university of groningen

faculty of science and engineering

Bachelor's Thesis Mathematics

Forecasting football match results with the ordered logit model

by

Guanqun Ma

Supervisors:
Dr. M.A. Grzegorczyk
Dr. W.P. Krijnen

March 2019

Abstract

Football result prediction is a fascinating area to apply statistical modelling techniques on. This thesis applied ordered logit model in the frequentist framework to predict the outcome of the first half of 2018-2019 Eredivisie season. The influential factors considered are cup performance, game importance, team strength index as well as home advantage. ELO rating system is implemented for calculating the team strength index. With the help of AIC/BIC, the model is optimized and then used for prediction. The model predict the probabilities of outcomes for each game and the outcome with the highest one is taken as the prediction.

Contents

Chapter 1

Introduction

Football result prediction has become more and more popular due to the increasing prize pool of the competitive leagues and the emerging betting market. With football betting being increasingly regulated, the incentive for obtaining accurate predictions from statistical models is increasing heavily. While working on this thesis, the machine learning journal even published an issue containing 7 articles on this topic.

One of the main reasons which make it hard to predict the result of a football match is the existence of draw as an outcome. Taking the past season from Eredivisie as an example, out of 306 games played in the 2017-2018 season of Eredivisie, 137 were won by the home team, 96 of those were won by the away team and the remaining 73 resulted in a draw which contribute to almost $\frac{1}{3}$ of the total results. It is rather intuitive that under a perfectly random setting (odds of each outcome are $\frac{1}{3}$), a random prediction would give an expected accuracy of 33.33% And under the same situation in other sports such as baseball, tennis, basketball etc, the accuracy would be 50%.

A fair amount of research has been done on football outcomes and betting accuracy, however, the value of accurate prediction is not limited to betting. Scarf & Shi[1] evaluated the importance of league games in England from 2001 to 2004. Knowing the strengths and weaknesses, as well as the overall form of the next few opponents and your own team, helps the manager decide what line-ups to put up against different opponents and resting star players during back to back games. These insights for the overall season will help the bottom teams avoid relegation, as well as the champion contenders to win the league.

Professional football is relatively new in the Netherlands as the first season started back in 1956. Although it's not as popular or as commercialized as the likes of BPL (Premier league), it has a huge fan base inside the Netherlands nonetheless. Plenty of research has already been done using the data from the top 5 major leagues in Europe, however the smaller leagues have not been getting a lot of attentions became more attractive to predict the result of the dutch league.

Despite extensive literature on the subject of football result prediction, the methodologies and models applied still lack variety. Another fact worth pointing out is that despite the amount of research that has been done on factors that influence the outcome, there hasn't been as many published papers done on prediction models. For financial purposes, it is common to keep the highly accurate prediction results in house and that also brings more interest to the topic of this thesis. Therefore we reached the research question of this paper: "Would we be able to successfully predict the outcome of a football match with generalized linear model with a high accuracy?"

The aim of this bachelor project is to determine which generalized linear model is most suitable to predict the result of a football match and if the accessible data is sufficient for a reasonable accuracy. In our case, the ordered probit model proposed by Hvattum & Arntzen[2] is used in combination with the ELO rating system developed by ELO[3]. The ELO rating system generates pre-game team ratings for both teams and is used as a team-quality measure representing one of the independent variables in the ordered logit model. In combination with multiple control variables, the result of the match is then predicted.

The following chapter provides some prior work done by other researchers using different models to predict the match result. The featured factors used in previous literature will also be presented. In the third chapter the data selection and the introduction to the general model used in the thesis is presented. After that the modified model is presented along with a detailed explanation of the selection procedure. Chapter 5 presents the empirical result and in the last chapter the discussion about the result and the limitation of the paper with potential future improvement is shown, along with the conclusion of the whole paper.

Chapter 2

Literature Review

2.1 Prior Work

There are two main distinct ways to predict football outcome. One could always opt to predict the result of the match directly, namely home win/draw/away win. Alternatively one could forecast the number of goals scored by the home team and the away team and then generate the result by comparing them.

The initial approach is to use the forecasting methods based on goal scoring. Maher [4] stated that the number of goals scored by the home team and the away team can be modelled into two independent poisson distributions with the parameters determined by each team's attacking and defending abilities. Dixon & Coles [5] applied the basic assumption of Maher and developed two poisson distributions with the means as a function of three parameters, namely, attack, defence and home advantage. However Dixon & Cole [5] also mentioned in their paper that there is no theoretical explanation of the independence assumption of the two poisson distributions. Sillanpää & Heino [6] stated that there hasn't been any theory published to establish the fact that you can safely assume that the two poisson distributions are independent. They also mentioned that the attack and defence parameters in the model should be a time-dependent variable instead of a constant, as team can have good/bad forms during a specific period. One such example would be a winning streak. Even though Dixon & Coles [5] accounted for the situation and added weight on the recent games to navigate the situation in their paper, if the distributions were to be dependent, it would still cause a lot of error.

Accounting for the correlation of the two distributions, Baio & Blangiardo [7] applied hierarchical models in the Bayesian framework with the assumption of the conditional independence of the number of goals scored. He used Poisson-logNormal models with three parameters, namely, home advantage, attack and defence for both teams. The home advantage is assumed to be a fixed constant hence the prior distribution for the home advantage parameter is selected as normal distribution with an extremely small variance. To be exact, 0.0001 is selected to represent the fixed effect. As for the attack and defence factor for individual teams t (t = 1,...,18), it is modelled as follows:

$$att_t \sim Normal(\mu_{att}, \tau_{att}), \qquad def_t \sim Normal(\mu_{def}, \tau_{def})$$

where the hyper-priors of the parameters are again modelled as flat prior distributions:

$$\mu_{att} \sim Normal(0, 0.0001), \qquad \mu_{def} \sim Normal(0, 0.0001),$$

$$\tau_{att} \sim Gamma(0.1, 0.1), \qquad \tau_{def} \sim Gamma(0.1, 0.1),$$

The conclusion drawn from his work is that the team with the best attack will win the league. However, the accuracy of the prediction is relatively low for the relegated team, reason being that it overestimates the team's performance.

Suzuki et al.[8] also proposed to use the Bayesian framework to predict the outcome of the 2006 world cup. In their paper the prediction of certain specialist and power rankings from the FIFA website is used as information towards building the prior distribution. Instead of using the existing data from previous seasons, Suzuki et al.[8] developed the model from scratch and uses negative binomial distribution. He also separated the model depending on whether it's the first game of the season. Karlis & Ntzoufras [9] proposed to use a non-informative prior, more specifically, they used normal prior distributions for the parameters with zero as the mean and a large variance to express prior ignorance and then predict the result using Skellam's distribution.

Football leagues reward points directly for results and not for goals, which means that the number of goals scored by the opposing teams are not directly related to the result. Although the data for goals scored does present a more extensive data set, it could possibly introduce more noise to the system[10]. Goddard & Asimakopoulos[11] stated that predicting the outcome via goal scoring model requires larger amount of parameters and a complex procedure. In order to avoid the potential problems mentioned above, numerous researchers switch to the prediction method based on result. Kyupers[12] applied ordered probit regression model to predict the outcome of the 1993 -1994 and 1994 -1995 season in the four divisions of the English football league. However, the theoretical background of the model is lacking in his paper. Koning[13] provided a much better model description and he also included control variables such as home advantage factor and strength indicator for each team. Goddard & Asimakopoulos [11] built the first model that gathers information from results of the past matches with various explanatory variables. They argued that the importance of a match, the travel distance of both teams, the level of participation in the domestic/international cup are all significant factors influencing the outcome of a match.

In order to account for recent performance, Goddard & Asimakopoulos[11] applied the lagged performance covariate model with a long-term performance variable and a short-term performance variable. It appears that both the indicators decrease with respect to time in contribution to the forecast accuracy. An interesting conclusion is drawn from the empirical result, that the results from matches played more than two seasons ago contributed nearly nothing to the prediction. The short-term performance indicator is constructed such that it represents a team's form within the last two months. The games played more than two months ago is determined to have little to no impact on the forecast accuracy. By adjusting the parameters the authors are able to control the impact of recent performance on the prediction. A surprising fact gathered from the empirical results is that the short-term indicator contributed less than the long-term ones. This is despite the fact that indicators decrease in contribution to the forecast accuracy with respect to time. However, a downside of this model is that it requires a lot of control variables as it's all divided into home and away teams as well as short-term and long-term factors.

