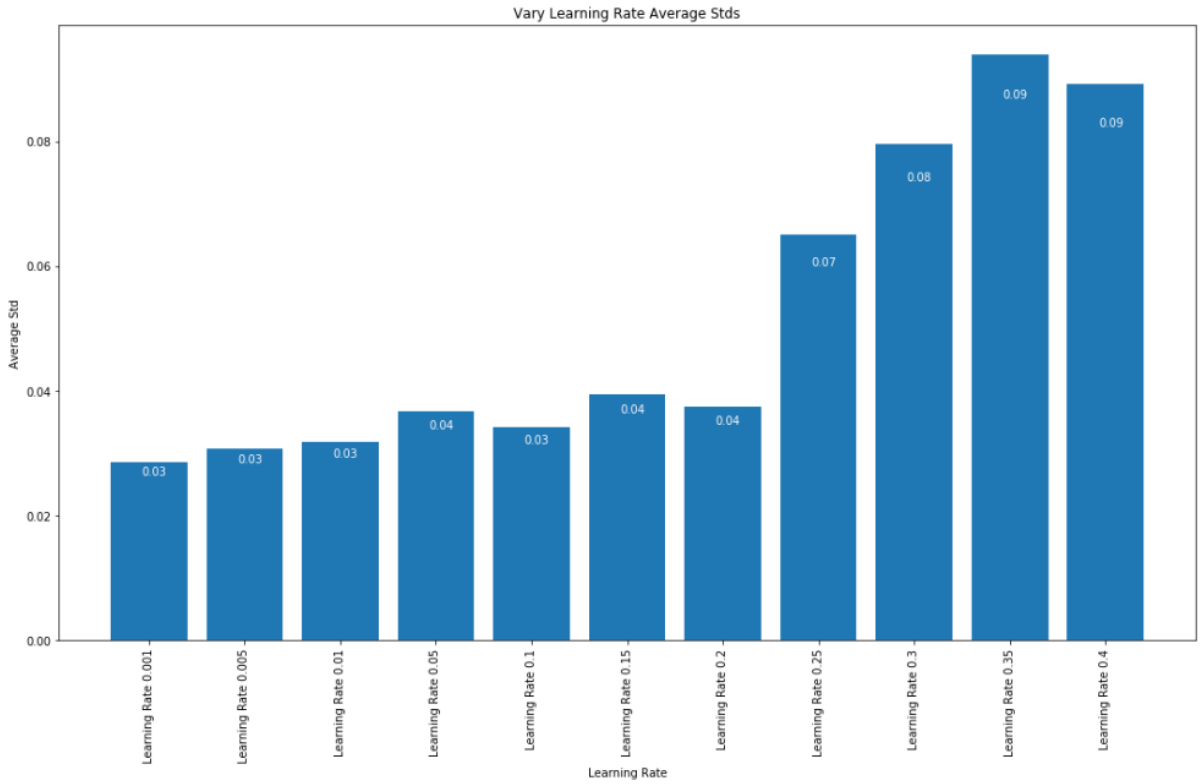

Figure V2: Overall accuracy std for train set (y) varying depending on learning rate (x)

Based on Figures V1 and V2, it appears that smaller learning rates result in higher accuracy. It can also be seen in Figure V2 that it seems with the higher learning rate, the consistency of the models starts to decline.The loss curves that can be obtained after training the models are considered to determine whether a particular learning rate is suitable for the problem.

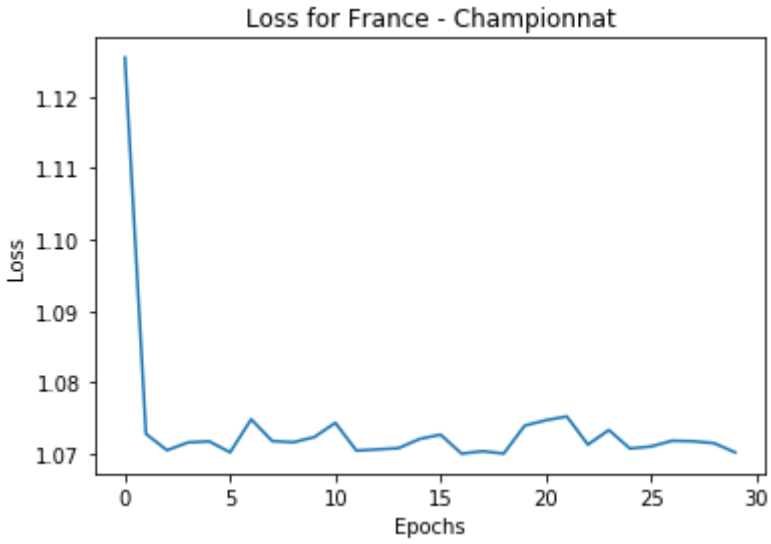

Figure V3: Loss history graph for training data for England Premiership with 0.1 learning rate for

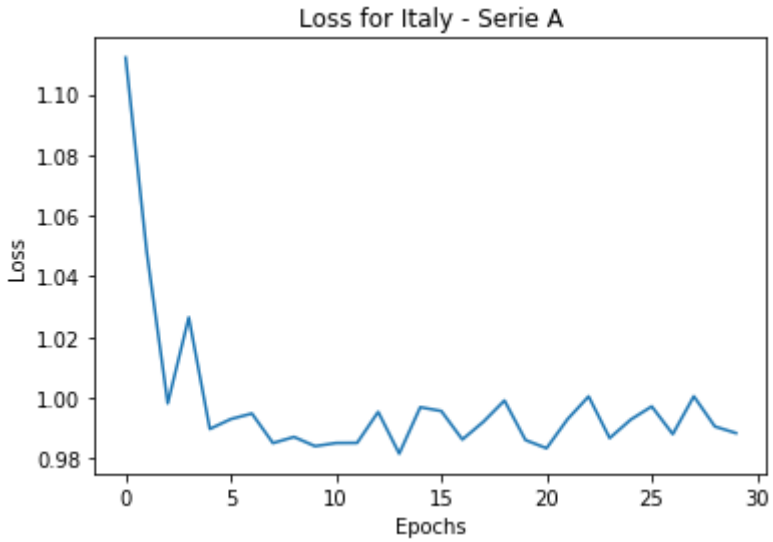

Figure V4: Loss history graph for training data for Italy Serie A with 0.1 learning rate

Based on the results in Figures V3 and V4, the learning rate of 0.1 is too high, which causes spikes in the loss during the training phase. It is also worth examining the accuracies and the standard deviation of these accuracies for each competition. To do so, the models are trained 50 times before calculating the deviation and average accuracy for each competition.

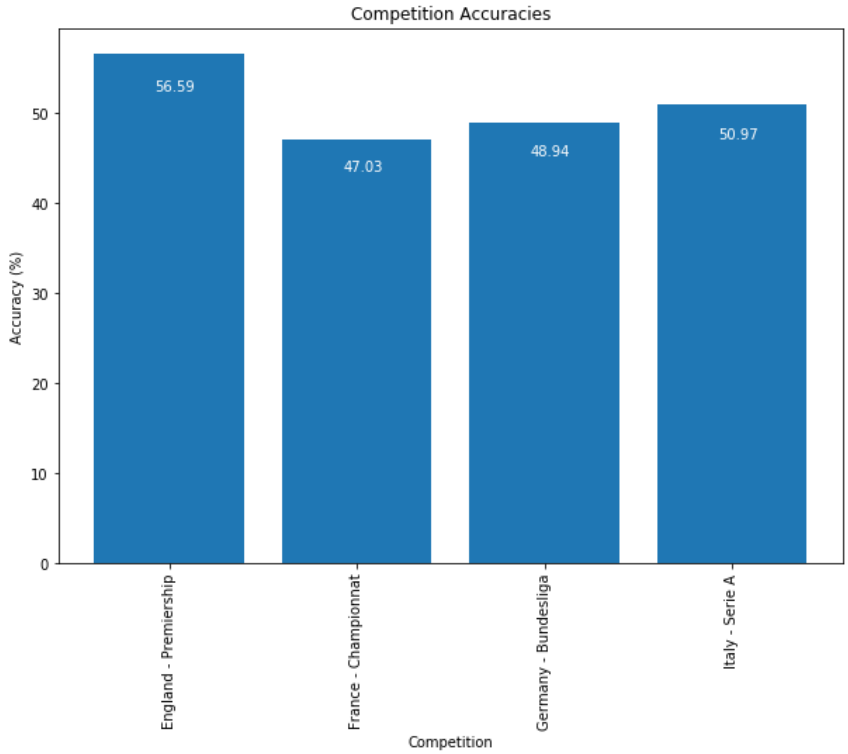

Figure V5: Overall competition accuracy with learning rate of 0.1 for tournaments 2018 - 2019

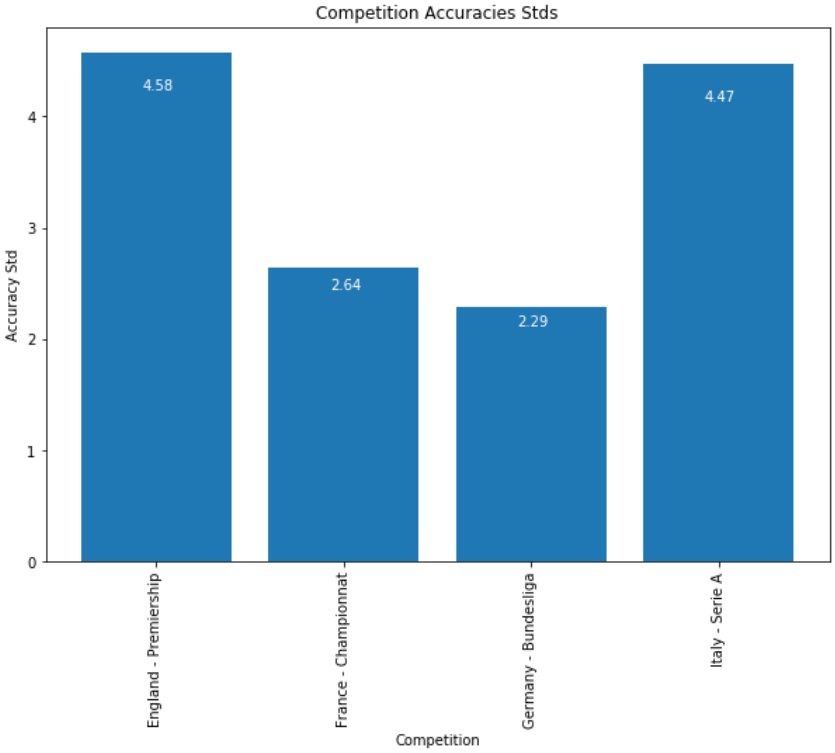

Figure V6: Overall competition accuracy std with learning rate of 0.1 for tournaments 2018 - 2019

All competitions have a relatively high amount of uncertainty, with England Premiership peaking with 4.58%. To confirm that a smaller learning rate may be beneficial for the models (which is what Figure V7 suggests), similar graphs with learning rate of 0.001 are obtained.

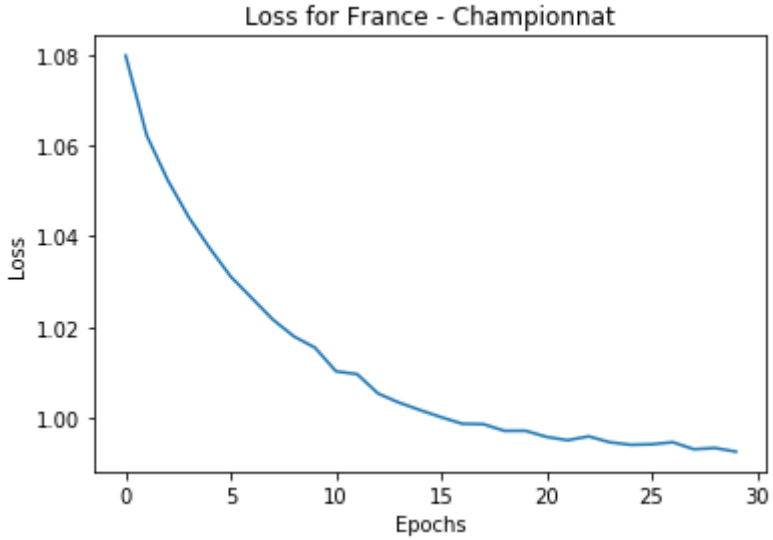

Figure V7: Loss history graph for training England Premiership with 0.001 learning rate

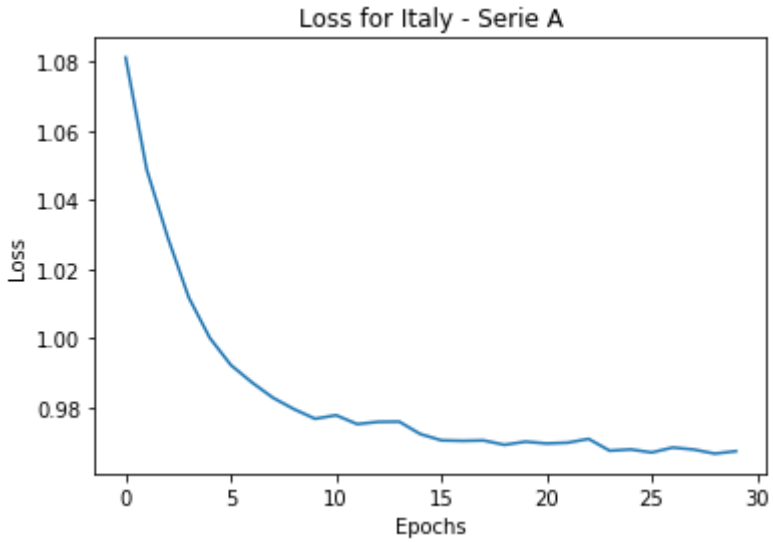

Figure V8: Loss history graph for training Italy Serie A with 0.001 learning rate

Updated loss graphs with 0.001 learning rate are presented in Figures V7 and V8. It seems that the lower learning rate leads to more stable training and better model generalisation.

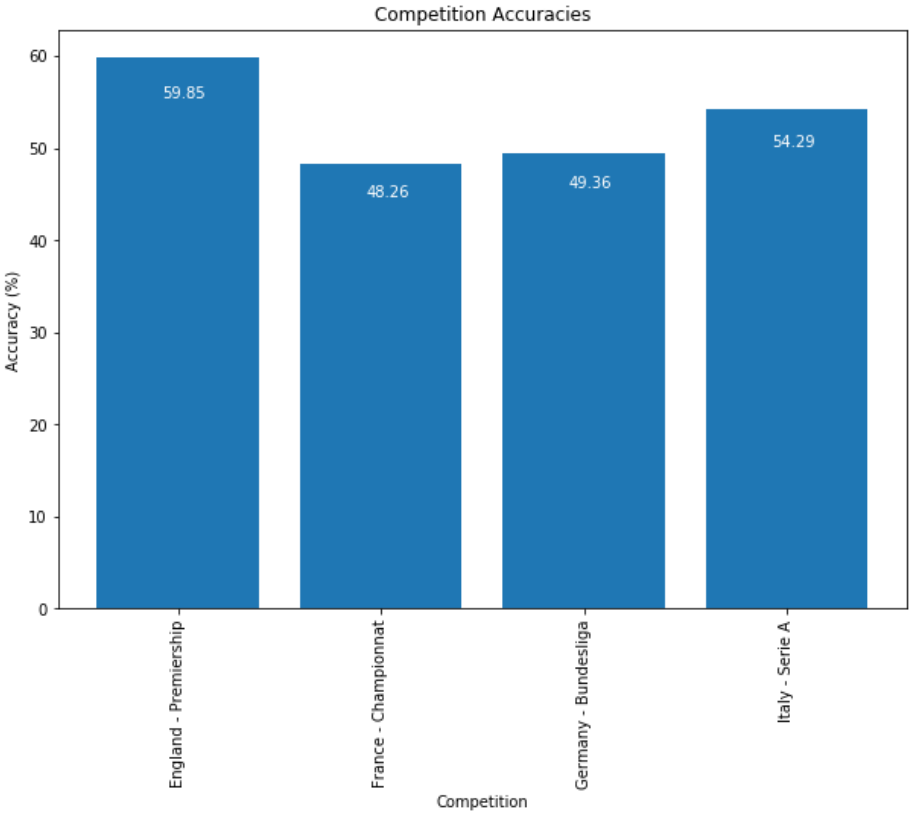

Figure V9: Overall competition accuracy with learning rate of 0.001 for tournaments 2018 - 2019

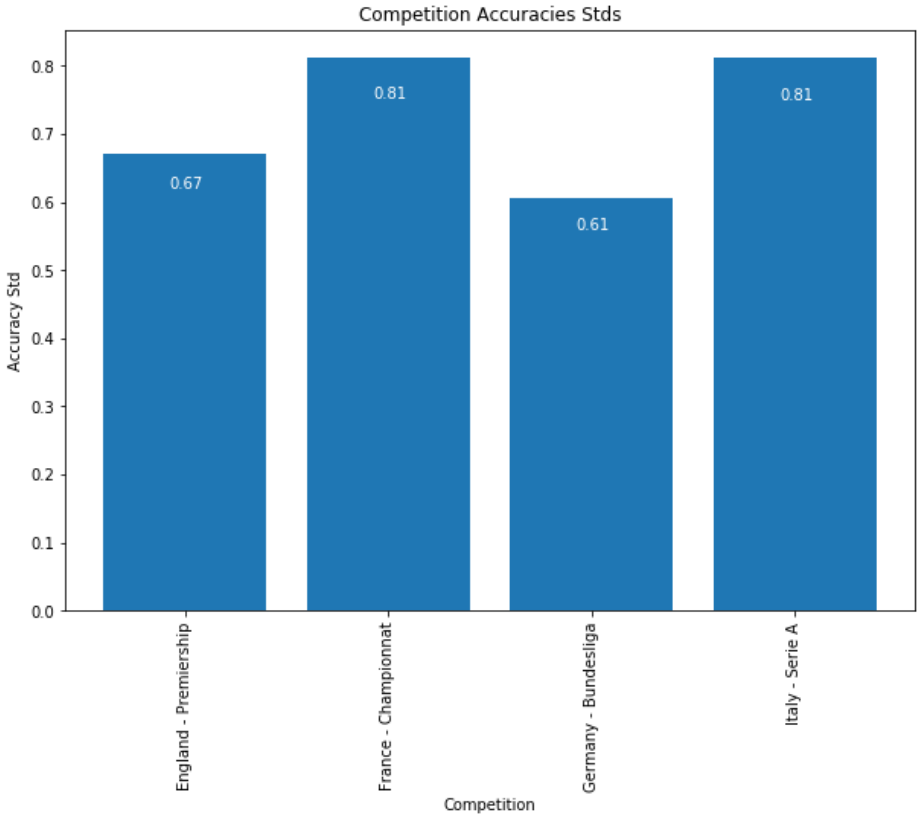

Figure V10: Overall competition accuracy std with learning rate of 0.001 for tournaments 2018 - 2019

The standard deviation of the models accuracy has gone down significantly, producing relatively stable models with the learning rate 0.001. It also seems that the accuracy has increased for all 2018 - 2019 tournaments. So, this learning rate is considered as best overall.

So far, the learning rate has been adjusted based on the average accuracy and standard deviation. However, the learning rate parameter can also be tuned on a per-competition basis. Throughout the model optimising process, both sets of parameters (global and per-competition) are found.

Next, the gridsearch approach was applied to England Premiership alone.

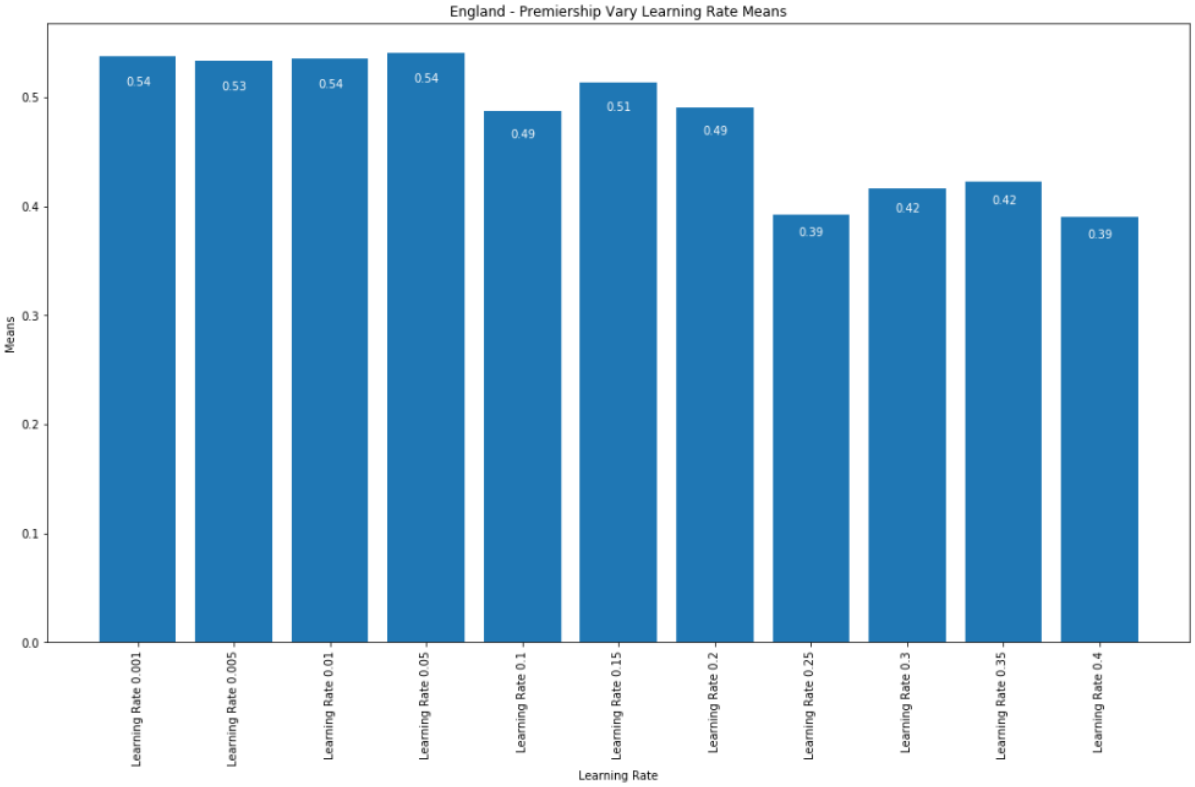

Figure V11: England Premiership accuracy for train set (y) varying depending on optimizer (x)

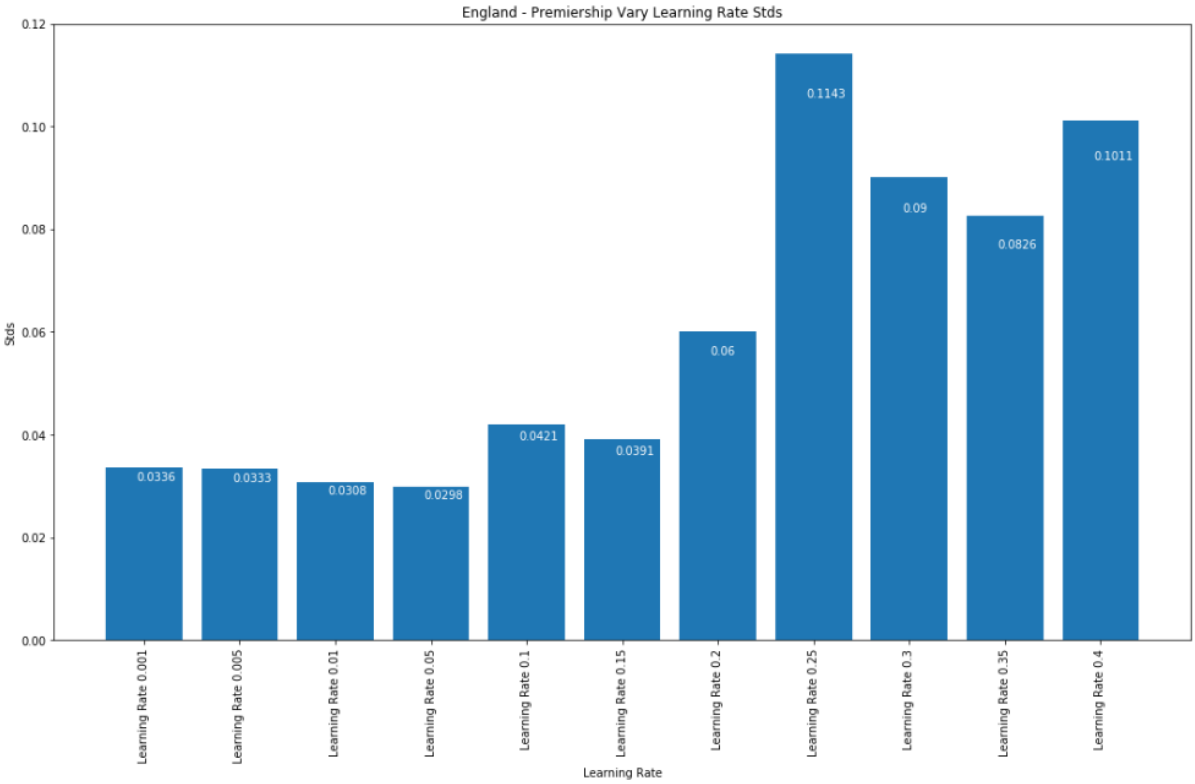

Figure V12: England Premiership accuracy std for train set (y) varying depending on optimizer (x)

The Figures V11 and V12 suggest that the optimal learning rate for England Premiership in terms of accuracy is 0.05. It is also the learning rate with the lowest standard deviation.

However, the difference in performance between models with learning rates of 0.05, 0.01, 0.005 and 0.001 is not comparatively high. It is also the case that small variabilities in the accuracy and standard deviation are expected since Figures V11 and V12 are obtained using 10 cross-fold validation. For a more analysis of the learning rate, the accuracy and loss graphs are produced, which can be obtained after a model has been trained.

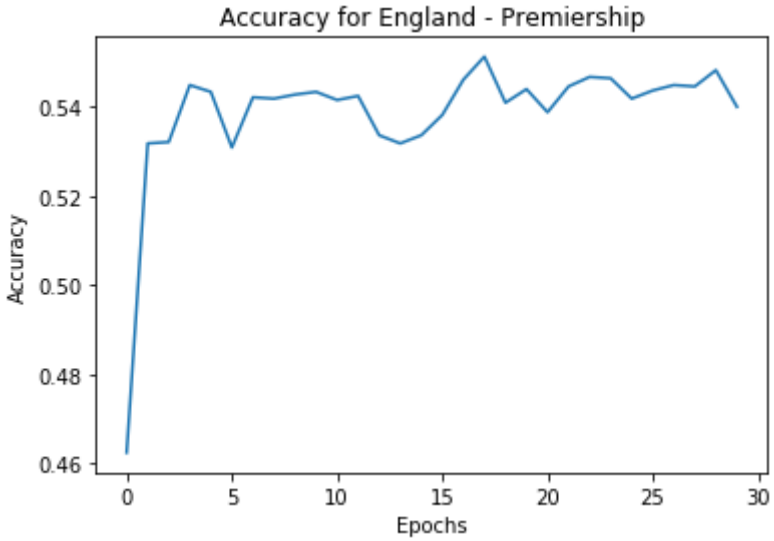

Figure V13: England Premiership accuracy change during the model training with learning rate 0.05

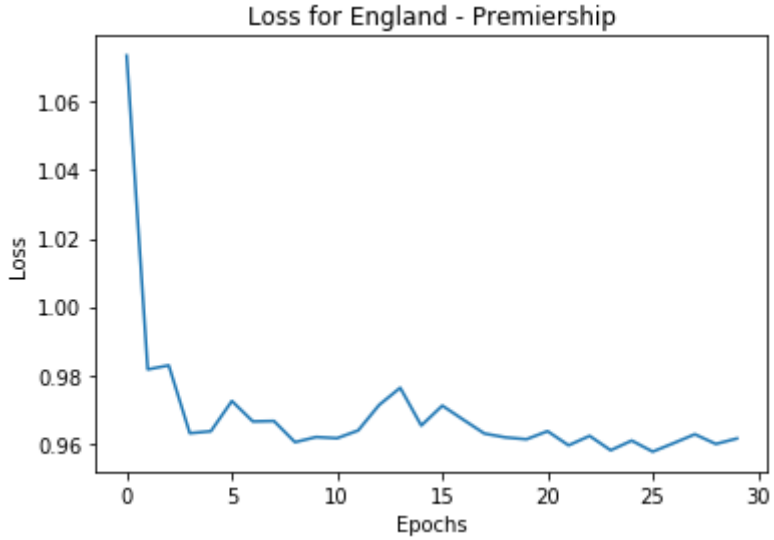

Figure V14: England Premiership loss change during the model training with learning rate 0.05

Evidently, the loss and accuracy during training experience spikes with a learning rate of 0.05. 1.89% standard deviation. For comparison, the same graphs with a learning rate of 0.001 are provided.

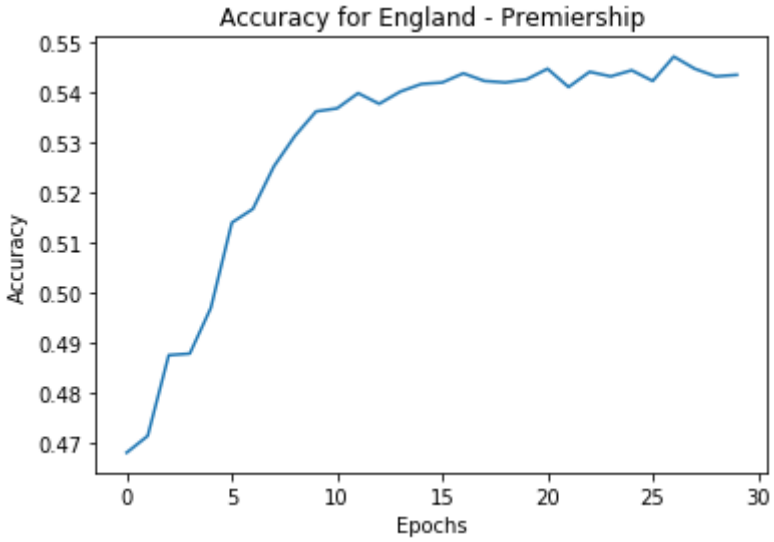

Figure V15: England Premiership accuracy change during the model training with learning rate 0.001

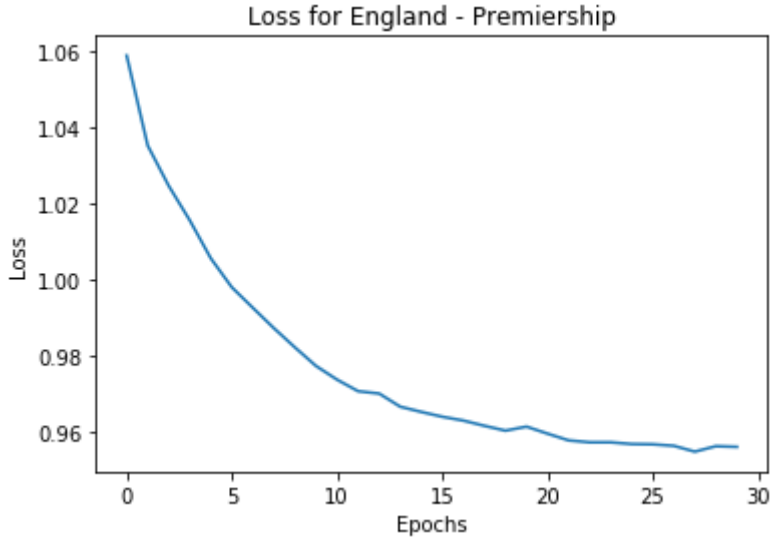

Figure V16: England Premiership loss change during the model training with learning rate 0.001

As can be seen in Figures V15 and V16, a lower learning rate results in a smoother training process. The standard deviation has also decreased from 1.89% to 0.86%, which is not trivial. These results indicate that the model is able to generalise better and produce more consistent outcomes when using a lower learning rate. A similar trend can be observed for France Championnat and Italy Serie A.

However, for Germany Bundesliga, the situation appears to be slightly different. Using a learning rate of 0.001, the loss graph still experiences spikes during the training process, which can be observed in Figure V17.

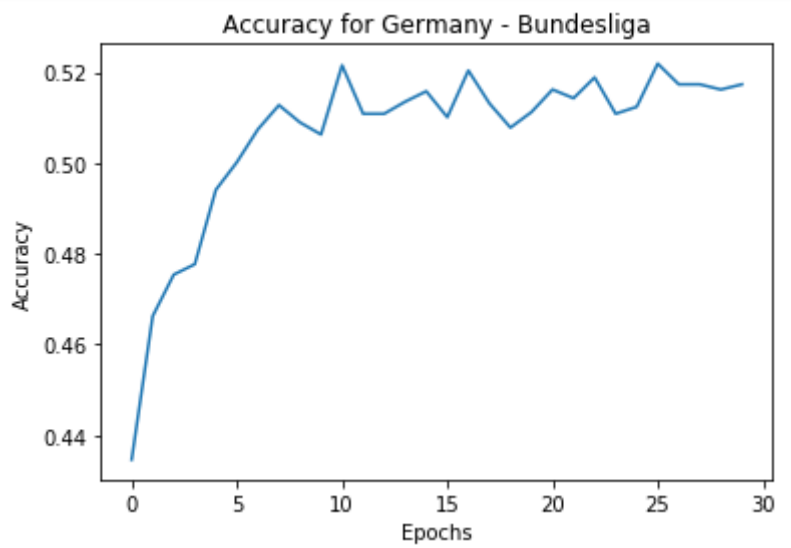

Figure V17: Germany Bundesliga accuracy change during the model training with learning rate 0.001

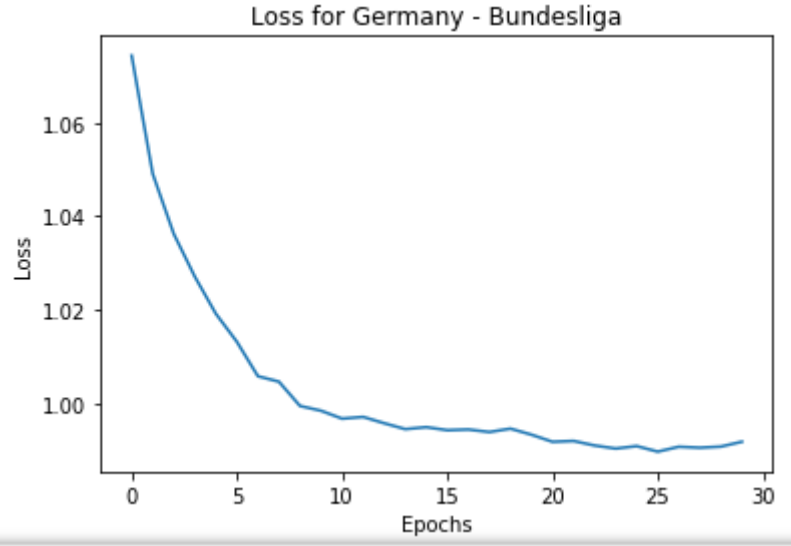

Figure V18: Germany Bundesliga loss change during the model training with learning rate 0.001

This suggests that decreasing the learning rate further may improve the stability of the model. An updated graph when using a learning rate of 0.0005 can be seen in Figure V19.

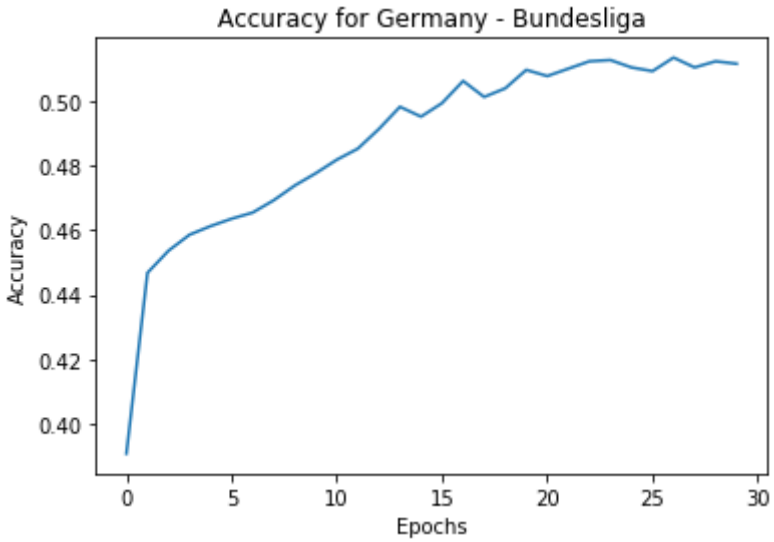

Figure V19: Germany Bundesliga accuracy change during the model training with learning rate 0.0005

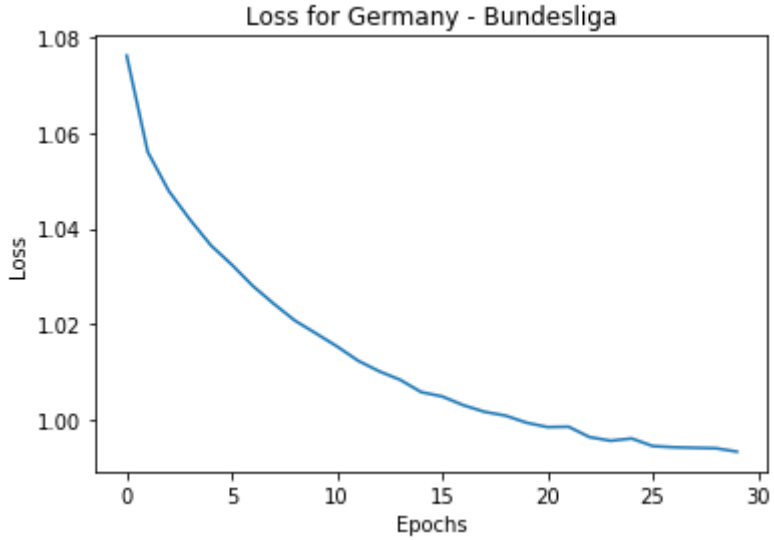

Figure V20: Germany Bundesliga loss change during the model training with learning rate 0.0005

As can be seen in Figures V19 and V20, the loss and accuracy change during the training process is now smoother. The standard deviation of the models has also decreased by 0.5%. Thus, a learning rate of 0.0005 is selected as optimal for Germany Bundesliga.

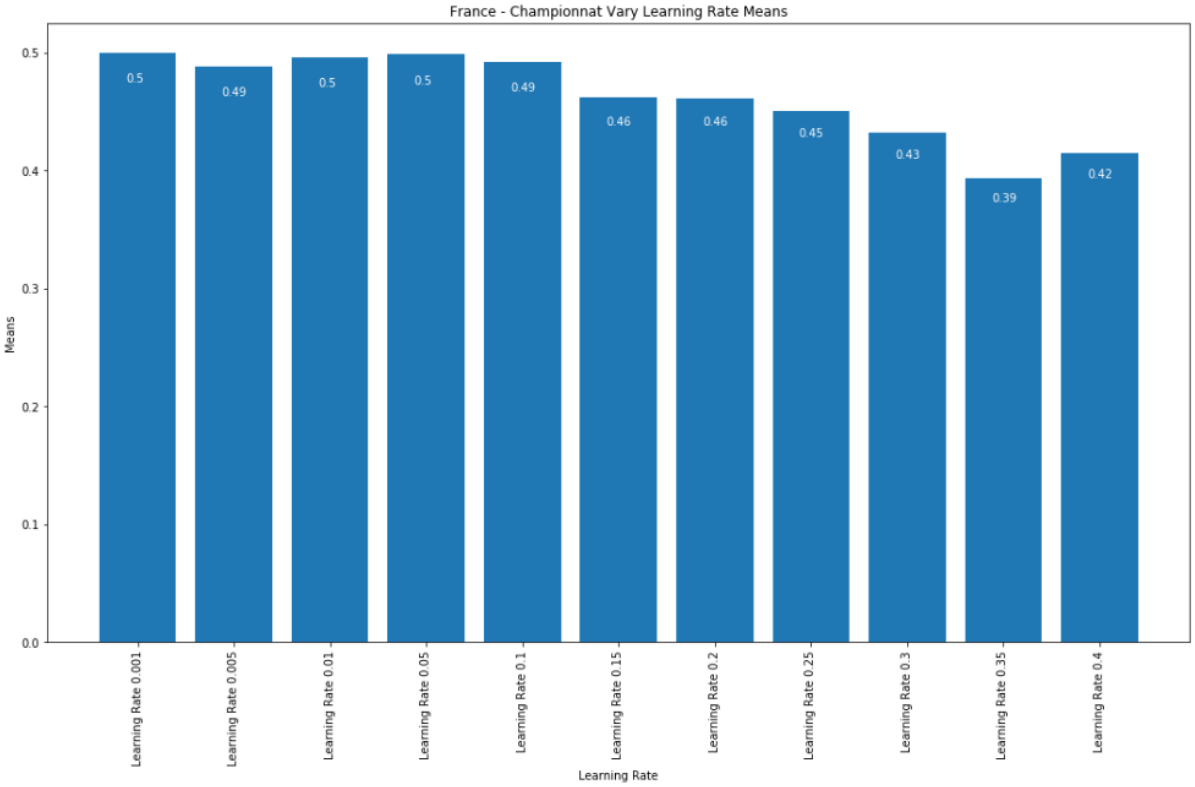

Figure V21: France Championnat accuracy for train set (y) varying depending on learning rate (x)

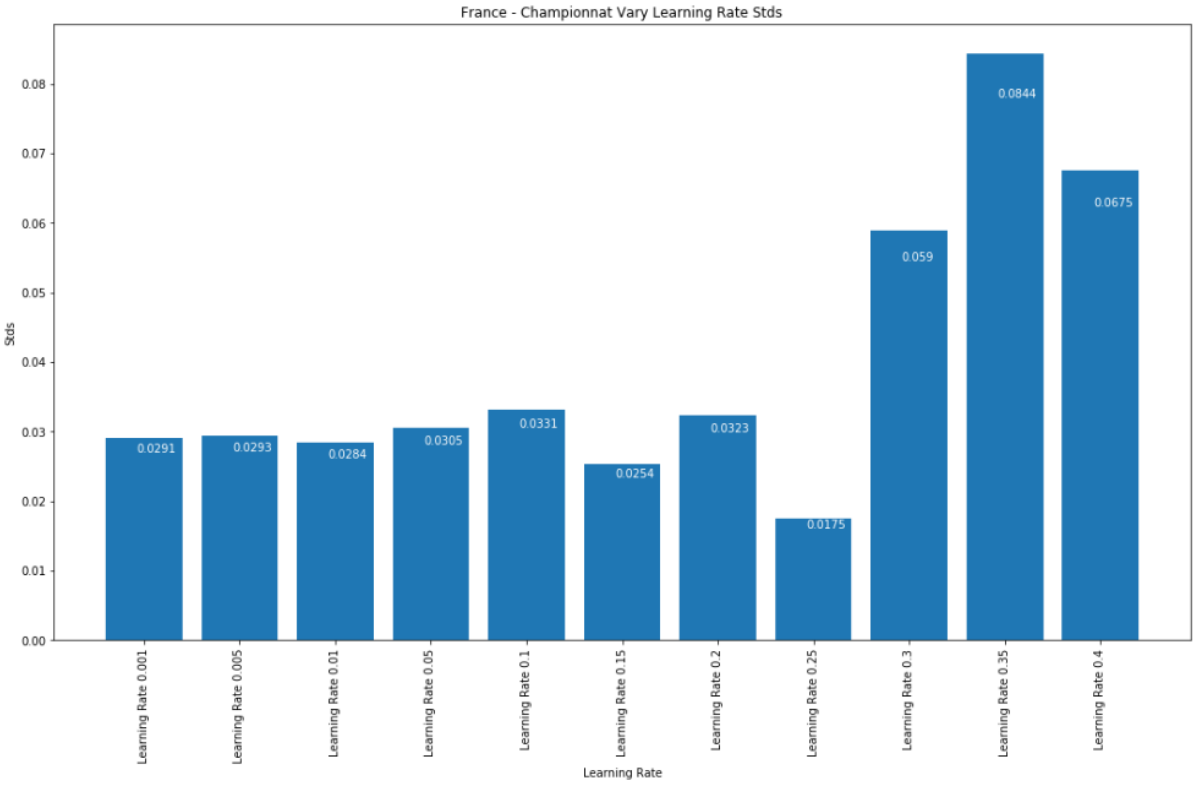

Figure V22: France Championnat accuracy std for train set (y) varying depending on learning rate (x)

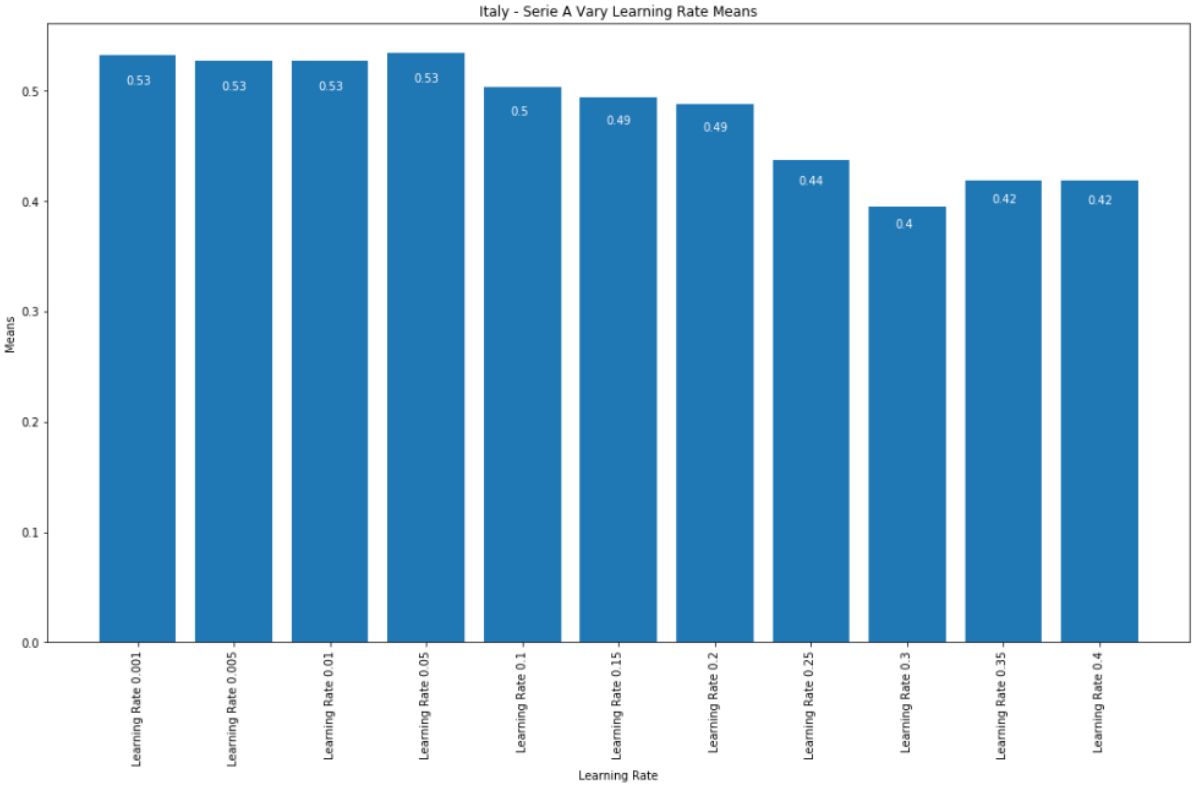

Figure V23: Italy Serie A accuracy for train set (y) varying depending on learning rate (x)

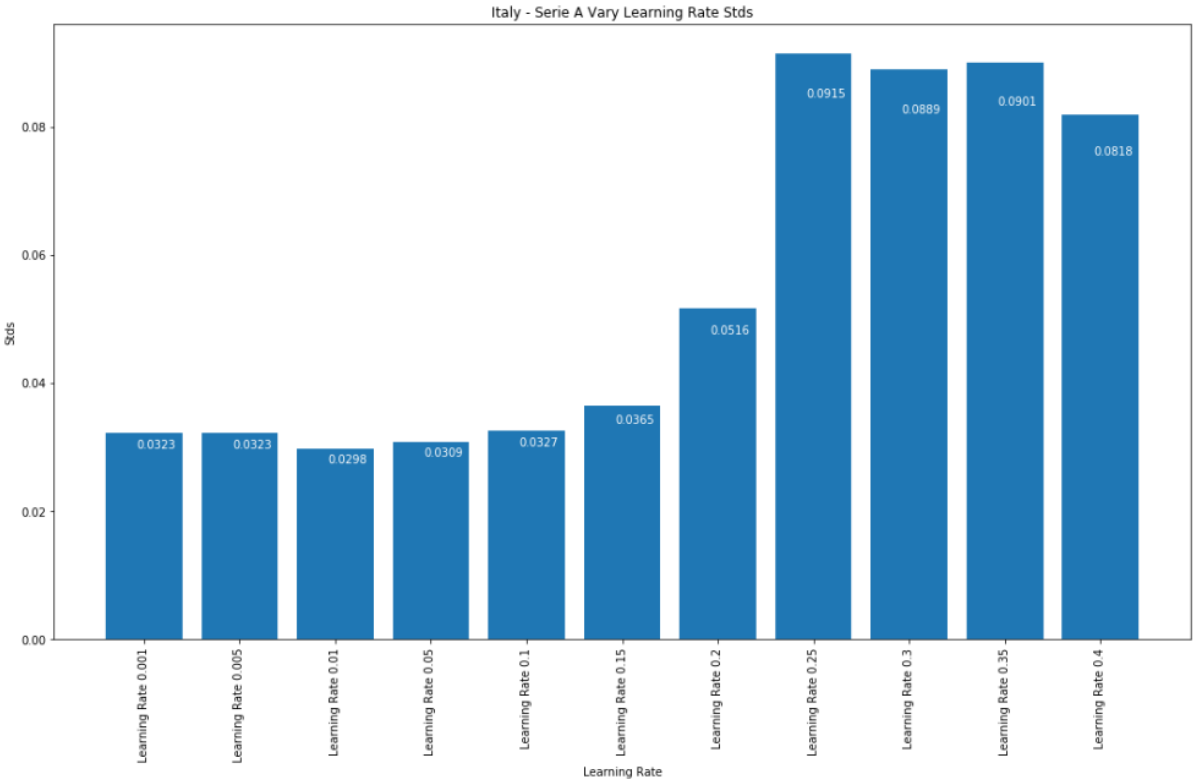

Figure V24: Italy Serie A accuracy std for train set (y) varying depending on learning rate (x)

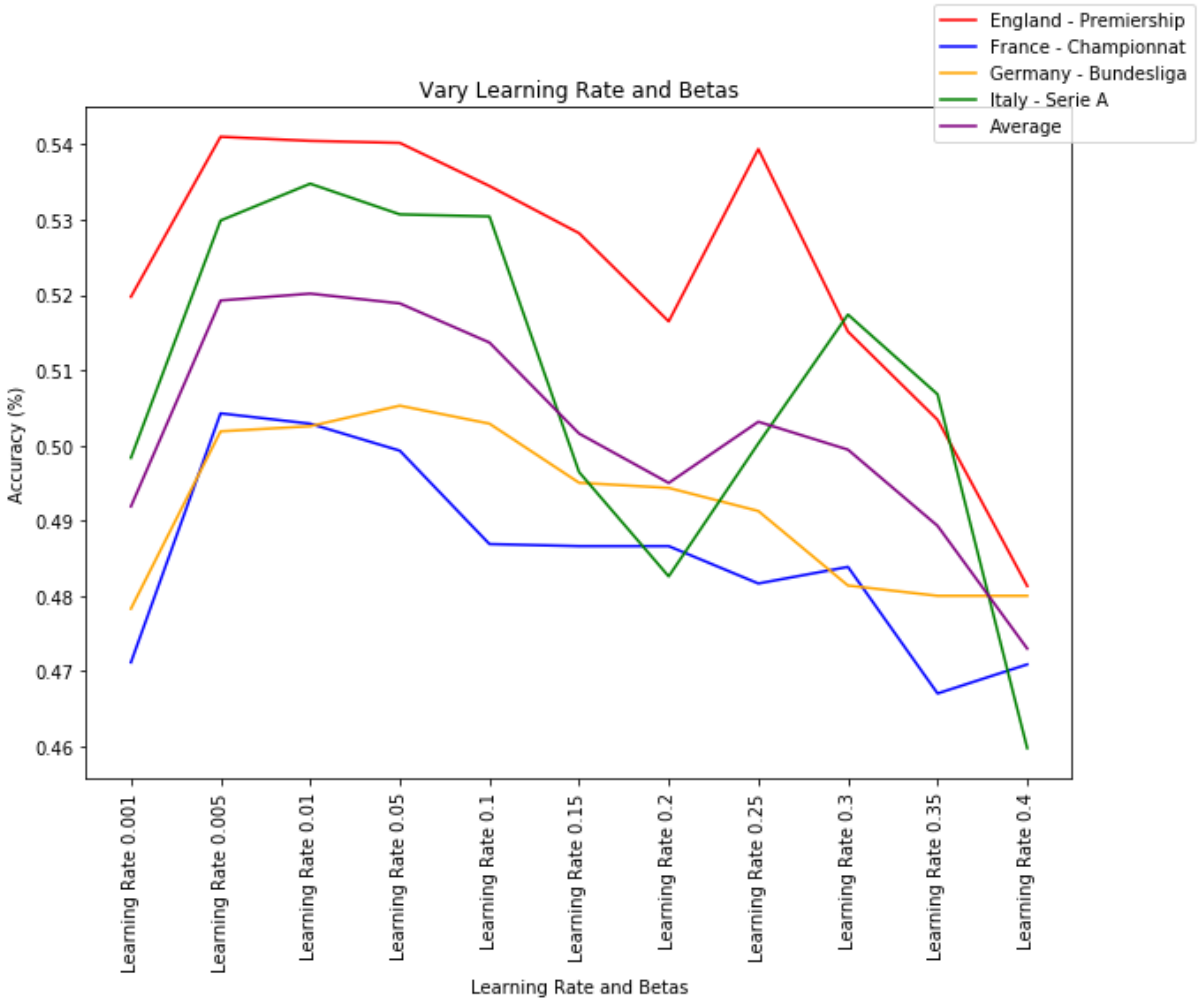

Figure V25: Accuracy for train set (y) varying depending on learning rate (x)

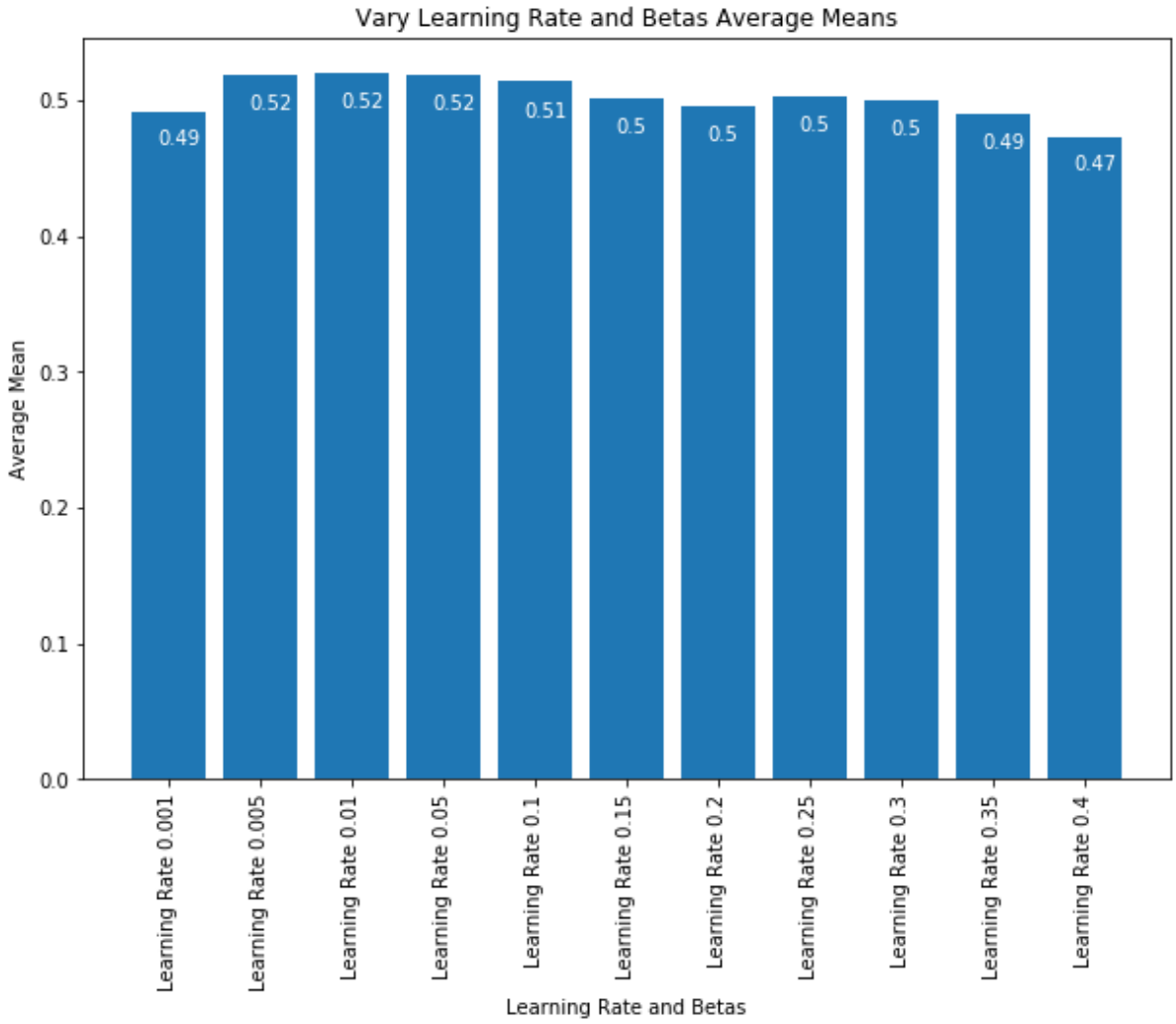

Figure V26: Overall accuracy for train set (y) varying depending on learning rate (x)

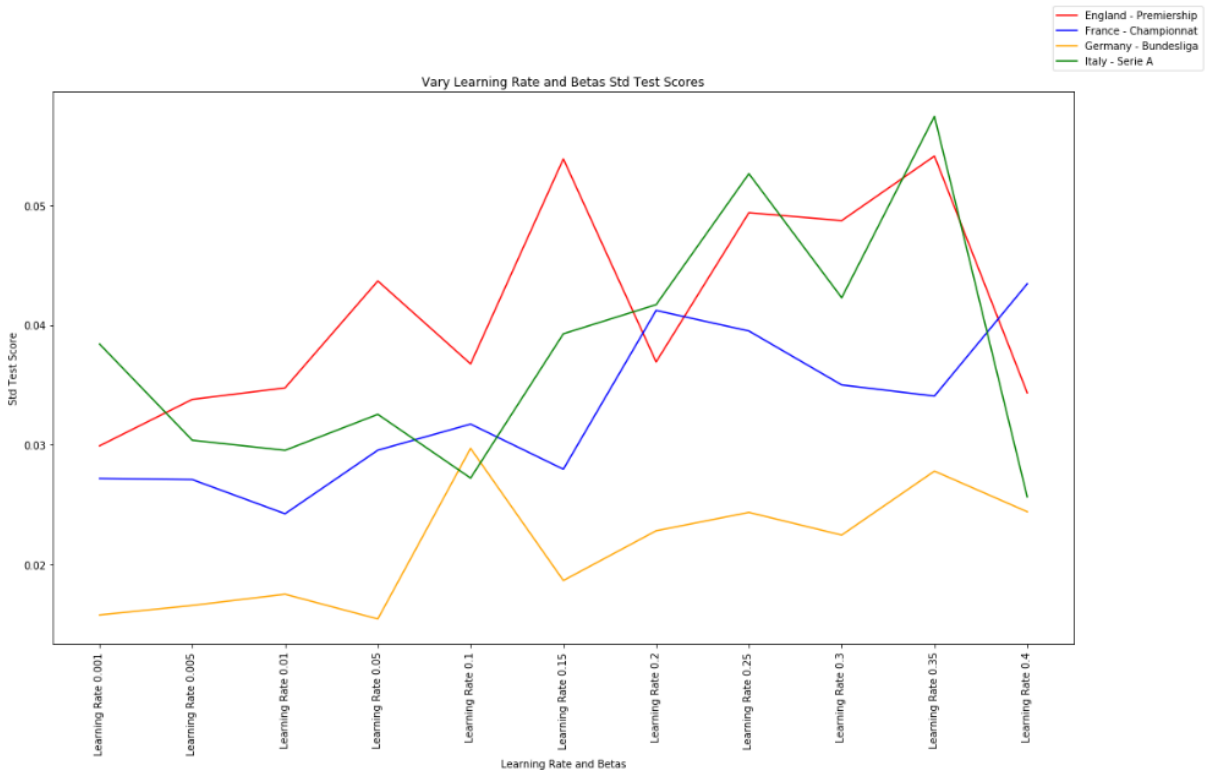

Figure V27: Accuracy std for train set (y) varying depending on learning rate (x)

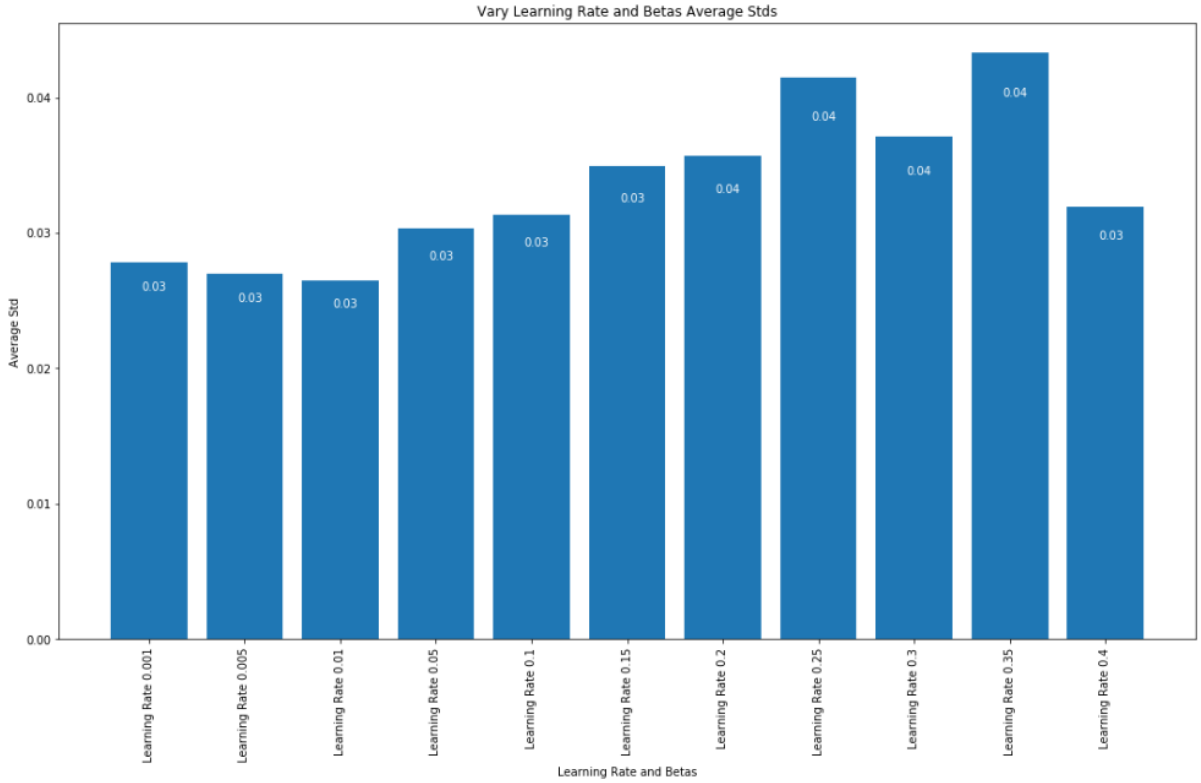

Figure V28: Overall accuracy std for train set (y) varying depending on learning rate (x)

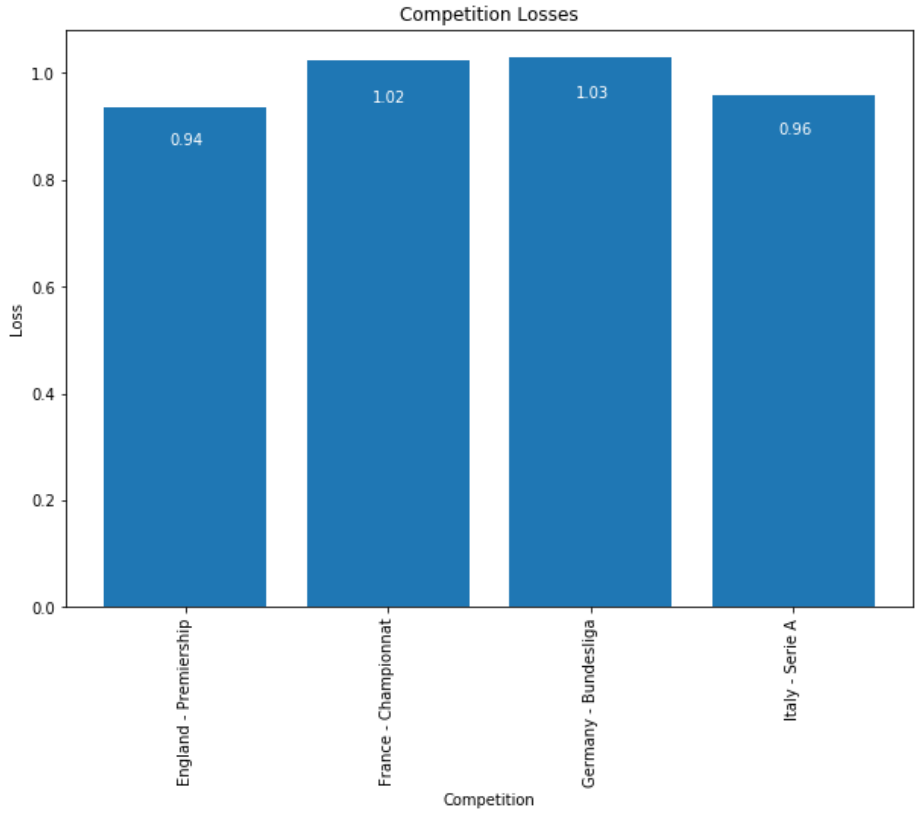

Figure V29: NN competition loss after training after tuning the learning rate

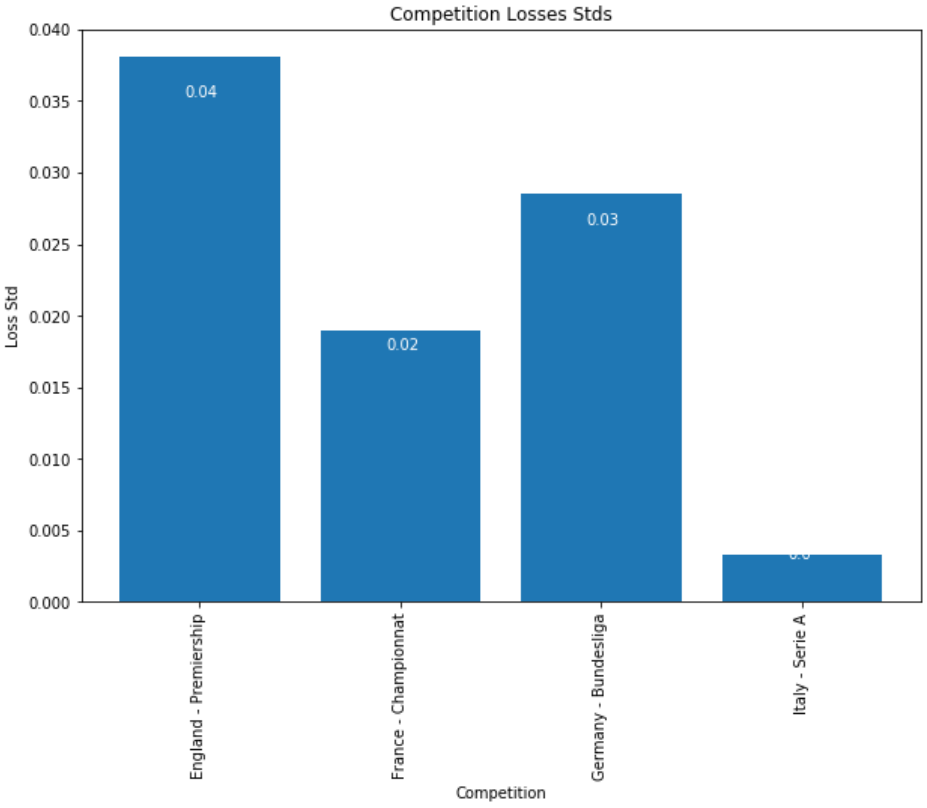

Figure V30: NN competition loss standard deviation after training after tuning the learning rate

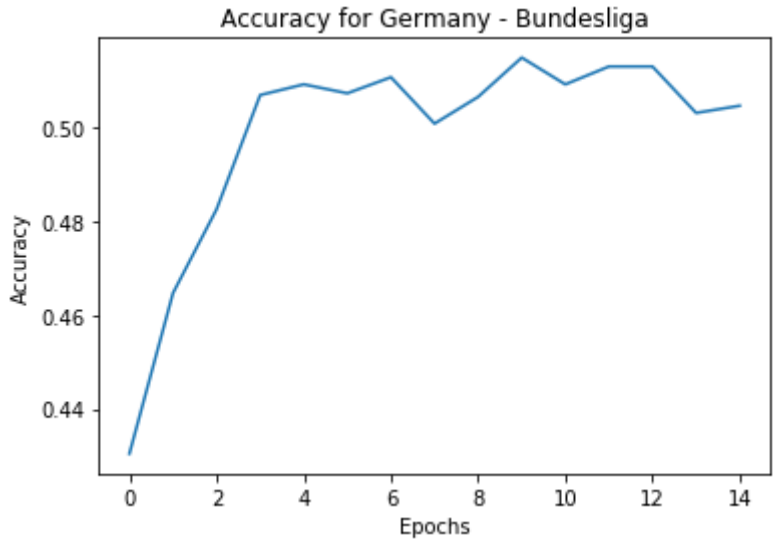

Figure V31: Germany Bundesliga train accuracy change with 0.05 learning rate

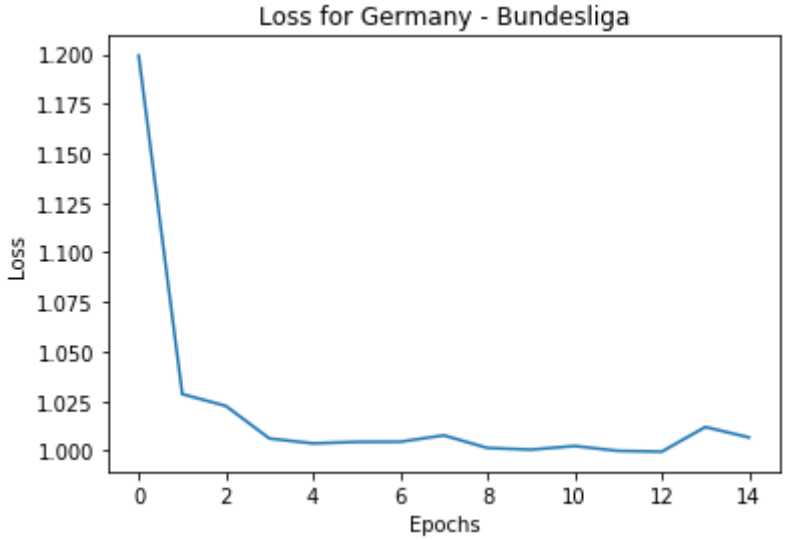

Figure V32: Germany Bundesliga train loss change with 0.05 learning rate

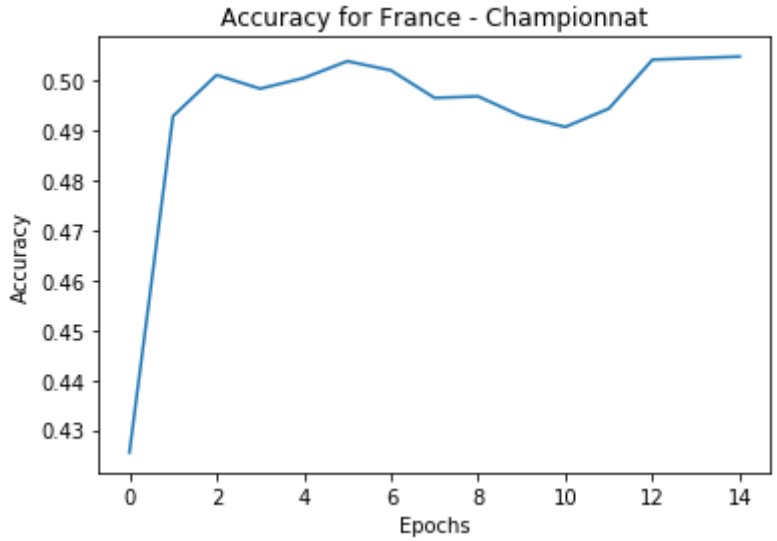

Figure V33: France Championnat train accuracy change with 0.1 learning rate

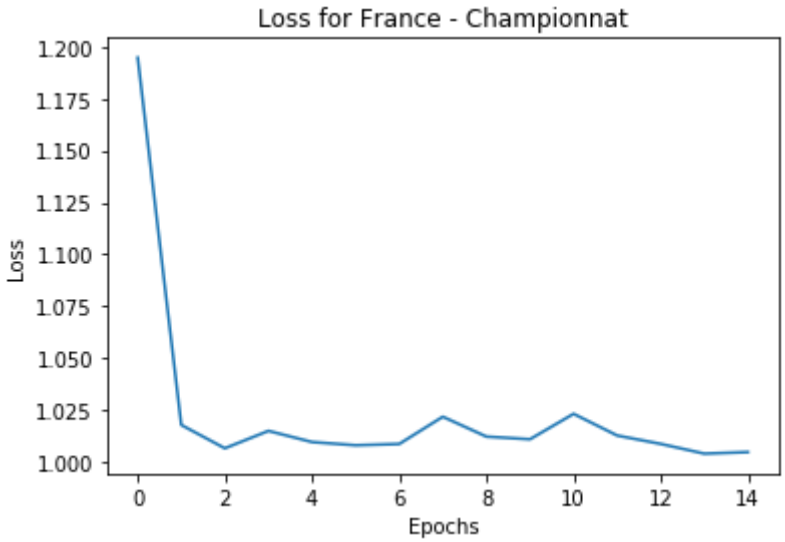

Figure V34: France Championnat train loss change with 0.1 learning rate

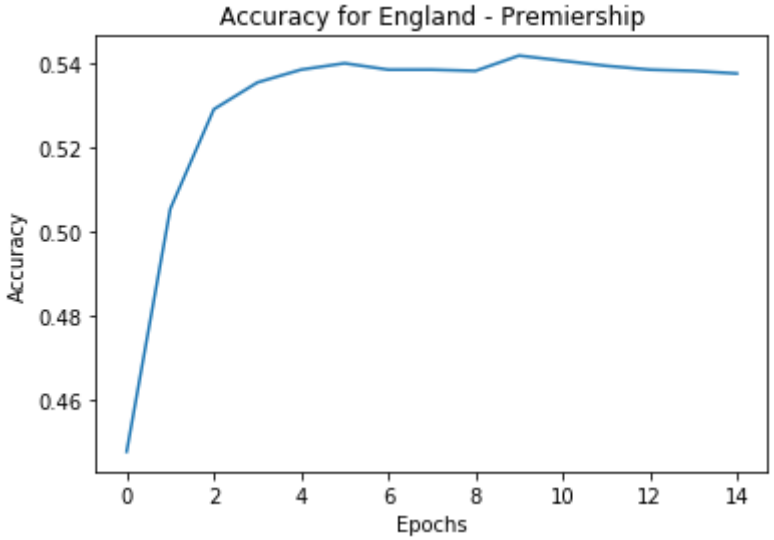

Figure V35: England Premiership train accuracy change with 0.1 learning rate

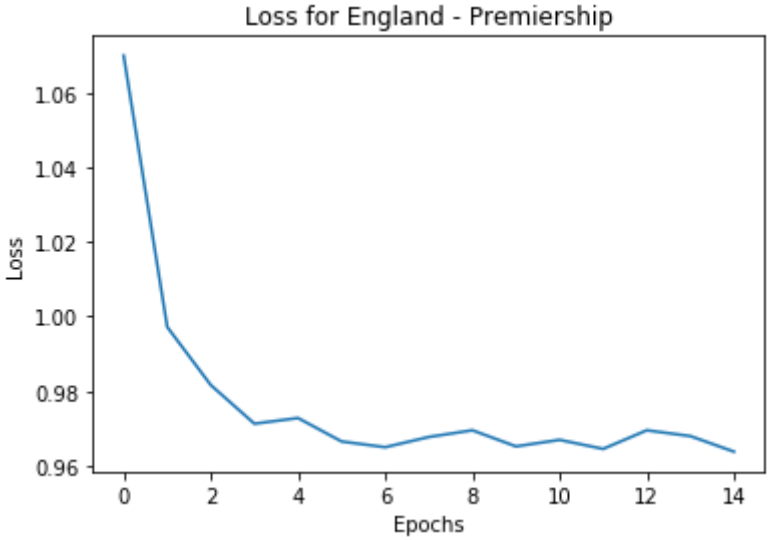

Figure V36: England Premiership train loss change with 0.1 learning rate

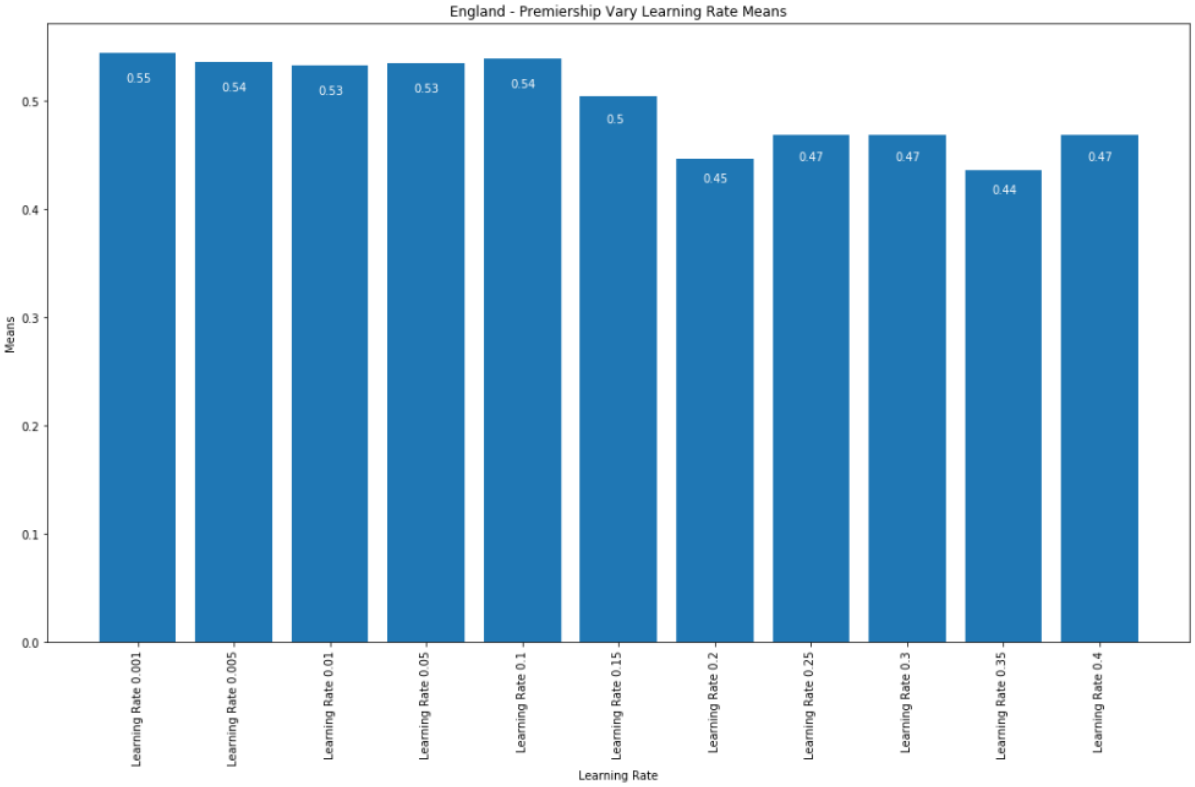

Figure V37: England Premiership average accuracy for NN with different learning rates

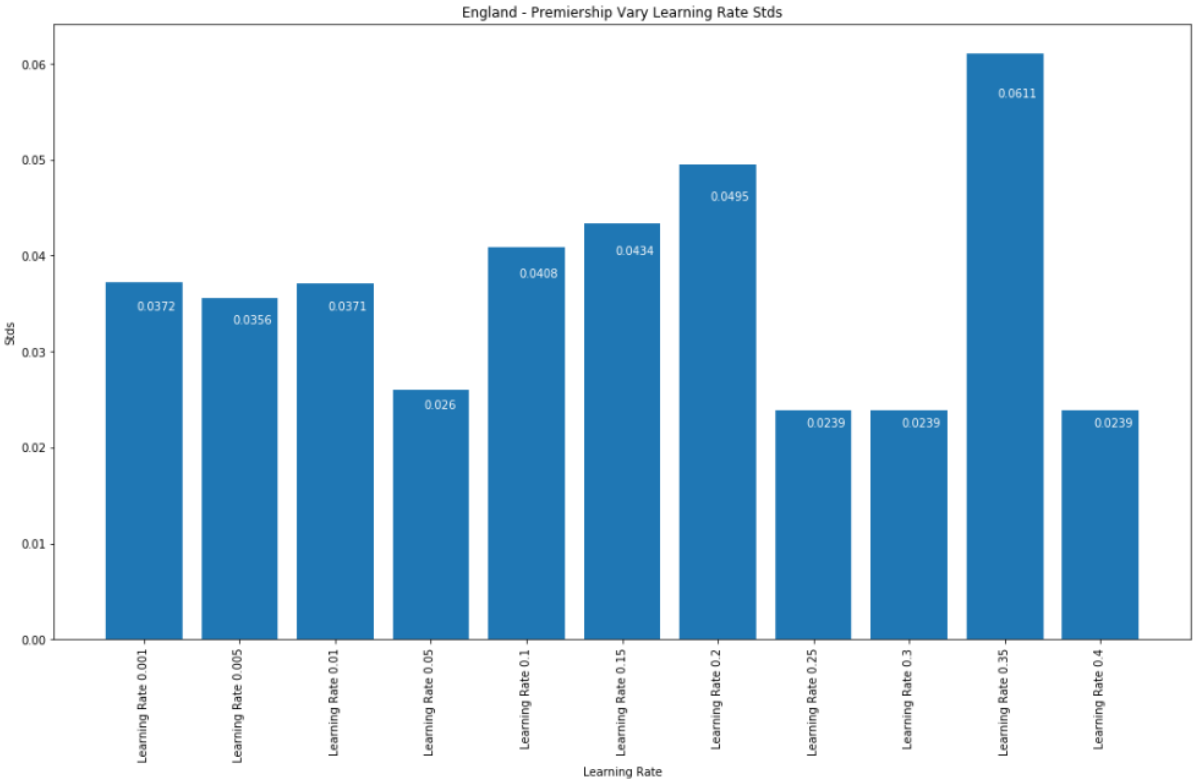

Figure V38: England Premiership average accuracy standard deviation for NN with different learning rates

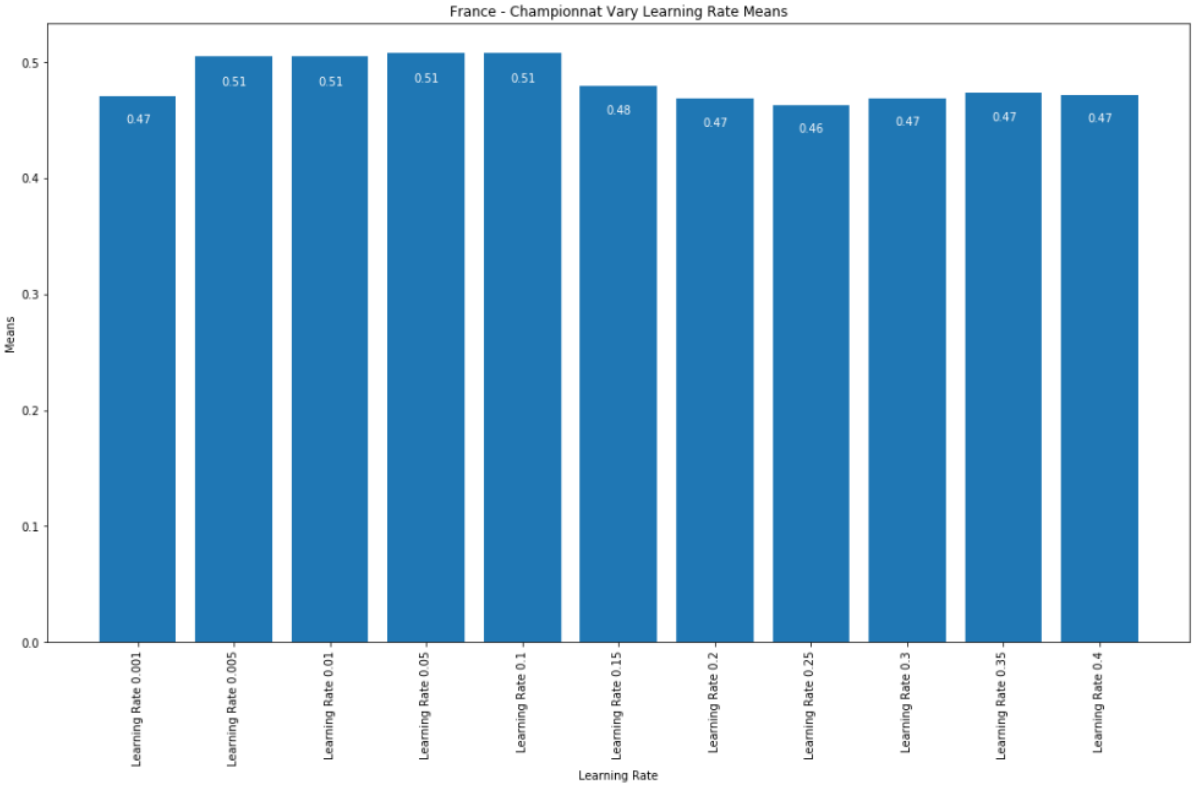

Figure V39: France Championnat average accuracy for NN with different learning rates

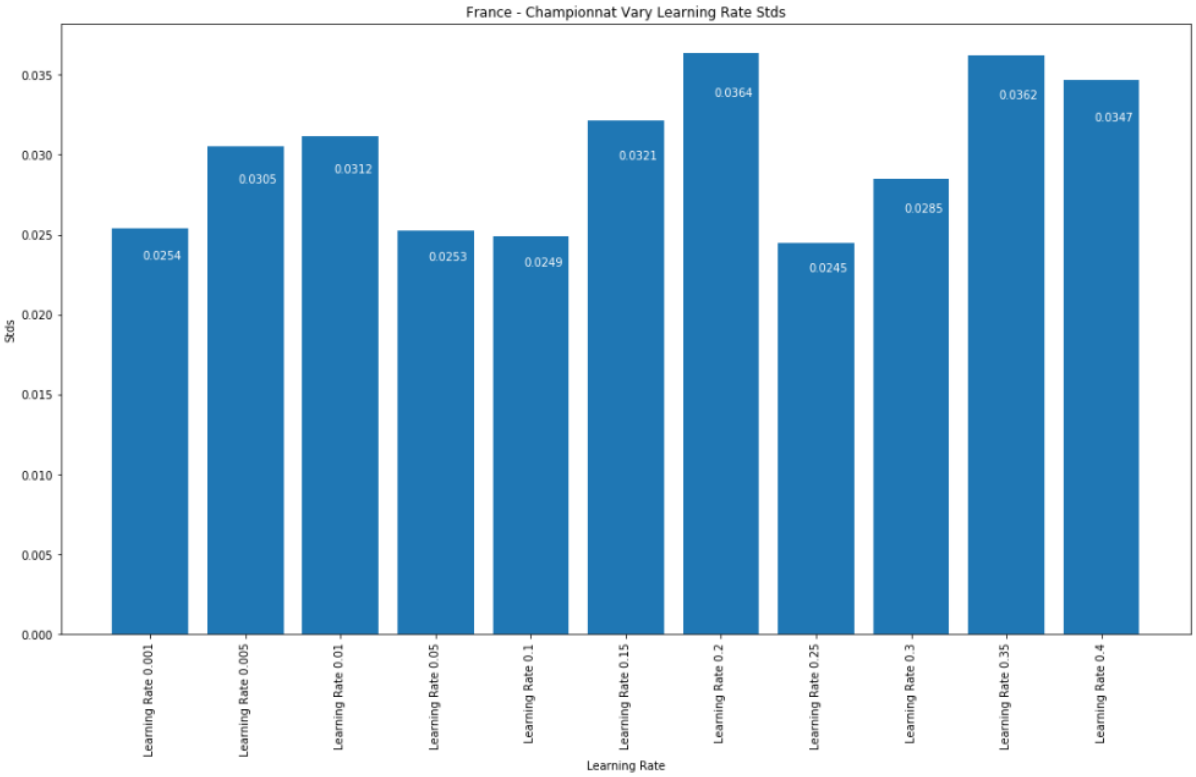

Figure V40: France Championnat average accuracy standard deviation for NN with different learning rates

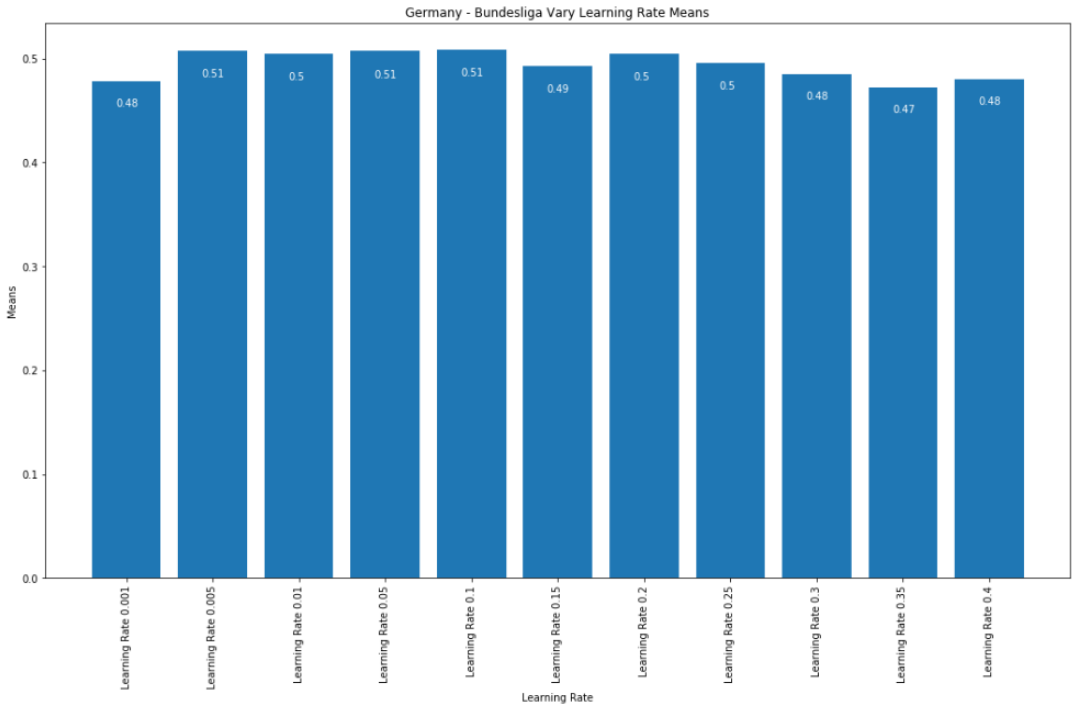

Figure V41: Germany Bundesliga average accuracy for NN with different learning rates

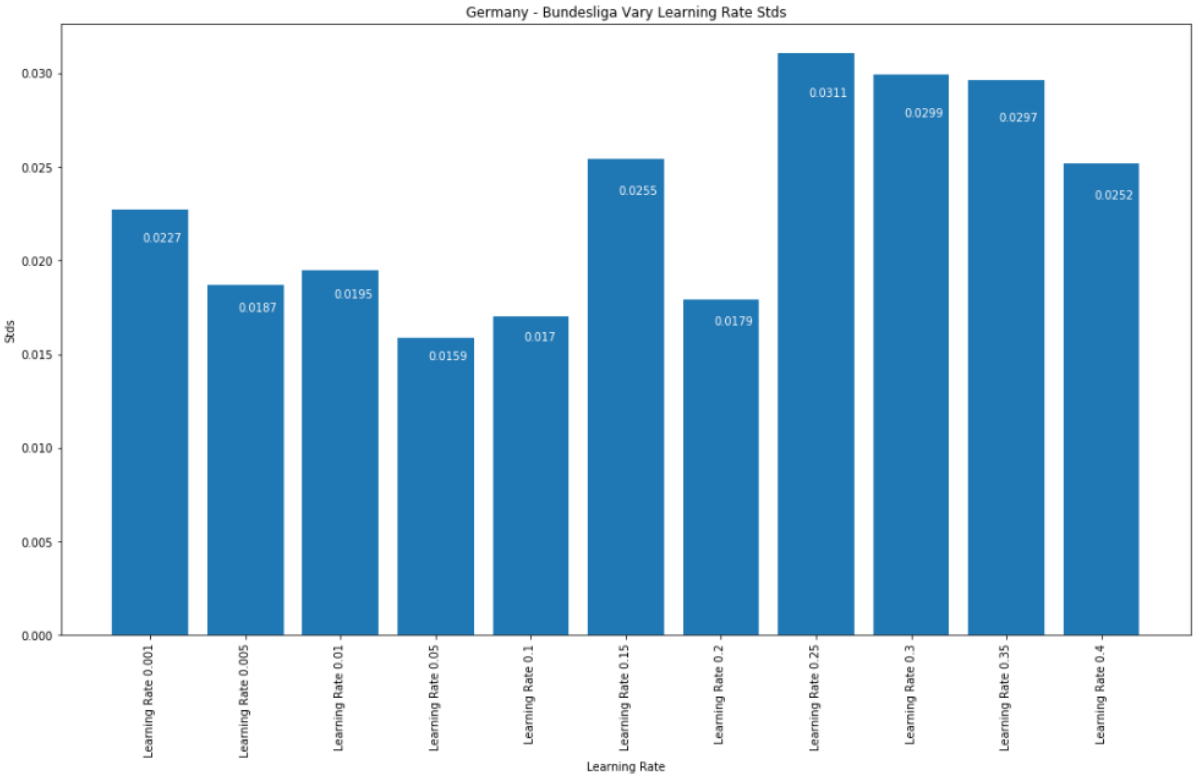

Figure V42: Germany Bundesliga average accuracy standard deviation for NN with different learning rates

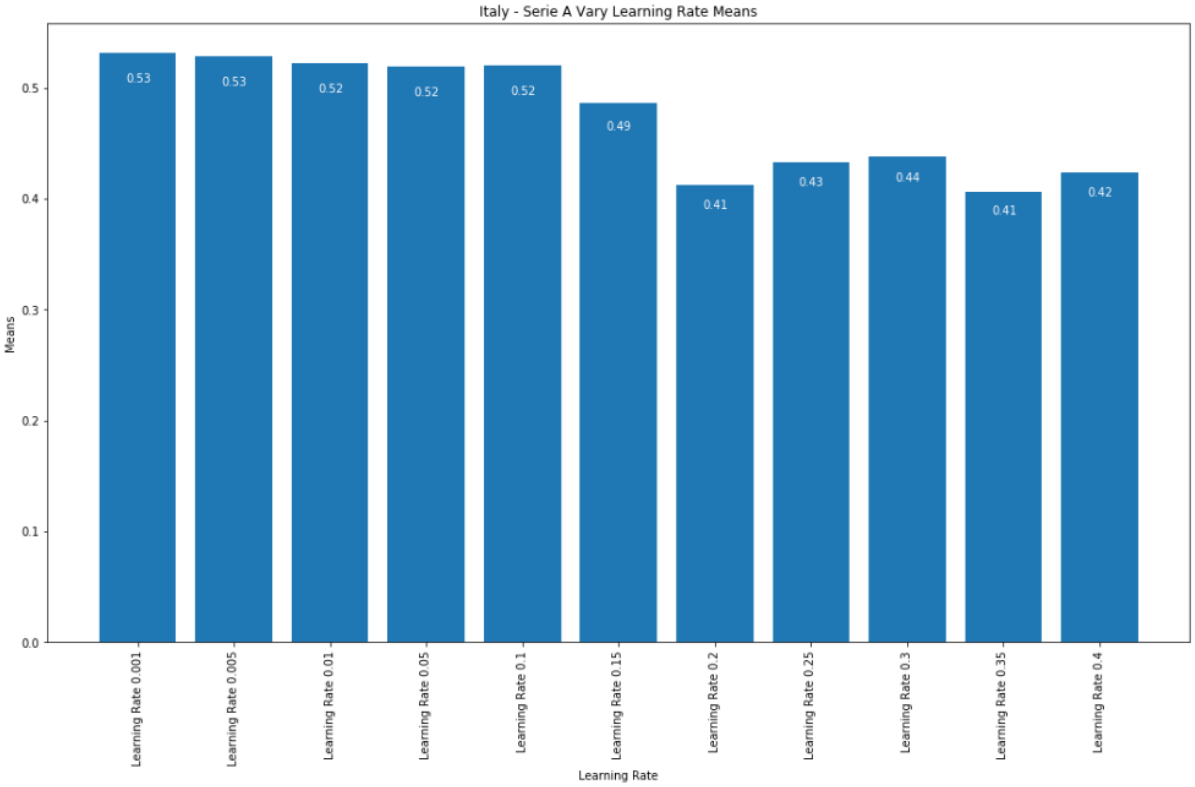

Figure V43: Italy Serie A average accuracy for NN with different learning rates

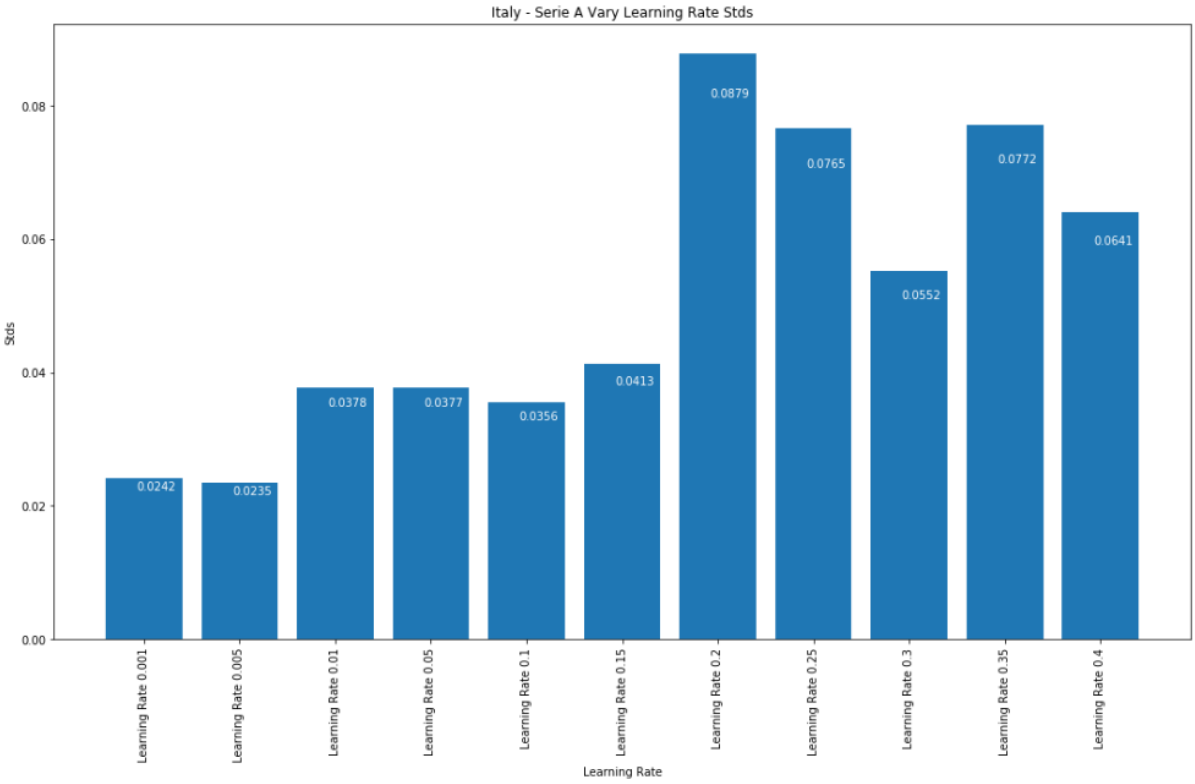

Figure V44: Italy Serie A average accuracy standard deviation for NN with different learning rates

Appendix W: Selecting Optimizer for Neural Network

There are several optimizers that come with the *keras* package, such as Stochastic gradient descent (SGD), RMSProp, Adagrad, Adadelta, Adam, Adamax and Nadam. For more information on each of those, refer to official Keras documentation. Using different optimizers, the models produce different accuracies, which are presented in Figure W1.