Hvattum & Arntzen [2] modified the idea of Goddard & Asimakopoulos and used the ELO model where they use relative performance levels between the two teams playing as the independent variable. The ELO rating system was initially developed to determine the skill difference between chess players[3], and later on it was extended for use in various other sports including football. The basic

ELO model contains the estimated and actual score, pre-game ratings and post-game ratings for the home and away team as well as a number of parameters. The actual score system is defined in such a way that a home win is rewarded by a score of 1, a draw gives a score of 0.5 and a loss contributes 0 to the score system. The ELO system then generates an expected score with a built in algorithm for both teams based on the pre-game ratings. Combining the expected score and the actual score, the post-game rating is reached. The post-game ratings for the teams are then stored and used as the pre-game rating for the upcoming match.

Hvattum & Arntzen [2] then applied the ordered probit model following the ELO rating system. They applied the process on the historical data of the top four divisions of the English league system, namely BPL, Champions league, league one and league two for 14 seasons. In their research, the data from the first two seasons are used for the calculation of the initial ratings for the ELO model. Then the next five seasons are used to estimate the parameters for the ordered probit regression. The last eight seasons are used in the testing phase.

Sillanpää & Heino [6] analyzed the differences between the ELO rating system and the lagged performance covariate model. An obvious advantage for the ELO system is the much simpler specification of the model. Both models apply a series of historical results to derive useful parameters so that they contain enough information to estimate the probability of the outcome of the future match. However, the ELO model only need the rating difference while the lagged performance model need around 30 parameters. The lagged performance variable model also failed to distinguish the results of tougher games from the easier ones. For instance, a 2:0 win against a significantly weaker opponent should be weighted differently than a 2:0 win against a more evenly matched opponent. They also argued that when appropriately parametrised, you would be able to see the difference in the expected scores between the two matches. A minor downside of the ELO model is the lack of consideration of the home advantage factor. While the lagged performance model introduced team-specific home advantage by adding the parameters that represent the recent home and away performance of a team. As for the accuracy, Hvattum & Arntzen[2] compared their result with that of Goddard & Asimakopoulos' model and concluded that it's more accurate to use the ELO model.

Sillanpää & Heino [6] improved on Hvattum and Arntzen's idea by adding a team-specific home advantage factor. They achieved that by splitting the team rating factor into home team rating and away team rating. In addition, they added extra control variables that could influence the prediction, namely, cup effects and pairwise home advantage.

In conclusion, due to the depth of the development history of the models, we opt for frequentist framework. Concerning the accuracy, the ordered logit model is chosen.

2.2 Featured factors

Dixon & Coles [5] indicated that in order to model the outcome of a football match accurately, factors such as different abilities of both teams, recent team performance, home team advantage should be included. Team abilities can be categorized into two groups, offensive and defensive ability. Logically speaking, teams with better characteristics are more likely to win. Team ability should be measured dynamically as it varies with time resulting from different factors such as managerial changes, team rotation changes, team morale etc.

Courneya & Carron[14] defined home advantage as "the consistent finding that home teams win over 50% of the games played under a balanced home and away schedule" in their paper. The existence of home advantage in football has been well documented. Clarke & Norman[15] estimated the home

advantage for individual teams in the English leagues taking into account the team abilities. 10 seasons of data is used to make sure the result is reliable and they calculated home advantage in terms of goals and wins for all 94 clubs in the English leagues. According to the result, home advantage has a great deal of influence on winning, however when it comes to goal margins, it is not that significant. Despite some exceptions, home ground advantage is worth just over 0.5 of a goal on average through out the four leagues.

Pollard & Pollard[16] analyzed various European leagues for evidence of a home advantage and provided a summary of partial aspects of home advantage. The main factors discussed in the paper are crowd factors (Nevill, Newell & Gale)[17], travel distance (Clarke & Norman[15]) and referee bias (Nevill, Balmer & Williams[18]). They concluded that travel distance is an insignificant factor for home advantage. The reason is that traveling for pro players has become easier and more and more comfortable through out the years. Crowd size is another factor concluded to have no direct influence to home advantage as Pollard & Pollard[16] found only a small fraction of difference in home advantage between the first and second divisions in the top five major leagues in Europe. However, there is a lot of evidence showing that the support of the crowd can influence the judgment of the referee.

Nevill et al[17] measured the factors associated with home advantage in English and Scottish football leagues for the 1992-1993 season. In their experiment, 'sending-off' and 'awarding penalties' are used to quantify the decision making by referees. Both variables are not only easily obtainable from all data set given on the official website, but also are likely to influence the match outcome. They observed that penalties are granted more often for the home team and more red cards are given against the away team. Although the number of penalties awarded do not always result in penalties scored, the frequency of the important decisions made by the referee, namely foul call with yellow card and sometimes even red card, which could sometimes be rewarded with penalties, indeed favoured the home side. Nevill et al[17] stated two possible explanations for the away team to receive more punishment. One of them being that the home crowds are able to cause psychological pressure on the away players, for instance, anxiety, which in turn cause them to panic and behave more aggressive on the pitch. An alternative explanation would be referee bias, the noise made by the home crowd could cause psychological pressure on the referee and influence his/her judgments and hence the decisions. They would be under the impression that the home team is always getting treated unfairly and question their own decision made before and could make compensation in the following foul call.

Later on Nevill et al[19] and Nevill et al[18] tested for referee bias through an experiment that makes the qualified referees watch video of a played game with and without the crowd noise. The result showed that referees from the experiment group with silent condition had fewer uncertain responses, chose more "no foul" option and the home players would receive more fouls comparing to the noise group. The presence of crowd noise influences referees when assessing tackles or challenges recorded on videotape. The avoidance of potential crowd displeasure is stated as a possible factor when decisions are made in favour of the home team.

Teams in the Eredivisie often participate in external competitions such as the KNVB cup and possibly champions league or europa league for the elite teams. Eredivisie is not one of the five major leagues, therefore the prize money from either the champions league or the Europa league is quite attractive for the teams. As for the KNVB cup, it is a battleground for the teams that are not able to achieve the qualification requirement for the europa league through playoff. The winners of the Toto KNVB Beker final will enter the 3rd qualifying round of Europa league for the upcoming season. Goddard & Asimakopoulos[11] analyzed the effect of cup involvement on the league performance for the teams in the English leagues. From a financial point of view, teams indeed have incentive to participate. They believe that this incentive to advance deep into the cup competition could be a distraction

for the teams and thus effect their league performance negatively. However, they also mentioned that if a team is successful in the cup competition, it could be a huge morale boost which could lead to an increased league performance. Whereas an early exit from the cup could hurt the team's morale and lead to a decreased league performance. The empirical results from the regression show a significantly positive relationship between cup involvement and the league performance.

Goddard[10][11] identified the importance of end-of-season matches. It could be tied with issues such as winning championship, promotion for the lower league teams, and relegation of the teams at the bottom of the table. He claimed that the match outcomes are affected by the incentives of the teams, namely if the match is more important for one side comparing to the other. Team with a higher incentive would obviously put more effort into the match. Also, it is also common for teams to train young players or try out new tactics if the match is of no importance when it comes to seeding in the league. Goddard defined two parameters $SIGH_{i,j}$ and $SIGA_{i,j}$ to represent the importance of a match for the home team and the away team respectively. $SIGH_{i,j}$ is set to be 1 if the match is important for the home team but not for the away team and 0 otherwise. $SIGA_{i,j}$ is set to be 1 if the match is important for the away team but not for the home team and 0 otherwise. A match is registered as important for the team if it is still possible to achieve the league goals, namely win the title, get a promotion, or avoid relegation, under the assumption that all the other teams that are competing for the same league goals gain average one point per game for the rest of the games.