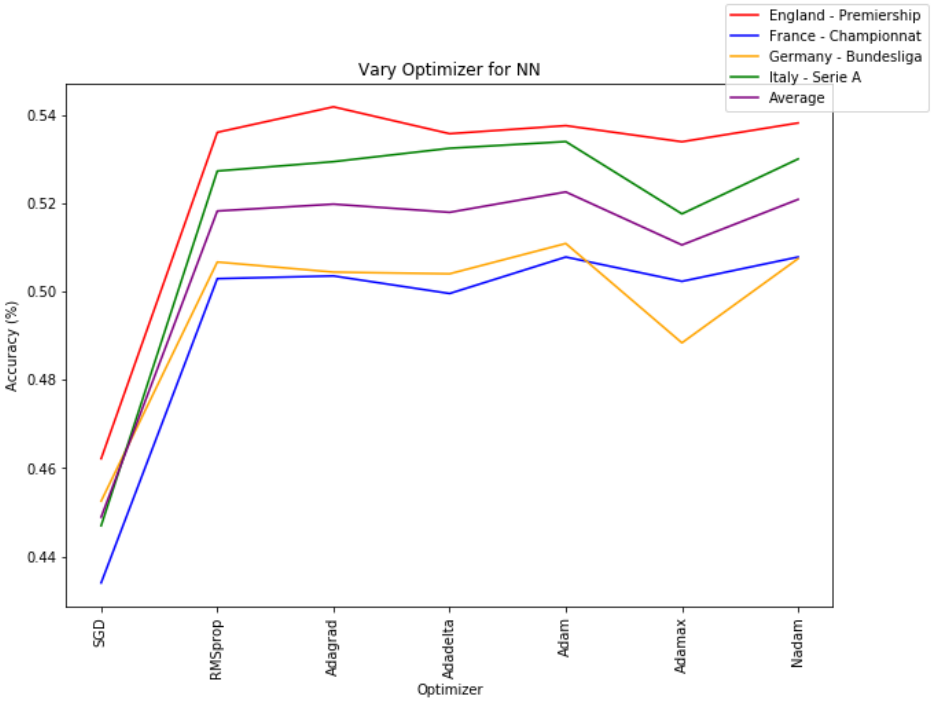

Figure W1: Competition accuracy for train set (y) varying depending on optimizer (x)

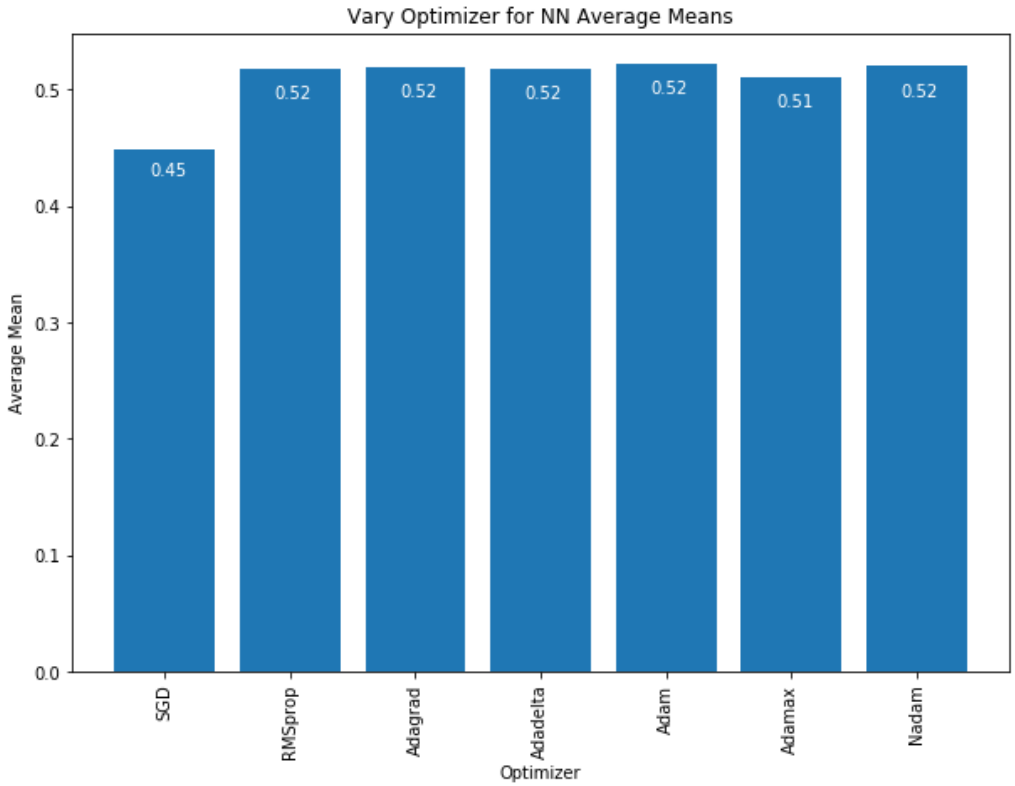

Figure W2: Overall accuracy for train set (y) varying depending on optimizer (x)

As can be seen in Figures W1 and W2, the best optimizer overall is Adam with an average accuracy of 52.25% (if judged solely by the average competition accuracy). Most optimizers have similar accuracies, except for SGD, which experienced a significantly lower performance of 45%. However, while all optimizers shared the learning rate, other available parameters that are optimizer-specific are left at default for the moment.

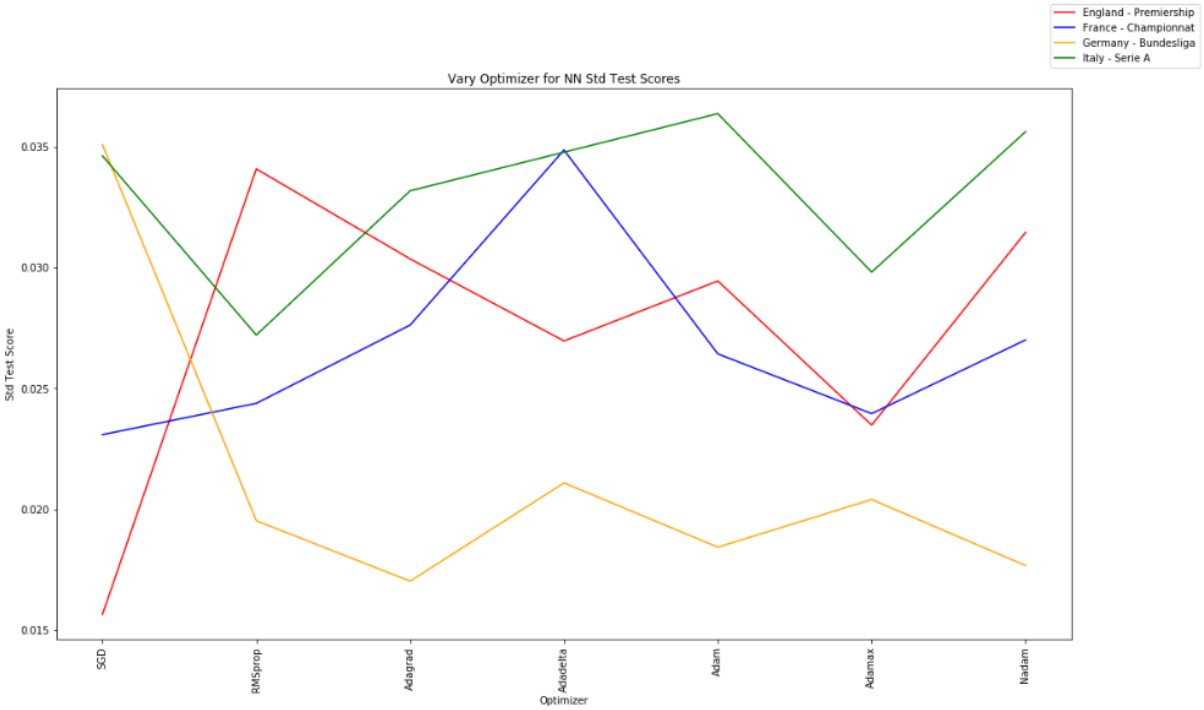

Figure W3: Competition accuracy std for train set (y) varying depending on optimizer (x)

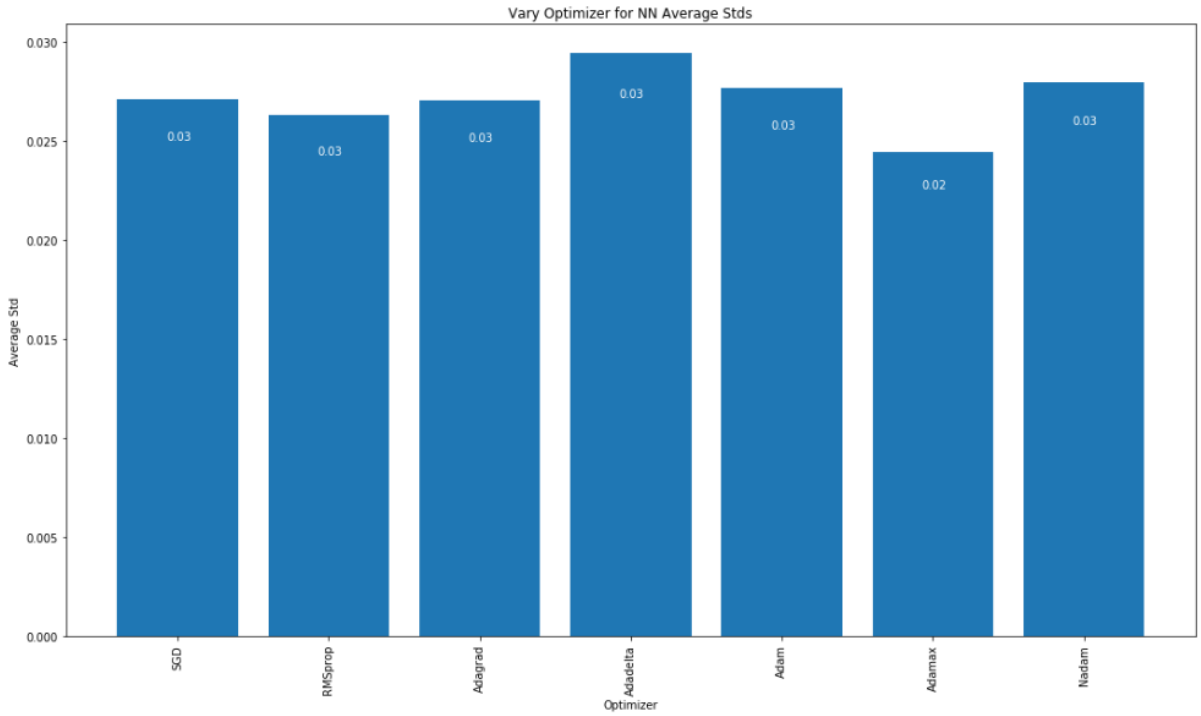

Figure W4: Overall accuracy std for train set (y) varying depending on optimizer (x)

Considering the standard deviation of accuracies for each optimizer, it is evident by Figures W3 and W4 that most of the deviations were at around 2.75% (+ or - 0.5%). The Adamax optimizer has the lowest standard deviation and is thus selected as optimal optimizer overall, given that its standard deviation compensates for the difference in accuracy. Determining accuracies and standard deviations for each competition in turn may result in a different picture.

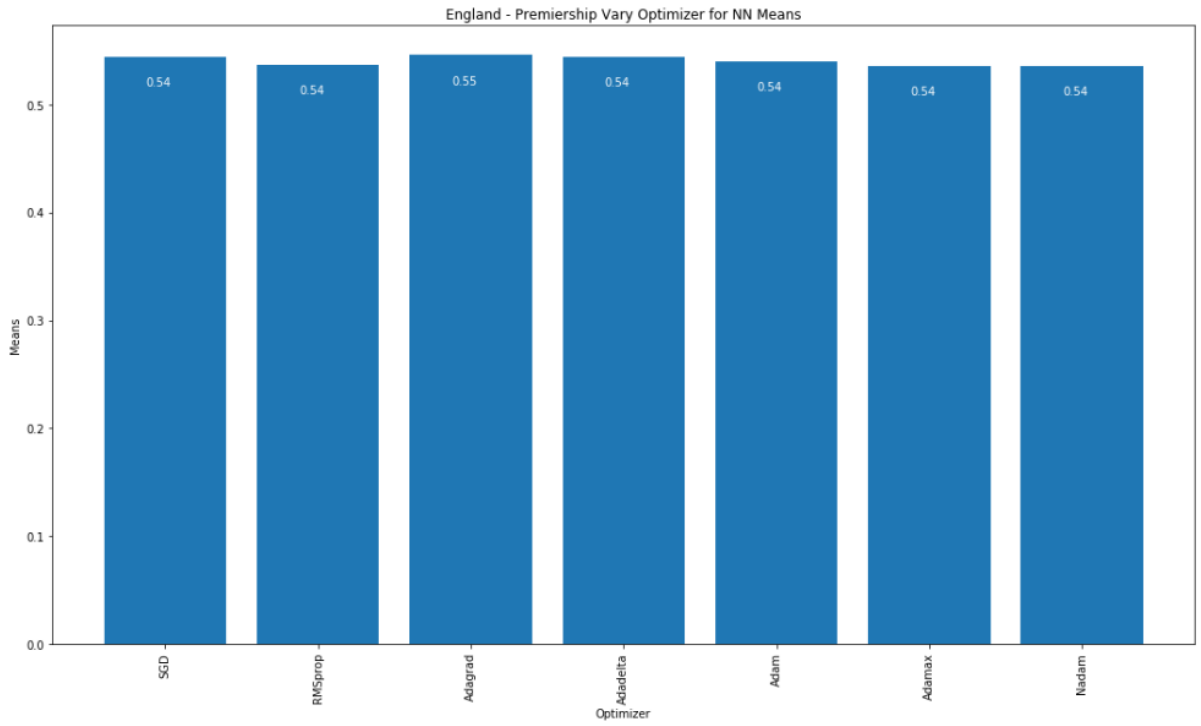

Figure W5: England Premiership accuracy for train set (y) varying depending on optimizer (x)

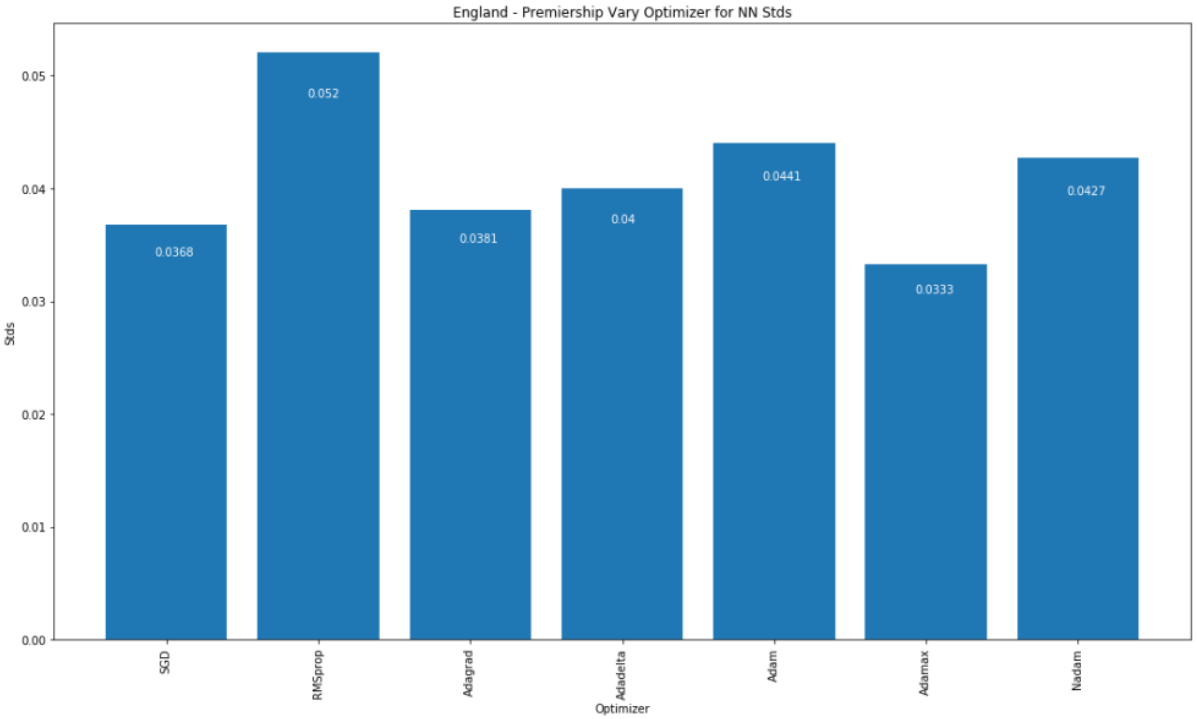

Figure W6: England Premiership accuracy std for train set (y) varying depending on optimizer (x)

As evident by Figure W5, all optimizers have very similar accuracy for the England Premiership. However, the standard deviation varied a significant amount, at least for the optimizers with default parameters (except for the shared learning rate). Adamax optimizer has the lowest standard deviation amongst all used optimizers and because of this is considered optimal for England Premiership. A similar trend can be observed in France Championnat.

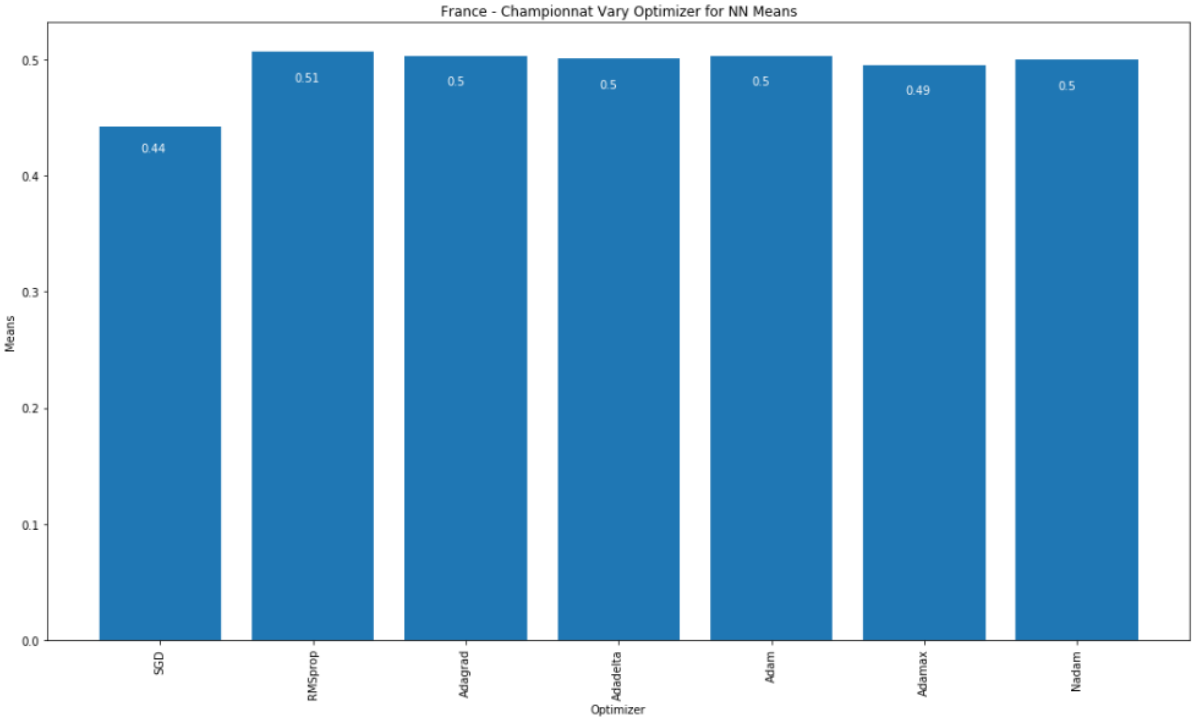

Figure W7: France Championnat accuracy for train set (y) varying depending on optimizer (x)

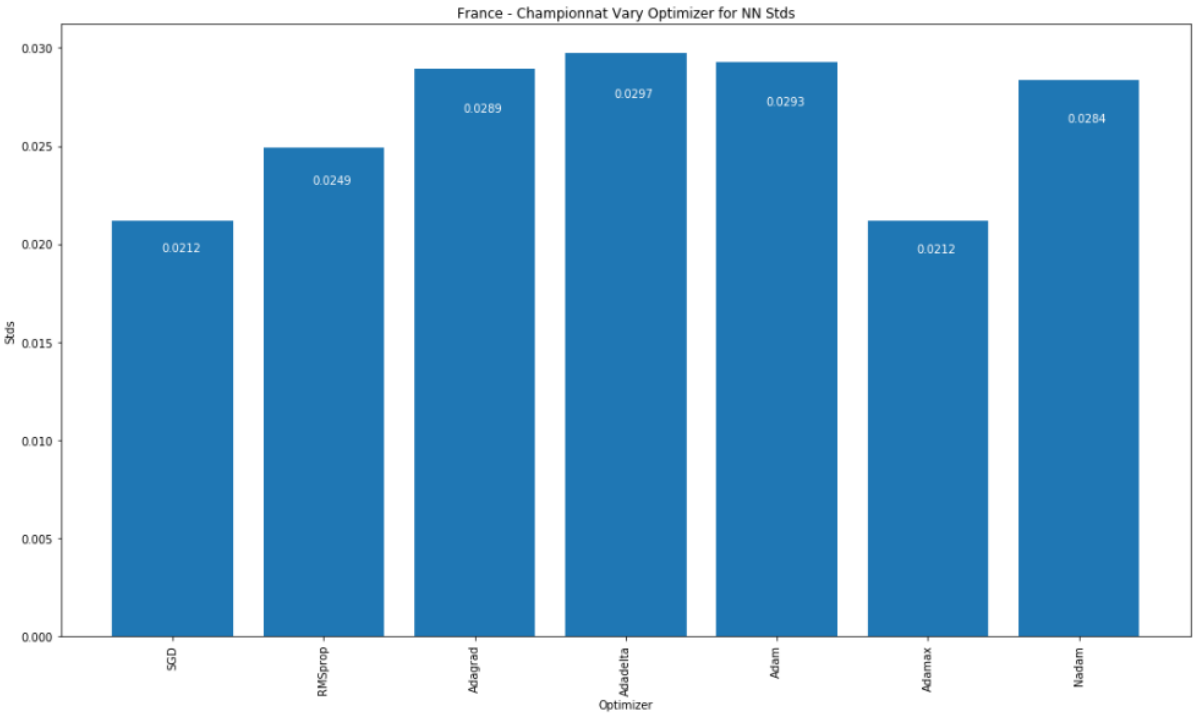

Figure W8: France Championnat accuracy std for train set (y) varying depending on optimizer (x)

France Championnat is different from the England Premiership in that SGD has a considerably lower accuracy of 44% in comparison to the 49% - 51% of other optimizers. Like with the England Premiership, Adamax optimizer has the lowest standard deviation, just as the SGD one. Since Adamax also has higher accuracy, it is selected as optimal for France Championnat.

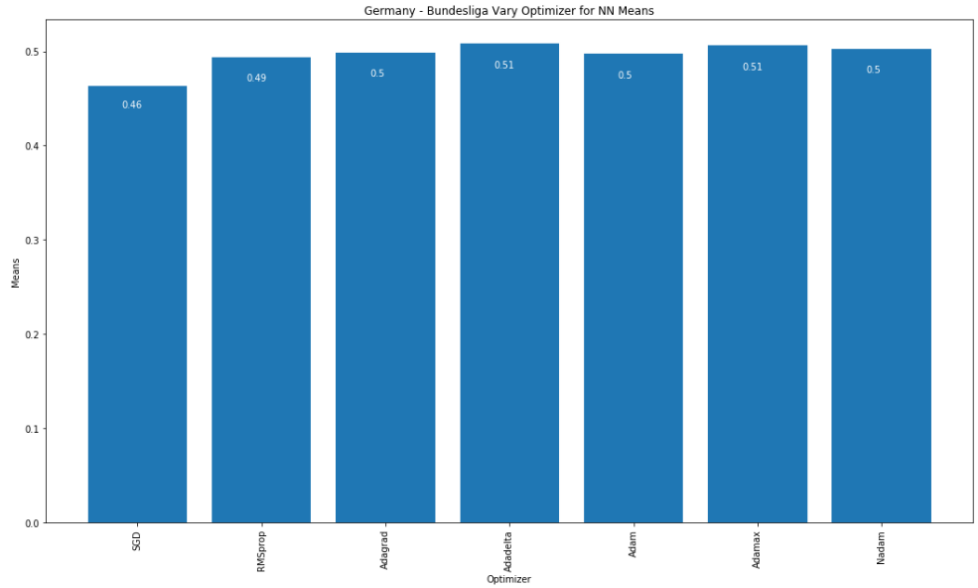

Figure W9: Germany Bundesliga accuracy for train set (y) varying depending on optimizer (x)

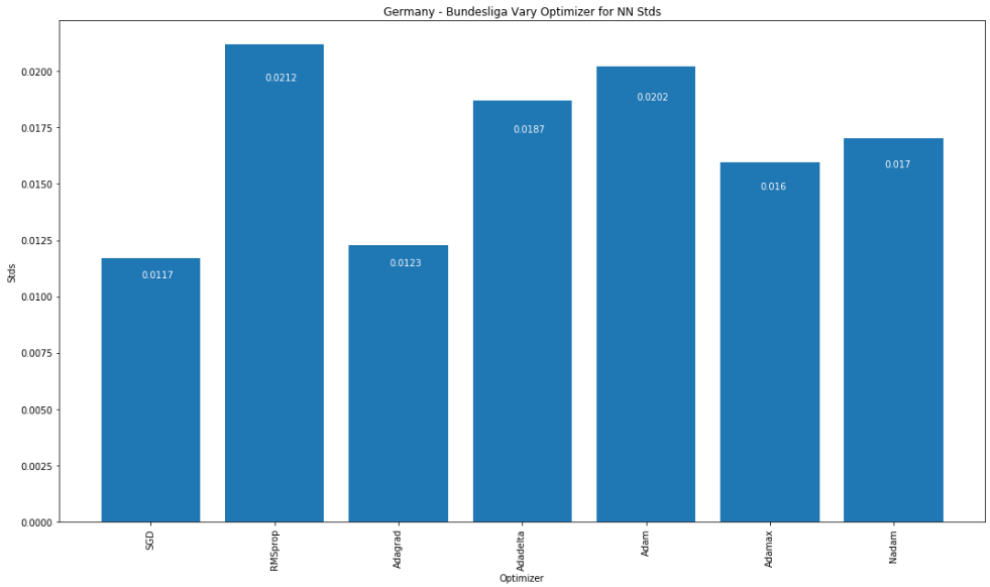

Figure W10: Germany Bundesliga accuracy std for train set (y) varying depending on optimizer (x)

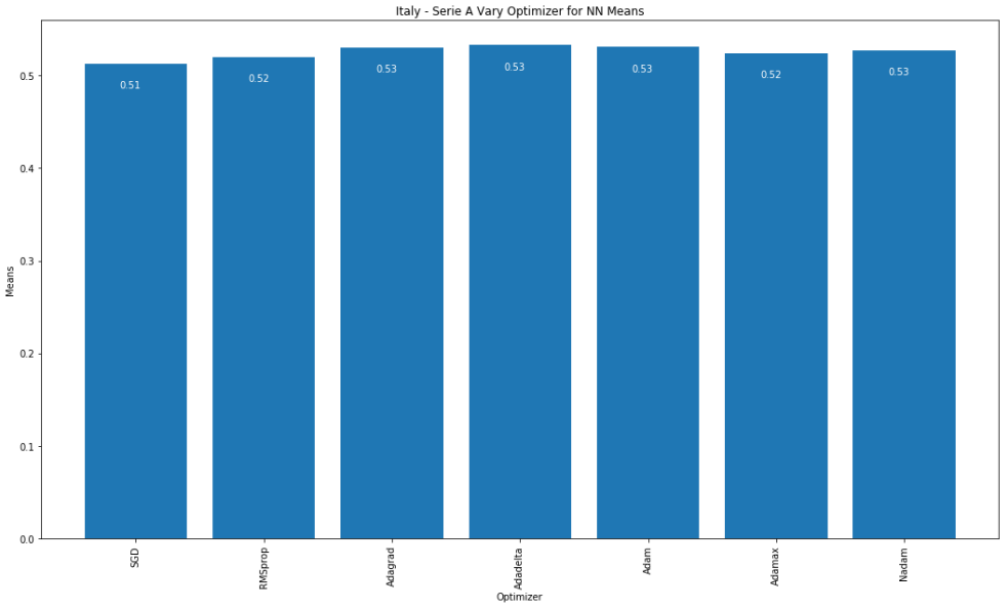

Figure W11: Italy Serie A accuracy for train set (y) varying depending on optimizer (x)

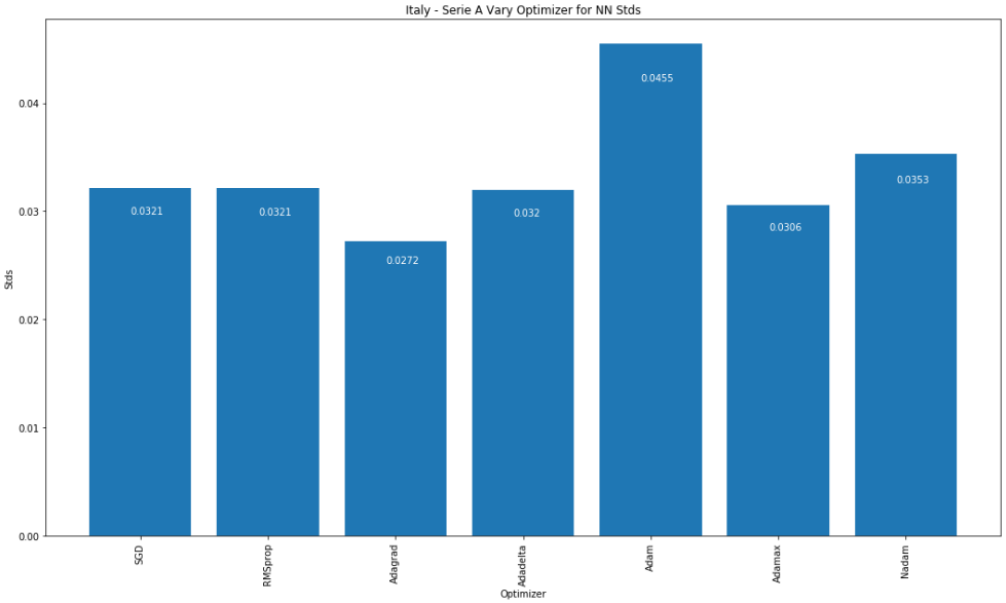

Figure W12: Italy Serie A accuracy std for train set (y) varying depending on optimizer (x)

Germany Bundesliga has similar accuracy differences as the ones that were observed in France Championnat while Italy Serie A is similar to England Premiership. However, what is different about both of them is the standard deviation. For both Germany Bundesliga and Italy Serie A, the best optimizer is Adagrad. Additional evidence of the performance of different optimizers can be observed in Figures W13 to W20.

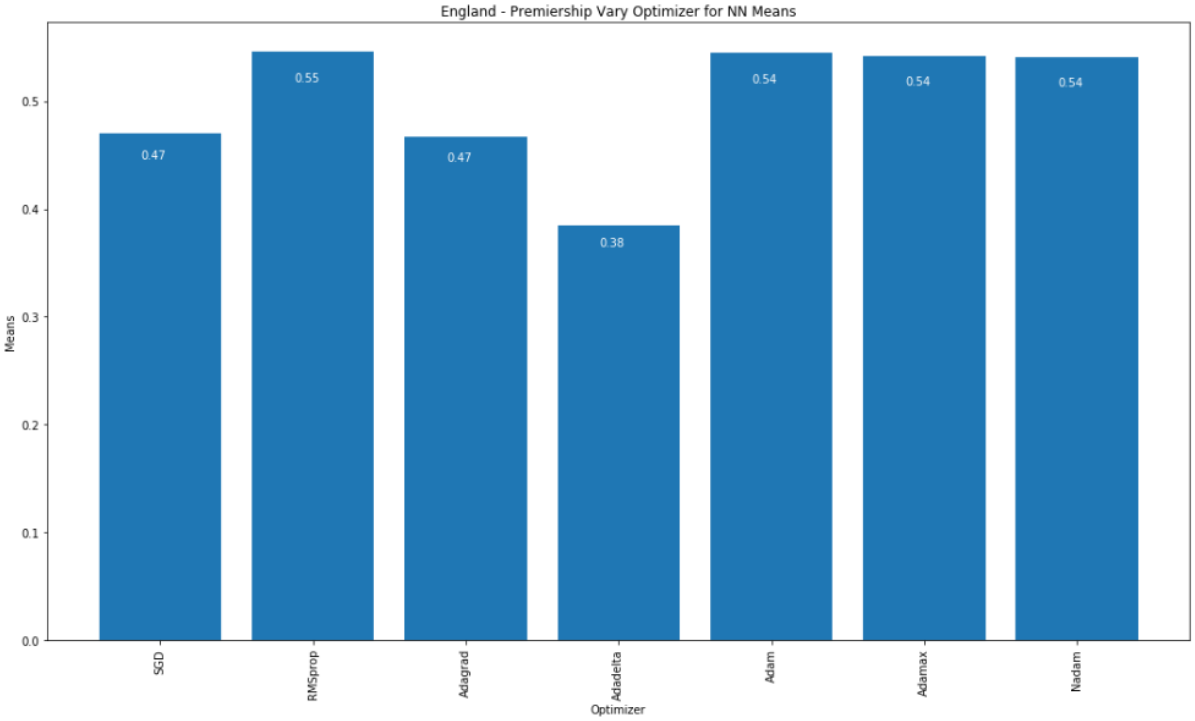

Figure W13: England Premiership accuracy for evaluation set (y) depending on optimizer (x)

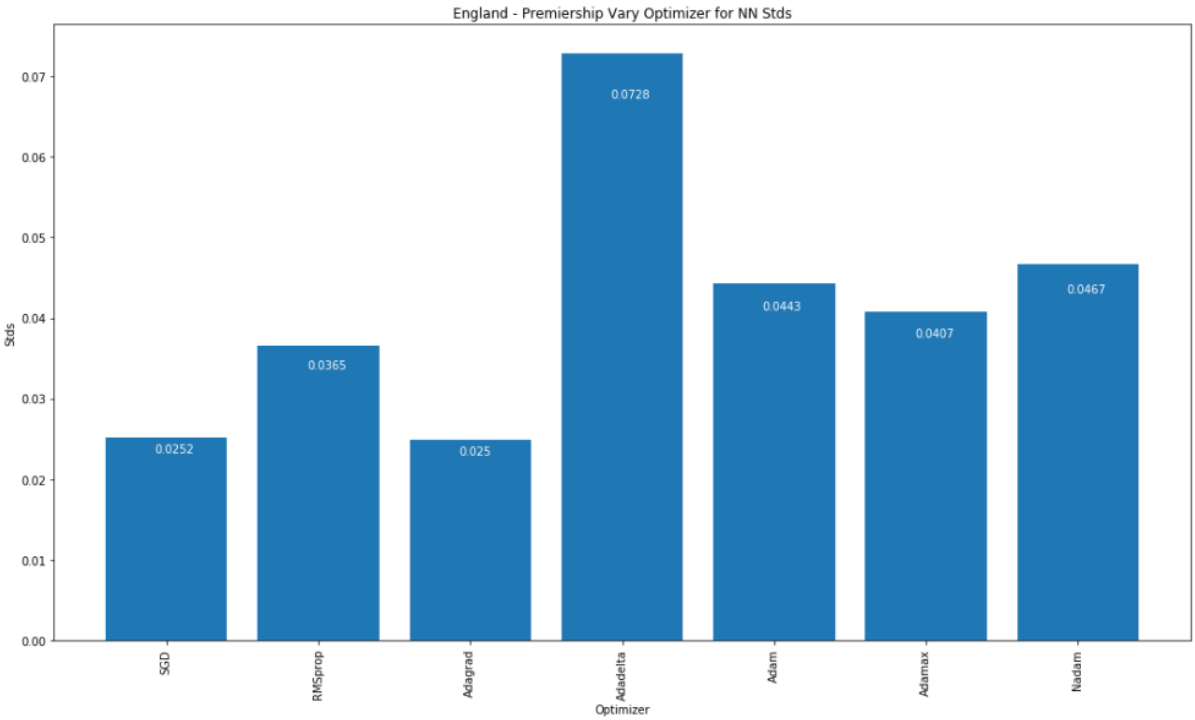

Figure W14: England Premiership accuracy standard deviation for evaluation set (y) depending on optimizer (x)

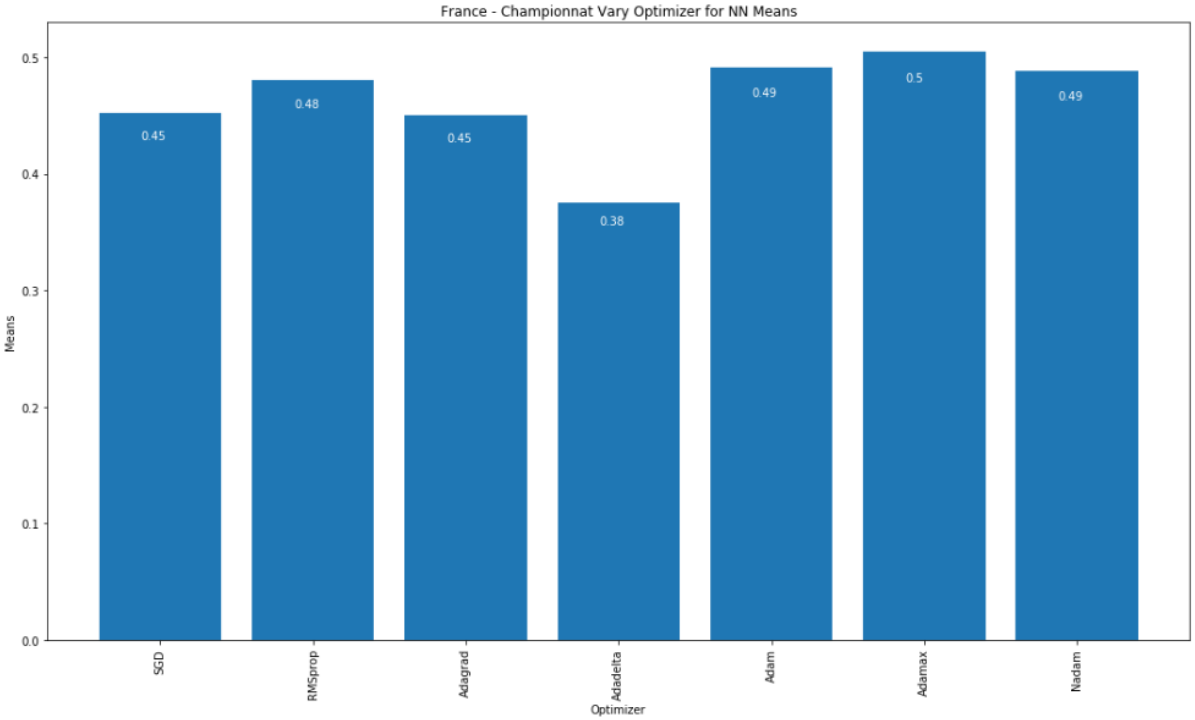

Figure W15: France Championnat accuracy for evaluation set (y) depending on optimizer (x)

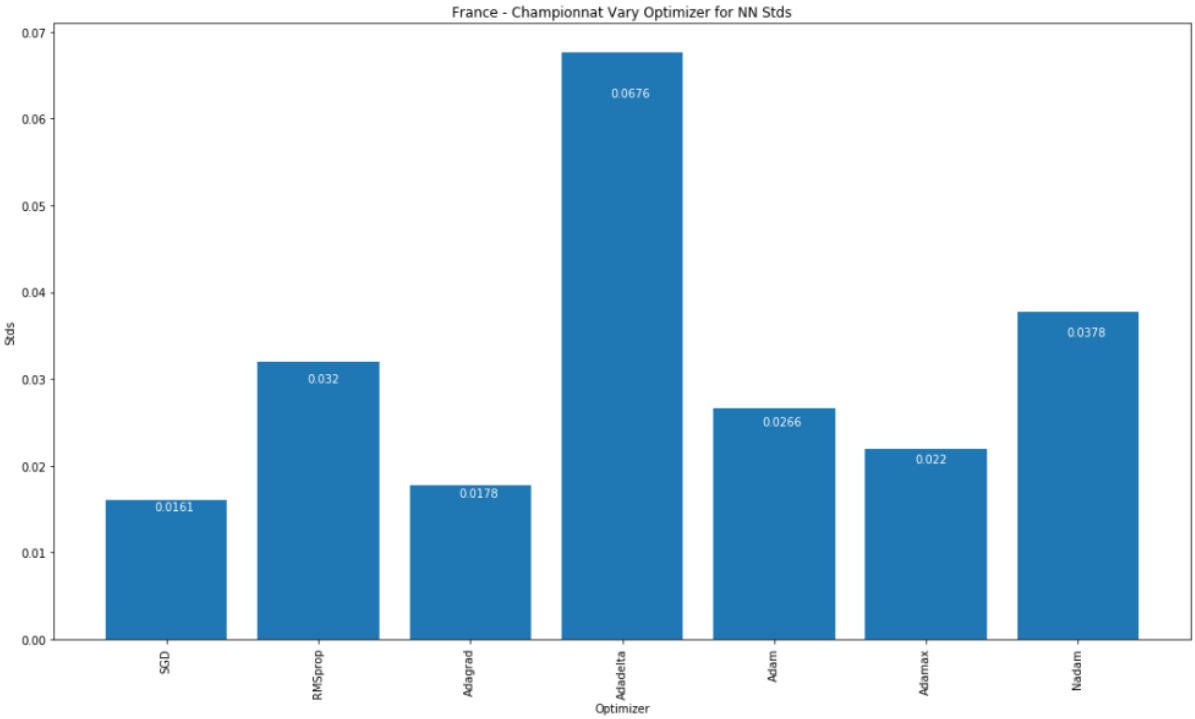

Figure W16: France Championnat accuracy standard deviation for evaluation set (y) depending on optimizer (x)

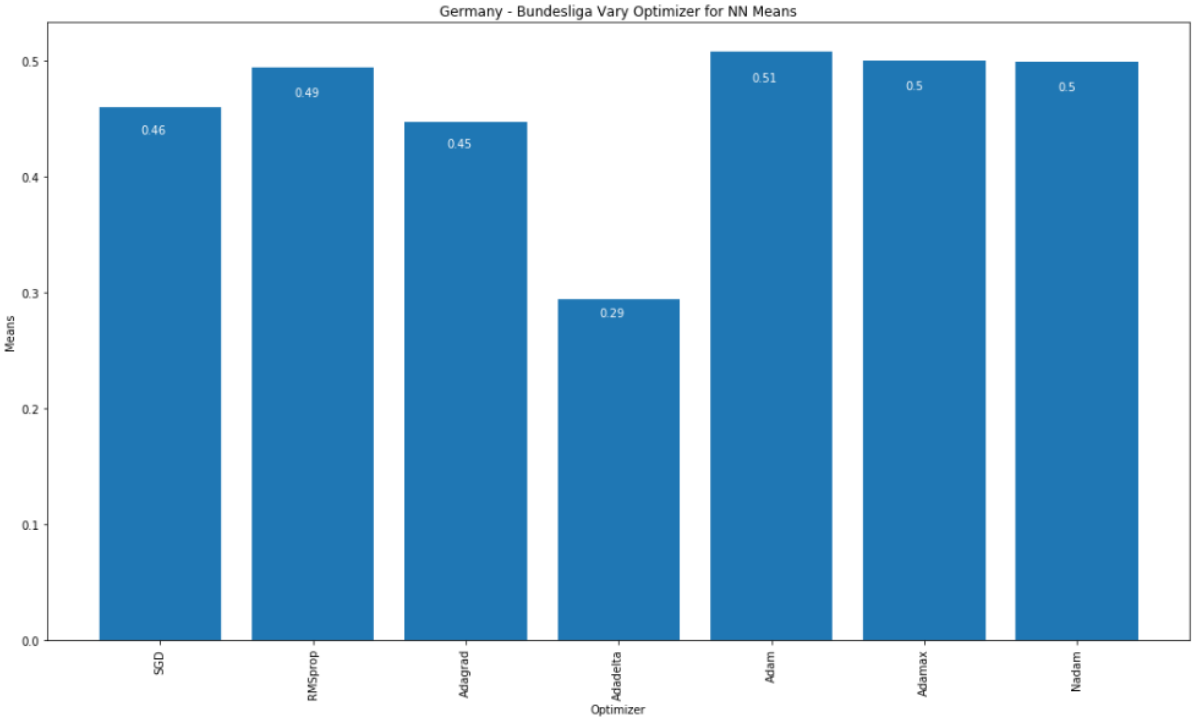

Figure W17: Germany Bundesliga accuracy for evaluation set (y) depending on optimizer (x)

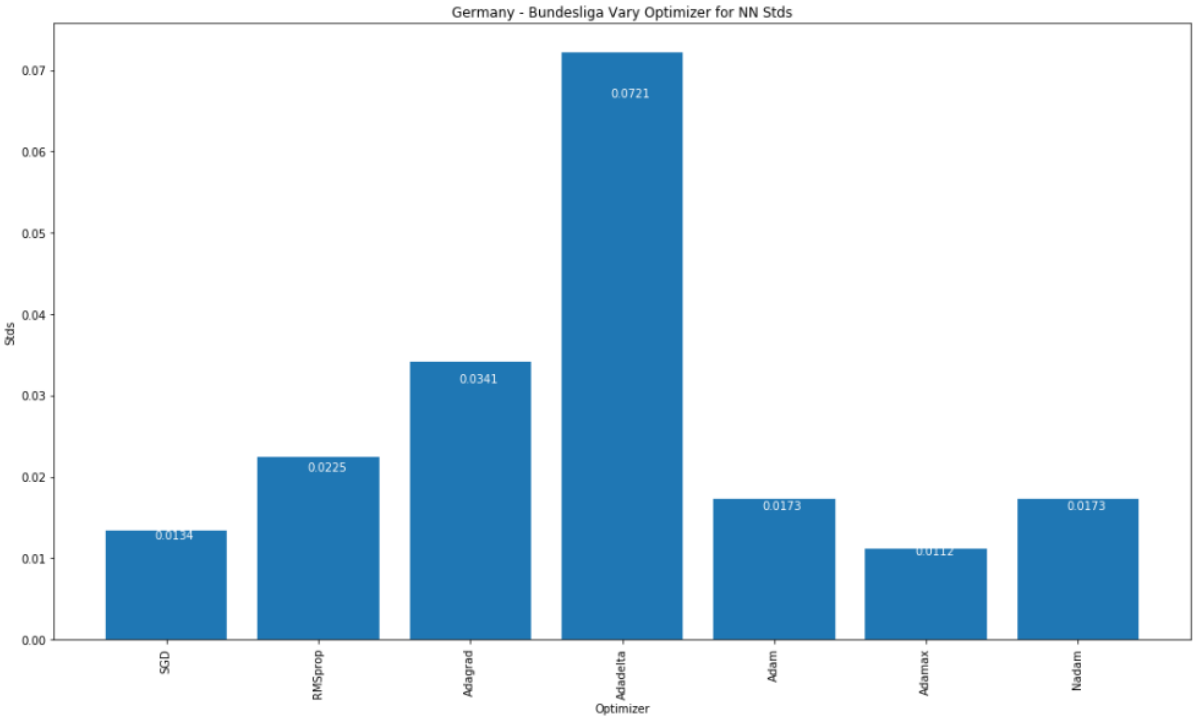

Figure W18: Germany Bundesliga accuracy standard deviation for evaluation set (y) depending on optimizer (x)

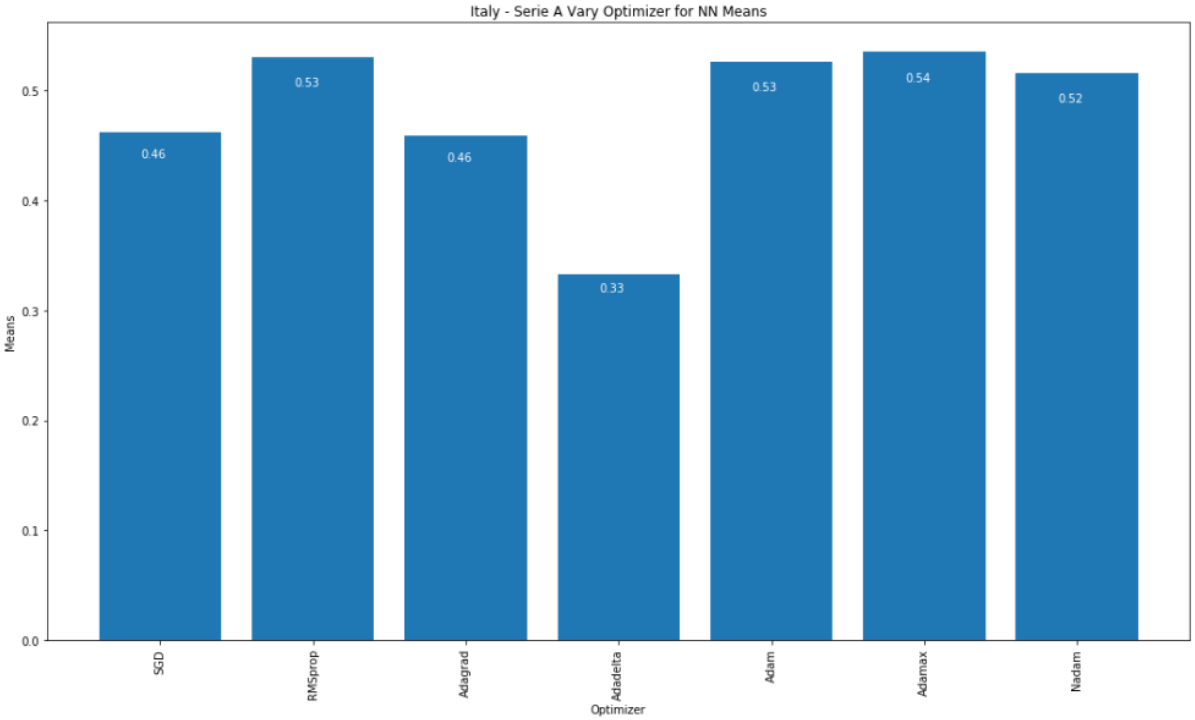

Figure W19: Italy Serie A accuracy for evaluation set (y) depending on optimizer (x)

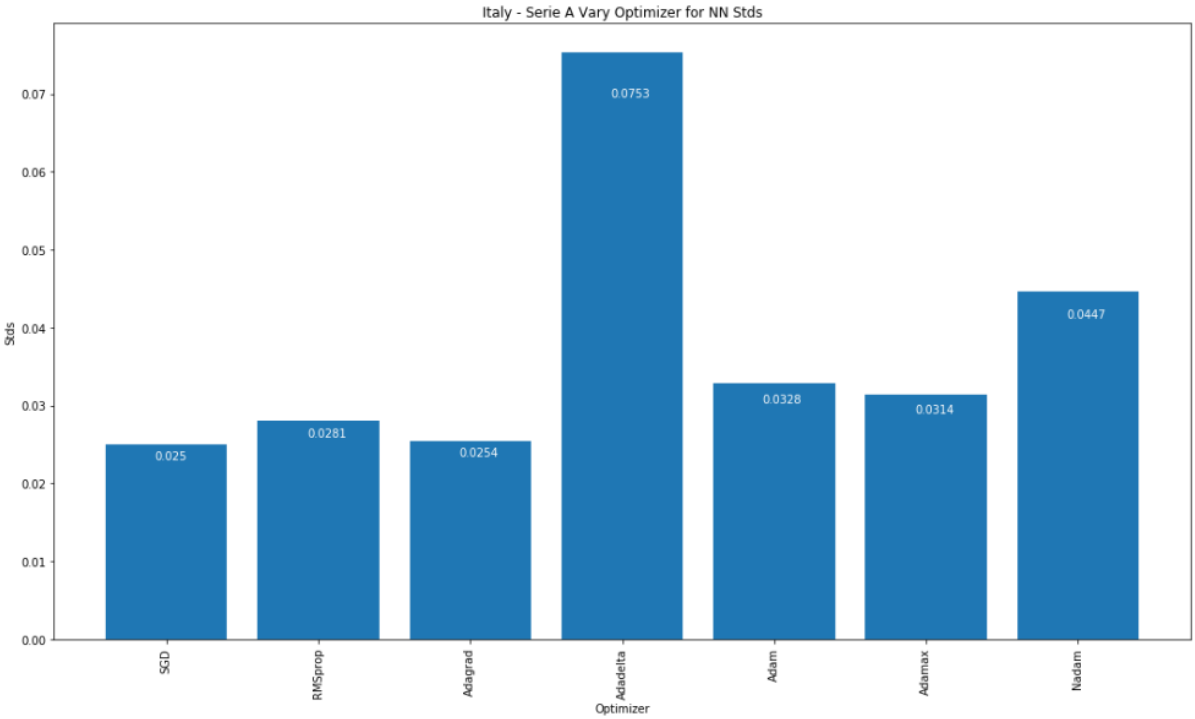

Figure W20: Italy Serie A accuracy standard deviation for evaluation set (y) depending on optimizer (x)

Appendix X: Batch Size and Number of Epochs Optimization for Neural Network

The results of varying both the batch size and number of epochs together can be observed in Figure X1.

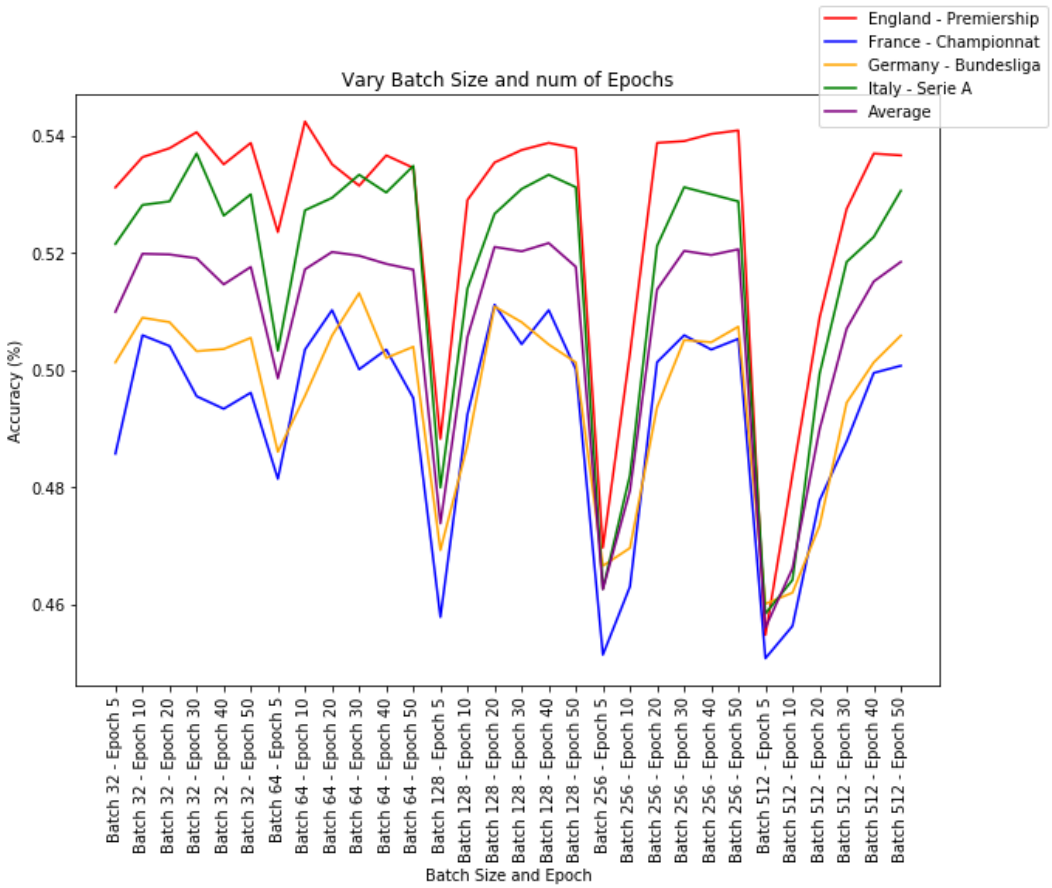

Figure X1: Competition accuracy for train set (y) varying depending on batch size and number of epochs (x)

The grid search determined that the optimal combination for batch size and number of epochs is 128 and 40 respectively (judging by the average accuracy). Considering the best values on a per-competition basis, it appears that for England Premiership, the optimal combination for batch size and number of epochs is 64 and 50 respectively. Of course, overfitting is an important concern here so using these batch size and epoch pairs, training graphs for loss and accuracy are produced. Based on those graphs, it appears that for some period of the training, the models are not improving their accuracy. For example, for England Premiership, the loss decreased by 0.01 in a period of 20 epochs (from 30 to 50), while it decreased by 2.02 in epochs 1 to 20. Moreover, the models did not improve the accuracy during the period from 30 to 50 epochs. Hence, to avoid the model from overfitting and to reduce training time by removing the unnecessary epochs, the optimal number of epochs for England Premiership is selected as 30.

Next, for France Championnat, the optimal values for the batch size and number of epochs as a result of a grid search was found to be 128 and 50 respectively. A similar trend that was

discovered for the England Premiership is also true for France Championnat. So, the same batch size and epochs combination of 128 and 30 is applied for France Championnat. Interestingly, for Germany Bundesliga, the best pair of values that was found with a grid search is 64 and 50, which was also found for Italy Serie A. Germany Bundesliga has the lowest learning rate amongst all considered competitions. Given this fact, the 50 epochs during the training process appears to be a good choice for this particular problem. This is motivated by the fact that up to about 50 epochs, the model is experiencing a drop in the loss function and an increase in accuracy that is not insignificant and does not look like a typical case of overfitting. However, the batch size of 64 does seem to increase the amplitude of the spikes in the training. For this reason, the batch size is left at 128. The statement about the batch size is true for Italy Serie A as well. Lastly, for Italy Serie A, the final number of epochs that is selected is 30 for the same reasons as with England Premiership and France Championnat - the model does not appear to improve significantly after about 30 epochs so any value above that is likely to only introduce some overfitting to the model. More evidence on the performance and training process of the Neural Network when using different batch size and number of epochs are presented in Figures X2 through X34.

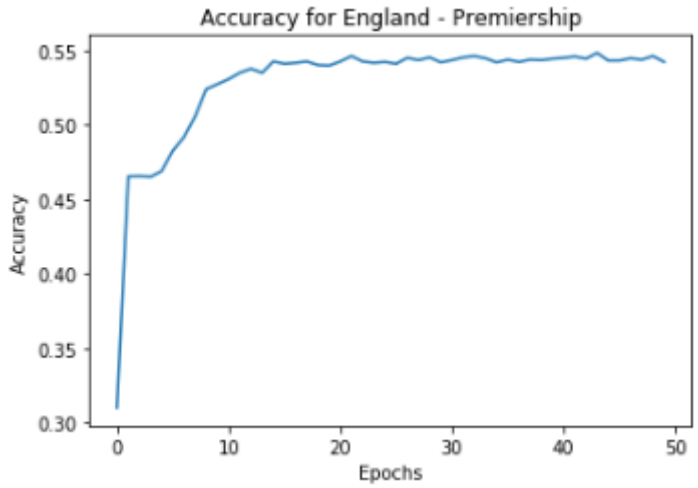

Figure X2: England Premiership accuracy change during the model training with batch size 128 and 40 epochs

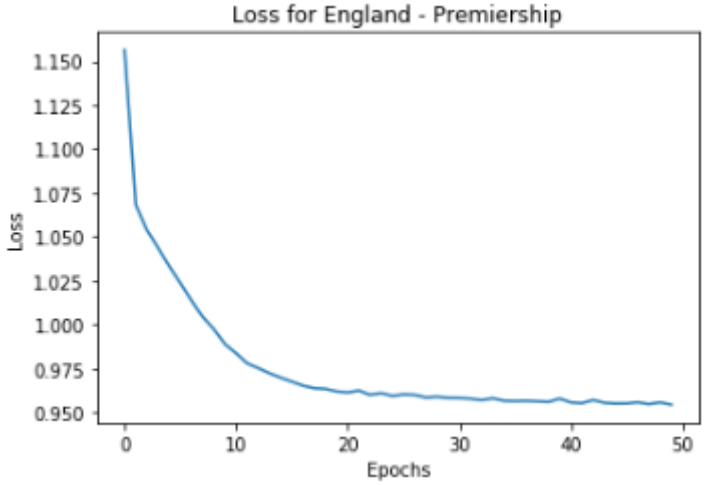

Figure X3: England Premiership accuracy change during the model training with batch size 128 and 40 epochs

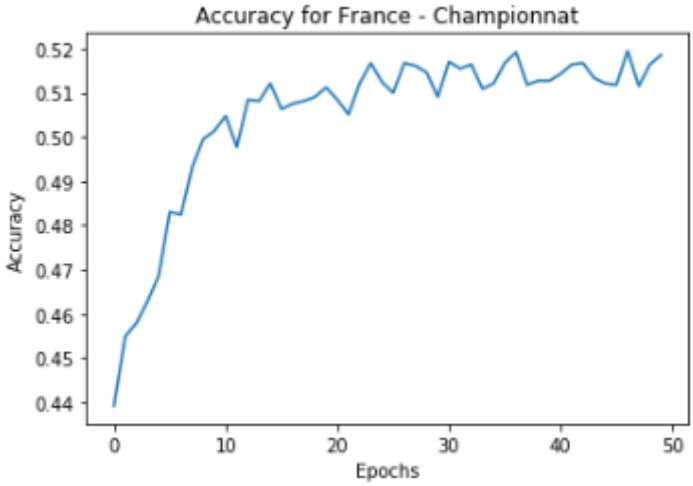

Figure X4: France Championnat accuracy change during the model training with batch size 128 and 40 epochs

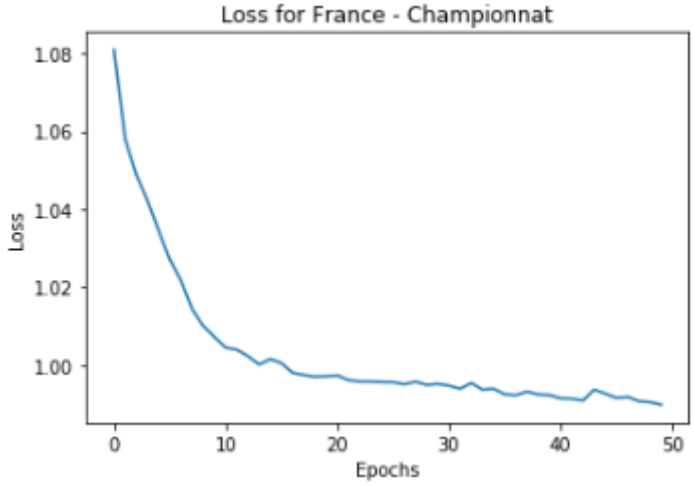

Figure X5: France Championnat accuracy change during the model training with batch size 128 and 40 epochs

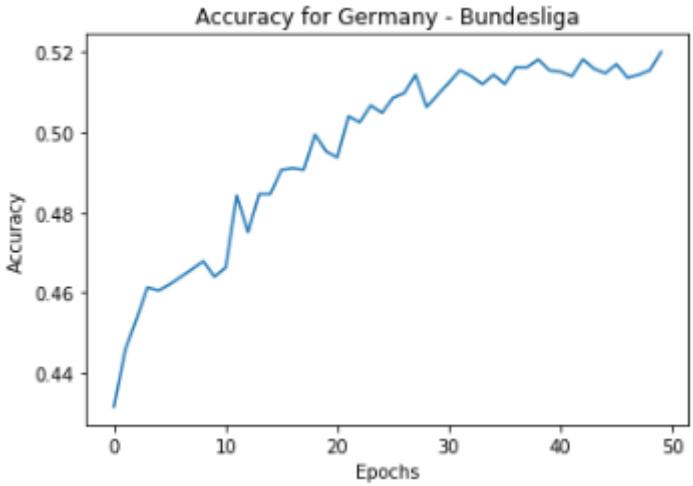

Figure X6: Germany Bundesliga accuracy change during the model training with batch size 128 and 40 epochs

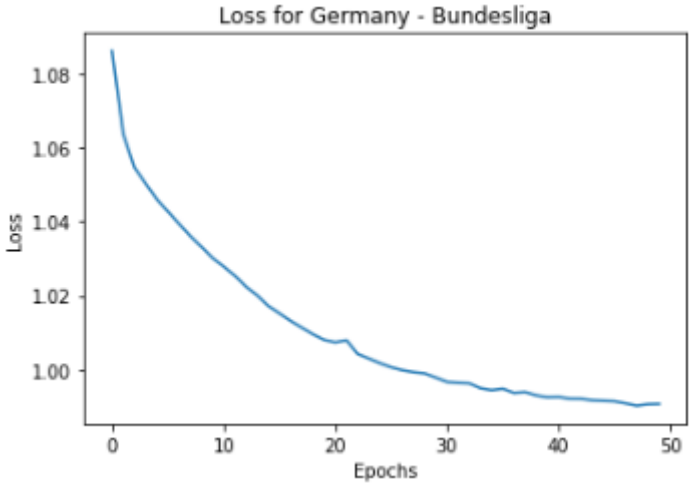

Figure X7: Germany Bundesliga accuracy change during the model training with batch size 128 and 40 epochs

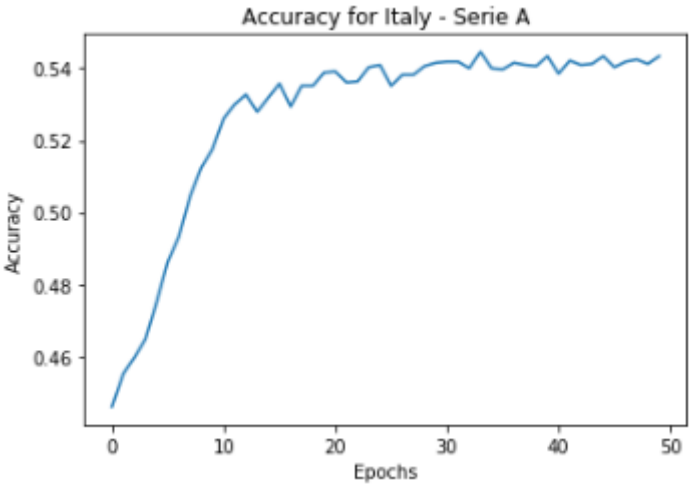

Figure X8: Italy Serie A accuracy change during the model training with batch size 128 and 40 epochs

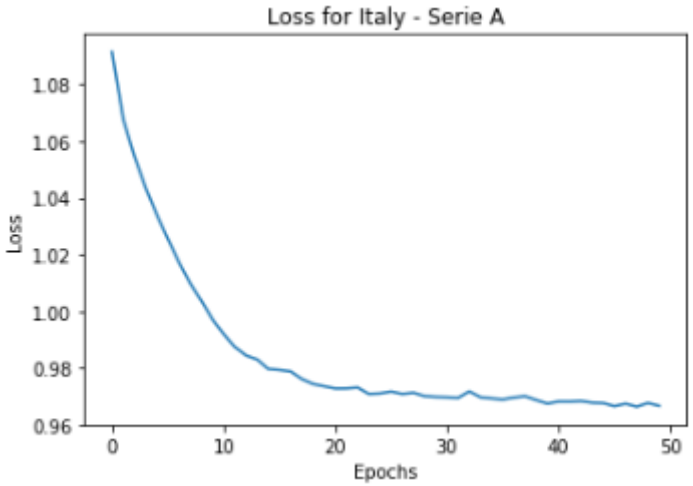

Figure X9: Italy Serie A accuracy change during the model training with batch size 128 and 40 epochs

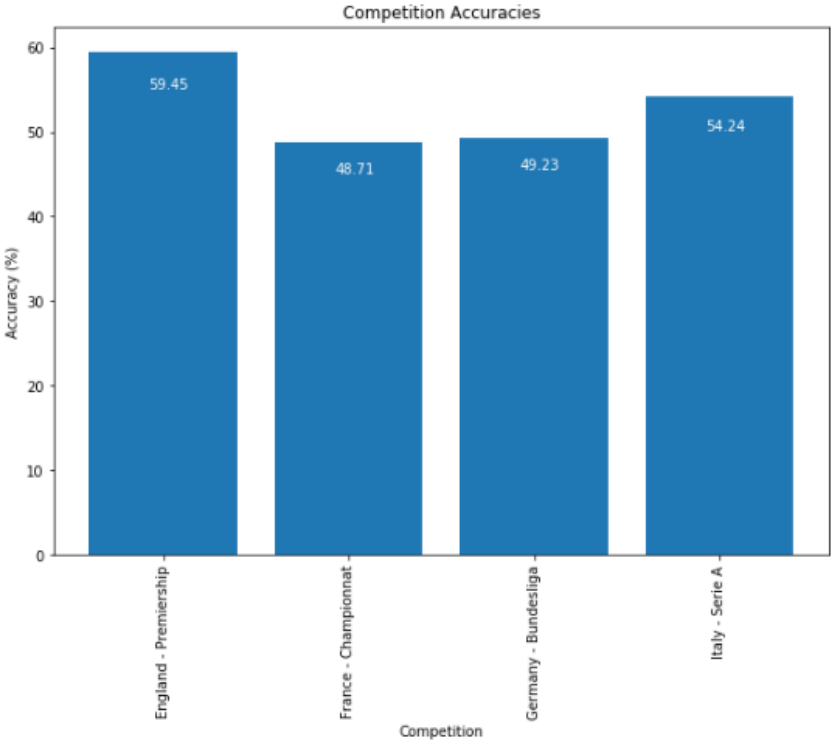

Figure X10: Competition accuracies for 2018 - 2019 tournaments with batch size 128 and 50 epochs

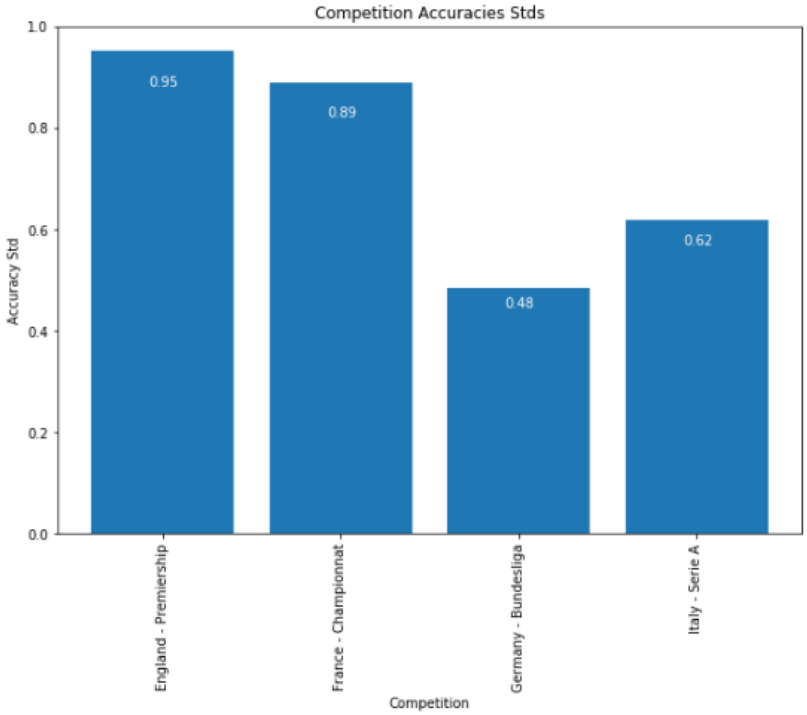

Figure X11: Competition accuracies std for 2018 - 2019 tournaments with batch size 128 and 50 epochs

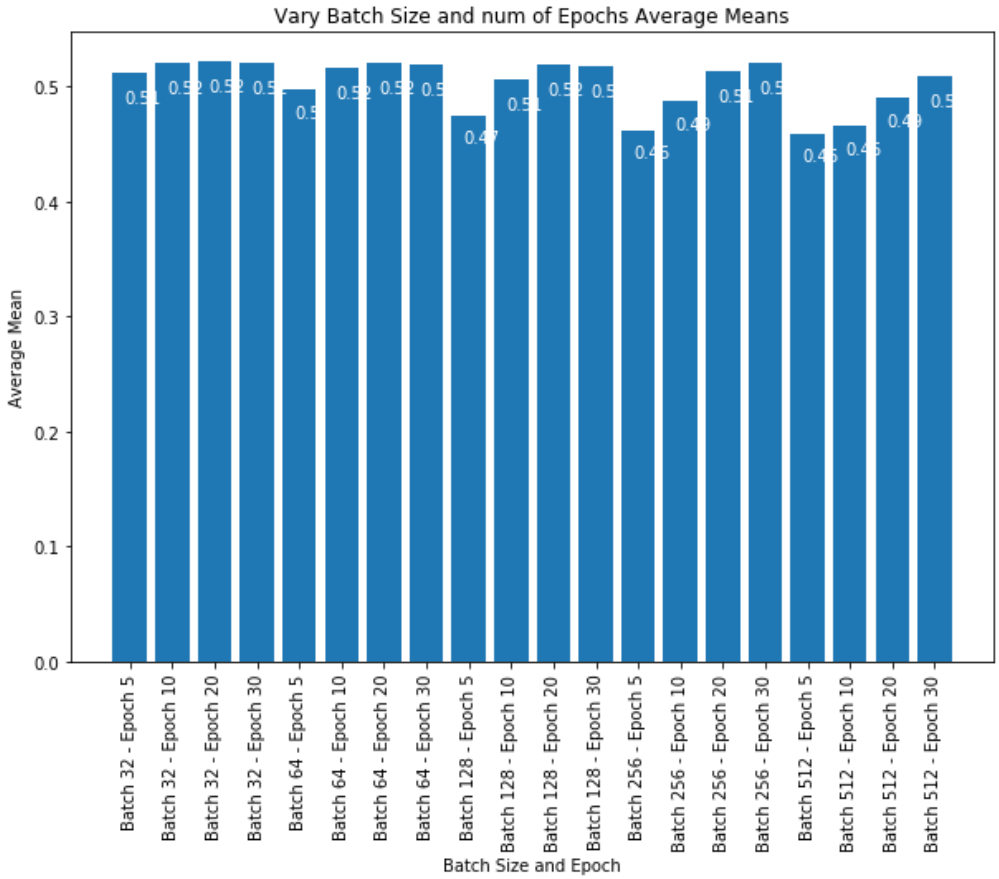

Figure X12: Overall accuracy for train set (y) varying depending on batch size and number of epochs (x)

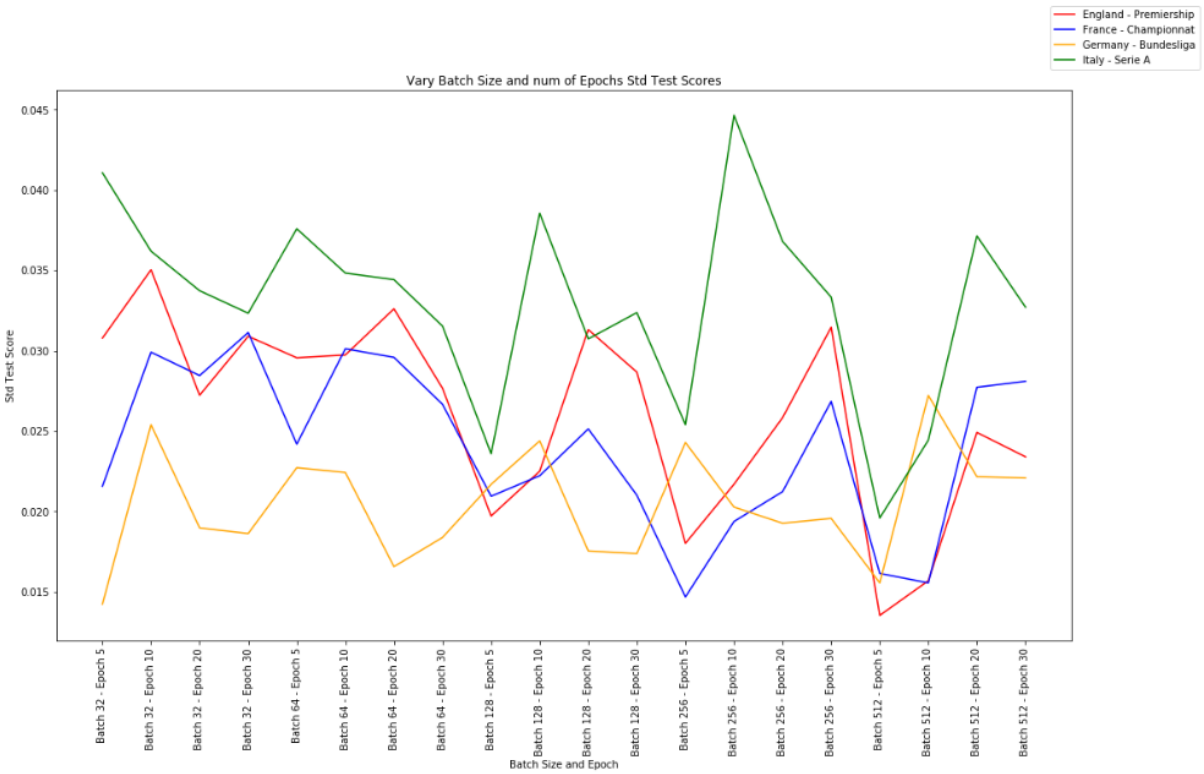

Figure X13: Standard Deviation for train set (y) varying depending on batch size and number of epochs (x)

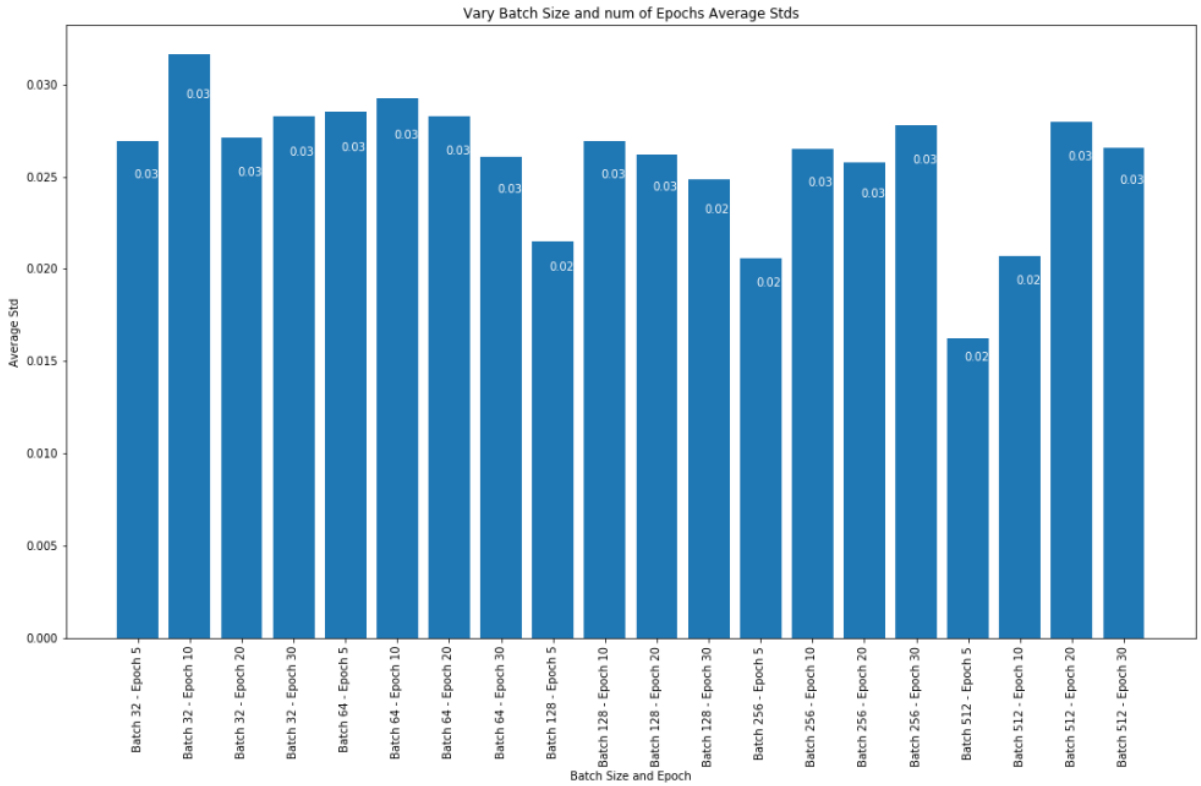

Figure X14: Standard Deviation for train set (y) varying depending on batch size and number of epochs (x)

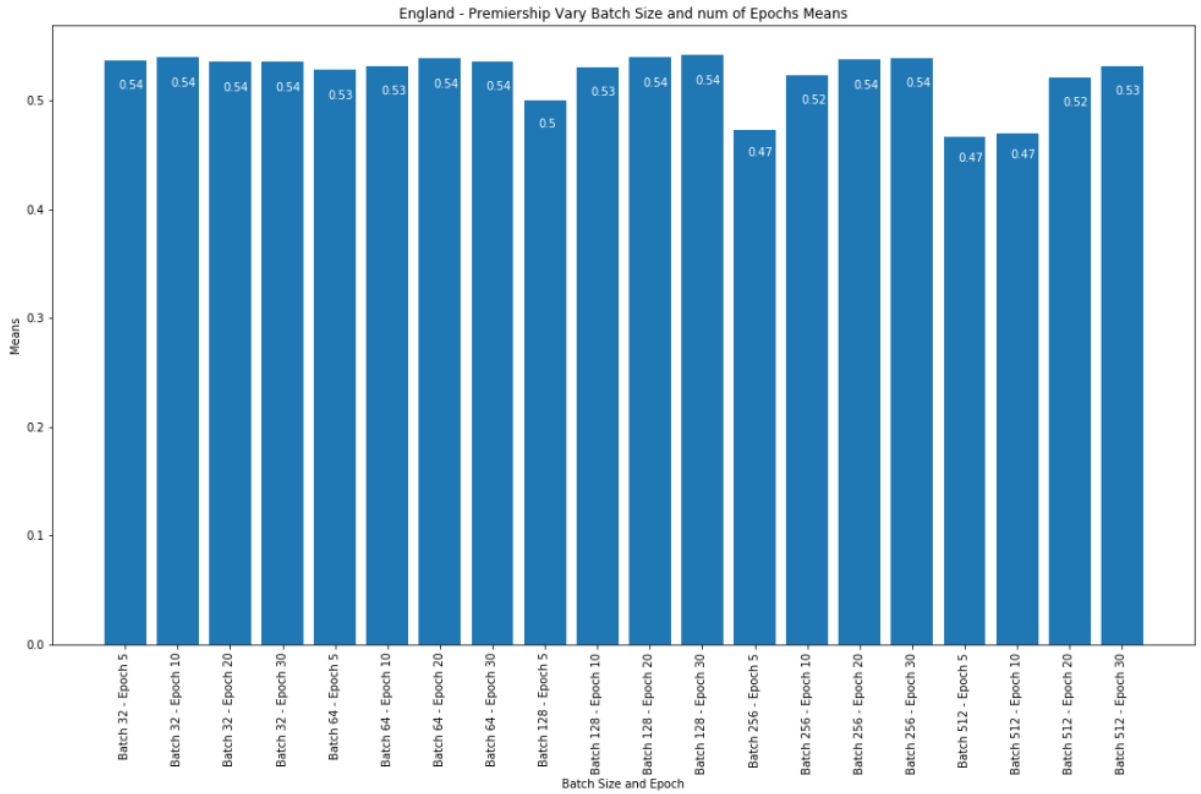

Figure X15: Accuracy for train set (y) varying depending on batch size and number of epochs (x) for England Premiership

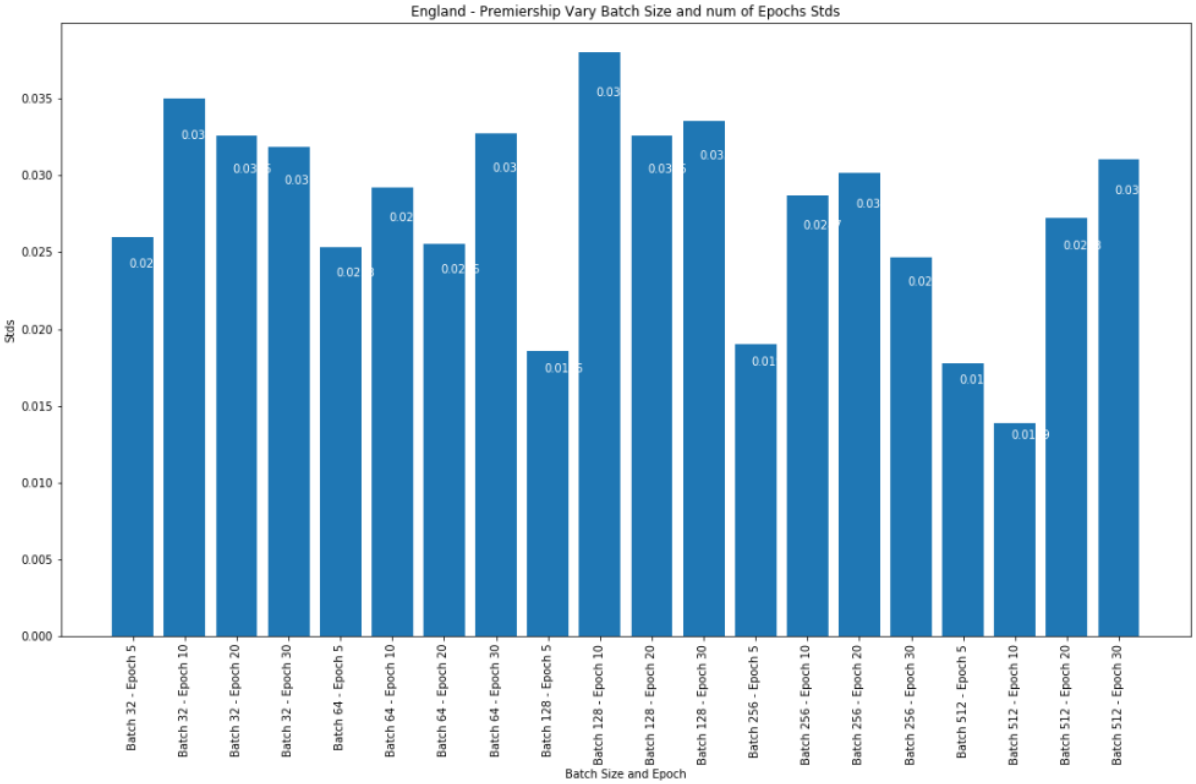

Figure X16: Accuracy Std for train set (y) varying depending on batch size and number of epochs (x) for England Premiership

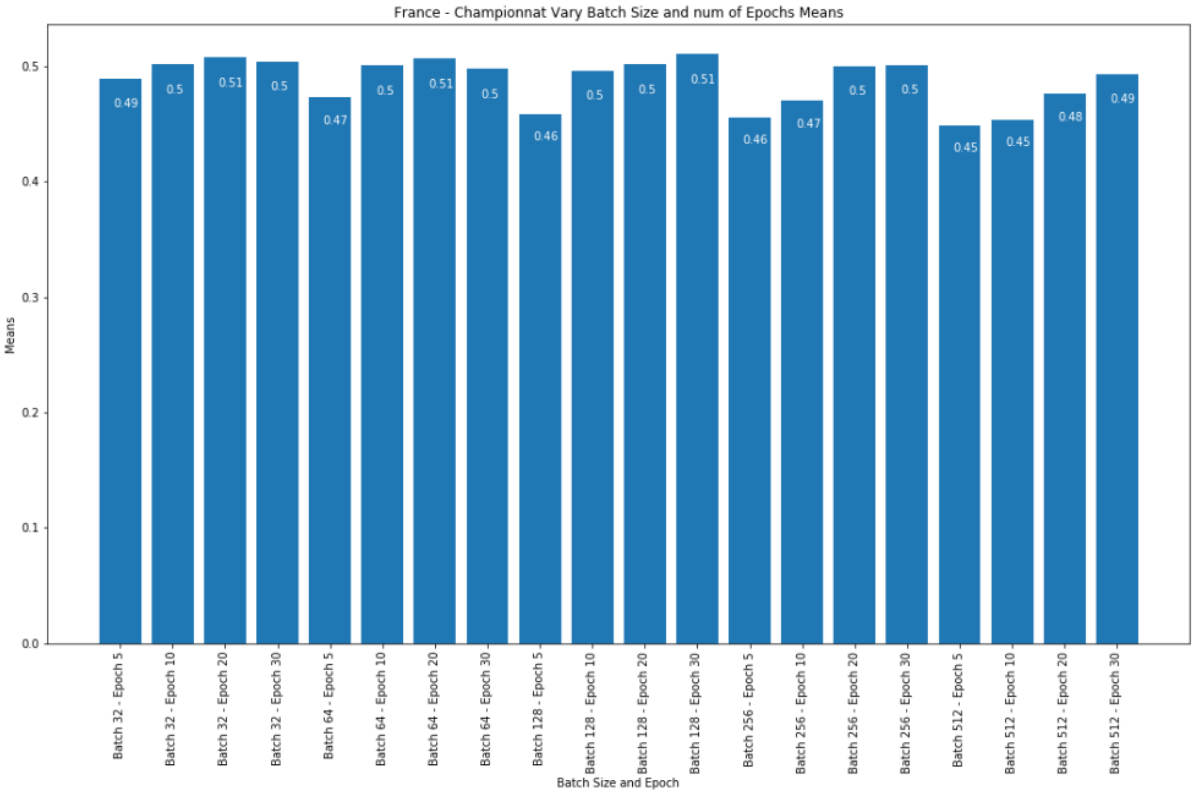

Figure X17: Accuracy for train set (y) varying depending on batch size and number of epochs (x) for France Championnat