Drawer & Fuller[20] studied the impact of players' injuries on club-performances, the results indicated that the injuries of key players significantly influence the team's performance. However, when only the Premier league is considered, the majority of the teams have enough depth in each position to prevent a performance dip due to the loss of a key player. Sillanpää & Heino[6] also stated that variables concerning player injuries are usually omitted from the result forecasting model due to the difficulty of data collection on this matter. They found no application of such a variable in historical statistical forecasting model and they also mentioned that it is not easy to measure how much an individual player is contributing to their team's performance at that given time before the injury. It has since become a common understanding that the injury problem will not be considered because it is a really complex variable to construct.

Chapter 3

Data and model introduction

3.1 Data selection

The Eredivisie is home to 18 clubs that each play each other at home and away, meaning the season is 34 games long. According to the rules, the team that finishes last place at the end of the season is relegated to the Eerste Divisie, which is the second level of the Dutch system, in line with the promotion of the Eerste Divise champion. The Dutch league operates a playoff system that's slightly more complex than in other countries. The two teams that finish above the bottom placed team in the Eredivisie go into a playoff with eight teams from the Eerste Divisie. The eight Eerste Divisie teams play down to two and these teams take on the Eredivise teams for a spot in the Eredivise the following year.

The primary dataset contains the match results from the Eredivisie from seasons 2016-2017 and 2017-2018, as well as matches from season 2018-2019 before the winter transfer window, totaling up to 765 matches. The dataset is collected from Footballdata.co.uk [21]. In which we select the full time result (FTR), with three categories (home win, draw or away win) along with the hometeam and awayteam. A secondary dataset consisting of the card stats (number of yellow and red cards received) was constructed. The data from season 2017-2018 and 2018-2019 is collected from Footballdata.co.uk [21], and the data from season 2016-2017 is manually constructed according to the information on worldfootball.net [22].

Another secondary data set consisting of the KNVB cup results from year 2016 to 2019 is constructed from the information on worldfootball.net [22]. This data is needed for the cup performance variable creation which will be shown later in the section. The initial ELO ratings (team strength factor) for each team in the Eredivisie is collected from clubelo.com [23]

The specific use of the information collected from these data set is demonstrated further in the following sections. Before diving into the specification of our model, the general idea of an ELO rating system and how the control variables are constructed will be shown first in the next section.

3.2 ELO system

The basic ELO model contains the estimated and actual score, pre-game ratings and post-game ratings for the home and away team as well as a number of parameters. The actual score system is

defined as follows: A home win is rewarded by a score of 1, a draw gives a score of 0.5 and a loss contribute 0 to the score system. The official notation is shown below:

$$\alpha^H = \begin{cases} 0 & \text{if the away team wins} \\ 0.5 & \text{if the result is a draw} \\ 1 & \text{otherwise} \end{cases}$$

Naturally it follows that the score for the away team is: $\alpha^A = 1 - \alpha^H$. In order to illustrate the formula representation of expected score and the strength index for both teams, the following parameters are introduced:

$$\begin{aligned} \ell_0^H &= \text{pre-game strength rating of home team} \\ \ell_1^H &= \text{post-game strength rating of home team} \\ \ell_0^A &= \text{pre-game strength rating of away team} \\ \ell_1^A &= \text{post-game strength rating of away team} \end{aligned}$$

The ELO calculation then assumes the following expression for the expected scores:

$$\gamma^H = \frac{1}{1 + c^{\frac{\ell_0^H - \ell_0^A}{d}}} \tag{3.1}$$

$$\gamma^A = 1 - \gamma^H = \frac{1}{1 + c^{\frac{\ell_0^A - \ell_0^H}{d}}} \tag{3.2}$$

Where c and d represent the appropriate scale for the ratings, and k is a parameter of change of the ratings. Note that k has to be chosen properly to reflect how much weight a match has on the strength rating of the team. If the value of k is too low, a team's rating will not be sensitive enough, while if the value of k is too high, the rating will not be stable enough to fit in the model.

The new rating for both teams after the match is calculated as follows:

$$\ell_1^H = \ell_0^H + k(\alpha^H - \gamma^H) \tag{3.3}$$

$$\ell_1^A = \ell_0^A + k(\alpha^A - \gamma^A) \tag{3.4}$$

For any upcoming match the pre-match ratings of both teams are taken as their post-game rating from their previous matches. Then they are used along with the given c and d to derive the expected scores for the upcoming match using equations 3.1 and 3.2. Subsequently the expected scores are taken into equations 3.3 and 3.4. Combining the expected scores, the selected k and the actual score after the game has been played, the post-game rating can then be calculated and stored to be used as pre-game rating for the upcoming match.

3.3 Ordered logit model

The ordered logit model describes the relation between a discrete dependent variable and the independent variables. During the process, a continuous and unobserved dependent variable is created to present how likely it is to observe a specific value of the actual dependent variable (observable).

The relation between the unobserved variable and the independent variables is assumed to be linear with a disturbance term.

The randomness/disturbance in ordered logit model is governed by the standardized logistic distribution. The probability density function of a logistic regression is shown below:

$$P(X = x) = \frac{e^{-\frac{x-\mu}{s}}}{s(1 + e^{-\frac{x-\mu}{s}})^2}$$

with variance being

$$var(x) = \frac{s^2\pi^2}{3}$$

Here s is a scaling factor and μ is the mean. In order to obtain the standardized logistic distribution, the mean μ is taken to be 0 and the scaling factor s is chosen in such a way that the variance becomes 1. Hence the resulting pdf for the standardized logistic distribution has the following pdf:

$$P(X = x) = \frac{e^{-\frac{\sqrt{3}x}{\pi}}}{s(1 + e^{-\frac{\sqrt{3}x}{\pi}})^2}$$

In order to presents a more clear expression of the unobserved dependent variable, we define the following notations:

$$\begin{aligned} y &= \text{observed dependent variable} \\ \mu_n &= \text{the endpoint of the n-th discrete state} \\ y^* &= \text{the unobserved exact value of the dependent variable} \\ x &= \text{vector for the independent variables} \end{aligned}$$

As mentioned above, the relationship between y^* and x is assumed to be linear with the notation shown below:

$$y^* = \beta^T x + \epsilon$$

where ϵ represents the disturbance term and β^T is a vector of parameters.

Now that we have established how y^* is created, we proceed to show the connection between y and y^*. In the ordered logit model, it is assumed that there exists some threshold points for y^* that divides the value of y^* into multiple categories. If a value lies in between certain threshold points, it implies that y is more likely to obtain the corresponding result of that category than the others. The category of responses is defined below:

$$y = \begin{cases} 0 & \text{if } \mu_1 < y^* \leq \mu_2 \\ 1 & \text{if } \mu_2 < y^* \leq \mu_3 \\ 2 & \text{if } \mu_3 < y^* \leq \mu_4 \\ ... \\ ... \\ ... \\ n & \text{if } \mu_n < y^* \end{cases}$$

The cdf of the logistic regression is needed to form an expression for y and it can be calculated from the pdf given above. The notation of the cdf is shown below:

$$P(X \leq x) = \frac{1}{1 + e^{-\frac{x-\mu}{s}}}$$

The probabilities for all the categories can then be obtained by combining the cdf and the representation for category of responses. It is assumed that the probability of observing a specific category of y depends on the independent variables and the threshold values. For an individual pair of the observed variable (y_i) and the independent variables (x_i), the notation is then shown below:

The notation is then shown below:

$$\begin{aligned}
(y = 0|x_i) &= \frac{1}{1 + e^{\beta^T x_i - \mu_1}} \\
(y = 1|x_i) &= \frac{1}{1 + e^{\beta^T x_i - \mu_2}} - \frac{1}{1 + e^{\beta^T X_i - \mu_1}} \\
&\ldots \\
(y = n|x_i) &= 1 - \frac{1}{1 + e^{\beta^T x_i - \mu_{n-1}}}
\end{aligned}$$

As can be seen from the expression above, the probability of obtaining a value for the observed dependent variable (y) is governed by logistic regression. Essentially the ordered logit regression is a combination of multiple logistic regressions each represent a category. Also worth noting is that each individual category is ordinal which is a key requirement for the model.