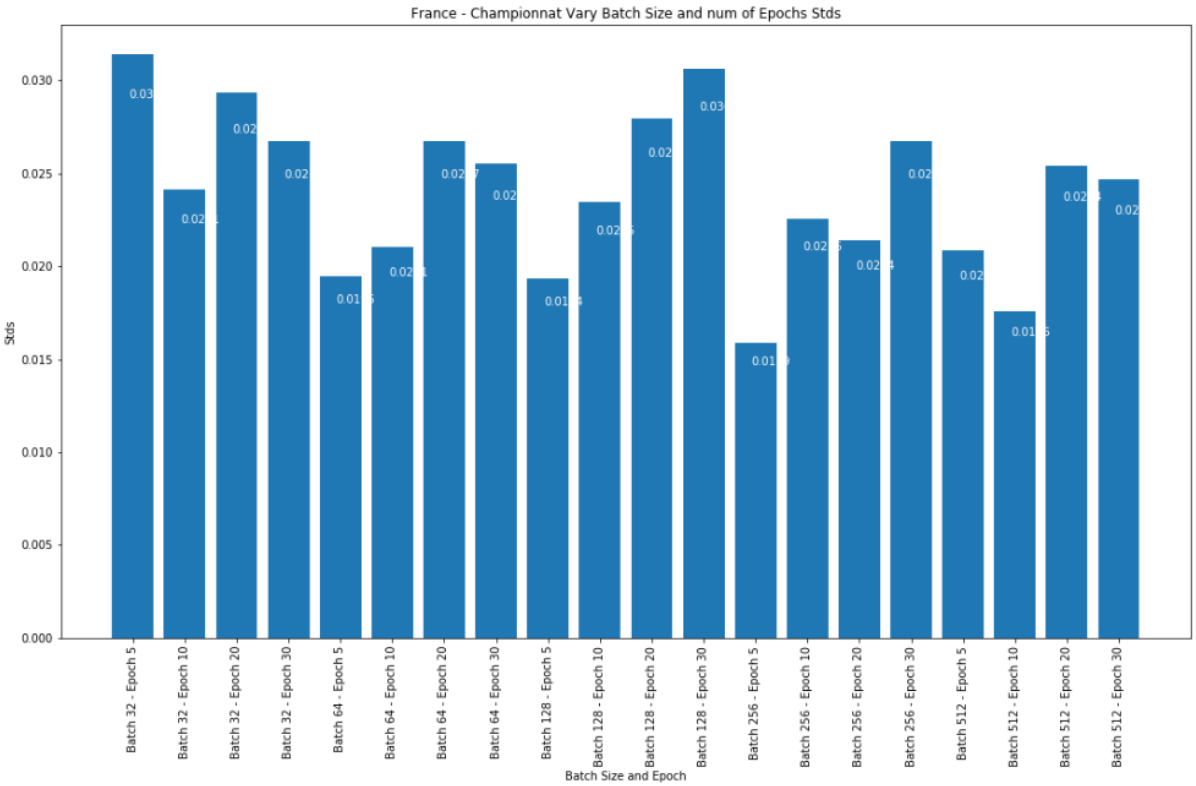

Figure X18: Accuracy Std for train set (y) varying depending on batch size and number of epochs (x) for France Championnat

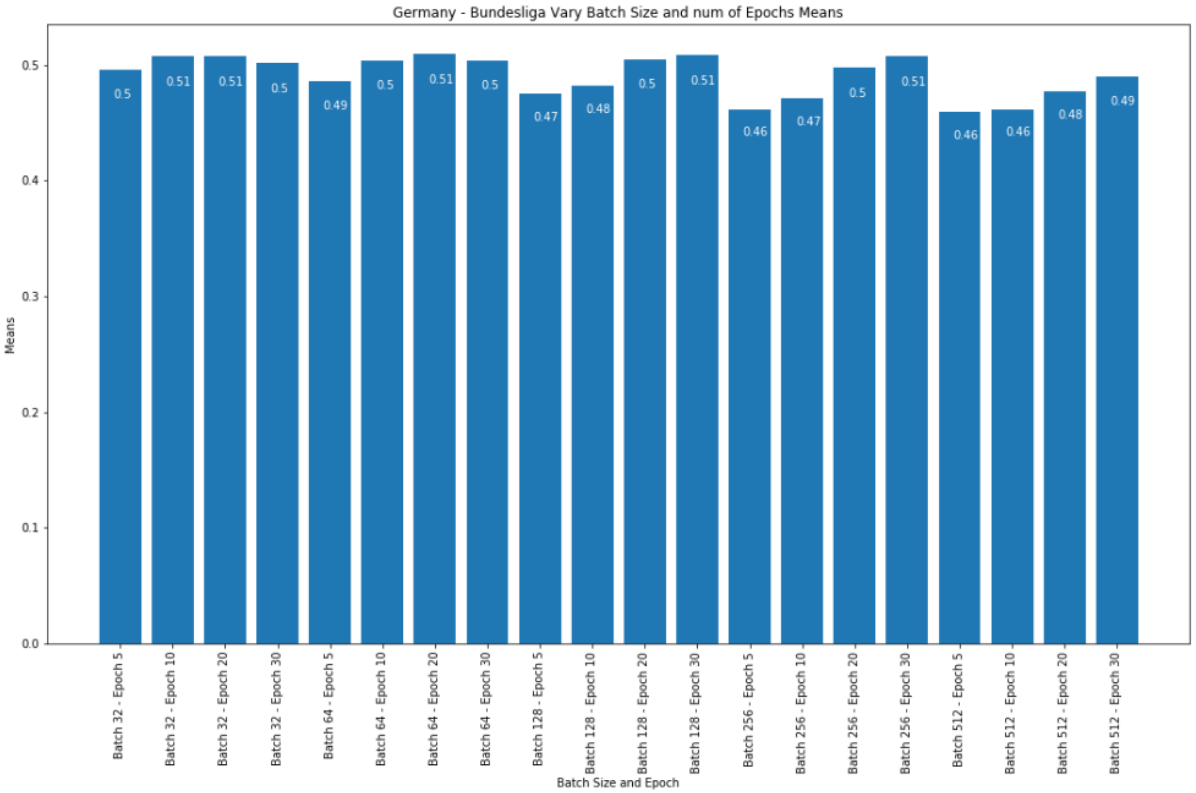

Figure X19: Accuracy for train set (y) varying depending on batch size and number of epochs (x) for Germany Bundesliga

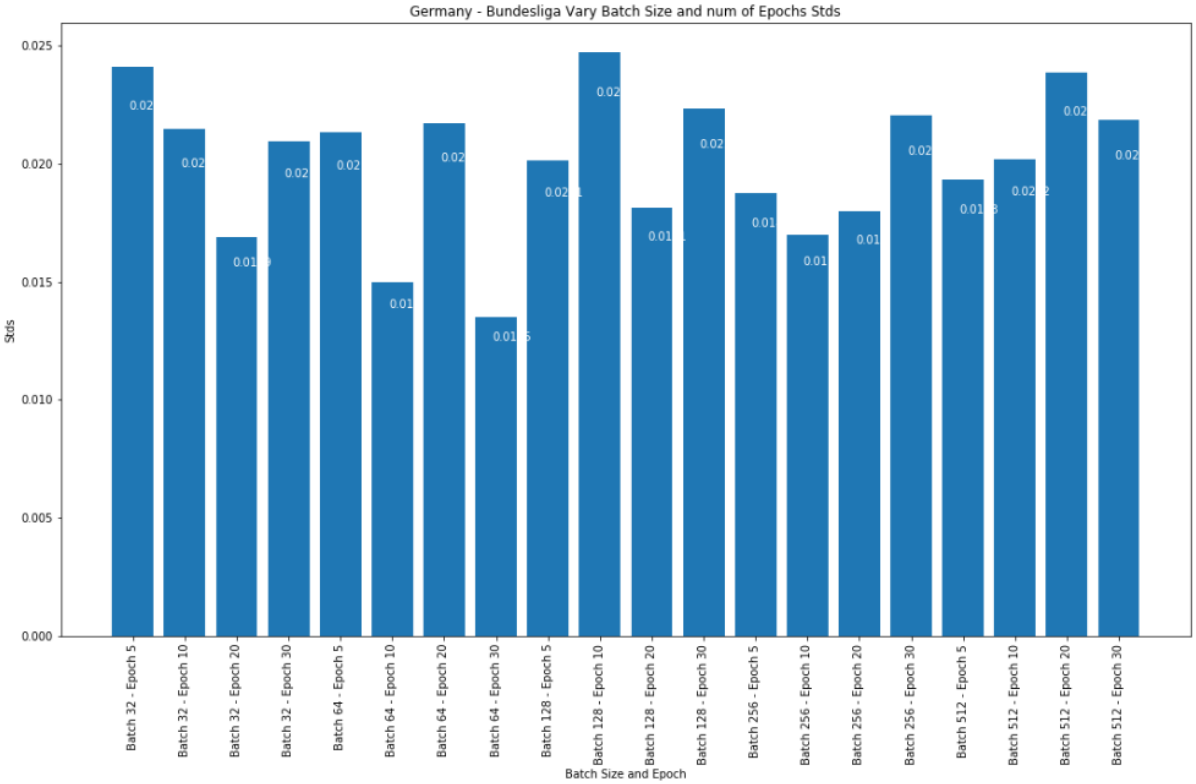

Figure X20: Accuracy Std for train set (y) varying depending on batch size and number of epochs (x) for Germany Bundesliga

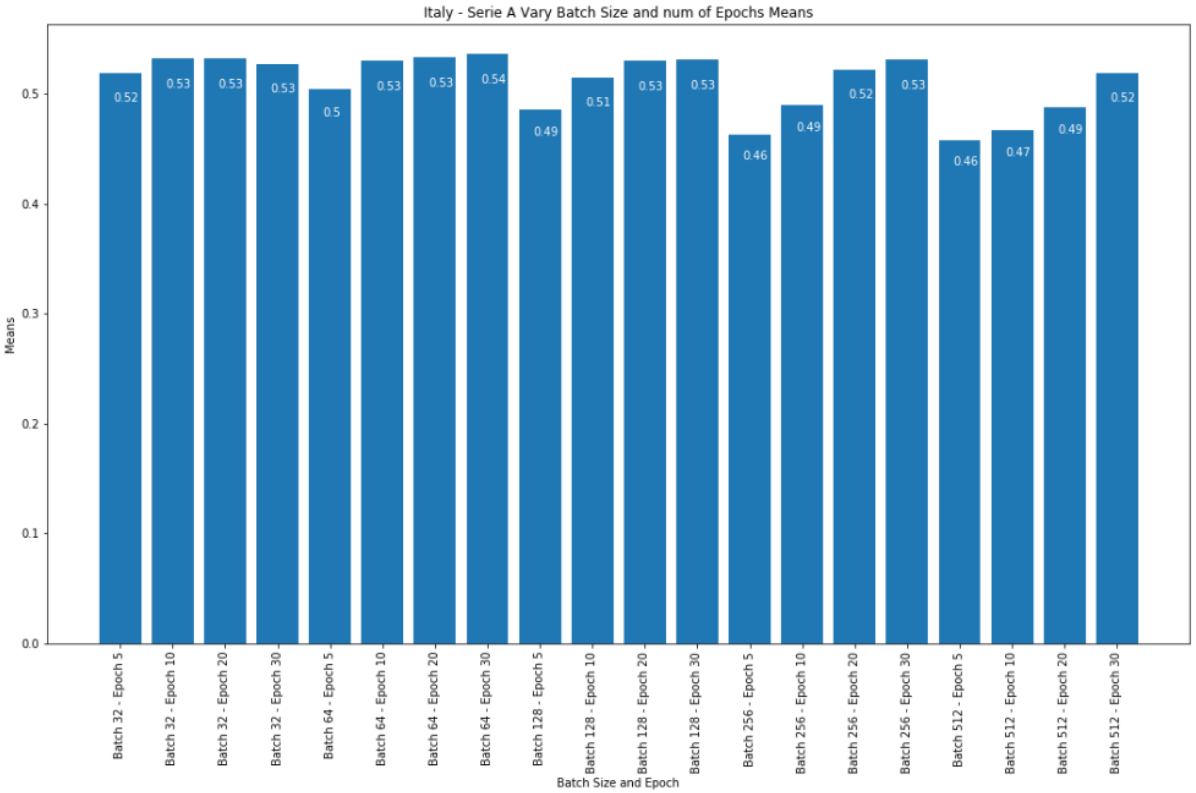

Figure X21: Accuracy for train set (y) varying depending on batch size and number of epochs (x) for Italy Serie A

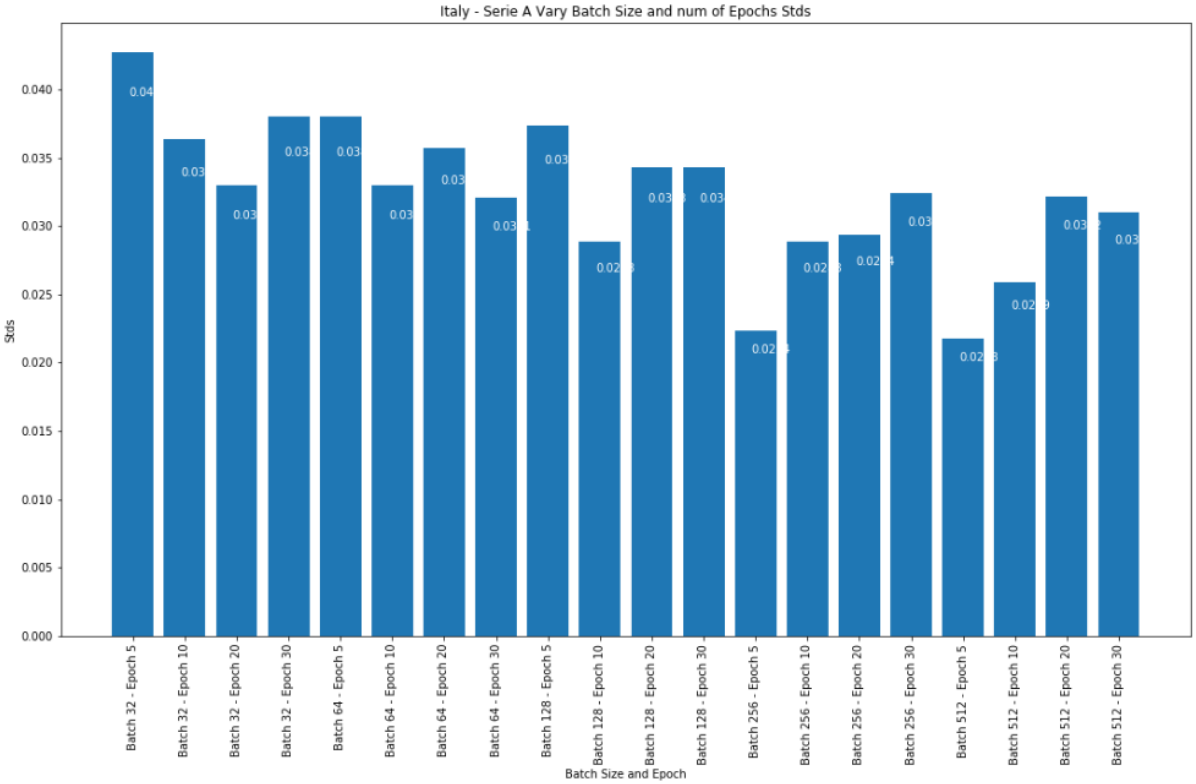

Figure X22: Accuracy for train set (y) varying depending on batch size and number of epochs (x) for Italy Serie A

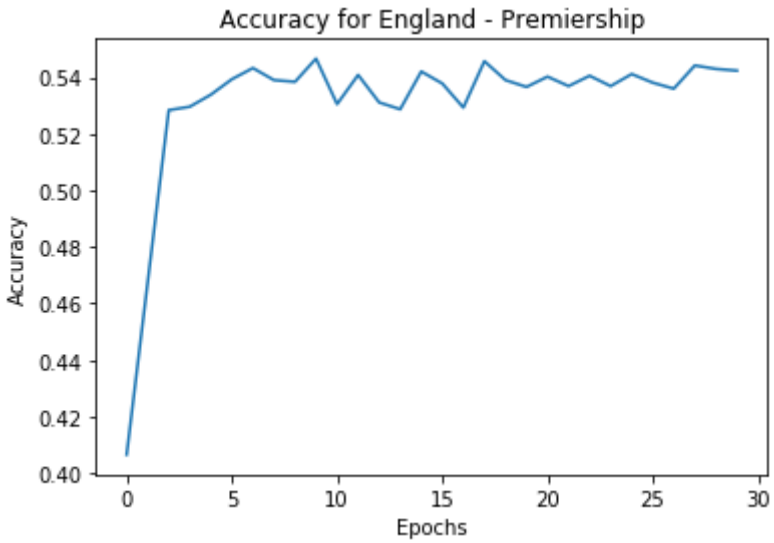

Figure X23: England Premiership train accuracy change with 128 batch size and 30 epochs

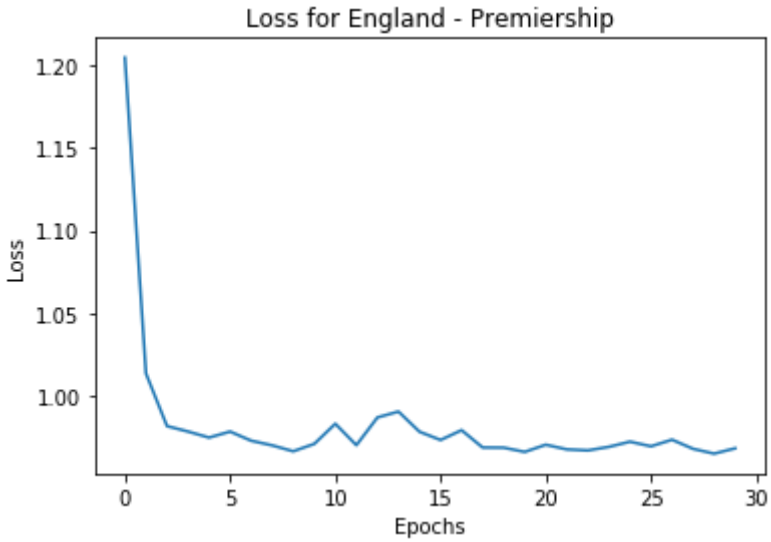

Figure X24: England Premiership train loss change with 128 batch size and 30 epochs

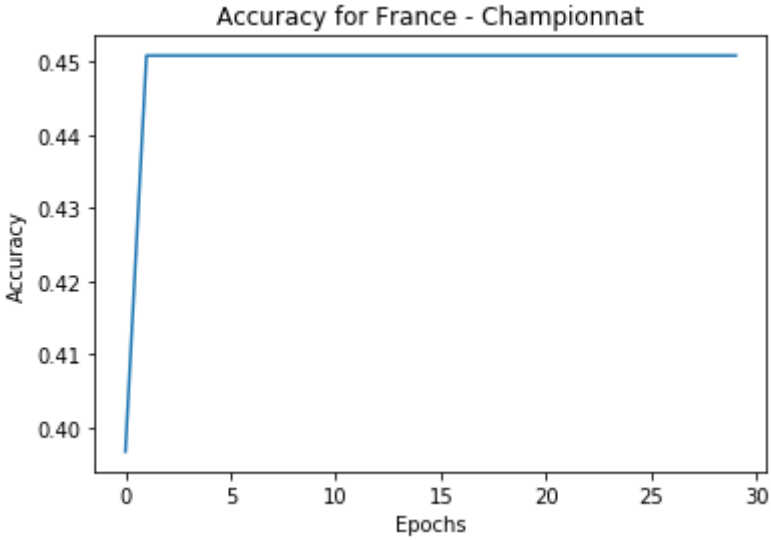

Figure X25: France Championnat train accuracy change with 128 batch size and 30 epochs

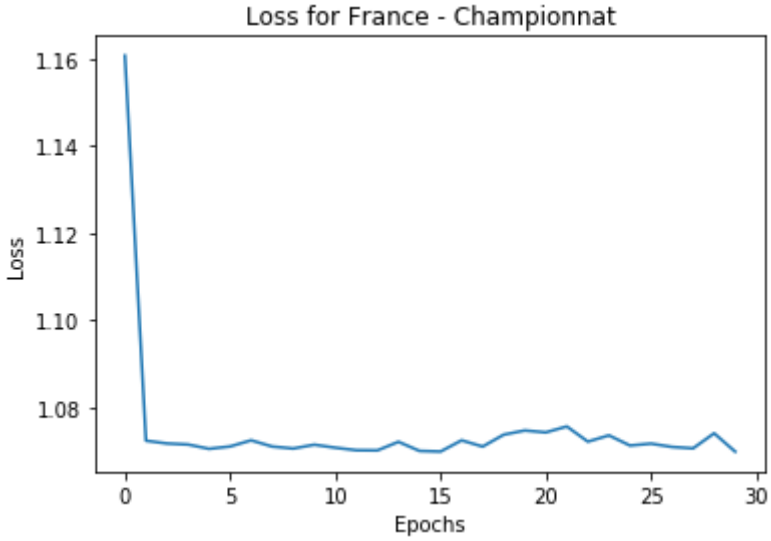

Figure X26: France Championnat train loss change with 128 batch size and 30 epochs

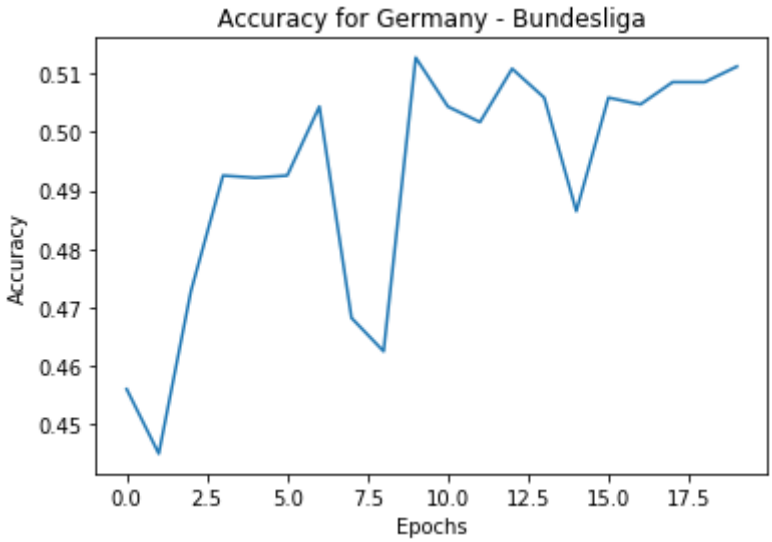

Figure X27: Germany Bundesliga train accuracy change with 64 batch size and 20 epochs

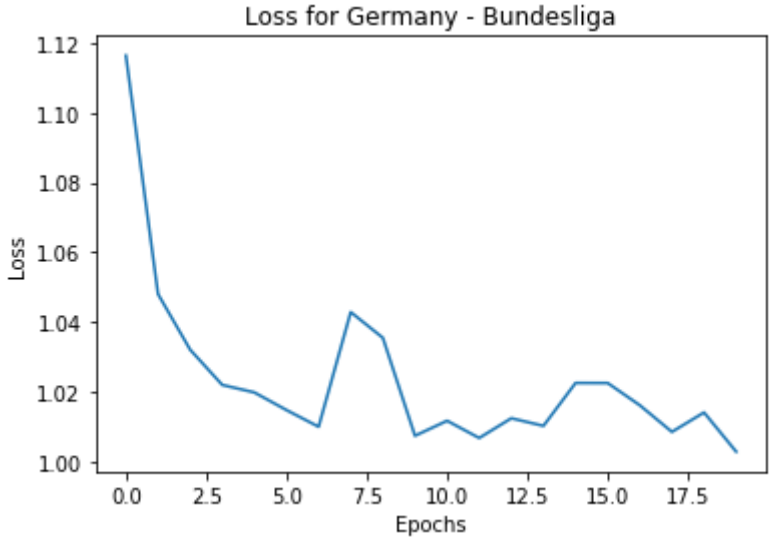

Figure X28: Germany Bundesliga train loss change with 64 batch size and 20 epochs

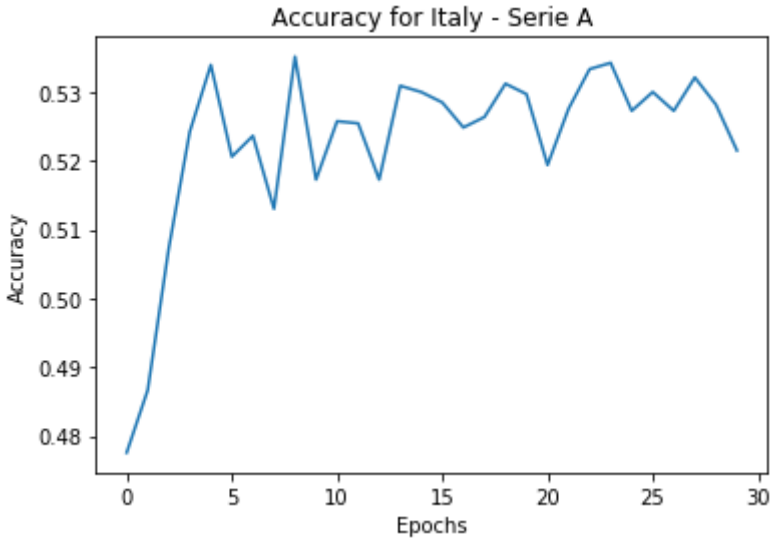

Figure X29: Italy Serie A train accuracy change with 64 batch size and 30 epochs

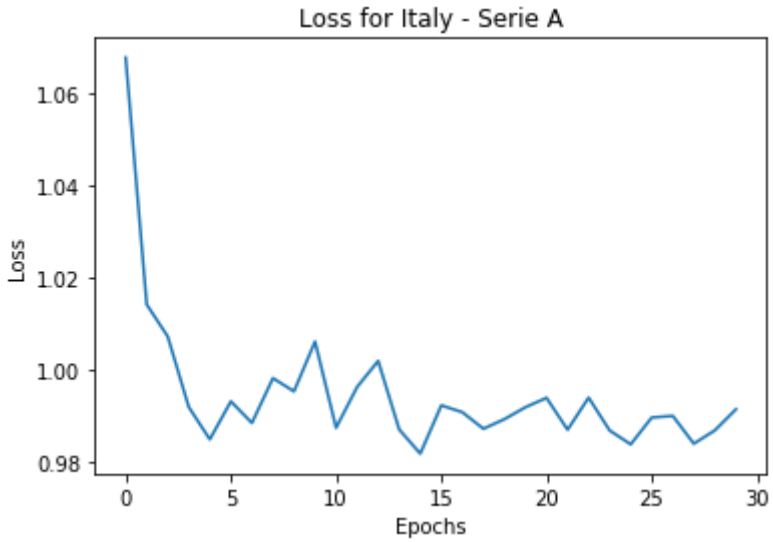

Figure X30: Italy Serie A train loss change with 64 batch size and 30 epochs

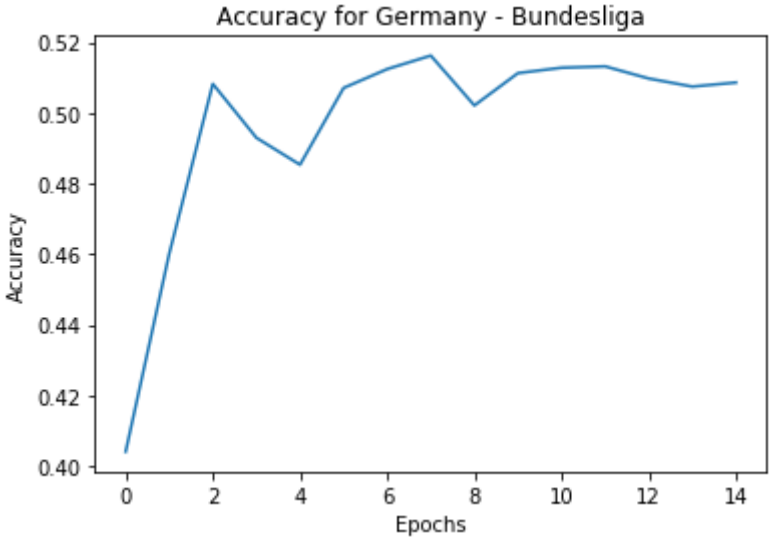

Figure X31: Germany Bundesliga train accuracy change with 128 batch size and 15 epochs

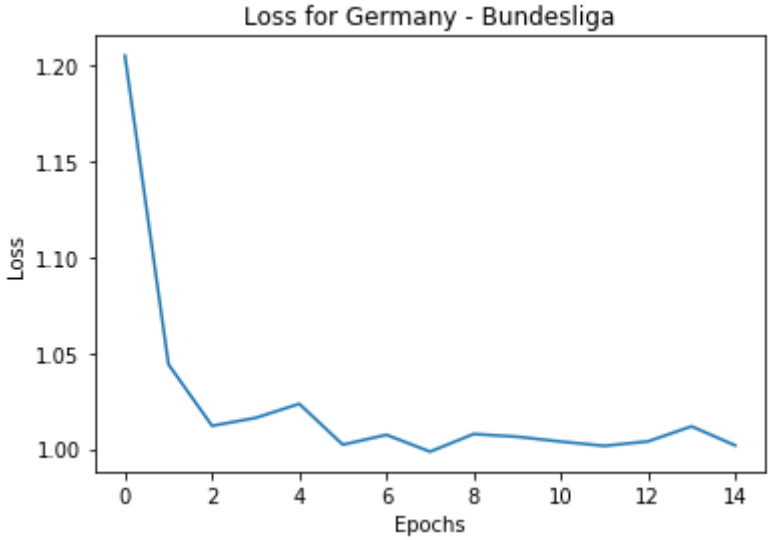

Figure X32: Germany Bundesliga train loss change with 128 batch size and 15 epochs

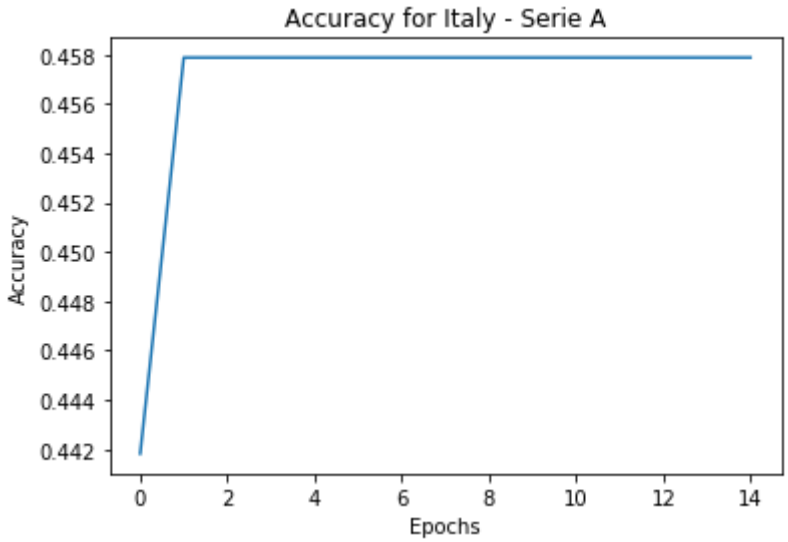

Figure X33: Italy Serie A train accuracy change with 128 batch size and 15 epochs

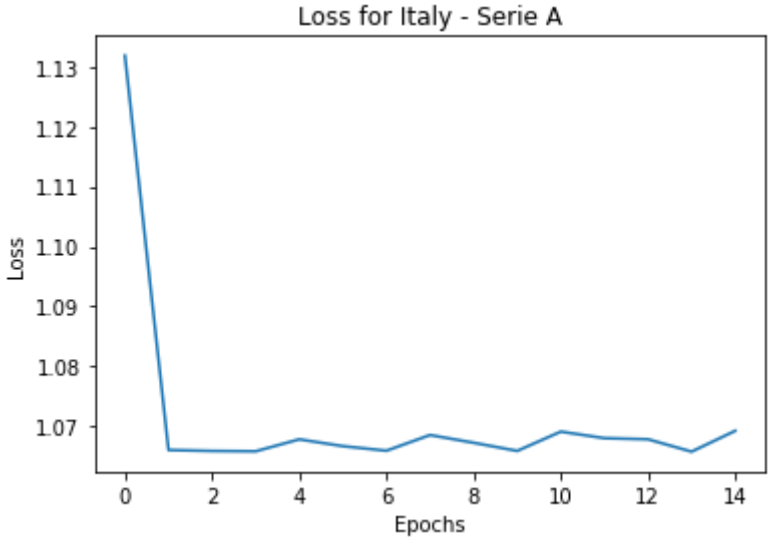

Figure X34: Italy Serie A train loss change with 128 batch size and 15 epochs

Appendix Y: Optimize Initialisation Mode for Neural Network

Keras provides multiple kernel initialisers that impact the initial values of the weights that the neural network's layers are initialised with.

The uniform initialiser uniformly initialises the weights, by default between -0.05 to 0.05. The lecun_uniform generates samples from a uniform distribution between positive and negative limits based on the units in the weight tensor. The normal initialiser simply generates weights using normal distribution. When it comes to zero initialiser, it simply sets all weights to 0 can lead to more stable results. However, it may also constantly get stuck in a local maxima. Next, the glorot_normal initialises the weights from a normal distribution, which is truncated based on the number of input and output units. Similar to glorot_normal is glorot_uniform, except the latter uses a uniform distribution. He_normal is an initialiser that generates the weights from a normal distribution, which is truncated based on the number of input units alone. Lastly, he_uniform is an initialiser that generates the weights from a normal distribution, which is truncated based on the number of input units alone.

Next, the results from creating a Neural Network using different initialisation modes are presented in Figures Y1 to Y4. If average accuracy for all competitions is used, the optimal initialisation mode is 'glorot_uniform' (which is confirmed by Figure Y1) with an accuracy of 52.28%. In terms of standard deviation, almost all kernel initialisation modes experienced similar levels of deviation except for the 'zero' initialiser, which has both the lowest standard deviation and the lowest accuracy. 'glorot_uniform' appears to have the highest variability amongst all initialisers, if only by 0.2%. 'glorot_normal' initialiser has an overall accuracy that is close to 'glorot_uniform' (less than 0.1% difference) and 0.2% lower standard deviation. It is difficult to conclude based on 10 k-fold cross-validation that 'glorot_uniform' is always better than 'glorot_normal' because the differences in performance are relatively low, but, given the computational constraints, it appears that this kernel initialisers is best on average.

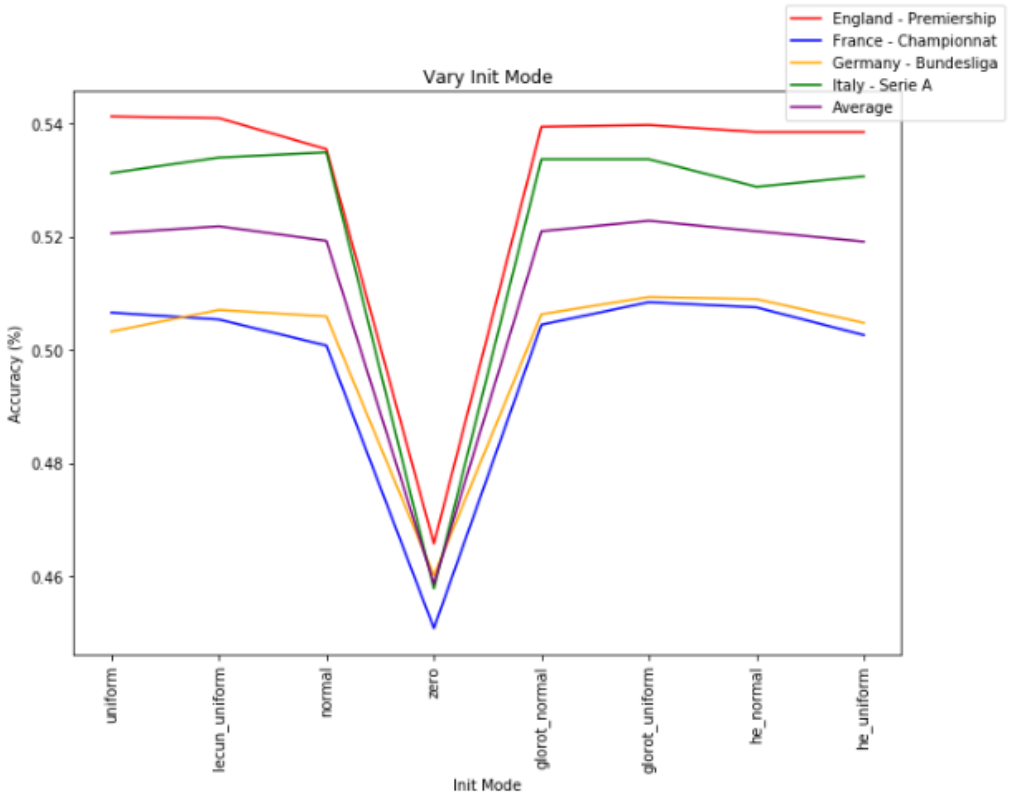

Figure Y1: Competition accuracy for train set (y) varying depending on kernel initialization mode (x)

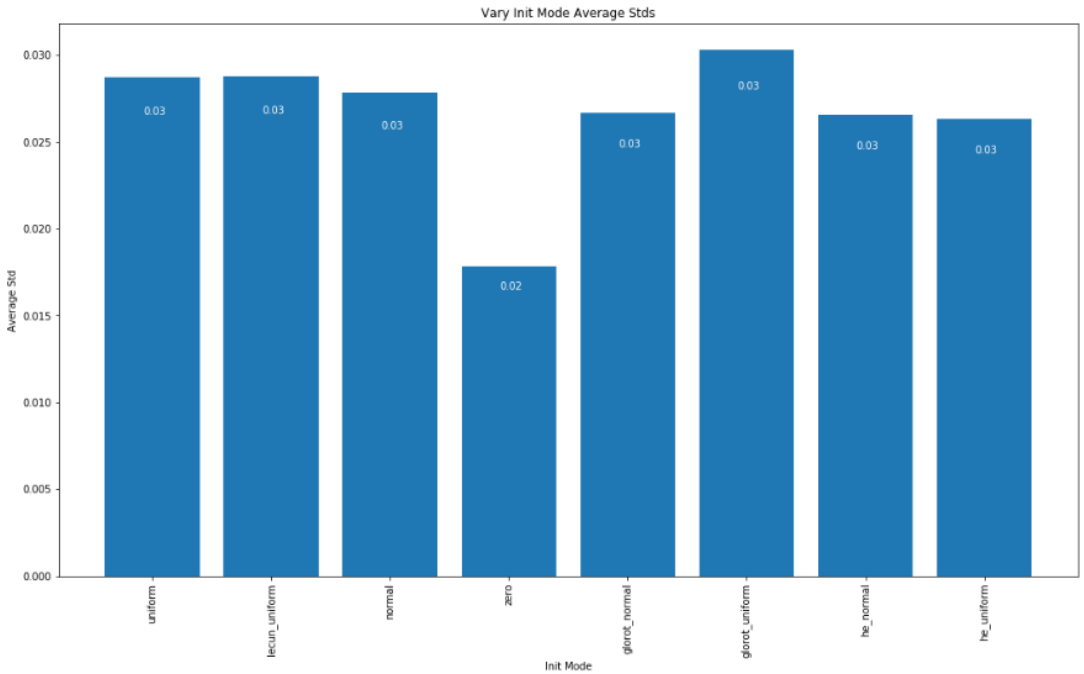

Figure Y2: Overall accuracy std for train set (y) varying depending on kernel initialisation mode (x)

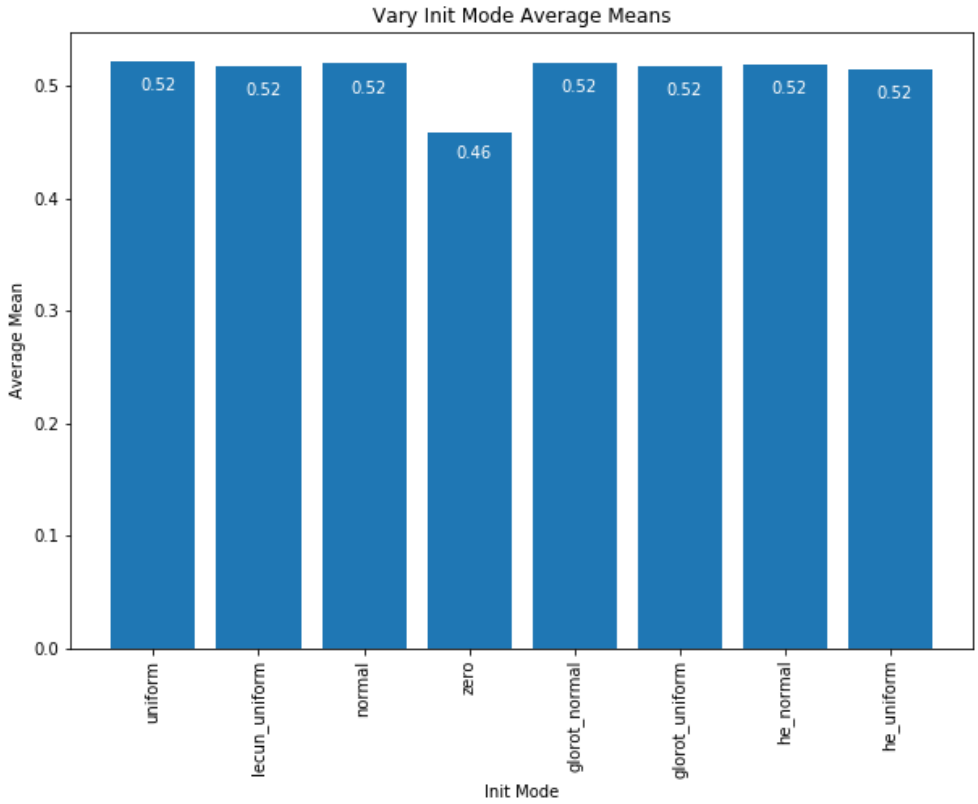

Figure Y3: Overall accuracy for train set (y) varying depending on kernel initialisation mode (x)

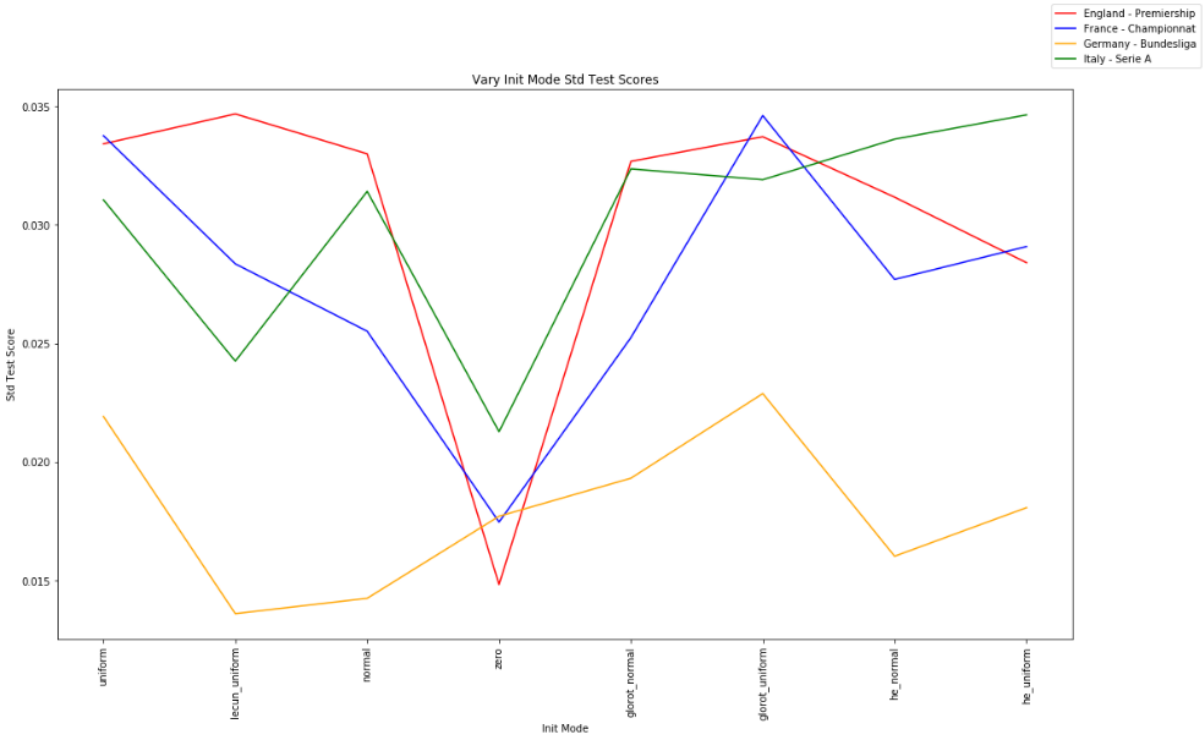

Figure Y4: Competition accuracy std for train set (y) varying depending on kernel initialisation mode (x)

Similarly to other hyperparameters, considering the best initialisation mode on a per-competition basis may result in a different set of parameters labelled as ‘optimal’. For the England Premiership, the initialisation mode with the highest accuracy is ‘normal’.