Chapter 4

Variables creation and model specification

4.1 Cup performance

The variable cup_i is created for both teams to demonstrate the effect of cup game participation on the league game result. Differing from the literature, Champions league and Europa league is omitted due to the fact that Eredivisie teams have been performing sub-par in recent years. Teams from Eredivisie either got eliminated rather early or do not qualify. Note that for the 2018-2019 season, the result for the games after the January transfer window are predicted under the assumption that the better team in ELO rating at that moment wins for simplicity. The official notation is defined as follows:

$$cup_h = \begin{cases} 0 & \text{if the hometeam is eliminated from KNVB cup} \\ 1 & \text{otherwise} \end{cases}$$

$$cup_a = \begin{cases} 0 & \text{if the awayteam is eliminated from KNVB cup} \\ 1 & \text{otherwise} \end{cases}$$

For simplicity, teams from the first league and amateur teams participating in the cup are assumed to always lose against a team from Eredivisie.

4.2 Game importance

The game importance factor $importance_i$ is defined as follows:

$$importance_h = \begin{cases} 1 & \text{if the game is important for the hometeam and not for the awayteam} \\ 0 & \text{otherwise} \end{cases}$$

$$importance_a = \begin{cases} 1 & \text{if the game is important for the awayteam and not for the hometeam} \\ 0 & \text{otherwise} \end{cases}$$

In the literature review section it is mentioned that a game is important if it is still possible for the the team to achieve the league goal. Here we elaborate on it and divide into two categories. Either

it is possible for a team to compete for top eight which lead to Champions league and Europa league qualification, or it leads to 13-15th placement which means the team is outside relegation and not in the mid-table. The assumption that teams that avoid relegation represent the teams in 13-15th place came from general knowledge of football and the term mid-table.

4.3 Model specification

Goddard, J. & Asimakopoulos [11] mentioned that games two years prior to the upcoming match are no longer relevant to the prediction. Additionally there has been some drastic change of rules and style of play of football since 2015, only the 2016-2017 and 2017-2018 season has been selected for training. After that the model is tested on the 2018-2019 season and evaluated on the accuracy of the first half of the season.

The general settings for the ELO model is adopted from github [24]. Hvattum & Arntzen [2] calibrated the parameter k from 12 seasons of English football matches. He manage to obtain the best k value that minimized the quadratic information loss and found it to be 20. As for c and d, they claimed that they serve only to set a scale for the ratings and opt with $c = 10$ and $d = 400$. We decided to use the same value for c and d, and after testing various k values, we opt to go with 20 as it also provided the best prediction accuracy for us.

Another setting worth mentioning is that for every new season there are potentially some teams getting promoted along with teams getting relegated. In this thesis, we assign the newly promoted team with the ELO rating of the relageted teams ordered by ranking. For example, the first place team in Eerste Divisie will get the rating of the best relegated team rating wise.

Now that we have introduced the necessary background knowledge for the general model as well as the variable construction, we express the general form of our model. Just to keep inline with the ELO system, we defined the category of responses as follows:

$$y = \begin{cases} 0 & \text{if the away team wins} \\ 0.5 & \text{if the result is a draw} \\ 1 & \text{if the home team wins} \end{cases}$$

Then from the ordered logit model, adopting the notations in the previous section, the probability for obtaining a particular value is set as follows:

$$\begin{aligned} (y = 0|x_i) &= \frac{1}{1 + e^{\beta^T x_i - \mu_1}} \\ (y = 0.5|x_i) &= \frac{1}{1 + e^{\beta^T x_i - \mu_2}} - \frac{1}{1 + e^{\beta^T x_i - \mu_1}} \\ (y = 1|x_i) &= 1 - \frac{1}{1 + e^{\beta^T x_i - \mu_2}} \end{aligned}$$

where

$$\begin{aligned} \beta^T &= \text{parameter vector for the independent variables} \\ x_i &= \text{independent variables used in the model} \\ \mu_1, \mu_2 &= \text{cut-off points for classification} \end{aligned}$$

Now simply define the ELO rating difference between home team and away team as ratingdiff, difference between yellow and red card recieved by home team and away team as yellowdiff and reddiff respectively, we can finally present the initial model for the training set:

$$\begin{aligned} \text{y} = \beta_1 \text{ratingdiff} + \beta_2 importance_h + \beta_3 importance_a + \beta_4 cup_h \\ + \beta_5 cup_a + \beta_6 \text{yellowdiff} + \beta_7 \text{Reddiff} + \epsilon \end{aligned}$$

4.4 Omitted variables

As discussed in the literature review, the impact of player's injuries is omitted in this thesis because of the fact that it is hard to quantify the impact of a specific player on team strength. Also, for the prediction of the later part of the season, the referee bias factor is omitted due to lack of data. This data can be collected from gambling websites however it is not clear how reliable the data is. The uncertainty of the data and the fact that it is not freely available, the variable is therefore out of consideration in this thesis.

Furthermore, the team-specific home advantage factor is omitted during the construction of the model. The reason is that it is hard to fit this factor into the exist ELO system. Although it could be done, it creates a rather complex mathematical model with more variables. Therefore, due to the time limit of this research, it is not taken into account at this moment.

4.5 Maximum likelihood estimation

Maximum likelihood estimation is applied to estimate the value for the parameters β and the cut-off points μ_i that describe the relationship between y and x most accurately. In order to express the likelihood function, we first introduce the following notations:

$$\begin{aligned} \hat{\beta^T} &= \text{estimator for the vector of parameters } \beta^T \\ \hat{\mu_n} &= \text{a set of estimators for all the threshold points } \mu_n \\ \hat{cdf}_{in} &= \text{cdf value for the standard logistic regression given equation } \hat{\beta^T} x_i - \mu_n \end{aligned}$$

Therefore the log-likelihood function can be calculated as follows:

$$
\begin{aligned}
lnL = & \sum_{i,y_i=1} ln(c\hat{df}_{i1}) \\
& + \sum_{i,y_i=2} ln(c\hat{df}_{i2} - ln(c\hat{df}_{i1}) \\
& \ldots \\
& + \sum_{i,y_i=n} ln(c\hat{df}_{in} - ln(c\hat{df}_{i(n-1)})
\end{aligned}
$$

Maximum likelihood estimators are the parameters that maximize the likelihood function above.

4.6 AIC and BIC

While using a statistical model to represent the relationship between the dependent and the independent variables, the result is never 100% accurate. To put in order words, some information contained in the data set will be lost using the representation. Therefore after the initial model is obtained, statisticians usually follow up with an analysis of information lose. After that, the model is altered to obtain the least amount of information loss. Two of the most popular methods are Akaike information criterion (AIC) and Bayesian information criterion (BIC). AIC is an evaluation criterion for the goodness of fit of the model if the parameters are estimated by the maximum likelihood method. Given a selection of models with different amount of parameters, the one with the lowest AIC value is the best fitting model for that data set. Konishi.S [25] states that if the exact distribution that represent the data exists close to a model within the selection, the bias associated with the log-likelihood of the model based on the maximum likelihood estimation of the parameters can be approximated by the number of parameters. AIC values both the simplicity and the goodness of fit of the model. To keep the model as simple as possible, it gives out a penalty for more parameters used in the model to reduce biasness. The official notation of AIC is defined as follows:

$$AIC = -2(ln\hat{L}) + 2k$$

where $ln\hat{L}$ is the maximum value of the log-likelihood function and k is the number of free parameters for each different models in the comparison.

BIC is closely related to AIC when it comes to definition, they both try to punish the overfitting of the model, namely too many parameters used. However, the difference is that BIC has a much heavier penalty on bias then AIC. The official notation of BIC is shown below:

$$BIC = -2(ln\hat{L}) + kln(n)$$

where n is the sample size that is sufficiently large, $ln\hat{L}$ is the maximum value of the log-likelihood function and k represents the number of parameters estimated by the model. As can be seen from the formula, the penalty on bias for BIC is $kln(n)$, which is way larger than $2k$ in AIC.

Konishi.S [25] stated that the expression above is obtained by approximating the marginal likelihood associated with the posterior probability of the model by Laplace's method for integrals. The detailed

derivation of the model is beyond the scope of this thesis. For a more detailed description of the procedure, see for example Konishi.S [25]

Chapter 5

Result

In this chapter we report the results of our testing. Firstly we report the result of the initial model. Then we demonstrate the AIC and BIC result for model selection. Finally we present the prediction result and the accuracy comparison with the actual result.

5.1 Result of the initial model

In this section we show the result of the logit model. As shown in table 5.1, rating difference has a positive effect on obtaining better results ($p < 0.001$). For one unit increase in rating difference, we expect a 0.056 increase in the expected value in the direction of getting to the home win category on the log odds scale, given that all of the other variables in the model are held constant. Red card difference is obtained by subtracting the number of red card received by the away team from that of the home team. It appears that red card difference has a negative effect on obtaining better results ($p < 0.001$), and for one unit increase in red card difference, we expect a 1.236 increase in the expected value in the direction of getting to the away win category on the log odds scale, given that all of the other variables in the model are held constant. As for the importance factor for away teams, it shows a minor positive effect on getting a better outcome ($p < 0.1$). For one unit increase in away importance factor (which means from 0 to 1), it should bring a 0.59 increase in the expected value in the direction of getting a home win on the log odds scale, given that all of the other variables in the model are held constant. As for the other proposed independent variables, namely importance factor for the home team, and cup participant factor for both teams as well as and for yellow card difference, it appears that no conclusion can be drawn from this regression result.

	Value	Std. Error	t value	p value
ratingdiff	0.005609623	0.0005627429	9.9683579	2.096665e-23
importance_h	-0.090653105	0.2563848086	-0.3535822	7.236520e-01
importance_a	0.590336893	0.2694139597	2.1911890	2.843812e-02
cup_h	-0.162265563	0.1288796472	-1.2590472	2.080133e-01
cup_a	-0.098359960	0.1291951148	-0.7613288	4.464607e-01
Yellowdiff	-0.069714104	0.0514522653	-1.3549278	1.754406e-01
Reddiff	-1.236454896	0.2539714518	-4.8684799	1.124600e-06
A|D	-0.631269881	0.2437290503	-2.5900478	9.596262e-03
D|H	0.653121637	0.2438422352	2.6784598	7.396160e-03

Table 5.1: Ordered logit model result

5.2 AIC and BIC result

In this section we first present the AIC result which selects the best fitting model by subtracting independent variables that are not significant to minimize noise. According to the "StepAIC" function in R, the result is shown in table 5.2.

	tryout	Df	AIC
1	none		1132.98
2	- Reddiff	1	1166.3
3	- ratingdiff	1	1261.2

Table 5.2: AIC result

The stepwise procedure based on AIC indicated that the model obtained from the significant test is already the best fit for the dataset. After that the BIC result was created by manually going through each options and the result is shown in table 5.3.

	tryout	Df	BIC
1	none		1150.644
2	- Reddiff	1	1179.576
3	- ratingdiff	1	1274.428

Table 5.3: BIC result

Again the BIC selection conclude that the original model is the best fit. Therefore we conclude our final model as follows:

$$\text{FTRordered} = \beta_1 \text{ratingdiff} + \beta_2 \text{Reddiff} + \epsilon$$

5.3 Prediction result and accuracy

Applying the final model from the previous section to the testing data, we obtain the prediction result of 55 away wins and 98 home wins with a 60 % accuracy. From the 153 games being predicted, the actual result consists of 53 away wins, 32 draws and 68 home wins. The confusion matrix is shown below:

	Reference		
Prediction	A	D	H
A	32	12	11
D	0	0	0
H	21	20	57

Table 5.4: Confusion matrix

As can be seen from the confusion matrix, no draw has been predicted. 33 out of 53 away wins is correctly predicted which gave a 62 % accuracy and 58 out of 68 home game is successfully predicted which gave a 85 % accuracy.

Table 5.5 shows the sum of the probabilities of outcomes in 153 matches, where you can see that the percentage of obtaining an outcome is similar to the actual distribution of the result.

count_away	48.7808 %
count_draw	35.1624 %
count_home	69.0511%

Table 5.5: sum of percentages of obtaining a specific result for each match for the whole testing set

The league table following the prediction result is shown in table 5.6. Comparing to the actual result table in 5.7, it can be seen that although the prediction accuracy is not desirable, the league table maintained a comparable result except some outliers such as PEC Zwolle and FC Groningen. The top three teams seperate themselves from the rest of the league and the bottom 6-7 teams are correctly predicted except the outliers mentioned above.

Prediction result					
rank	Team	homewin	draw	awaywin	point
1	PSV Eindhoven	17	0	0	51
2	Ajax	16	0	1	48
3	AZ Alkmaar	15	0	2	45
4	Feyenoord	12	0	5	36
5	PEC Zwolle	11	0	6	33
6	Heerenveen	10	0	7	30
7	ADO Den Haag	10	0	7	30
8	FC Groningen	10	0	7	30
9	FC Utrecht	9	0	8	27
10	Heracles Almelo	8	0	9	24
11	Vitesse	7	0	10	21
12	Willem II	6	0	11	18
13	FC Emmen	6	0	11	18
14	NAC Breda	5	0	12	15
15	VVV-Venlo	4	0	13	12
16	De Graafschap	4	0	13	12
17	Fortuna Sittard	2	0	15	6
18	SBV Excelsior	1	0	16	3

Table 5.6: Prediction league table before winter transfer

Actual result					
rank	Team	homewin	draw	awaywin	point
1	PSV Eindhoven	16	0	1	48
2	Ajax	15	1	1	46
3	Feyenoord	11	3	3	36
4	FC Utrecht	8	4	5	28
5	Vitesse	7	5	5	26
6	Heracles Almelo	8	2	7	26
7	AZ Alkmaar	7	4	6	25
8	VVV-Venlo	6	5	6	23
9	Fortuna Sittard	5	5	7	20
10	Heerenveen	5	5	7	20
11	ADO Den Haag	5	5	7	20
12	Willem II	5	4	8	19
13	FC Emmen	4	5	8	17
14	SBV Excelsior	4	4	9	16
15	FC Groningen	4	3	10	15
16	PEC Zwolle	4	3	10	15
17	NAC Breda	4	3	10	15
18	De Graafschap	3	3	11	12

Table 5.7: Actual result gathered from worldfootball.net [22]

Chapter 6

Discussion and Conclusion

6.1 Conclusion

In this paper the ordered logit model is proposed to predict the outcome of the 2018-2019 Eredivisie season. It is based on the direct result prediction approach, and with limited amount of resources, the model didn't perform with the desired accuracy. With some in-depth analysis on the result, it is concluded that with the currently collected dataset and the proposed control variables, it is not possible statistically to predict a draw. Despite the undesirable result, the paper provides a more detailed explanation on the variable creation with the code attached in the appendix. The detailed reasoning for the categories of each control variables are also discussed in detail. It is worth noting that outside of the matches with draw as an outcome, the prediction is quite accurate.

In conclusion, it is not possible with the current data set to obtain an accurate prediction with the ordered logit model.

6.2 Discussion

The result obtained from the initial model is not satisfactory and with good reasons. First of all, the game importance factor is not supposed to be useful until the later half of the season due to the fact that in the first half, it is possible for every team to achieve their league goal mathematically. An alternative experiment would be training on the 2015-2016 and 2016-2017 season and predict on the 2017-2018 season so that we would obtain the result of the whole season. However, as mentioned in the literature review, the game has had a significant change in style of play which makes the game data prior to the 2016 season less valuable. Combining with the fact that disregarding the draws, the prediction accuracy is not too bad, it is decided to stick with the original experiment.