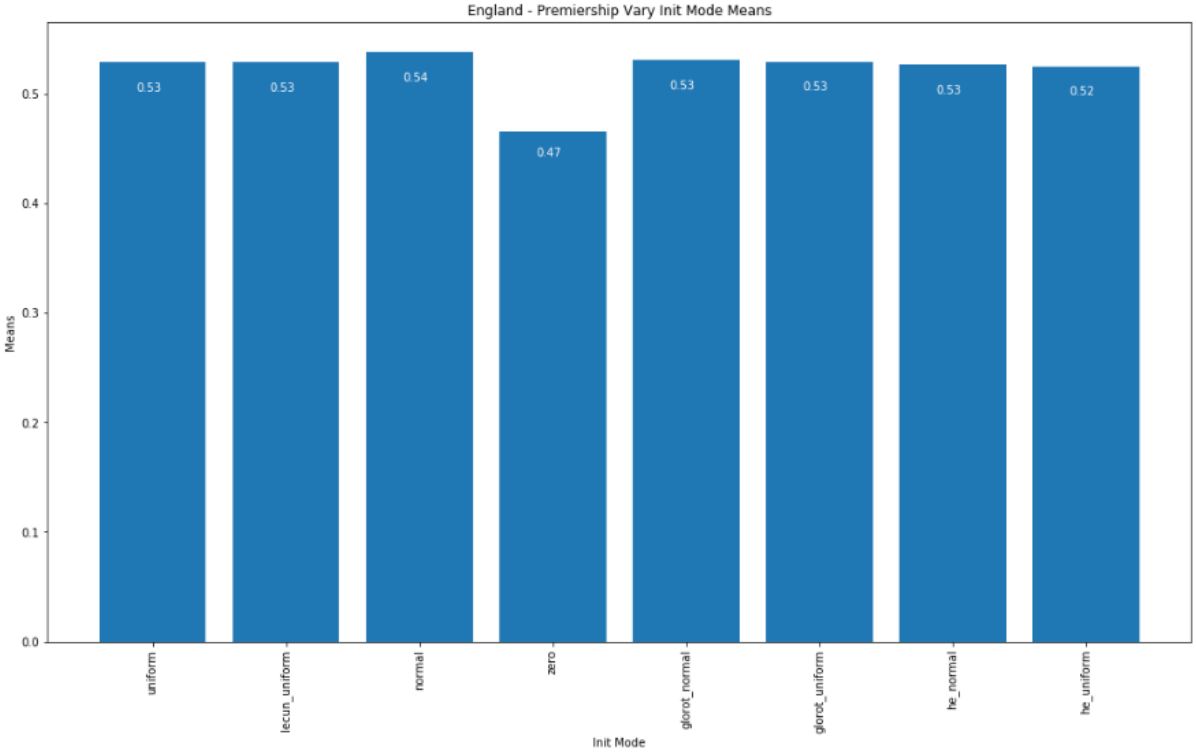

Figure Y5: England Premiership accuracy for train set (y) varying depending on kernel initialisation mode (x)

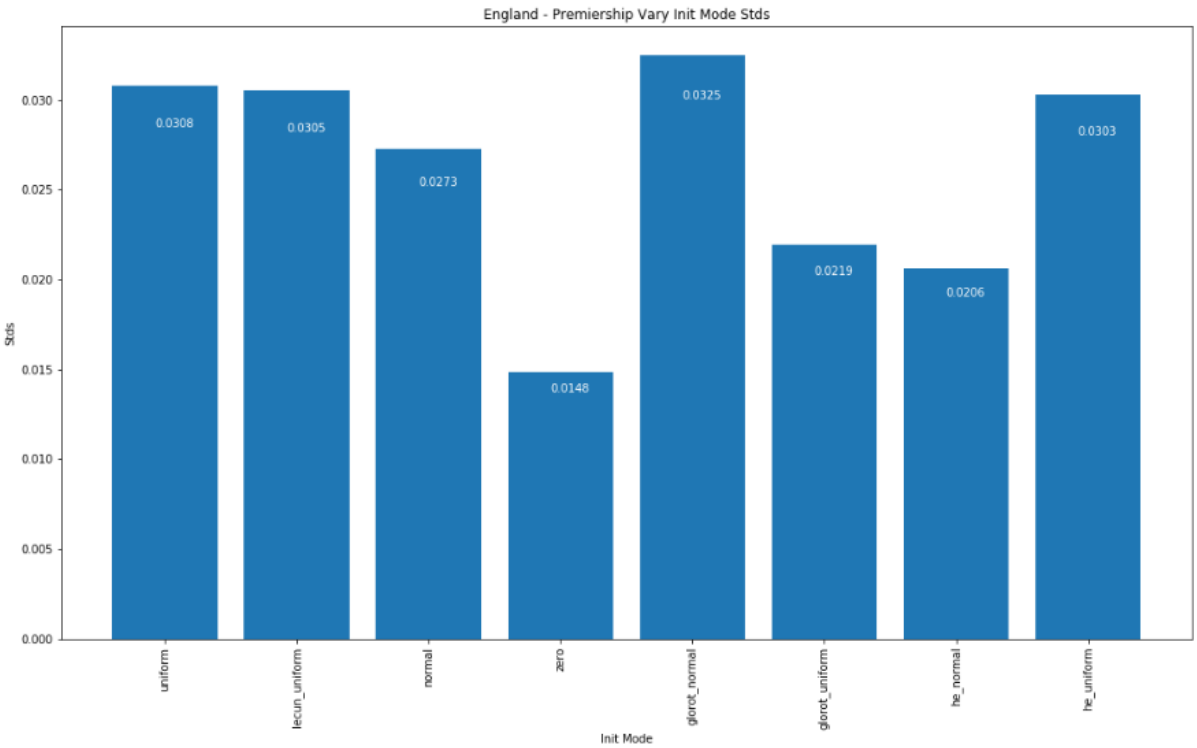

Figure Y6: England Premiership accuracy std for train set (y) varying depending on kernel initialisation mode (x)

As observed in Figure Y6, while ‘normal’ does have the highest accuracy, almost all other initialisers perform almost at the same level, except for the ‘zero’ initialiser. This is something that has been observed with the average accuracies as well. In regards to the standard deviation, as confirmed by Figure X, amongst the high-performing initialisers, ‘glorot_uniform’ and ‘he_normal’ have the lowest standard deviations. So it appears that the default initialiser actually performs well for the England Premiership.
Next, the effects of varying the kernel initialiser for France Championnat are examined.

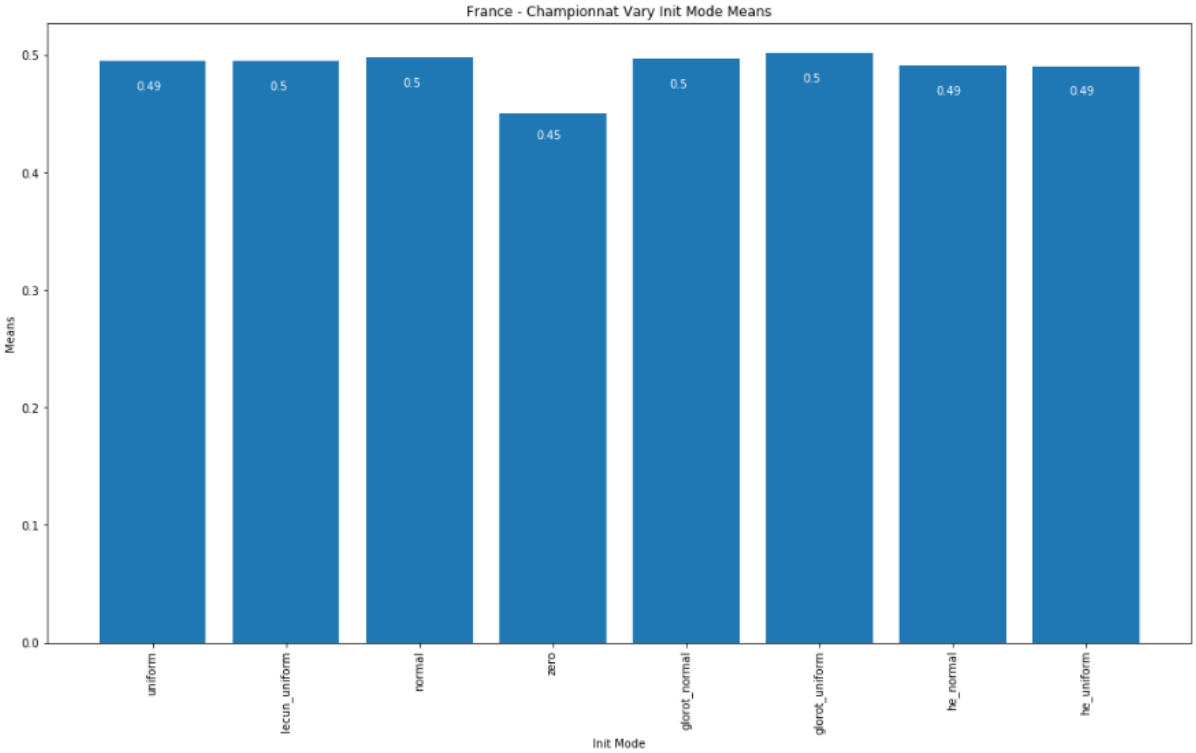

Figure Y7: France Championnat accuracy for train set (y) varying depending on kernel initialisation mode (x)

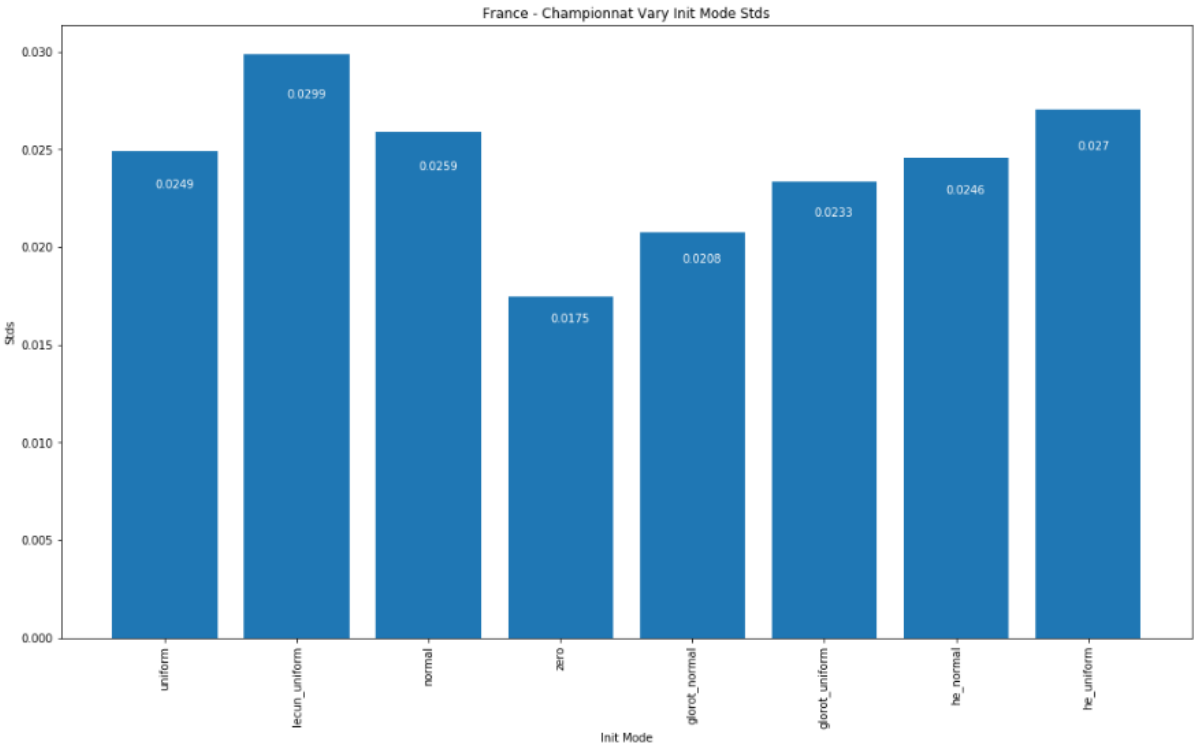

Figure Y8: France Championnat accuracy std for train set (y) varying depending on kernel initialisation mode (x)

For France Championnat, the initialiser with the highest accuracy is 'glorot_uniform', which can be seen in Figure Y7. As with England Premiership, 'zero' initialiser performed worse than other initialisers while all other have relatively similar performances. The consistency of the models with difference initialisers can be observed in Figure Y8. It is evident that 'glorot_uniform' has second lowest standard deviation amongst the high-performing initialisers, right after 'glorot_normal'. The difference in performance between these two is 0.5%, with 'glorot_uniform' having the higher accuracy. As for the standard deviation, 'glorot_normal' has 0.24% higher deviation.

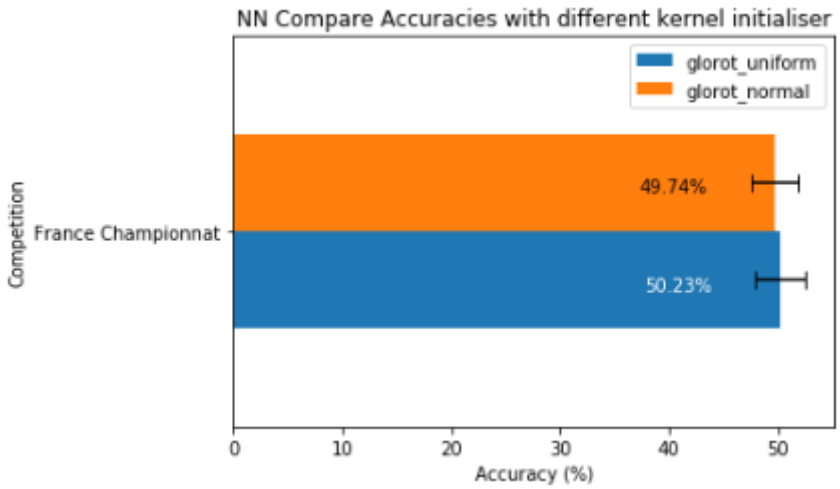

Figure Y9: France Championnat performance comparison for 'glorot_uniform' and 'glorot_normal'

As can be observed in FIgure Y9, the ‘glorot_uniform’ initialiser does indeed perform slightly better than ‘glorot_normal’, even given the difference in the standard deviation.

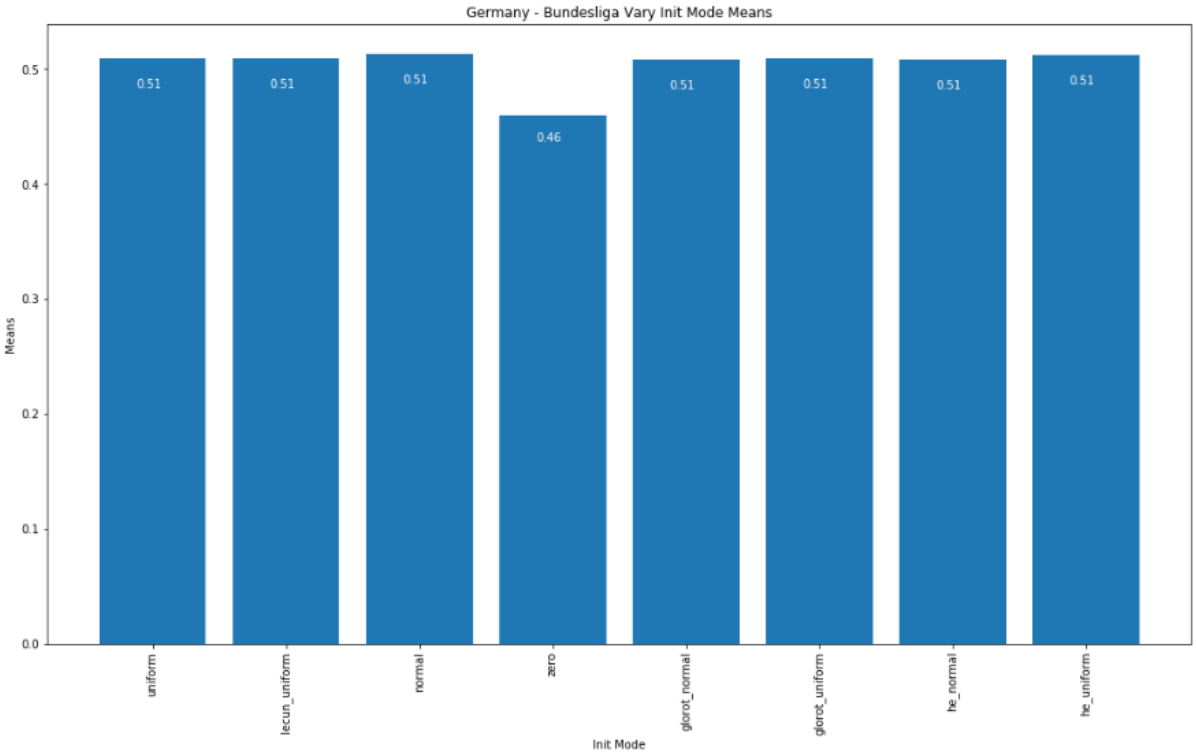

Figure Y10: Germany Bundesliga accuracy for train set (y) varying depending on kernel initialisation mode (x)

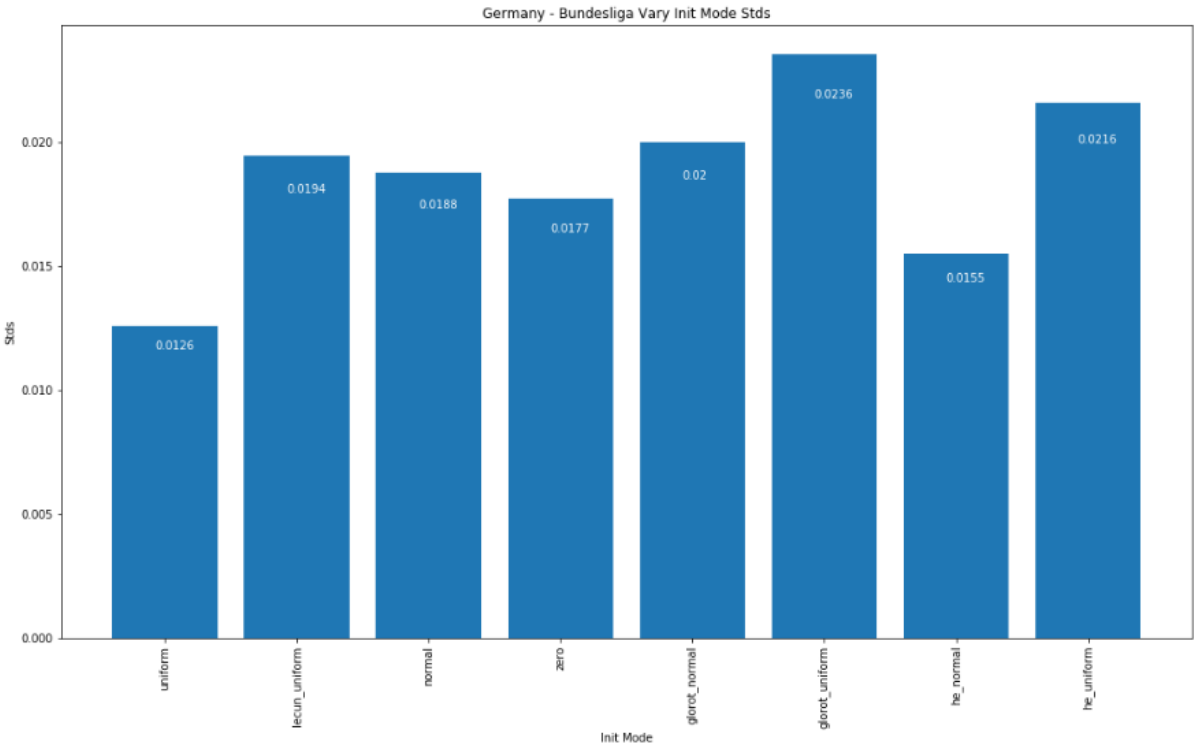

Figure Y11: Germany Bundesliga accuracy std for train set (y) varying depending on kernel initialisation mode (x)

The accuracies for Germany Bundesliga follow the same pattern that was observed with other competitions, with the ‘normal’ kernel initialiser having the highest accuracy. Interestingly, as can be seen in Figure Y12, ‘uniform’ initialiser has the lowest standard deviation with a considerable difference in respect to the other initialisers.

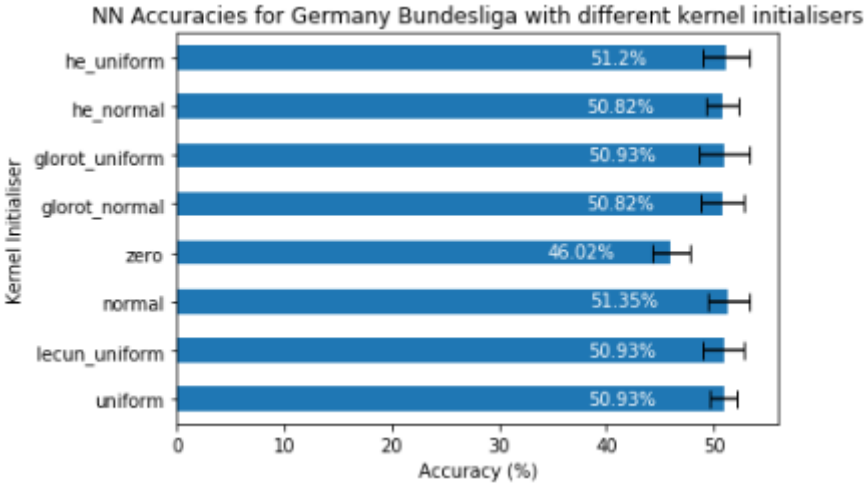

Figure Y12: Germany Bundesliga performance comparison for models with different kernel initialiser using averages of 10 models per kernel initialiser

As can be seen in Figure Y12, the low standard deviation of ‘uniform’ appears to make it the best for Germany Bundesliga. Lastly, the kernel initialiser is selected for Italy Serie A.

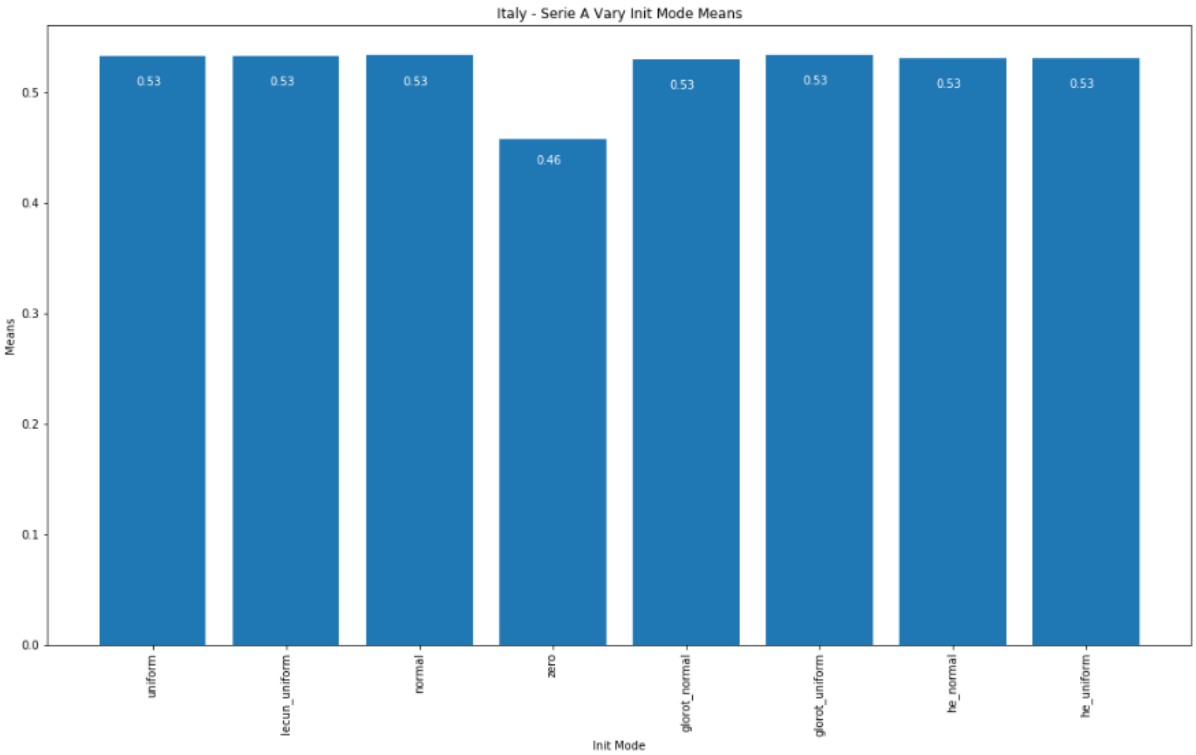

Figure Y13: Italy Serie A accuracy for train set (y) varying depending on kernel initialisation mode (x)

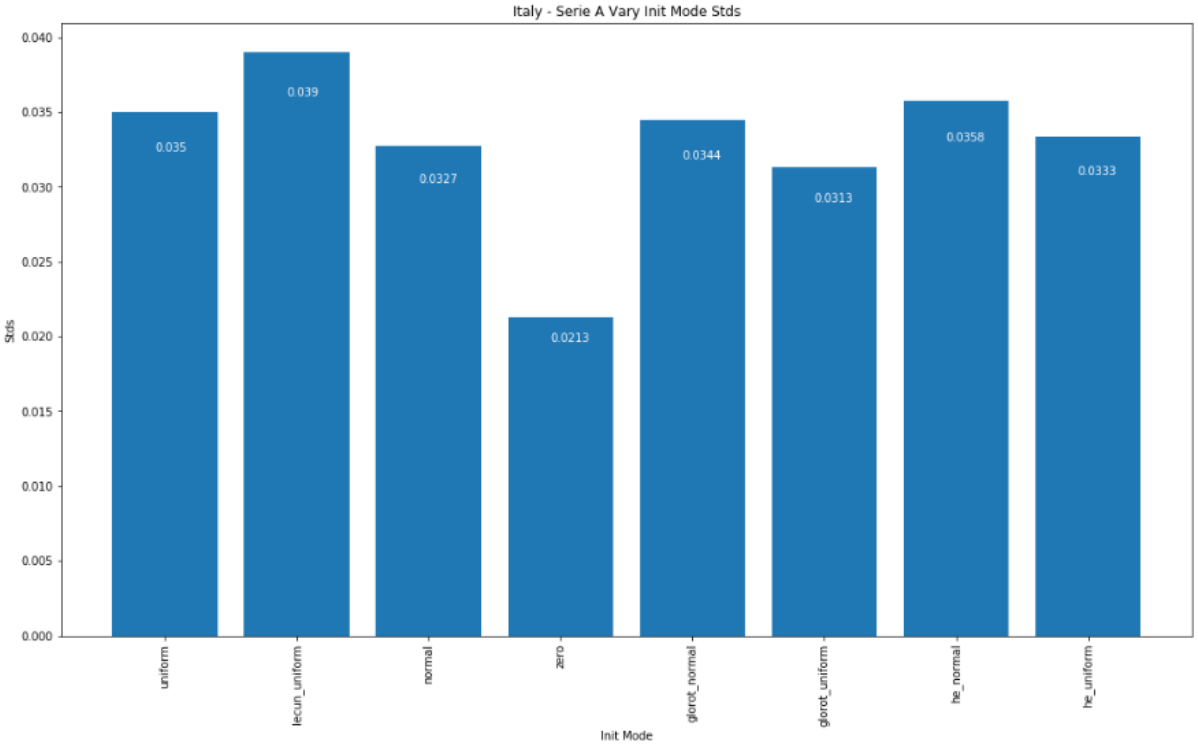

Figure Y14: Italy Serie A accuracy std for train set (y) varying depending on kernel initialisation mode (x)

As evident by Figures Y13 and Y14, the ‘glorot_uniform’ kernel initialiser has the highest accuracy and the lowest standard deviation amongst the high-performing initialisers, making it an obvious choice for Italy Serie A.

Appendix Z: Selecting Architecture for Neural Network

Accuracy and standard deviation for multiple architectures was calculated and compared against each other. Up to 3 hidden layers were used. The number of neurons for the first layer varied between 10, 20, 25, 30, 35, 40, 45, 50, 60 and 70. In regards to the second layer, the number of neurons were 0 (meaning the layer was not used), 5, 10, 15, 20, 25, 30, 35 and 40. Lastly, the third hidden layer was attempted with values of 0, 3, 5, 10, 15, 20 and 25. A total of 2,520 architectures with 630 for each competition was used, which made it impossible to produce a single graph to compare the accuracies and standard deviations. Instead, each architecture got assigned a score, which was calculated as follows:

Quality = accuracy - 1.5 * standard deviation ***(Q1)***

Architecture with two hidden layers, each of which have 10 neurons, resulted in the highest quality score overall. For the England Premiership, 20 neurons in the first hidden layer and 20 in the second maximised this value. Next, for France Championnat, a 3-layer architecture was used with 30, 5 and 5 neurons in the first, second and third hidden layers respectively. Germany Bundesliga had two hidden layers, similar to England Premiership, except both hidden layers had 25 neurons. Lastly, Italy Serie A contained three hidden layers with 20, 20 and 10 neurons. The updated accuracies and standard deviation can be observed in Figures Z1 and Z2. Interestingly, the standard deviation has increased significantly with the updated architectures for all competitions except England Premiership.

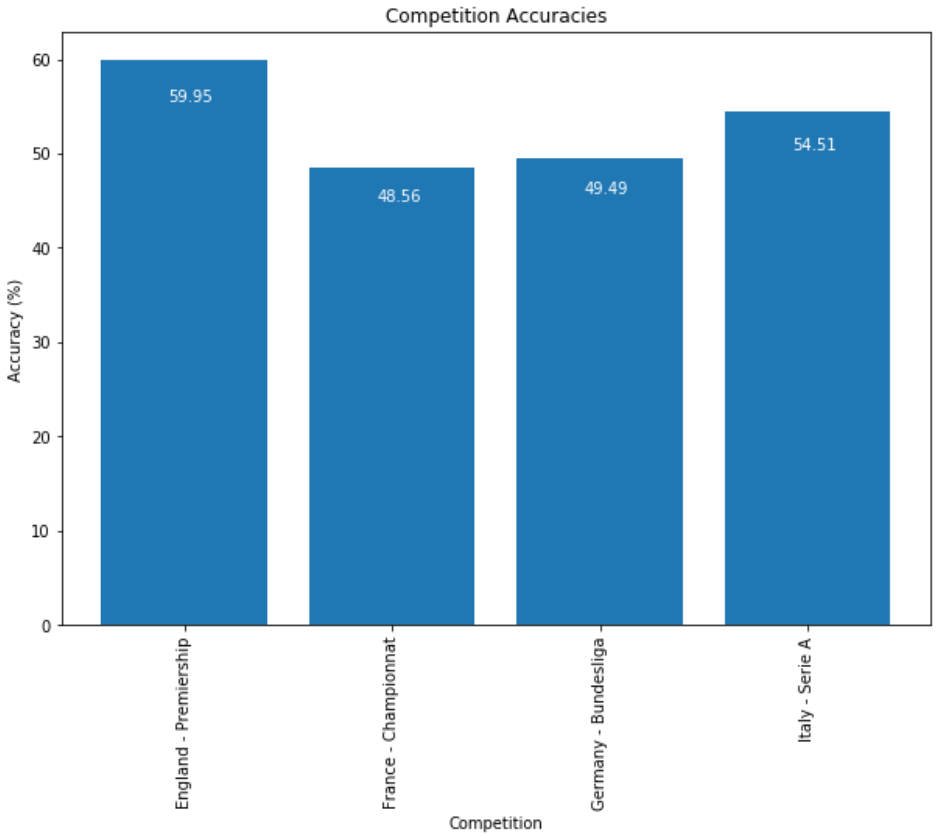

Figure Z1: Overall competition accuracy with per-competition architecture for tournaments 2018 - 2019

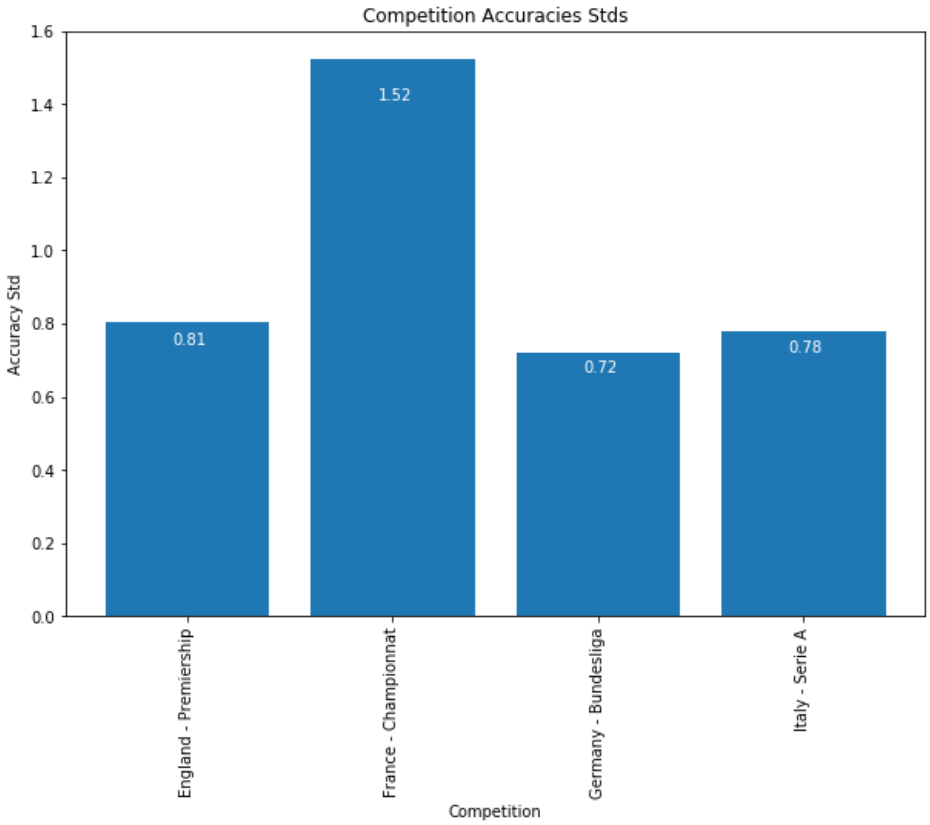

Figure Z2: Overall competition accuracy std with per-competition architecture for tournaments 2018 - 2019

To determine the cause behind the increase in standard deviation, the loss and accuracy graphs during model training are obtained, which can be seen in Figure X to Y.

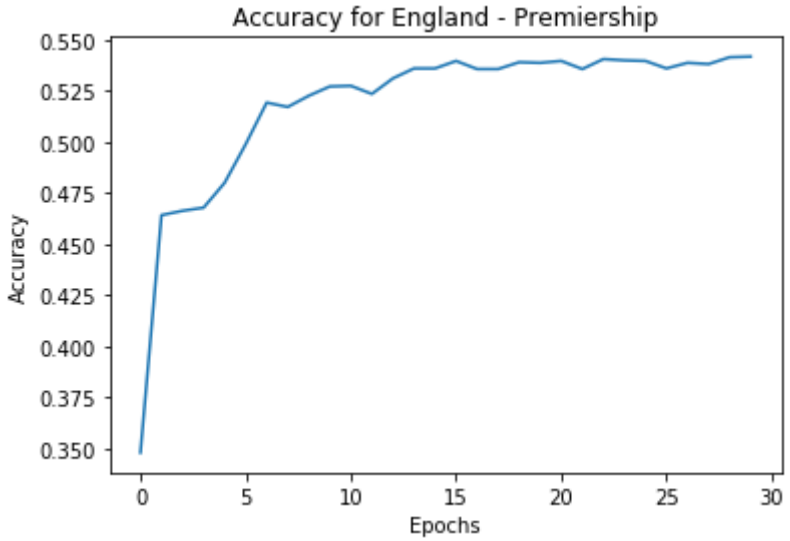

Figure Z3: England Premiership train loss change with updated architecture and 0.001 learning rate

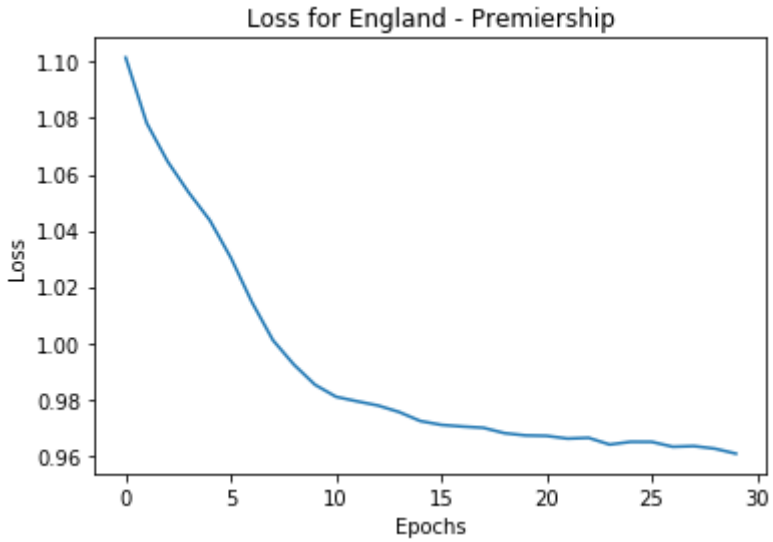

Figure Z4: England Premiership train loss change with updated architecture and 0.001 learning rate

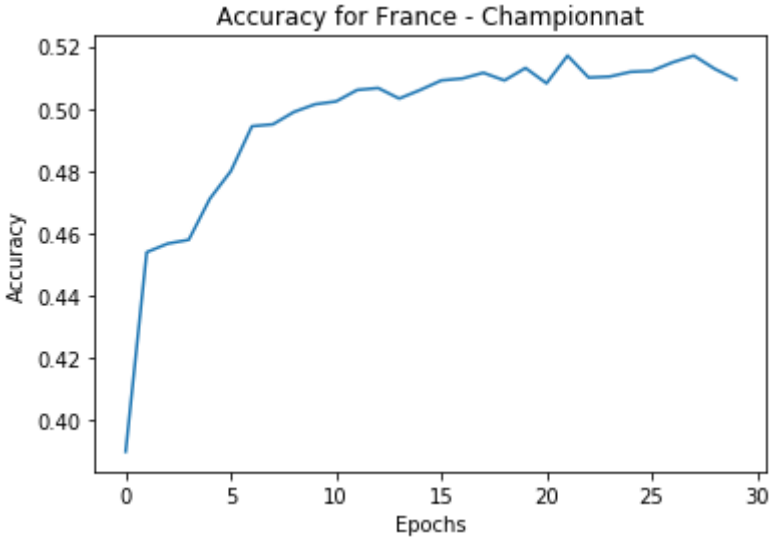

Figure Z5: France Championnat train loss change with updated architecture and 0.001 learning rate

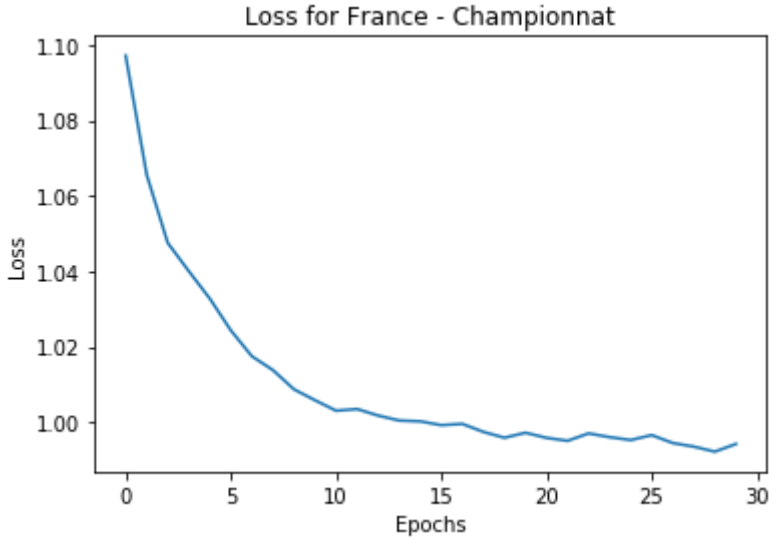

Figure Z6: France Championnat train loss change with updated architecture and 0.001 learning rate

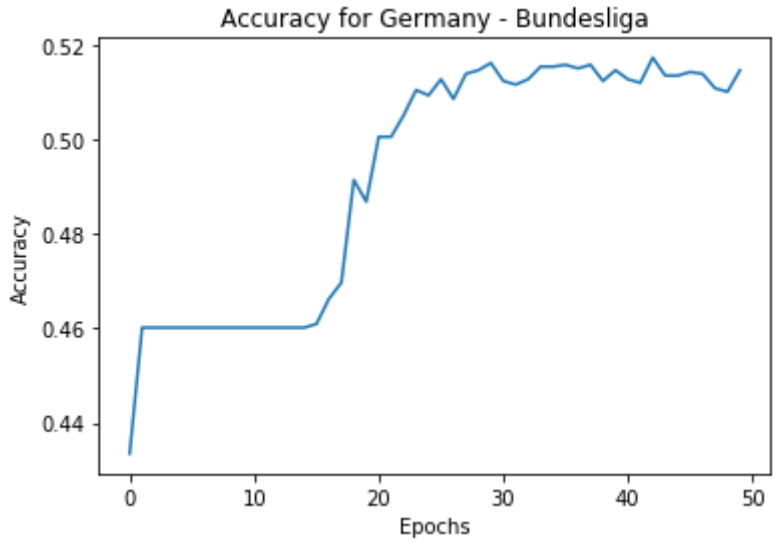

Figure Z7: Germany Bundesliga train loss change with updated architecture and 0.0005 learning rate

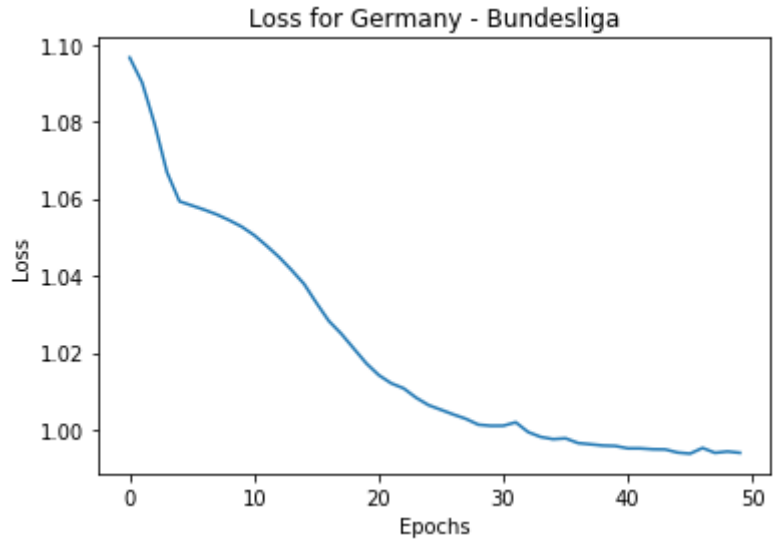

Figure Z8: Germany Bundesliga train loss change with updated architecture and 0.0005 learning rate

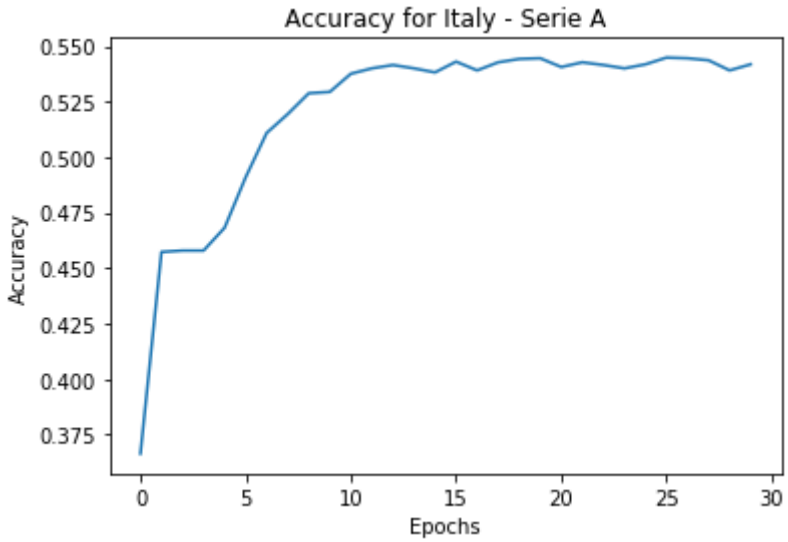

Figure Z9: Italy Serie A train loss change with updated architecture and 0.001 learning rate

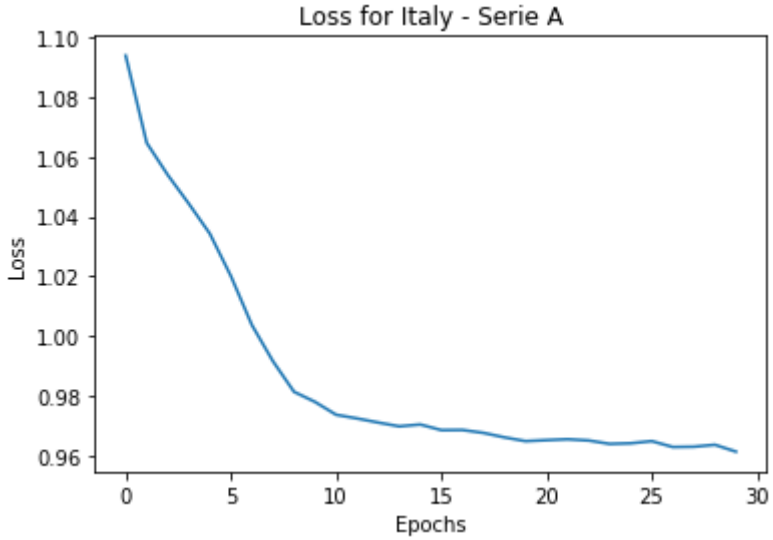

Figure Z10: Italy Serie A train loss change with updated architecture and 0.001 learning rate

To better determine whether the effects of overfitting on the Neural Network, a validation set of 20% is used. Graphs with both the train and validation sets performances can be seen in Figures Z11 and Z12.

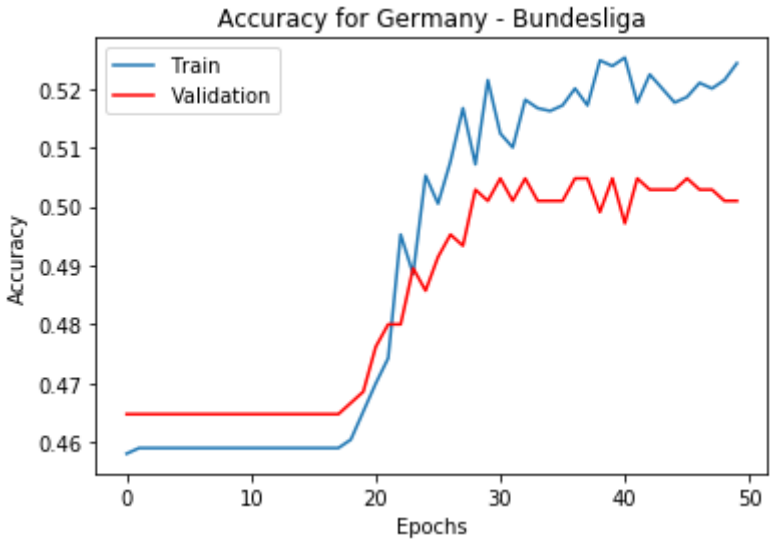

Figure Z11: Accuracy for training and validation sets during training for Germany Bundesliga with learning rate 0.0005 and 50 epochs

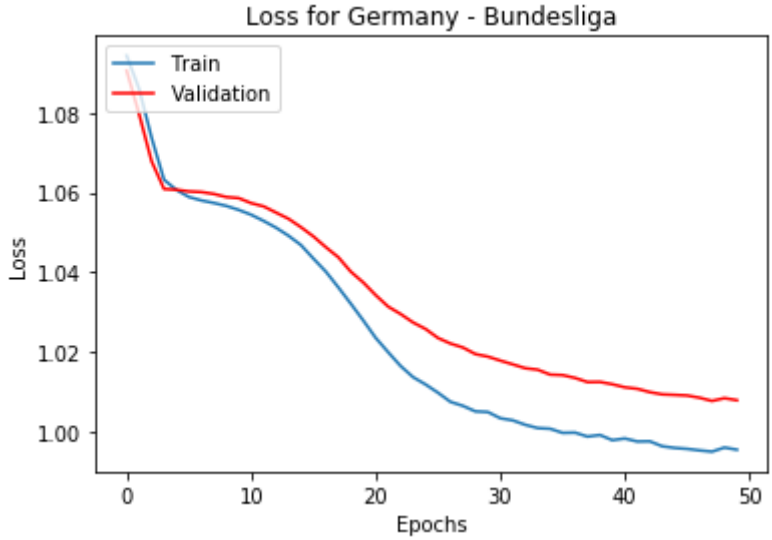

Figure Z12: Loss for training and validation sets during training for Germany Bundesliga with learning rate 0.0005 and 50 epochs

It appears that models that are using updated architectures have spikes in loss during the training. One assumption is that the new architectures are more complex (all have 2 or 3 hidden layers as opposed to just 1 that was used before), so an adjustment to the learning rate, batch size and number of epochs could improve the performance. For France Championnat, the learning rate is decreased to 0.00025 and the number of epochs is increased to 70.

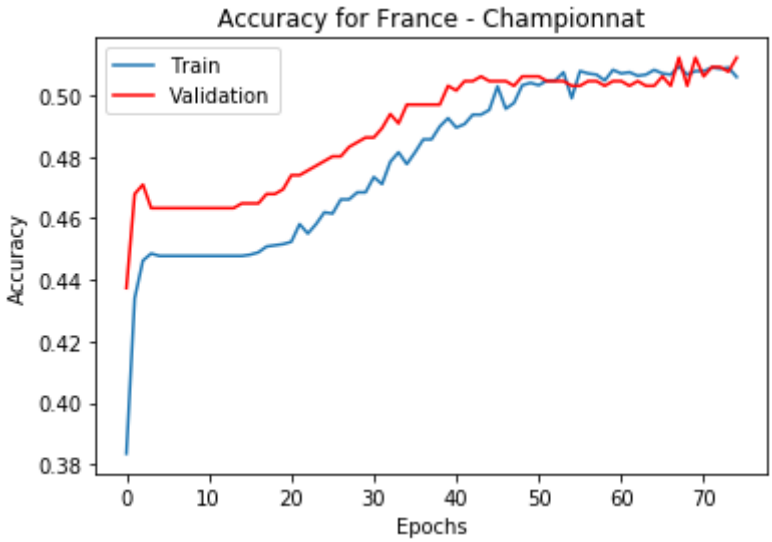

Figure Z13: Accuracy for training and validation sets during training for France Championnat with learning rate 0.00025 and 70 epochs

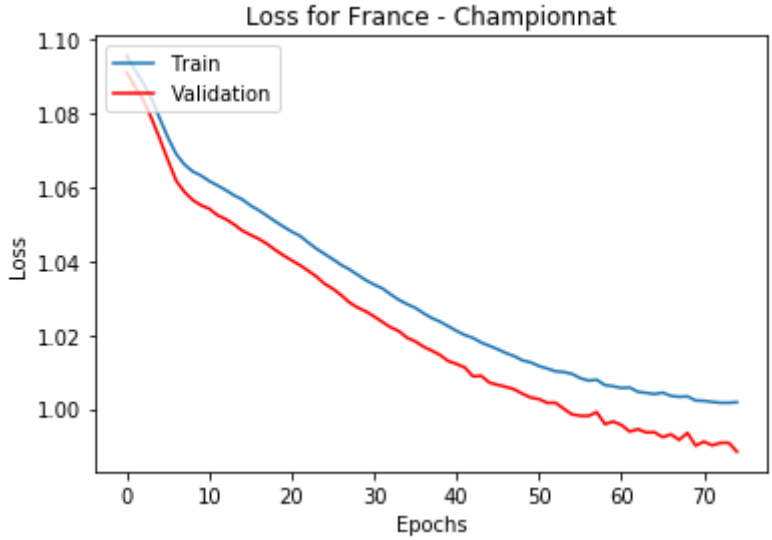

Figure Z14: Loss for training and validation sets during training for France Championnat with learning rate 0.00025 and 70 epochs

For Germany Bundesliga, the current set of parameters clearly overfits the model. So, for Germany Bundesliga, the number of epochs is reduced down to 10 to avoid overfitting the model.
Lastly, for Italy Serie A, the learning rate is decreased from 0.001 to 0.0005 and the number of epochs is increased to 40.

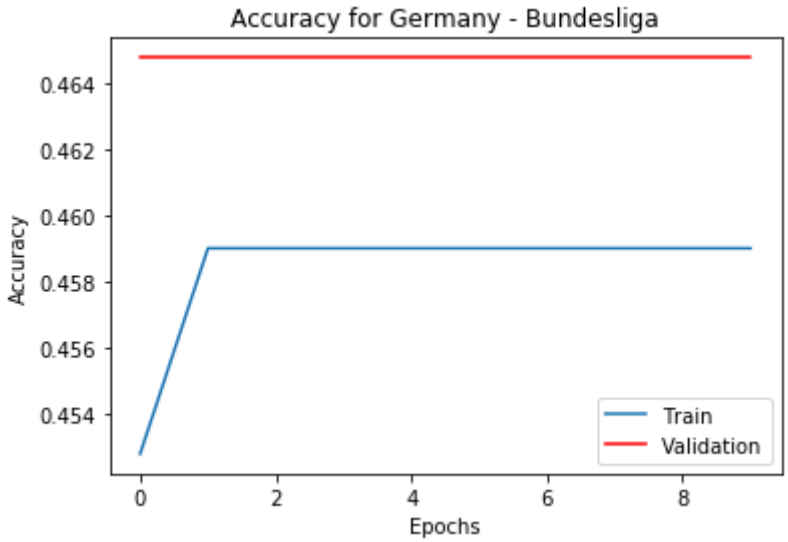

Figure Z15: Accuracy for training and validation sets during training for Germany Bundesliga with learning rate 0.0005 and 10 epochs

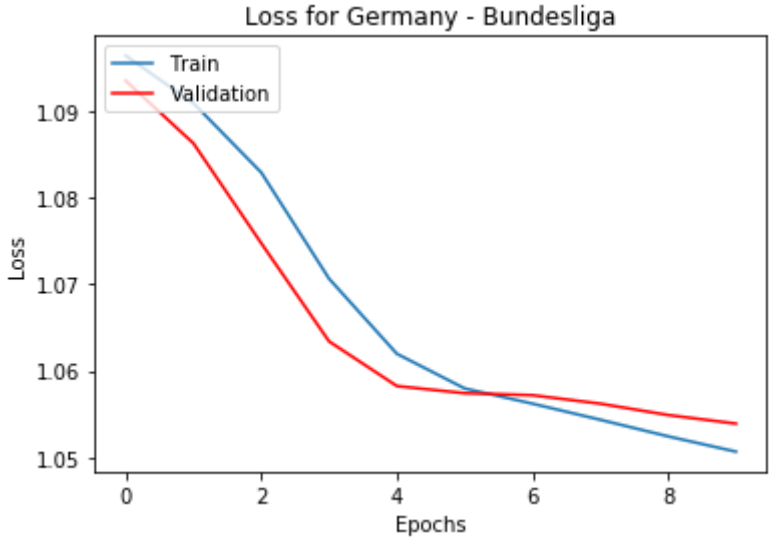

Figure Z16: Loss for training and validation sets during training for Germany Bundesliga with learning rate 0.0005 and 10 epochs

Appendix AA: Choosing Activation Function for Neural Network

Using the overall accuracy (Figure AA1) as a measure of activation functions' performance, softplus, softsign, relu and tanh all have similar accuracies with softsign peaking at 52.29%. Amongst these functions, softsign has the lowest standard deviation, closely followed by tanh.

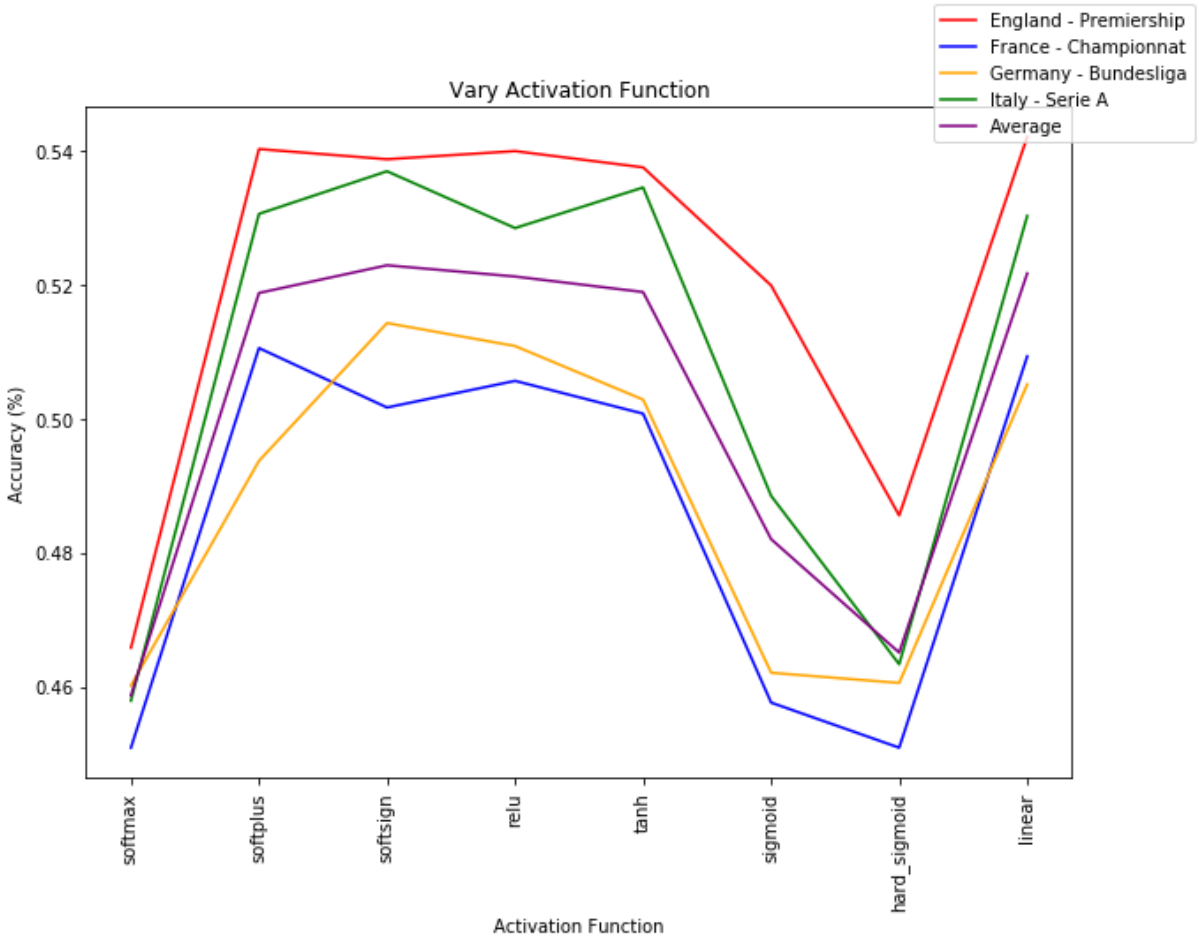

Figure AA1: Competition accuracy for train set (y) varying depending on activation function (x)

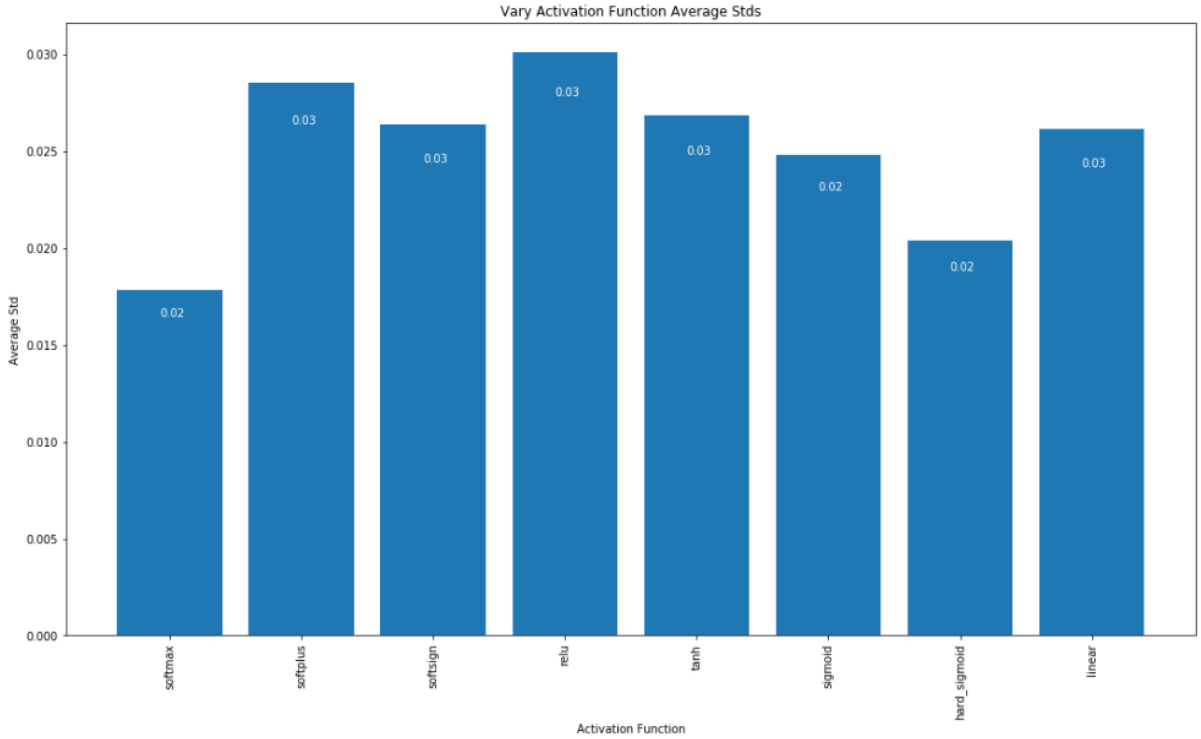

Figure AA2: Overall accuracy std for train set (y) varying depending on activation function (x)

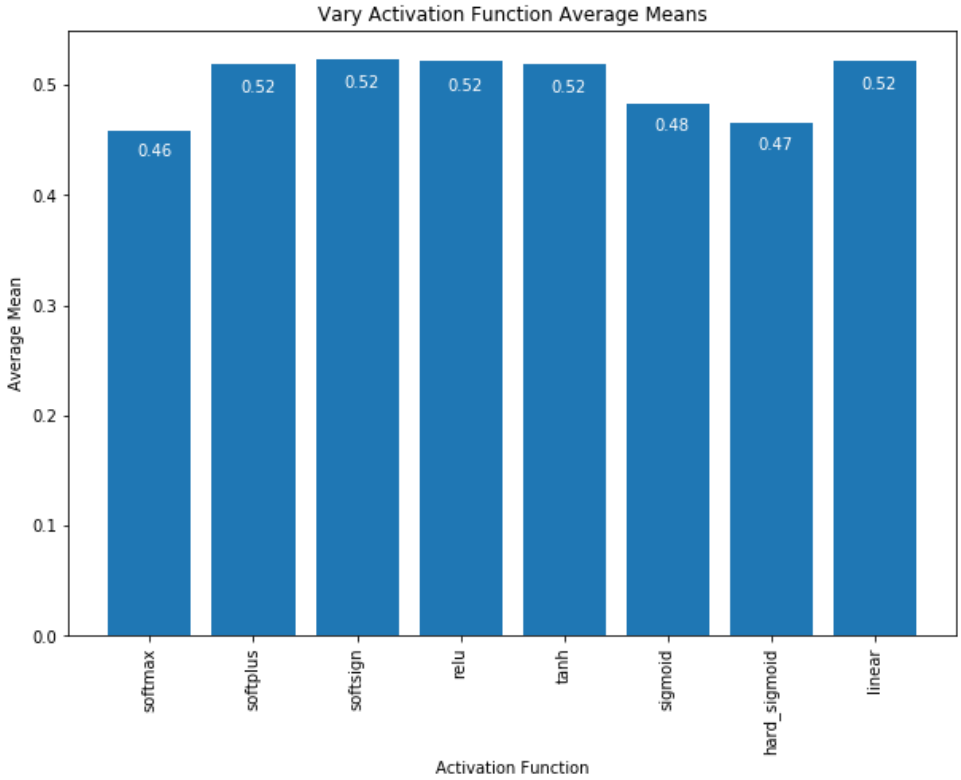

Figure AA3: Overall accuracy for train set (y) varying depending on activation function (x)

Considering the activation function performance on a per-competition basis, the England Premiership has the highest accuracy with the tanh, closely followed by softsign, tanh and, surprisingly, linear, which can be seen in Figure X.

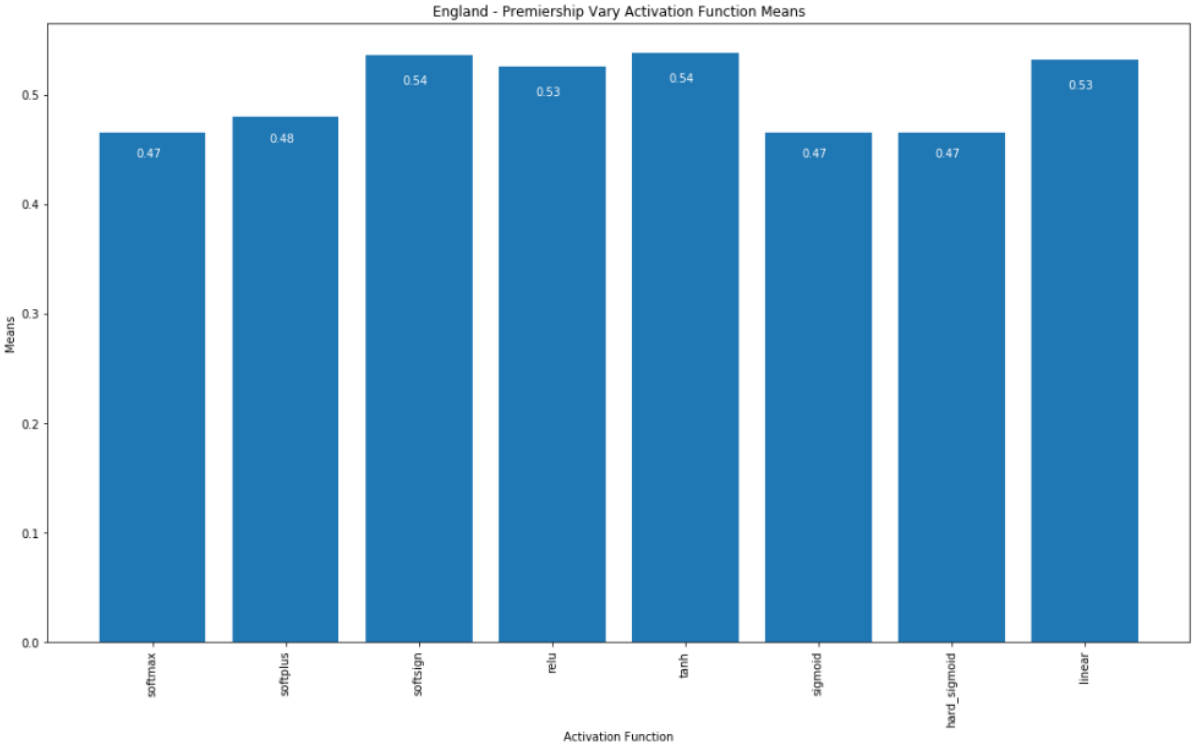

Figure AA4: England Premiership accuracy for train set (y) varying depending on activation function (x)

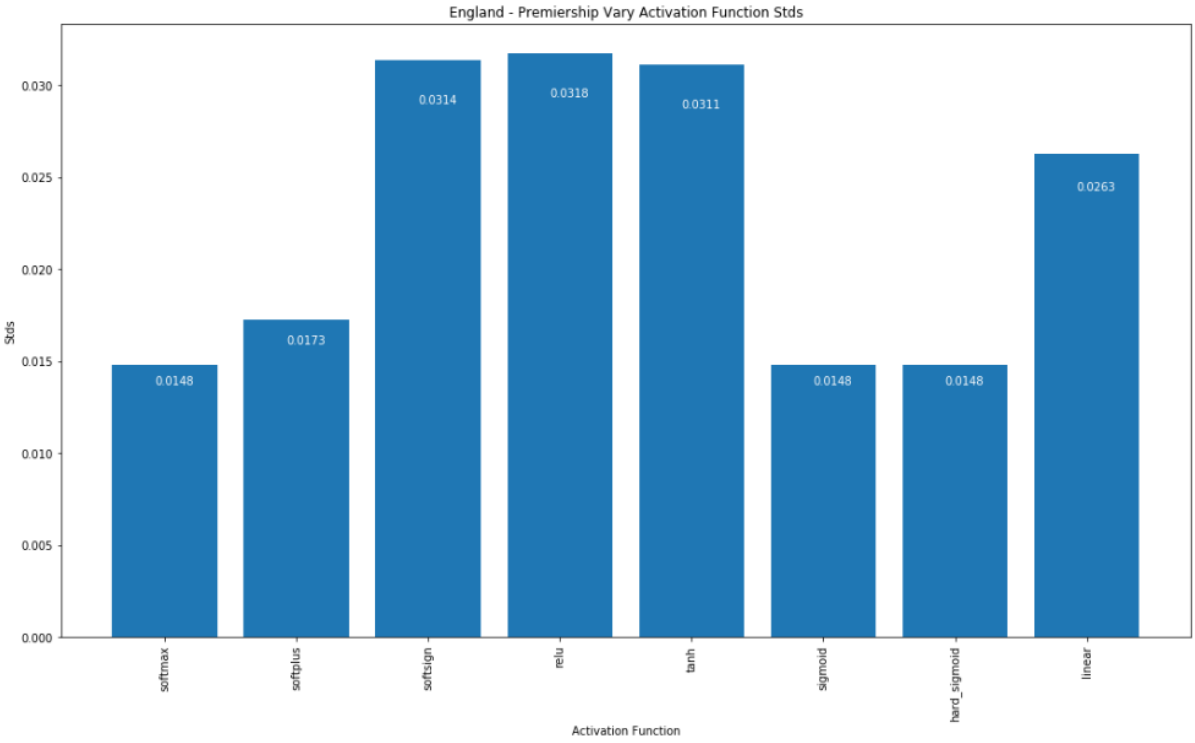

Figure AA5: England Premiership accuracy std for train set (y) varying depending on activation function (x)

Considering the standard deviation along with the accuracy of the activation functions linear appears to be the best choice for England Premiership with tanh being close second. It is worth observing how these functions perform with 2018 - 2019 tournaments to confirm whether Figures AA4 and AA5 are accurate. Using the linear activation function, the

accuracy and standard deviation is 60.06% and 0.66% respectively. Switching to tanh, these metrics are 60.08% and 0.76%.

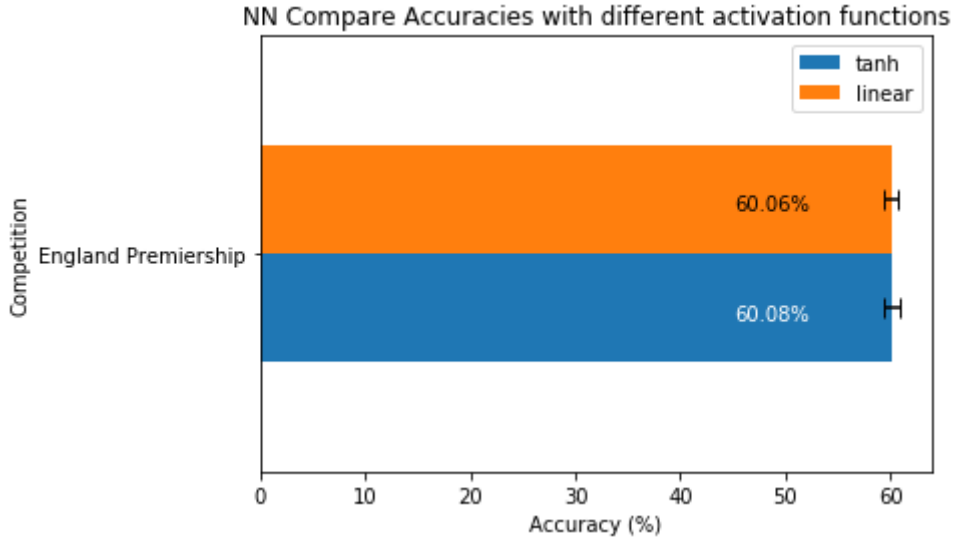

Figure AA6: England Premiership accuracy and standard deviation comparison for tanh and linear activation functions

As can be seen in Figure AA6, both functions have similar performance for the England Premiership. Tanh is selected as the best option as it is a more common choice amongst the activation functions.

A similar trend with accuracies as with England Premiership can be observed in France Championnat, which can be seen in Figure AA7. In Figure AA8, the standard deviation of the high-performing functions makes RelU appear like the best choice for activation function. The difference in accuracy between RelU and linear function is 2%, while the change in standard deviation is 0.15%. Both functions are selected for further examination using the 2018 - 2019 tournaments.

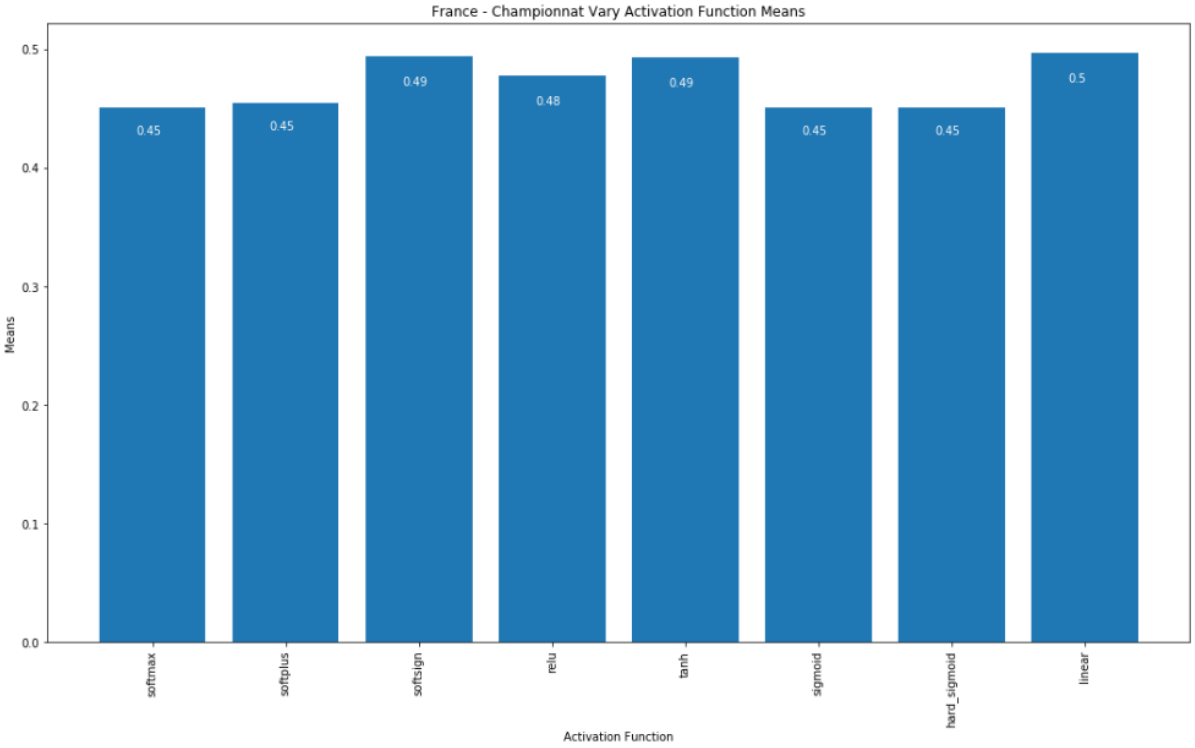

Figure AA7: France Championnat accuracy for train set (y) varying depending on activation function (x)

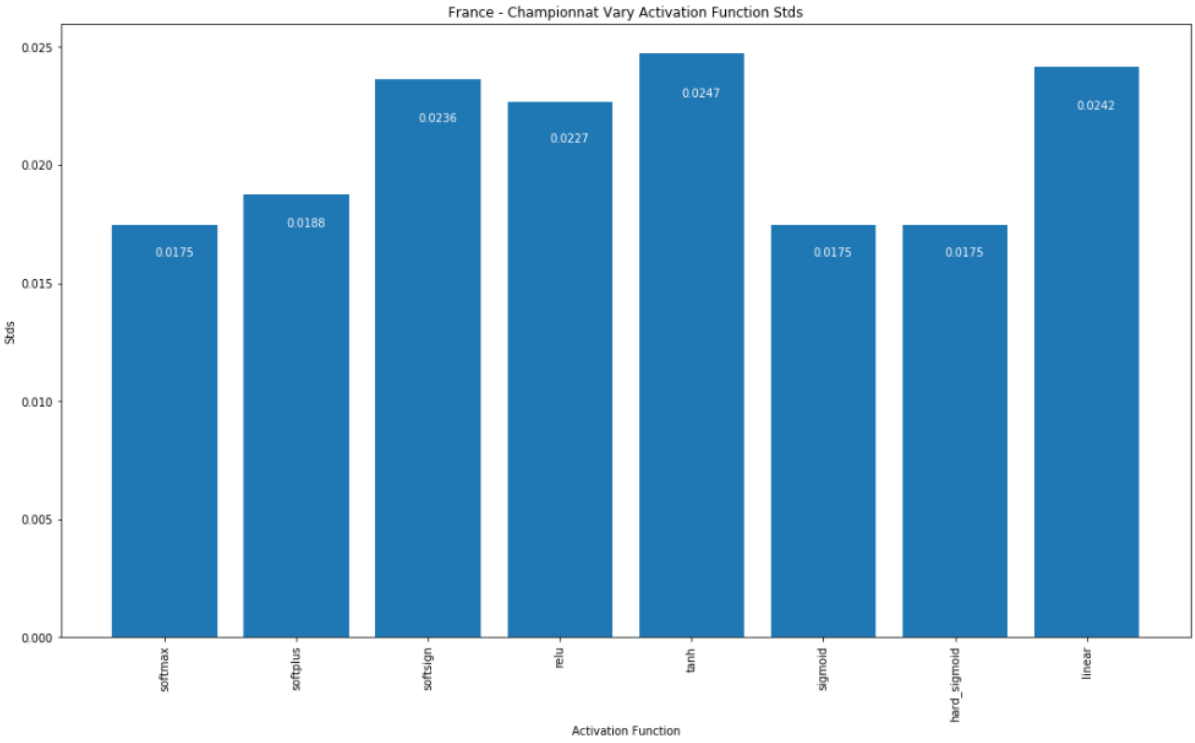

Figure AA8: France Championnat accuracy std for train set (y) varying depending on activation function (x)

For the linear function, the accuracy is 48.13% while the standard deviation is 0.77%. Using relu, the prediction accuracy is 48.76% and the deviation of 1.04%.

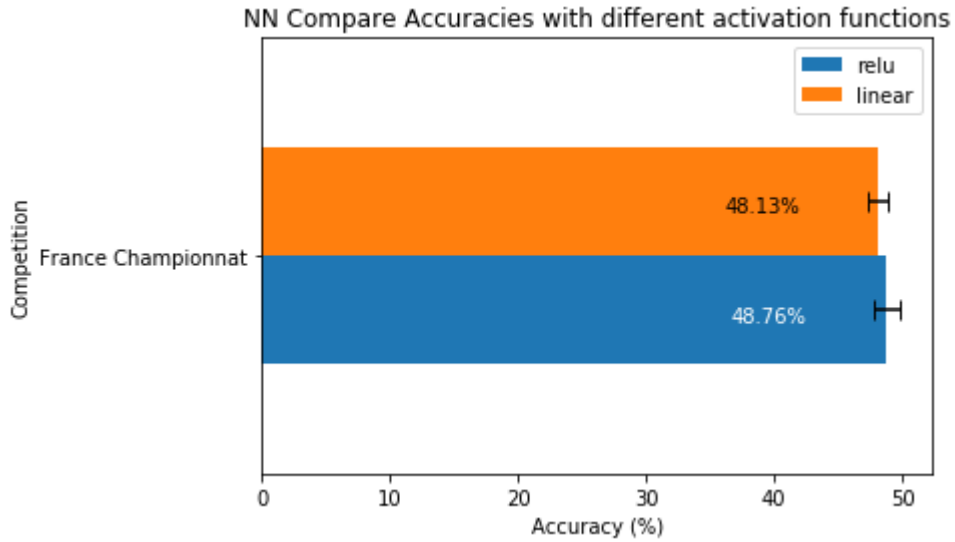

Figure AA9: France Championnat accuracy and standard deviation comparison for relu and linear activation functions

A close comparison of both activation functions is presented in Figure X. It is evident that relu appears to be a better option for France Championnat and is thus selected as optimal.

Next, different activation functions did not appear to have significant effects on Germany Bundesliga. Thus, a default RelU function is selected.
Lastly, for Italy Serie A, in terms of accuracy, a similar trend to England Premiership and France Championnat can be observed. All functions performed similarly in terms of standard deviation, with linear having the lowest value of 1.86% and softsign, on the other end, 2.17%. Based on these figures, tanh seems like the best choice for Italy Serie A.

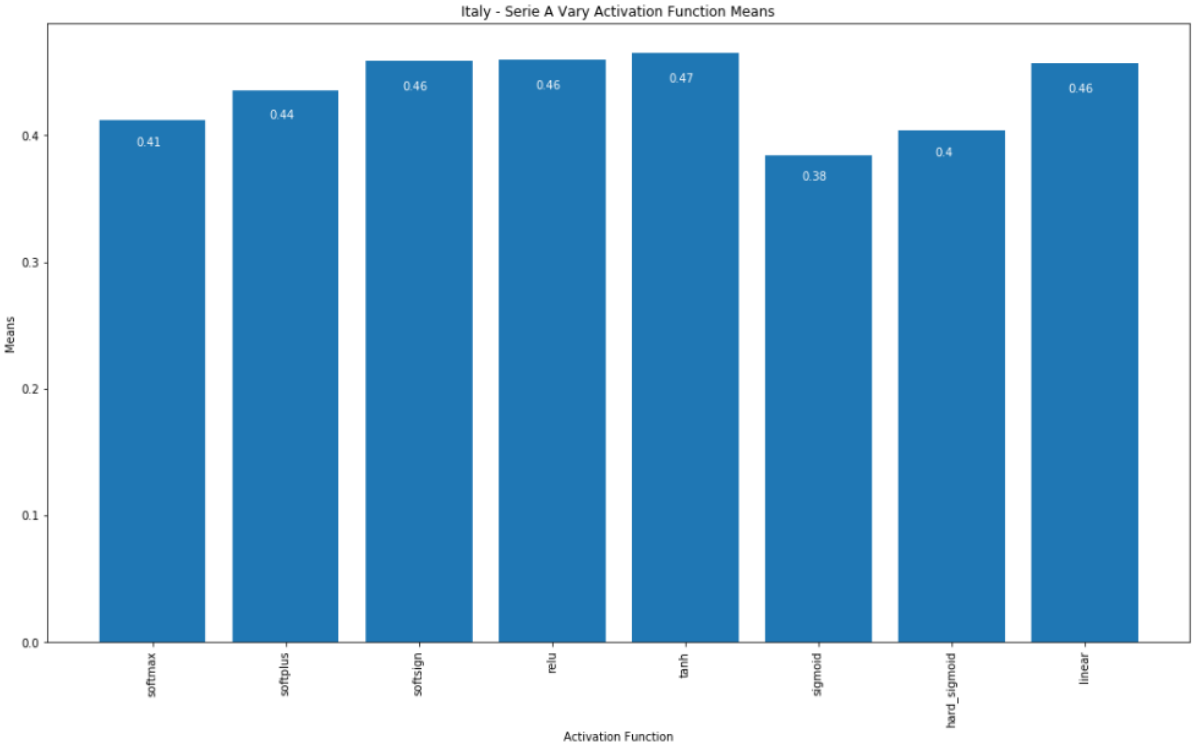

Figure AA10: Italy Serie A accuracy for train set (y) varying depending on activation function (x)

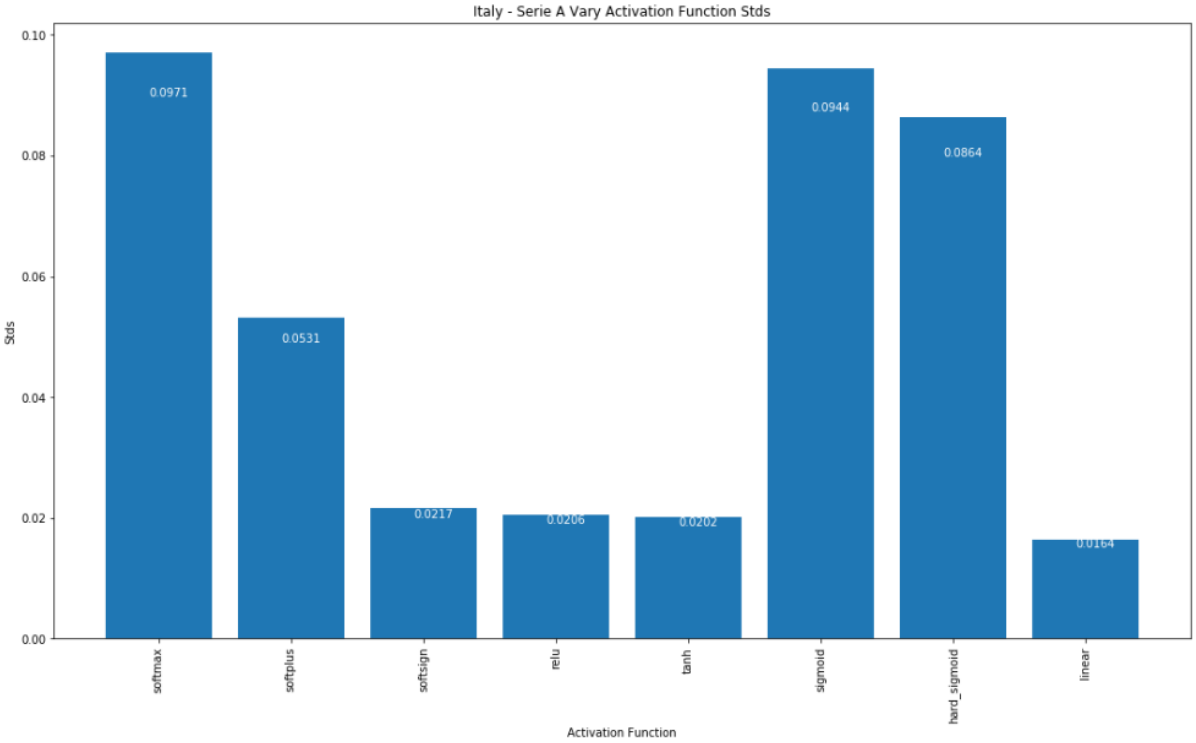

Figure AA11: Italy Serie A accuracy std for train set (y) varying depending on activation function (x)

Appendix AB: Ensemble Neural Network Approach

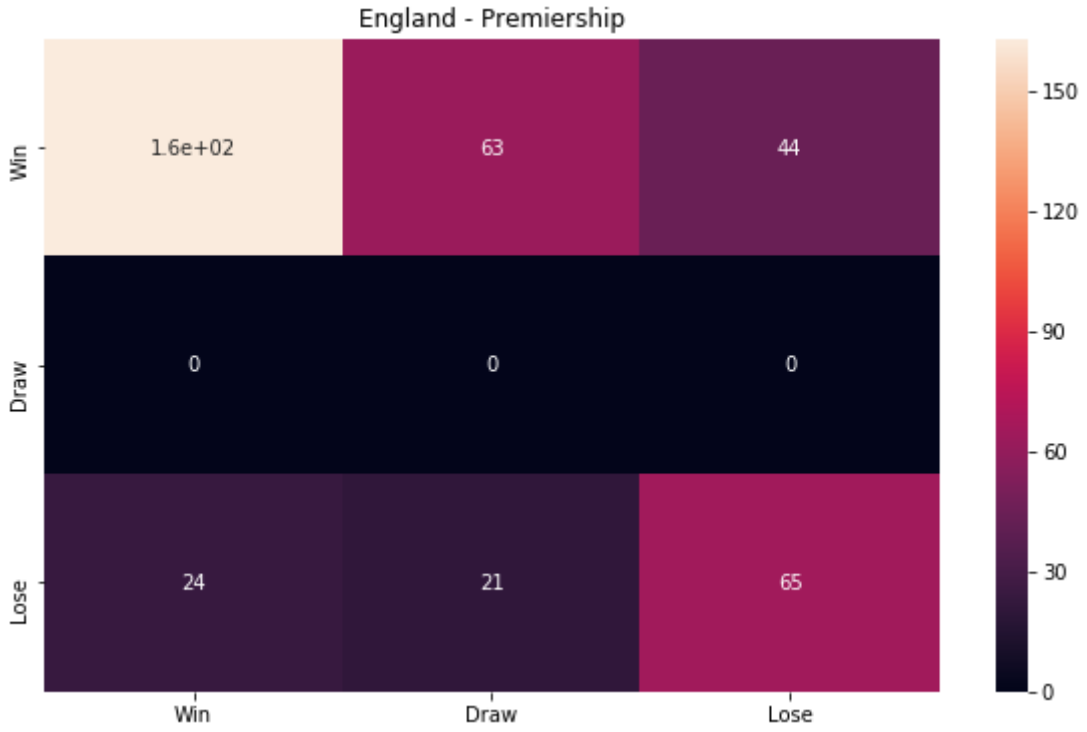

Figure AB1: England Premiership confusion matrix for 2018 - 2019 tournament using a Neural Network model

```
Precision:      Win 0.87        Draw 0.00       Lose 0.60
Recall:         Win 0.60        Draw 0.00       Lose 0.59
F1:     Win 0.71        Draw 0.00       Lose 0.59
```

Figure AB2: England Premiership metrics for 2018 - 2019 tournament using a Neural Network model

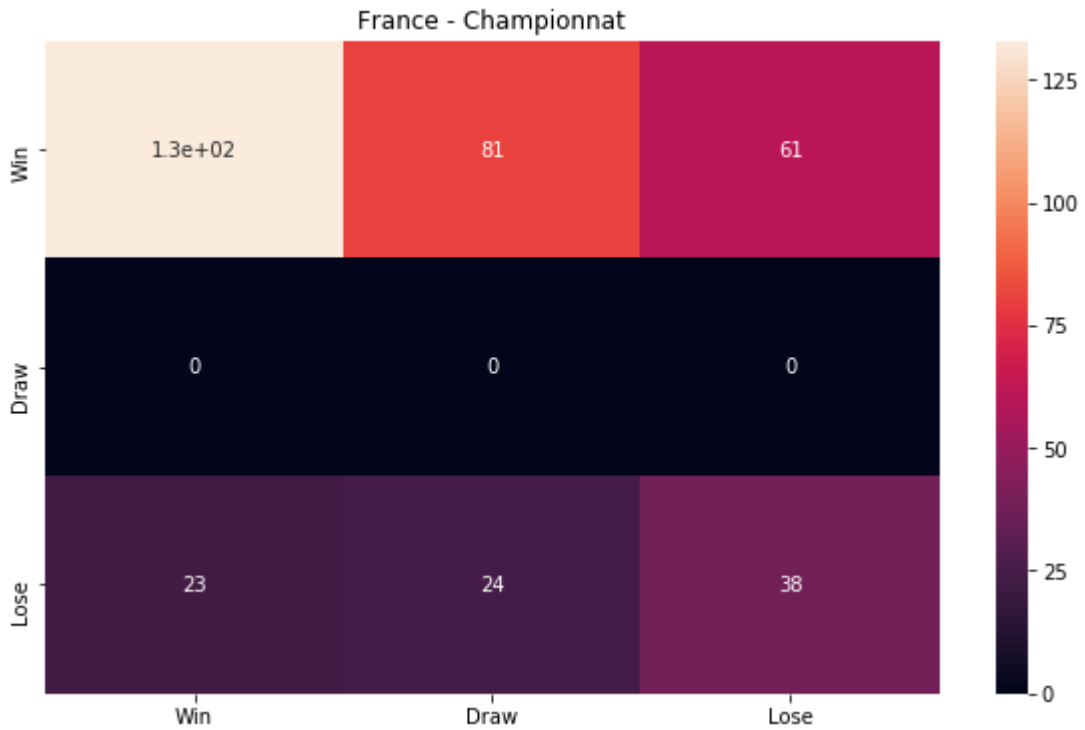

Figure AB3: France Championnat confusion matrix for 2018 - 2019 tournament using a Neural Network model

```
Precision:      Win 0.85        Draw 0.00       Lose 0.38
Recall:         Win 0.48        Draw 0.00       Lose 0.45
F1:     Win 0.62        Draw 0.00       Lose 0.41
```

Figure AB4: France Championnat metrics for 2018 - 2019 tournament using a Neural Network model

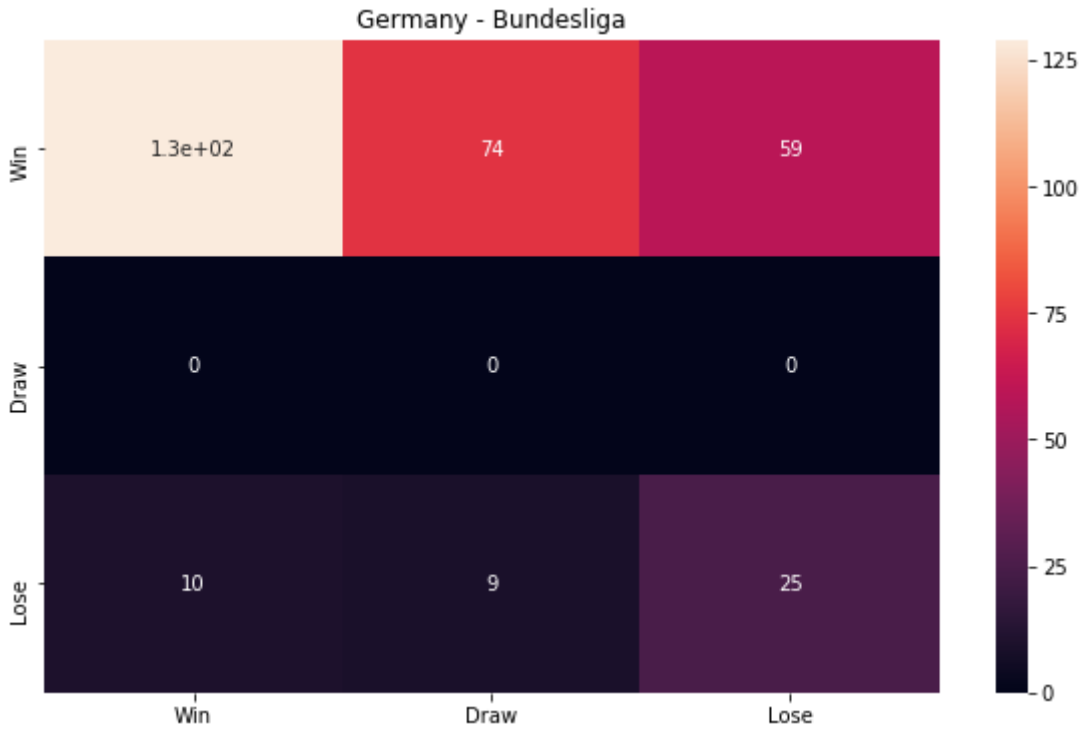

Figure AB5: Germany Bundesliga confusion matrix for 2018 - 2019 tournament using a Neural Network model

```
Precision:      Win 0.93        Draw 0.00       Lose 0.30
Recall:         Win 0.49        Draw 0.00       Lose 0.57
F1:     Win 0.64        Draw 0.00       Lose 0.39
```

Figure AB6: Germany Bundesliga metrics for 2018 - 2019 tournament using a Neural Network model

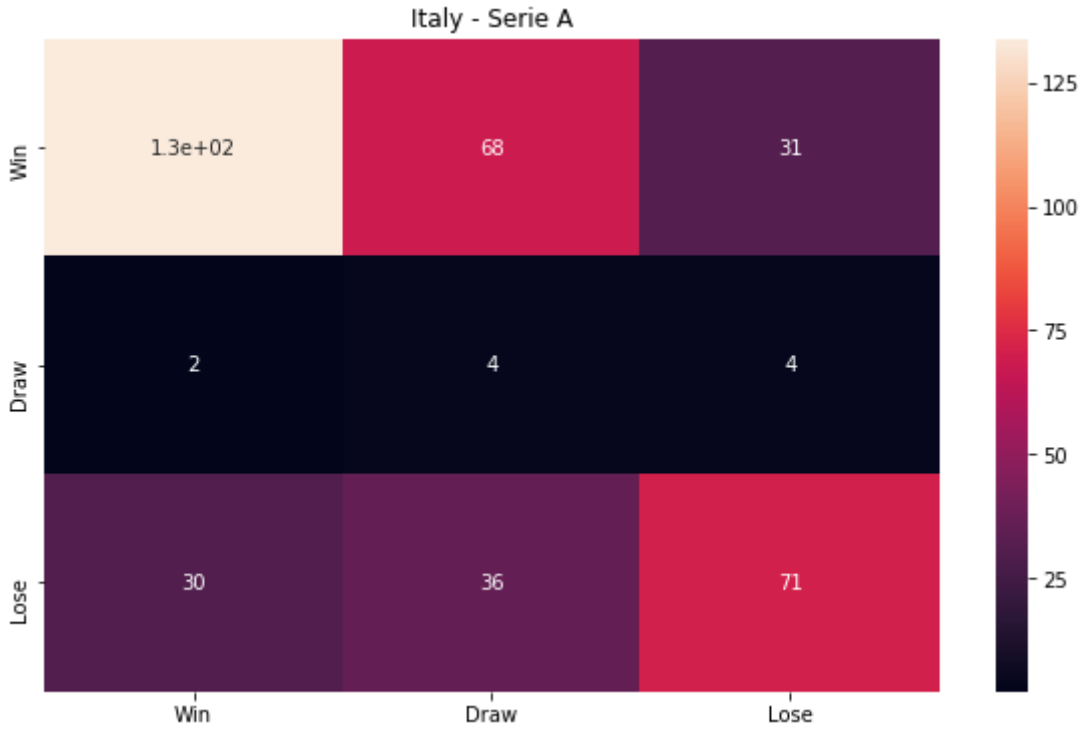

Figure AB7: Italy Serie A confusion matrix for 2018 - 2019 tournament using a Neural Network model

```
Precision:      Win 0.81        Draw 0.04       Lose 0.67
Recall:         Win 0.58        Draw 0.40       Lose 0.52
F1:     Win 0.67        Draw 0.07       Lose 0.58
```

Figure AB8: Italy Serie A metrics for 2018 - 2019 tournament using a Neural Network model

Appendix AC: Average Neural Network vs. Competition-based Neural Network

In Table AC1, accuracy of Average Neural Network that was trained using hyperparameters that maximise overall accuracy is compared with Competition-based Neural Network that consists of 4 Neural Networks, which were trained using hyperparameters optimised for the given competition. Evidently, Competition-based Neural Networks outperform Average Neural Network.