Secondly, the data for the dutch league is not as reliable as the other leagues for prediction purposes. As can be seen from historical data, the league is heavily dominated by the top 3-5 teams. Taking the 2016-2017 season as an example, Feyenoord finished with 82 points, Ajax ended the season with 81 points and PSV finished with 76 points while the rest of the league either barely obtained 50 points or even less. It can also see from the actual league table in the result section that the top three teams already distance themselves quite far from the rest of the league only halfway through the season. This creates a huge difference in strength ratings and narrowing down to a single match, it is not easy to explain a draw between a top 3 team against a weak team. To illustrate this situation

visually, the prediction of the second game of the season (Ajax vs Heracles) is shown in table 6.1. As can be seen, there is a 78 % chance that Ajax should win that game however it is easier for the weaker team to force a draw if they set their mind to it. It is also worth noting that the percentage of the probabilities in each category add up to present accurately the distribution of the matches, which means that the model is correctly predicting the outcome statistically. However, as demonstrated in the example in table 6.2 and the scatter plot in figure 6.1, it is also clear that the probability of obtaining a draw is never higher than both of the other two outcomes hence it is never predicted by the program.

	mean	2.5%	97.5%
A	0.075	0.051	0.104
D	0.147	0.111	0.185
H	0.778	0.716	0.833

Table 6.1: result prediction of the match between Ajax and Heracles in probabilities

	mean	2.5%	97.5%
A	0.339	0.297	0.382
D	0.305	0.261	0.348
H	0.356	0.301	0.399

Table 6.2: result prediction of the match between Heerenveen and Vitesse in probabilities

To further examine our assumption that the sub-par result happened because of the quality of the data, a simulation study is conducted. The result of the matches in the training set is generated in such a way that if the rating differences between teams are lower than 75 then it's a draw, if the rating of the home team is more than 75 points higher than the rating of the away team it is a home win, and the rest is away wins. By doing this, the simulation study consists of 205 away wins, 215 draws and 192 home wins . The prediction consists of 58 away wins, 40 draws and 55 home wins where the confusion matrix is shown below in table 6.3. As the confusion matrix provides an almost perfect prediction, it is concluded that the model functions well.

	Reference		
Prediction	A	D	H
A	58	0	0
D	6	26	8
H	0	0	55

Table 6.3: Confusion matrix for simulation study where the cut-points for the categories are preset

The cup performance factor didn't show a super significant effect as expected, and although the difference between red card received is determined to be influential for the result of the match, it alone is probably not a good enough indicator for the home advantage factor. The reason is that hardly any team receives red cards during the match which makes the variable less informative.

6.3 Future improvement

The set of control variables could use some improvement in the future. Although it is determined to be an effective way to represent home advantage using the card data, it is not a very successful set of parameters in this experiment. Therefore finding a more reliable variable for home advantage factor could be a big improvement for future research. The cup performance factor could be designed in a

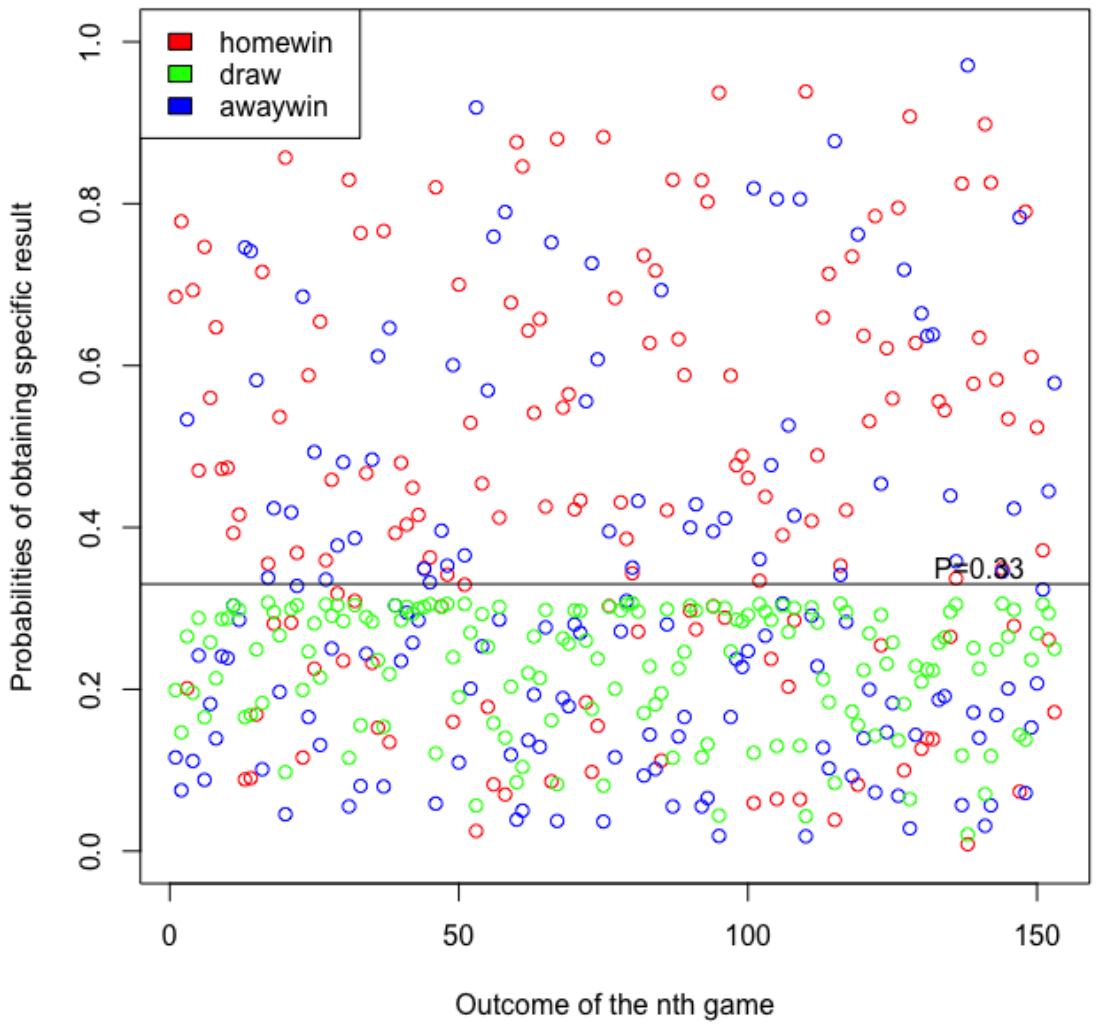

FIGURE 6.1: Scatter plot for probabilities of outcome for all games, where blue represent the probability of obtaining an away win, red represent the probability of obtaining a home win and green represent the probability of obtaining a draw. The black line draw a boundary whether the probability of an outcome is over 1/3

more detailed fashion. Specifying to the game level, the amount of games that the cup performance could influence could be preset to a certain number. The effect of each individual cup action (advance to round of 16/8/4/2 or getting relegated) can also be set differently.

There are also some assumptions made along the construction of the initial model for simplicity. For instance the result of the cup game is predicted by simply comparing the ELO rating of the two teams, and it is assumed that the lower league teams will always lose. As can be seen from the historical data, that is, in majority of the cases but not all. A future improvement would be to assign ratings for the lower league teams according to their performance in their own competition as well as the cup competition.

Another assumption being made is that there won't be a rating change during the summer and winter transfer window. This assumption is obviously not reliable however it appears to be extremely hard

to consider the impact of squad change to the team rating down to specific player level. A reliable way to consider the rating change is to purchase data from gambling websites where an in-depth analysis on how player effect team strength has been done.

Bibliography

[1] Scarf, P. A. & Shi, X. The importance of a match in a tournament. *Computers Operations Research*, 35:2406–2418, 2018.

[2] Hvattum, L. M. & Arntzen, H. Using ELO ratings for match result prediction in association football. *International Journal of Forecasting.*, 26:460–470, 2010.

[3] ELO, A. E. *The Rating of Chessplayers, Past and Present.* New York: Arco Pub, 1978.

[4] Maher, M. J. Modelling association football scores. In *Statist. Neerland.*, page 109–118, 1982.

[5] Dixon, M. J. & Coles, S. G. Modeling association football scores and inefficiencies in the football betting market. In *Appl. Statist.*, page 265–280, 1997.

[6] Sillanpää, V. & Heino, O. Forecasting football match results - a study on modeling principles and efficiency of fixed-odds betting markets in football. Master's thesis, Aalto University School of Business, 2013.

[7] Baio, G. & Blangiardo, M. Bayesian hierarchical model for the prediction of football results. *Journal of Applied Statistics*, 37(2):253–264, 2010.

[8] Suzuki, A. K. , Salasat, L. E. B., Leite, J. G. & Louzado-Neto, F. A Bayesian approach for predicting match outcomes: The 2006 (Association) Football World Cup. *Journal of the Operational Research Society*, 61:1530–1539, 2010.

[9] Karlis, D.& Ntzoufras, I. Bayesian modelling of football outcomes: using the Skellam's distribution for the goal difference. *IMA Journal of Management Mathematics*, 20(2):133–145, 2008.

[10] Goddard, J. Regression models for forecasting goals and match results in association football. *International Journal of Forecasting.*, 21:331–340, 2005.

[11] Goddard, J. & Asimakopoulos, I. forecasting football match results and the efficiency of fixed-odds betting. *International Journal of Forecasting.*, 23:51–66, 2004.

[12] Kuypers, T. Information and efficiency: an empirical study of a fixed odds betting market. *Applied Economics.*, 32:1353–1363, 2000.

[13] Koning, R. Balance in competition in Dutch soccer. *The Statistician.*, 49:419–431, 2000.

[14] Courneya, K. S. & Carron, A. V. Modelling association football scores and inefficiencies in the football betting market. *Journal of Sport and Exercise Psychology*, 14(1):13–27, 1992.

[15] Clarke, S. R.& Norman, J. M. Home ground advantage of individual clubs in English soccer. *Statistician*, 44(4):509–521, 1995.

[16] Pollard, R. & Pollard, G. Home Advantage in Soccer: A Review of its Existence and Causes. *International Journal of Soccer and Science*, 3(1):28–38, 2005.

[17] Nevill, A. M. , Newell, S. M. & Gale, S. Factors associated with home advantage in English and Scottish Soccer matches. *Journal of Sports Sciences*, 14(2):181–186, 1996.

[18] Nevill, A. M., Balmer, N. J. & Williams, A.M. The influence of crowd noise and experience upon refereeing decisions in football. *Psychology of Sport*, 3:261–272, 2002.

[19] Nevill, A. M., Balmer, N. J. & Williams, A.M. Home advantage in sport: An overview of studies on the advantage of playing at home. *Sports Medicine*, 28:221–236, 1999.

[20] Drawer, S. & Fuller, C. W. An economic framework for assessing the impact of injuries in professional football. *Safety Science*, 40:537–556, 2002.

[21] Footballdata.co.uk, 2019. URL `http://www.football-data.co.uk/netherlandsm.php`.

[22] worldfootball.net, 2019. URL `https://www.worldfootball.net/competition/ned-eredivisie/`.

[23] clubelo, 2019. URL `http://clubelo.com/`.

[24] clubelo, 2014. URL `https://gist.github.com/i000313/90b1f3c556b1b1ad8278`.

[25] Kitagawa.G Konishi.S. *Information Criteria and Statistical Modeling.* Springer, 2008.