Competition	Average NN	Competition-based NN	Difference
England Premiership	59.58%	60.2%	+0.62%
France Championnat	45.78%	47.67%	+1.89%
Germany Bundesliga	47.71%	50.03%	+2.32%
Italy Serie A	54.74%	54.07%	-0.67%

Table AC1: Competition accuracy comparison of Average Neural Network vs. Competition-based Neural Network for 2018 - 2019 tournaments

Appendix AD: Football Neural Network with Other Sports

Figures AD1 through AD4 present performance information of the Neural Network that was created for football predictions and then applied to NFL season 2018. The Neural Network achieved 64.59% accuracy and 0.58% standard deviation. The bookie, which was Bet365, had 65.08% for NFL 2018. Full metrics for both the Neural Network and the bookie are given in Figures AD1 through AD4.

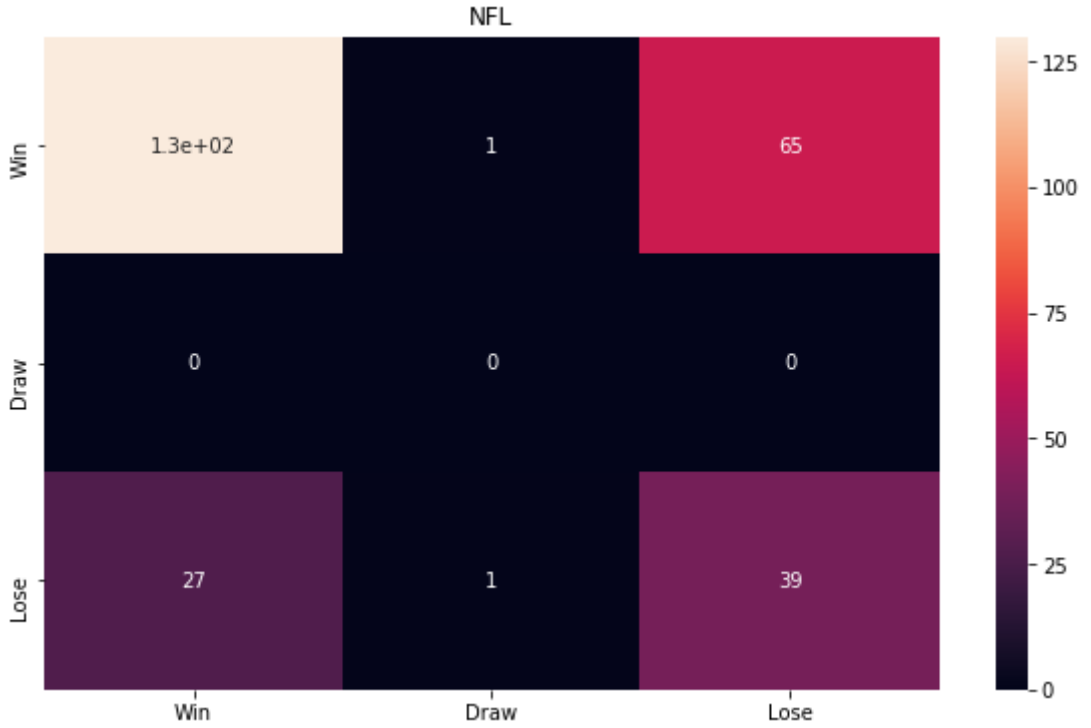

Figure AD1: Neural Network NFL confusion matrix for 2018

```
Precision:      Win 0.83        Draw 0.00       Lose 0.38
Recall:         Win 0.66        Draw 0.00       Lose 0.58
F1:     Win 0.74        Draw 0.00       Lose 0.46
```

Figure AD2: Neural Network NFL metrics for 2018

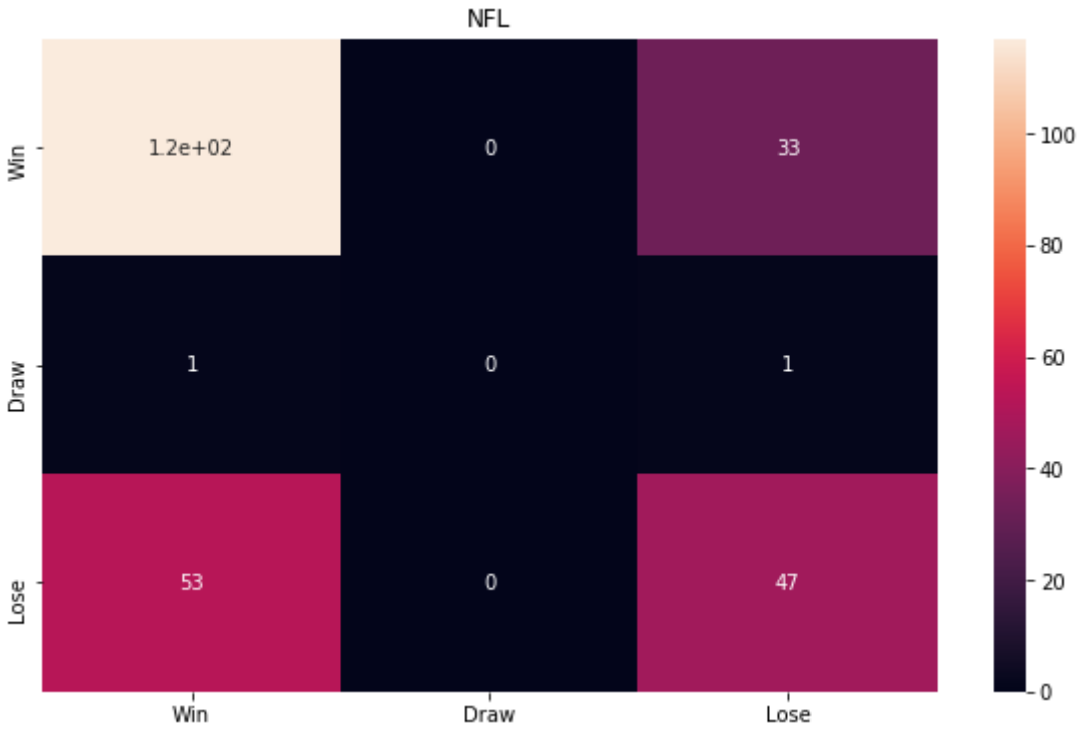

Figure AD3: Bookie NFL confusion matrix for 2018

```
MAE 0.9506098447702063
RMSE 0.9509345042003235
Precision:      Win 0.68        Draw 0.00       Lose 0.58
Recall:         Win 0.78        Draw 0.00       Lose 0.47
F1:     Win 0.73        Draw 0.00       Lose 0.52
```

Figure AD4: Bookie NFL metrics for 2018

Another sport that the Neural Network is trained and tested on is NBL (season 2018 - 2019). It appears that the Neural Network mostly chooses to predict the win class and so constantly misclassifies loses. For this sport, the Neural Network had 59.53% prediction accuracy. The model also had 0.41% standard deviation. On the other hand, Bet365 had 63.33% accuracy, significantly outperforming the model.

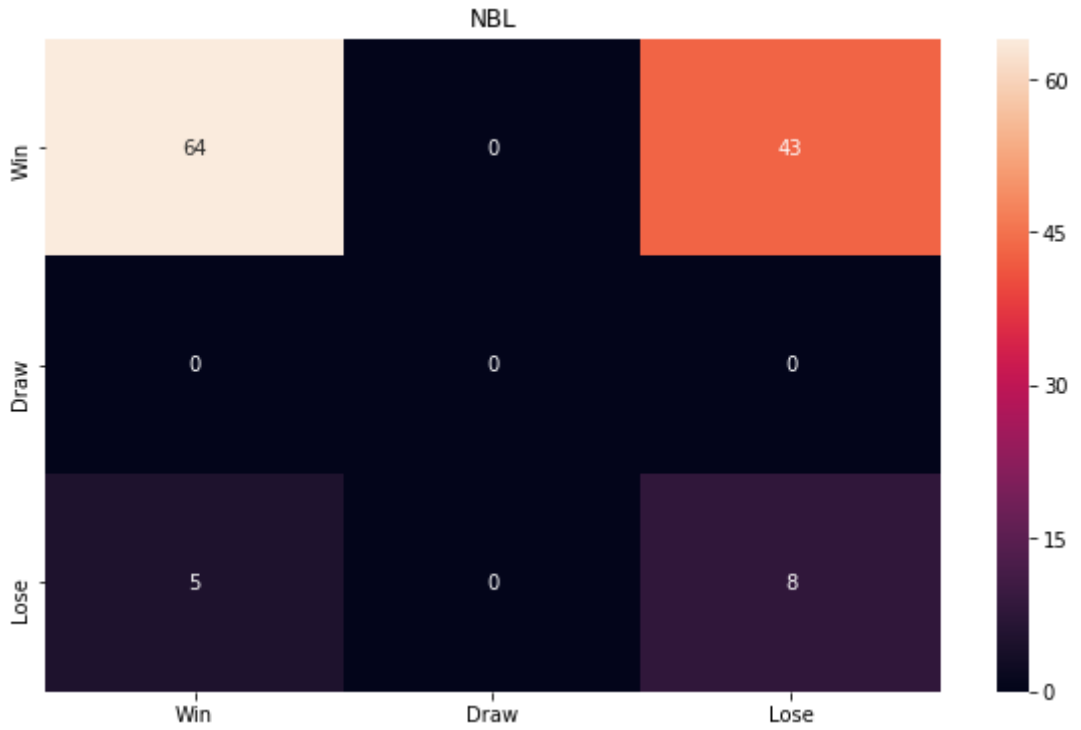

Figure AD5: Neural Network NBL confusion matrix for 2018 - 2019

```
Precision:      Win 0.93        Draw 0.00       Lose 0.16
Recall:         Win 0.60        Draw 0.00       Lose 0.62
F1:     Win 0.73        Draw 0.00       Lose 0.25
```

Figure AD6: Neural Network NBL metrics for 2018 - 2019

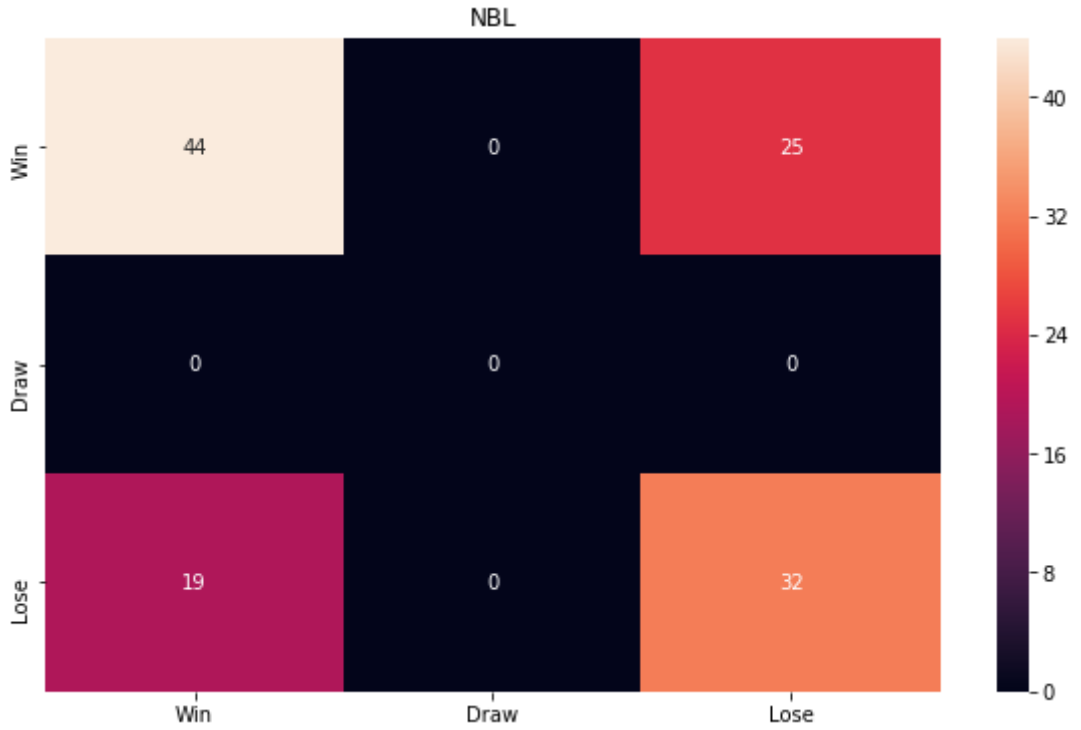

Figure AD7: Bookie NBL confusion matrix for 2018 - 2019

```
MAE 0.8726321195144724
RMSE 0.874568081016361
Precision:      Win 0.70        Draw 0.00       Lose 0.56
Recall:         Win 0.64        Draw 0.00       Lose 0.63
F1:     Win 0.67        Draw 0.00       Lose 0.59
```

Figure AD8: Bookie NBL metrics for 2018 - 2019

Next competition that the Neural Network was evaluated on is Super Rugby. Here, the model reached the accuracy of 67.38%, while the standard deviation was 0.97%. Bet365 achieved 62.7% prediction accuracy for the same competition. The exact figures are presented in Figures AD9 to AD12.

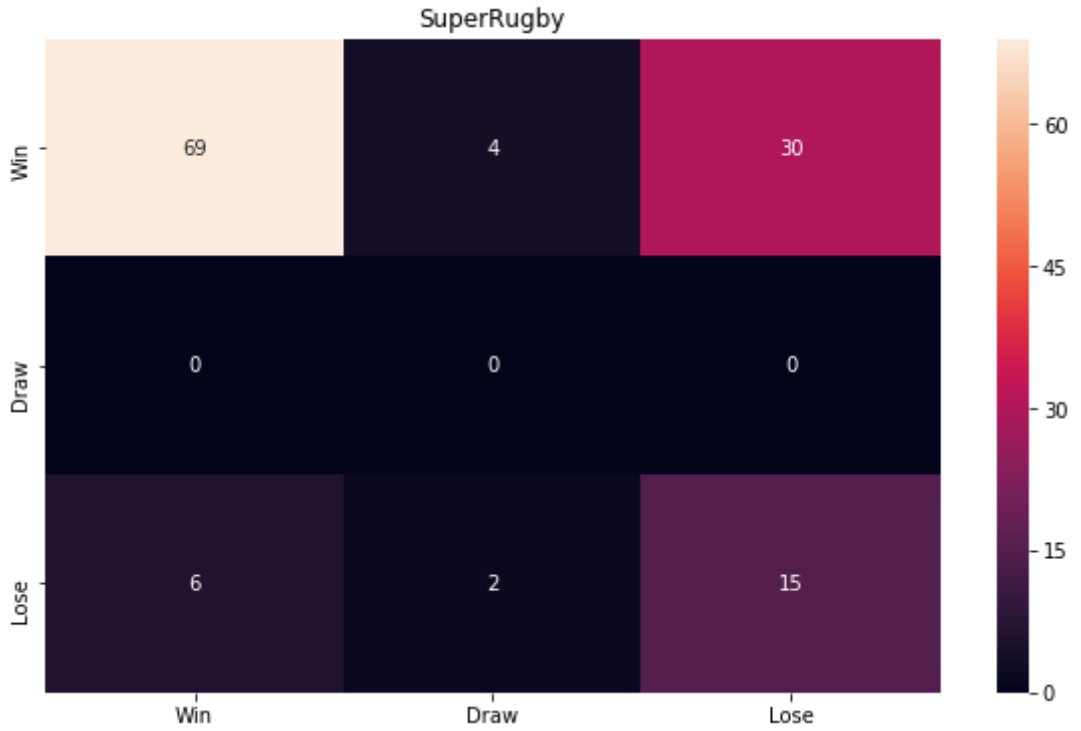

Figure AD9: Neural Network Super Rugby confusion matrix for 2019

```
Precision:      Win 0.91        Draw 0.00       Lose 0.38
Recall:         Win 0.68        Draw 0.00       Lose 0.65
F1:     Win 0.78        Draw 0.00       Lose 0.48
```

Figure AD10: Neural Network Super Rugby metrics for 2019

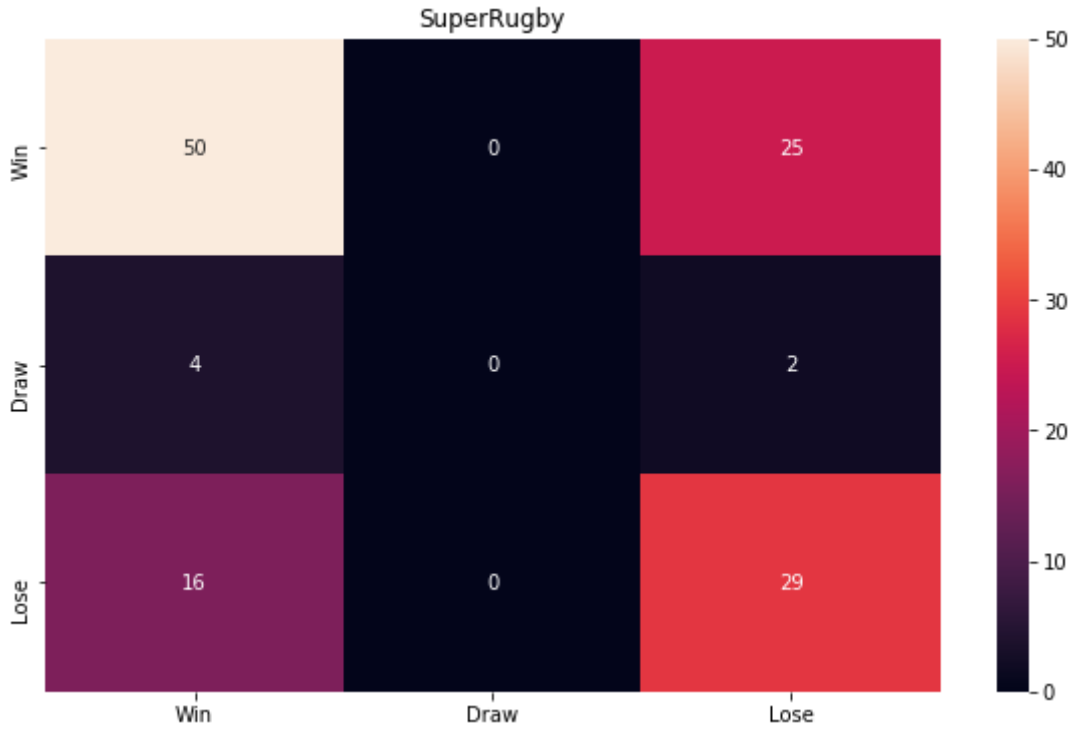

Figure AD11: Bookie Super Rugby confusion matrix for 2019

```
MAE 0.9307418785661291
RMSE 0.9363384462284559
Precision:      Win 0.71        Draw 0.00       Lose 0.52
Recall:         Win 0.67        Draw 0.00       Lose 0.64
F1:     Win 0.69        Draw 0.00       Lose 0.57
```

Figure AD12: Bookie Super Rugby metrics for 2019

The last competition that the Neural Network was used on is Twenty20 Big Bash. Even more so than with NBL, the amount of data seems insufficient for the model to generalise. Here, Bet365 had 45.88% accuracy, while the Neural Network reached 44.7% accuracy. The model was deterministic for this competition as it only predicted one class for all matches. The performance metrics for both the NN and the bookie are given in Figures AD13 through AD16.

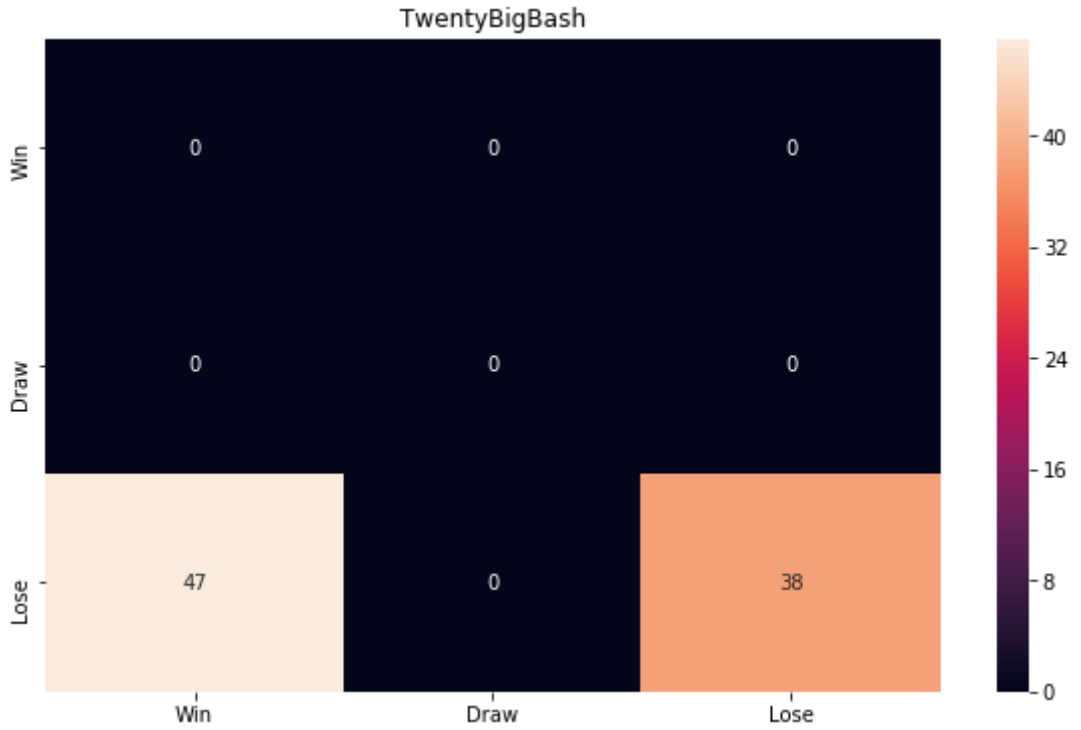

Figure AD13: Neural Network Twenty20 Big Bash confusion matrix for 2017 - 2018 and 2018 - 2019

```
Precision:      Win 0.00        Draw 0.00       Lose 1.00
Recall:         Win 0.00        Draw 0.00       Lose 0.45
F1:     Win 0.00        Draw 0.00       Lose 0.62
```

Figure AD14: Neural Network Twenty20 Big Bash metrics for 2017 - 2018 and 2018 - 2019

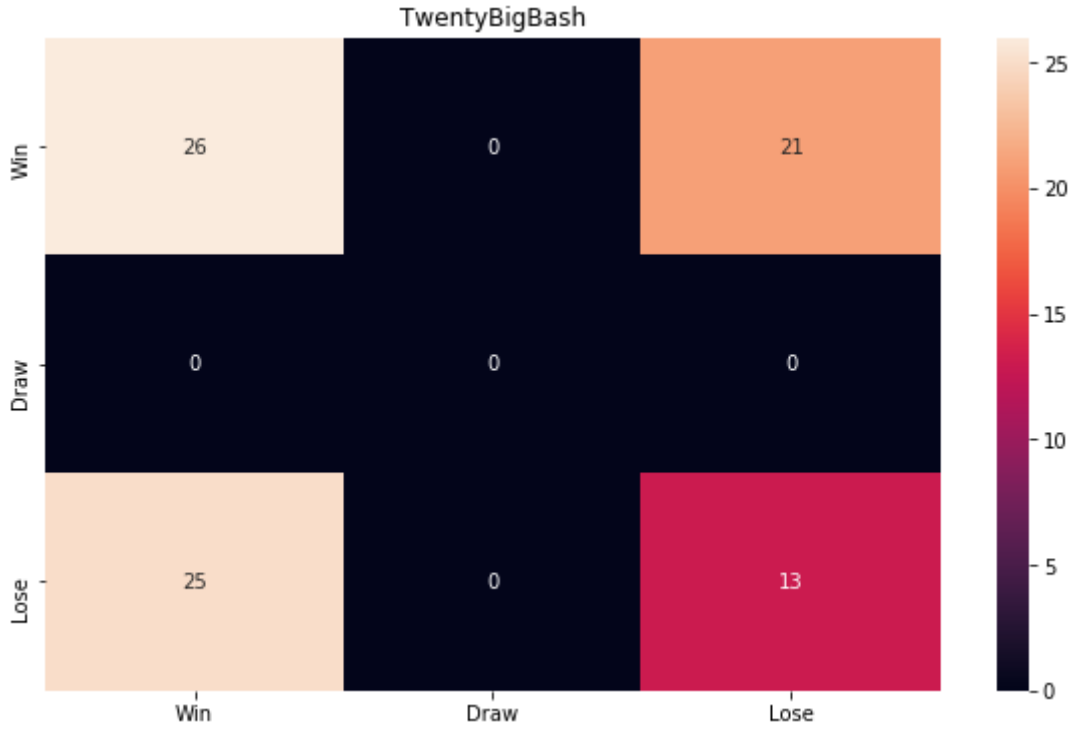

Figure AD15: Bookie Twenty20 Big Bash confusion matrix for 2017 - 2018 and 2018 - 2019

```
MAE 0.7890875395635517
RMSE 0.7963087397795303
Precision:      Win 0.51        Draw 0.00       Lose 0.38
Recall:         Win 0.55        Draw 0.00       Lose 0.34
F1:     Win 0.53        Draw 0.00       Lose 0.36
```

Figure AD16: Bookie Twenty20 Big Bash metrics for 2017 - 2018 and 2018 – 2019

Appendix AE: Neural Network Feature Set Analysis

Performance metrics for Neural Network developed in Section 3.4.11 are presented in Figures AE1 and AE2. Evidently, the results are similar to those obtained using the Average NN and Competition-based NNs (see Appendix AC). As with the other Neural Networks, the model metrics are obtained by utilising ensemble approach with 50 models. To determine the standard deviation, 25 independent iterations of the train/test cycle were recorded.

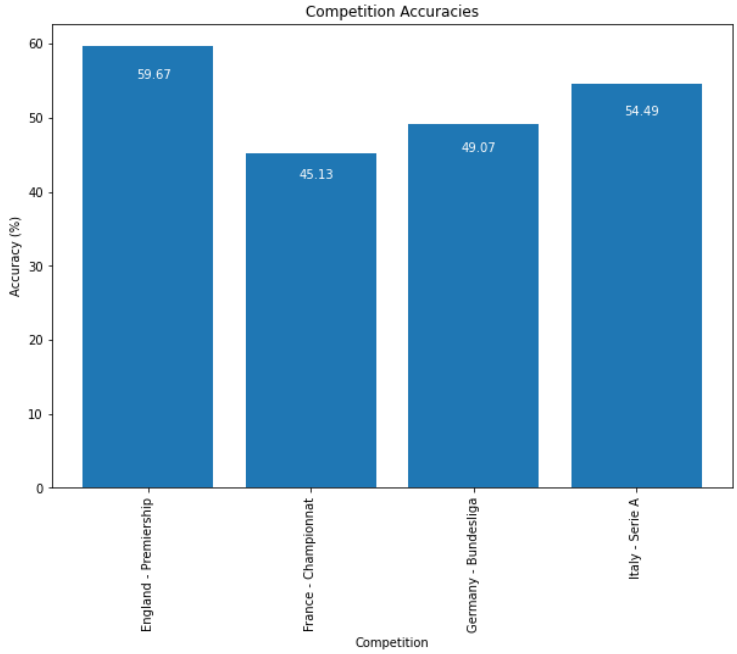

Figure AE1: competition accuracies using Neural Network with past performance features

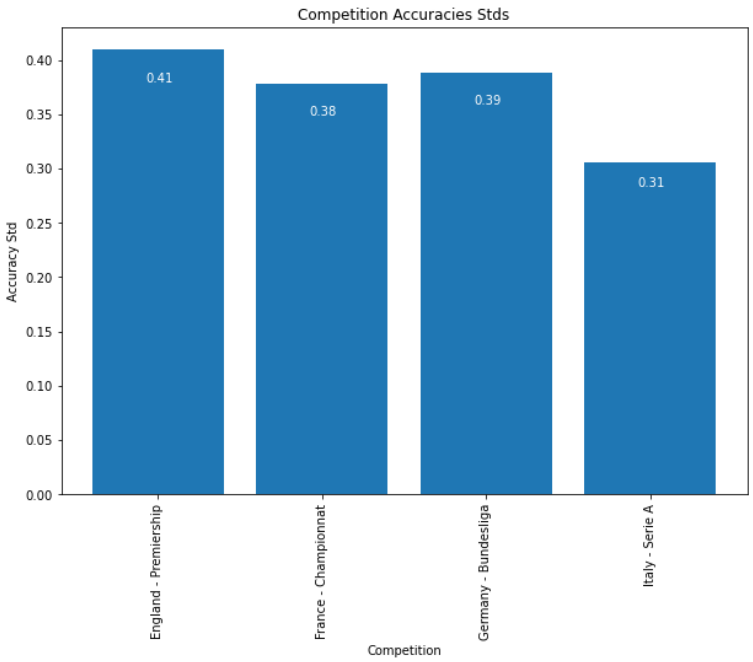

Figure AE2: competition accuracies standard deviations using Neural Network with past performance features

Next, sample confusion matrices for England Premiership, France Championnat, Germany Bundesliga and Italy Serie A are presented in Figures AE3 through AE8. Additionally, these contain more detailed information about performance, including F1, Precision and Recall.

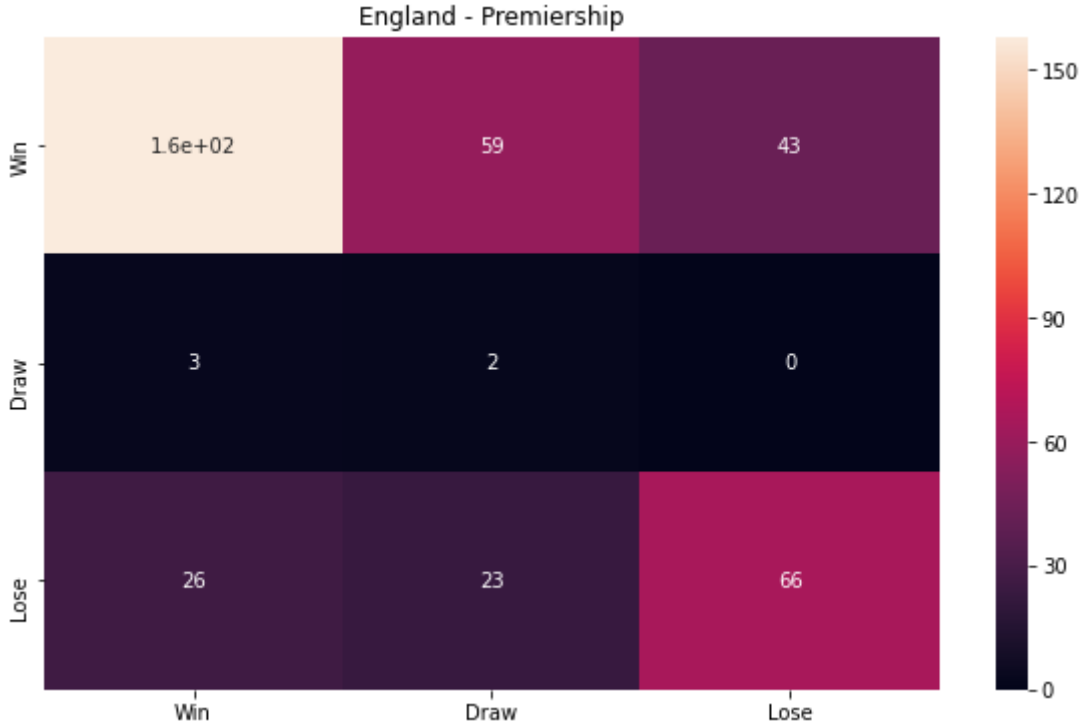

Figure AE3: England Premiership Confusion Matrix for Neural Network with past performance features

```
Accuracy: 59.47%
Precision:      Win 0.84        Draw 0.02       Lose 0.61
Recall:         Win 0.61        Draw 0.40       Lose 0.57
F1:     Win 0.71        Draw 0.04       Lose 0.59
```

Figure AE4: England Premiership performance metrics for Neural Network with past performance features

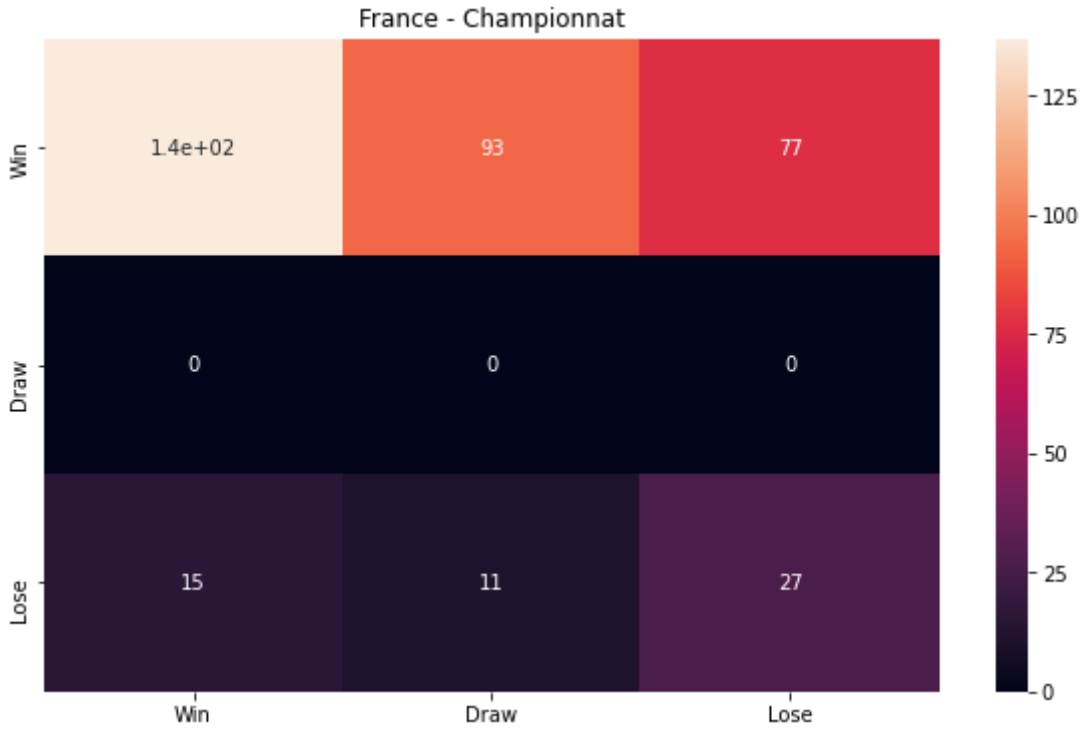

Figure AE5: France Championnat Confusion Matrix for Neural Network with past performance features

```
Accuracy: 45.56%
Precision:      Win 0.90        Draw 0.00       Lose 0.26
Recall:         Win 0.45        Draw 0.00       Lose 0.51
F1:     Win 0.60        Draw 0.00       Lose 0.34
```

Figure AE6: France Championnat performance metrics for Neural Network with past performance features

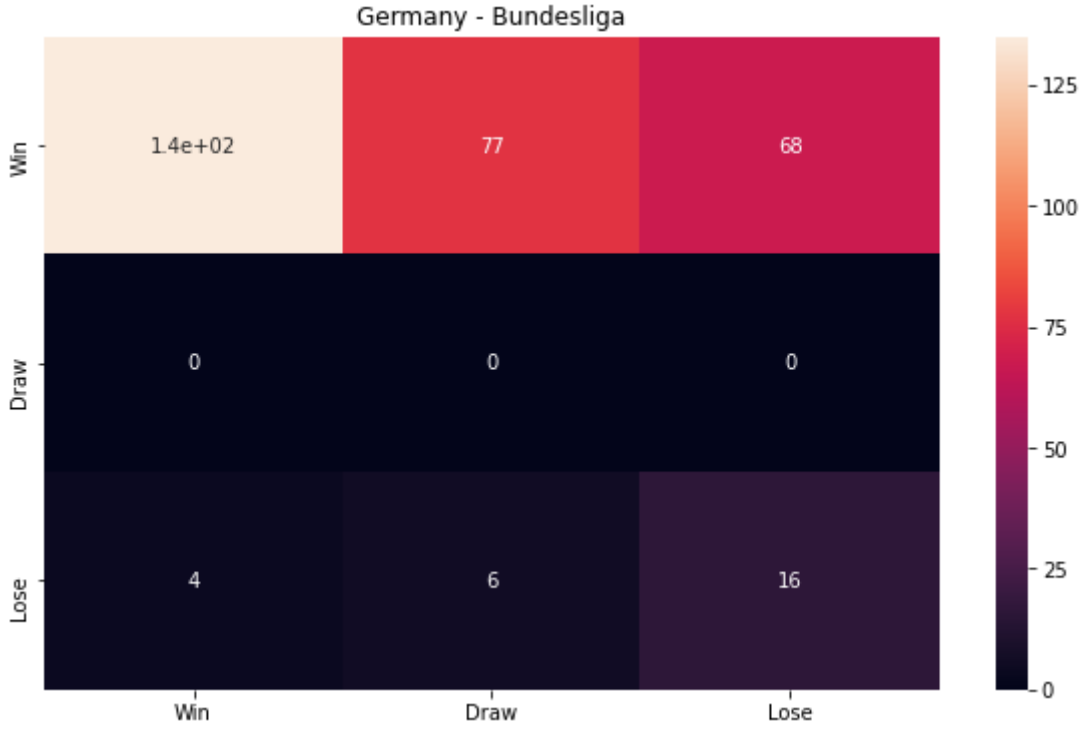

Figure AE7: Germany Bundesliga Confusion Matrix for Neural Network with past performance features

```
Accuracy: 49.35%
Precision:      Win 0.97        Draw 0.00       Lose 0.19
Recall:         Win 0.48        Draw 0.00       Lose 0.62
F1:     Win 0.64        Draw 0.00       Lose 0.29
```

Figure AE8: Germany Bundesliga performance metrics for Neural Network with past performance features

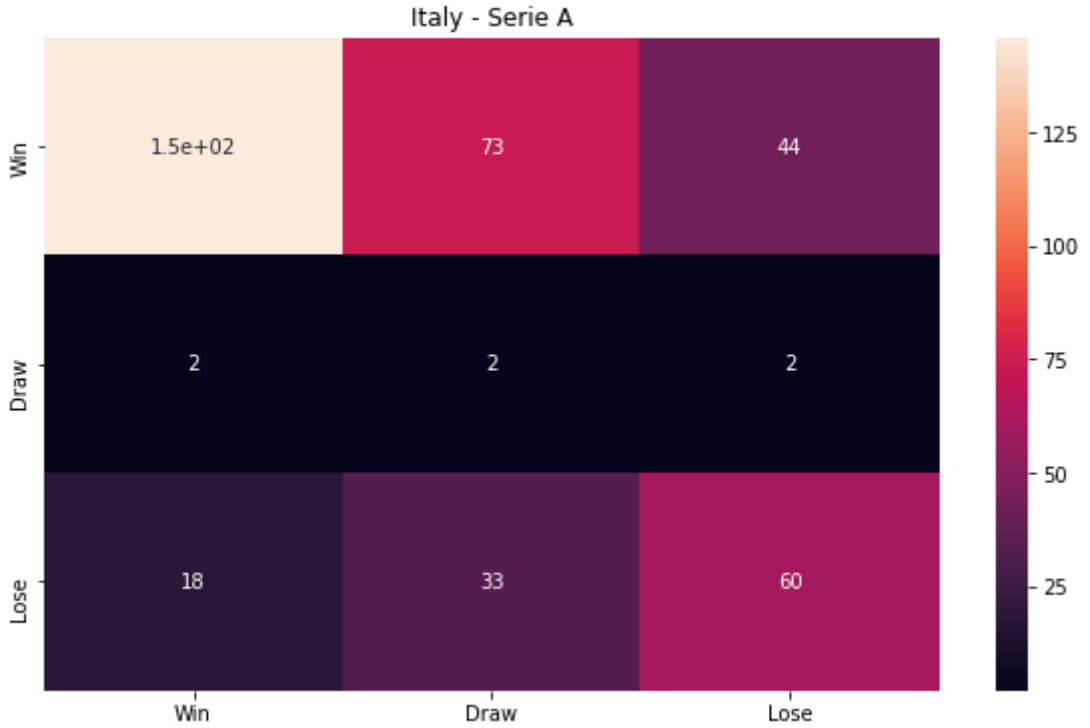

Figure AE9: Italy Serie A Confusion Matrix for Neural Network with past performance features

```
Accuracy: 54.74%
Precision:      Win 0.88        Draw 0.02       Lose 0.57
Recall:         Win 0.56        Draw 0.33       Lose 0.54
F1:     Win 0.68        Draw 0.04       Lose 0.55
```

Figure AE10: Italy Serie A performance metrics for Neural Network with past performance features

Appendix AF: Evaluation of Different Models

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	0.87	0.94	0.87	0.75
France Championnat	0.76	0.90	0.85	0.72

Germany Bundesliga	0.86	0.82	0.93	0.64
Italy Serie A	0.81	0.89	0.81	0.74

Figure AF1: Win precision comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Win Precision	0.825	0.8875	0.865	0.7125

Figure AF2: Win precision comparison of different models overall

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	0.63	0.55	0.6	0.65
France Championnat	0.48	0.46	0.48	0.5
Germany Bundesliga	0.5	0.47	0.49	0.56
Italy Serie A	0.54	0.50	0.58	0.6

Figure AF3: Win recall comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Win Recall	0.5375	0.495	0.5375	0.5775

Figure AF4: Win recall comparison of different models overall

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	0.64	0.28	0.60	0.55
France Championnat	0.45	0.22	0.38	0.31
Germany Bundesliga	0.42	0.25	0.3	0.3

Italy Serie A	0.62	0.42	0.67	0.56

Figure AF5: Lose precision comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Lose Precision	0.5325	0.2925	0.4875	0.43

Figure AF6: Lose precision comparison of different models overall

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
England Premiership	0.57	0.49	0.59	0.63
France Championnat	0.4	0.50	0.45	0.56
Germany Bundesliga	0.53	0.38	0.57	0.56
Italy Serie A	0.51	0.55	0.52	0.57

Figure AF7: Lose recall comparison of different models on different tournaments

Tournament 2018 - 2019	Elo	TrueSkill	Neural Network	Bet365
Average Lose Recall	0.5025	0.4675	0.5325	0.58

Figure AF8: Lose recall comparison of different models overall

Appendix AG: SAGE

SAGE

Response ID Completion date
514292-514283-60276149 28 May 2020, 11:46 (BST)
1 Applicant Name Anton Bendrikov
1.a University of Surrey email address
ab01719@surrey.ac.uk
1.b Level of research Undergraduate
1.b.i Please enter your University of Surrey supervisor's name. (If you have more than one supervisor, enter the details of the supervisor who will check this submission).
Paul Krause
1.b.ii Please enter your supervisor's University of Surrey email address. (if you have more than one supervisor, enter the details of the supervisor who will check this submission)
p.krause@surrey.ac.uk
1.c School or Department Computer Science
2 Project title Using Non-Standard Models for Football Predictions: Elo, TrueSkill, Neural Networks
4 Are you making an amendment to a project with a current University of Surrey/NHS REC/other favourable ethical opinion in place?
NO
5 Does your research involve any animals, animal data or animal derived tissue, including cell lines?
NO
6 This question is deliberately left blank.
Please click here to continue
7 Does your project involve* human participants, their data and/or any human tissue?
NO
8 Does your funder, collaborator or other stakeholder require a mandatory ethics review (e.g. Institutional Review Board (IRB) review) to take place at the University of Surrey?
NO
9 Does your project process personal data1? Processing covers any activity performed with personal data, whether digitally or using other formats, and includes contacting, collecting, recording, organising, viewing, structuring, storing, adapting, transferring, altering, retrieving, consulting, marketing, using, disclosing, transmitting, communicating, disseminating, making available, aligning, analysing, combining, restricting, erasing, archiving, destroying.

NO

10 Does your project require the processing of special category2 data?

NO

11 If you are an undergraduate or Masters student, are you ONLY using name and contact details for recruitment purposes, and no other personal data is being collected as listed in questions 9 and 10 above?

NO

12 Does your project involve any type of human tissue? This includes Human Tissue Authority (HTA) relevant, or irrelevant tissue (e.g. non-cellular such as plasma or serum), any genetic material, samples that have been previously collected, samples being collected directly from the donor or obtained from another researcher, organisation or commercial source.

NO

13 Does your research involve exposure of participants to any hazardous materials e.g. chemicals, pathogens, biological agents or does it involve any activities or locations that may pose a risk of harm to the researcher or participant?

NO

14 Will you be accessing any organisations, facilities or areas that may require prior permission? This includes organisations such as schools (Headteacher authorisation), care homes (manager permission), military facilities etc. If you are unsure, please contact RIGO.

NO

15 Will you be working with any collaborators or third parties to deliver any aspect of the research project?

NO

16 Will you be travelling to non-UK countries for any of your research activities?

NO

17 Will any research activities be conducted outside of the UK?

NO

18 Does your research involve lone working?

NO

19 Certain types of research require ethics approval from a nationally recognised research ethics committee (REC) which operates to standards set out by the Department of Health's Governance Arrangements for Research Ethics Committees. Recognised research ethics committees (REC) include NHS RECs and the MoDREC. Does your research involve any of the following? (select all that apply)

None of the above

20 Have you selected any of the options between A-O from question 19?

NO

21 Does your project require ethics review from another institution?

NO

28 Declarations

*I confirm that I have read the University's Code on Good Research Practice and ethics policy and all relevant professional and regulatory guidelines applicable to my research and that I will conduct my research in accordance with these. I confirm that I have provided accurate and complete information regarding my research project I understand that a false declaration or providing misleading information will be considered potential research misconduct resulting in a formal investigation and subsequent disciplinary proceedings liable for reporting to external bodies I understand that if my answers to this form have indicated that I must submit an ethics and governance application, that I will NOT commence my research until a Favourable Ethical Opinion is issued and governance checks are cleared. If I do so, this will be considered research misconduct and result in a formal investigation and subsequent disciplinary proceedings liable for reporting to external bodies. I understand that if any of my responses to the governance questions have requested additional documents, that these will be provided with my ethics and governance application if my project is to proceed. I understand that if I have selected any options from Qu 22-27 I MUST submit an ethics and governance application (EGA) for review in order to proceed with this research project UNLESS I am an undergraduate or Masters student, in which case I have completed Qu 29 below.

29 If I am conducting research as a student:

I confirm that I have discussed my responses to the questions on this form with my supervisor to ensure they are correct.