Appendix A

Model training

In this appendix, the code for training the ordered logit model is shown. The code include the construction of ELO function, the initialization of the parameters necessary as well as the function for training. After that the AIC and BIC selection is applied and the final model is obtained in the end.

```
require(foreign)
require(ggplot2)
require(MASS)
require(reshape2)
require(glm.predict)
require(stringr)
require("nnet")
require('arm')
require("caret")
#defining ELO function
eloRating=function(playerARating, PlayerBRating, k=20) {

  # Expected score for player A and for player B.
  EA <- (1 / (1 + 10^((PlayerBRating - playerARating)/400)))
  EB <- (1 / (1 + 10^((playerARating - PlayerBRating)/400)))

  # RAn = RA + K * (SA - EA)
  newRatingPlyAWins  <- playerARating + k * (1 - EA)
  newRatingPlyADraws <- playerARating + k * (0.5 - EA)
  newRatingPlyADefeated  <- playerARating + k * (0 - EA)

  # RBn = RB + K * (SB - EB)
  newRatingPlyBWins  <- PlayerBRating + k * (1 - EB)
  newRatingPlyBDraws <- PlayerBRating + k * (0.5 - EB)
  newRatingPlyBDefeated  <- PlayerBRating + k * (0 - EB)

  chanceToWin <- round(data.frame(chanceToWin=c(EA, EB)) * 100, digits=0)
  playerAWins  <- round(data.frame(playerAWins=c(newRatingPlyAWins, newRatingPlyBDefeated)), digits=0)
  playerDraw  <- round(data.frame(draw=c(newRatingPlyADraws, newRatingPlyBDraws)), digits=0)
  playerBWins  <- round(data.frame(playerBWins=c(newRatingPlyADefeated, newRatingPlyBWins)), digits=0)

  df <- cbind(chanceToWin, playerAWins, playerDraw, playerBWins)
  rownames(df) <- c('playerA', 'playerB')
  return(df)
}

#inputing data season 2016-2017 with result
data1617 <- read.table(file = "~/Desktop/data1617.csv", header=TRUE, sep=",",stringsAsFactors = FALSE)
selecting <- c("HomeTeam", "AwayTeam", "FTR")
ELOtraining <- data1617[selecting]


#inputing data season 2017-2018 with result
data1718 <- read.table(file = "~/Desktop/data1718.csv", header=TRUE, sep=",",stringsAsFactors = FALSE)
ELOtraining[307:612,1] <- data1718["HomeTeam"]
```

```
ELOtraining[307:612,2] <- data1718["AwayTeam"]
ELOtraining[307:612,3] <- data1718["FTR"]

#card difference data 2016-2017 with result
card1617 <- read.table(file = "~/Desktop/cardfoul1617.csv", header=TRUE, sep=",")
card1617$Yellowdiff <- card1617$HY - card1617$AY
card1617$Reddiff <- card1617$HR - card1617$AR


#inputting the card stats into ELOtraining for second season
ELOtraining$Yellowdiff <- card1617$Yellowdiff
ELOtraining$Reddiff <- card1617$Reddiff
ELOtraining$Yellowdiff[307:612] <- data1718$HY - data1718$AY
ELOtraining$Reddiff[307:612] <- data1718$HR - data1718$AR

# initialization of ELO ratings
team_dict <- list(Feyenoord=1545, Ajax=1696, Roda=1341, Utrecht=1471, Heracles=1450,
                  Vitesse=1483, Heerenveen=1422,
                  Groningen=1425, Twente=1448, Nijmegen=1378, Excelsior=1314, Zwolle=1433)
team_dict["Den Haag"] <- 1429
team_dict["Willem II"] <- 1343
team_dict["Sparta Rotterdam"] <- 1300
team_dict["PSV Eindhoven"] <- 1737
team_dict["Go Ahead Eagles"] <- 1300
team_dict["AZ Alkmaar"] <- 1555


# initialization of team scores
team_score <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Twente=0, Nijmegen=0, Excelsior=0,Zwolle=0)
team_score["Den Haag"] <- 0
team_score["Willem II"] <- 0
team_score["Sparta Rotterdam"] <- 0
team_score["PSV Eindhoven"] <- 0
team_score["Go Ahead Eagles"] <- 0
team_score["AZ Alkmaar"] <- 0


# initialization of team rounds
team_round <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Twente=0, Nijmegen=0, Excelsior=0, Zwolle=0)
team_round["Den Haag"] <- 0
team_round["Willem II"] <- 0
team_round["Sparta Rotterdam"] <- 0
team_round["PSV Eindhoven"] <- 0
team_round["Go Ahead Eagles"] <- 0
team_round["AZ Alkmaar"] <- 0


#initialization of hometeam_potential
hometeam_potential <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Twente=0, Nijmegen=0, Excelsior=0, Zwolle=0)
hometeam_potential["Den Haag"] <- 0
hometeam_potential["Willem II"] <- 0
hometeam_potential["Sparta Rotterdam"] <- 0
hometeam_potential["PSV Eindhoven"] <- 0
hometeam_potential["Go Ahead Eagles"] <- 0
hometeam_potential["AZ Alkmaar"] <- 0

#initialization of awayteam_potential
awayteam_potential <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                           Vitesse=0, Heerenveen=0,
                           Groningen=0, Twente=0, Nijmegen=0, Excelsior=0, Zwolle=0)
awayteam_potential["Den Haag"] <- 0
awayteam_potential["Willem II"] <- 0
awayteam_potential["Sparta Rotterdam"] <- 0
awayteam_potential["PSV Eindhoven"] <- 0
awayteam_potential["Go Ahead Eagles"] <- 0
awayteam_potential["AZ Alkmaar"] <- 0
```

```
# initialization of team cup participant
team_cup <- list(Feyenoord=1, Ajax=1, Roda=1, Utrecht=1, Heracles=1,
                 Vitesse=1, Heerenveen=1,
                 Groningen=1, Twente=1, Nijmegen=1, Excelsior=1, Zwolle=1)
team_cup["Den Haag"] <- 1
team_cup["Willem II"] <- 1
team_cup["Sparta Rotterdam"] <- 1
team_cup["PSV Eindhoven"] <- 1
team_cup["Go Ahead Eagles"] <- 1
team_cup["AZ Alkmaar"] <- 1

for (i in 1:306) {
  level <- ELOtraining[i, 3]
  hometeam = paste(ELOtraining[i, 1])
  awayteam = paste(ELOtraining[i, 2])
  home_rating = team_dict[[hometeam]]
  away_rating = team_dict[[awayteam]]
  home_score = team_score[[hometeam]]
  away_score = team_score[[awayteam]]
  home_round = team_round[[hometeam]]
  away_round = team_round[[awayteam]]
  home_cup = team_cup[[hometeam]]
  away_cup = team_cup[[awayteam]]
  hometeam_point = home_score+ (34 - home_round)* 3
  awayteam_point = away_score+ (34 - away_round)* 3


  # avoid relegation is worth playing for team not yet avoiding relegation
  # top 8 is worth playing for, for team not yet guarented top 8 but guarented not being relegated
  # champion is worth playing for, for team guarented top 8

  for (name in names(hometeam_potential)){
    if (name == hometeam) {
      hometeam_potential[[name]] = 0
    }else {
      team_point = team_score[[name]]+ 34 - team_round[[name]]

      #home_importance
      if (team_point < hometeam_point){
          hometeam_potential[[name]] = 1
      }else {
          hometeam_potential[[name]] = 0
        }
    }
  }

  for (name in names(awayteam_potential)){
      if (name == awayteam) {
        awayteam_potential[[name]] = 0
      }else{
        team_point = team_score[[name]]+ 34 - team_round[[name]]

      #away_importance
        if (team_point < awayteam_point){
          awayteam_potential[[name]] = 1
        } else {
          awayteam_potential[[name]] = 0
        }


      }
  }
  homecount = 0
  for(name in names(hometeam_potential)){
    homecount = homecount + hometeam_potential[[name]]
  }
  awaycount = 0
  for(name in names(awayteam_potential)){
    awaycount = awaycount + awayteam_potential[[name]]
  }

  home_importance = 0
  away_importance = 0
  if (homecount>9){
```

```
  home_importance = 1
}else if (homecount<7 & homecount>3){
  home_importance = 1
}else{
  home_importance = 0
}

if (awaycount>9){
  away_importance = 1
}else if (awaycount<7 & awaycount>3){
  away_importance = 1
}else{
  away_importance = 0
}

# cup participant
if (i>52){
  team_cup["Roda"] <- 0
  team_cup["Willem II"] <- 0
  team_cup["Nijmegen"] <- 0
  team_cup["Twente"] <- 0
}
if (i>90){
  team_cup["PSV Eindhoven"] <- 0
  team_cup["Go Ahead Eagles"] <- 0
  team_cup["Groningen"] <- 0
  team_cup["Excelsior"] <- 0
}
if(i>144){
  team_cup["Den Haag"] <- 0
  team_cup["Zwolle"] <- 0
  team_cup["Ajax"] <- 0
  team_cup["Heracles"] <- 0
}
if(i>171){
  team_cup["Utrecht"] <- 0
  team_cup["Heerenveen"] <- 0
  team_cup["Feyenoord"] <- 0
}
if(i>216){
  team_cup["Sparta Rotterdam"] <- 0
}
if(i>288){
  team_cup["AZ Alkmaar"] <- 0
}



ELOtraining$home_rating[i] <- home_rating
ELOtraining$away_rating[i] <- away_rating
ELOtraining$ratingdiff[i] <- home_rating - away_rating
#ELOtraining$home_score[i] <- home_score
#ELOtraining$away_score[i] <- away_score
#ELOtraining$home_round[i] <- home_round
#ELOtraining$away_round[i] <- away_round
ELOtraining$home_importance[i] <- home_importance
ELOtraining$away_importance[i] <- away_importance
ELOtraining$home_cup[i] <- home_cup
ELOtraining$away_cup[i] <- away_cup

newRatings = eloRating(team_dict[[hometeam]], team_dict[[awayteam]])
team_round[[hometeam]] <- team_round[[hometeam]] + 1
team_round[[awayteam]] <- team_round[[awayteam]] + 1
if (level == "H") {
  team_dict[[hometeam]] <- newRatings[1, 2]
  team_dict[[awayteam]] <- newRatings[2, 2]
  team_score[[hometeam]] <- team_score[[hometeam]] + 3
  team_score[[awayteam]] <-  team_score[[awayteam]]
} else if (level == "A") {
  team_dict[[hometeam]] <- newRatings[1, 4]
  team_dict[[awayteam]] <- newRatings[2, 4]
  team_score[[hometeam]] <- team_score[[hometeam]]
  team_score[[awayteam]] <-  team_score[[awayteam]] + 3
```

```
  } else if (level == "D") {
    team_dict[[hometeam]] <- newRatings[1, 3]
    team_dict[[awayteam]] <- newRatings[2, 3]
    team_score[[hometeam]] <- team_score[[hometeam]] + 1
    team_score[[awayteam]] <-  team_score[[awayteam]] + 1
  }
}

# initialization of ELO ratings on August 10th 2017 for new teams promoted
team_dict["VVV Venlo"] <- team_dict["Nijmegen"]
team_dict["NAC Breda"] <- team_dict["Go Ahead Eagles"]

# initialization of team scores
team_score <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Twente=0, Excelsior=0,Zwolle=0)
team_score["Den Haag"] <- 0
team_score["Willem II"] <- 0
team_score["Sparta Rotterdam"] <- 0
team_score["PSV Eindhoven"] <- 0
team_score["AZ Alkmaar"] <- 0

#added on August 10th 2017
team_score["VVV Venlo"] <- 0
team_score["NAC Breda"] <- 0


# initialization of team rounds
team_round <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Twente=0, Excelsior=0, Zwolle=0)
team_round["Den Haag"] <- 0
team_round["Willem II"] <- 0
team_round["Sparta Rotterdam"] <- 0
team_round["PSV Eindhoven"] <- 0
team_round["AZ Alkmaar"] <- 0

#added on August 10th 2017
team_round["VVV Venlo"] <- 0
team_round["NAC Breda"] <- 0



#initialization of hometeam_potential
hometeam_potential <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                           Vitesse=0, Heerenveen=0,
                           Groningen=0, Twente=0, Excelsior=0, Zwolle=0)
hometeam_potential["Den Haag"] <- 0
hometeam_potential["Willem II"] <- 0
hometeam_potential["Sparta Rotterdam"] <- 0
hometeam_potential["PSV Eindhoven"] <- 0
hometeam_potential["AZ Alkmaar"] <- 0

#added on August 10th 2017
hometeam_potential["VVV Venlo"] <- 0
hometeam_potential["NAC Breda"] <- 0

#initialization of awayteam_potential
awayteam_potential <- list(Feyenoord=0, Ajax=0, Roda=0, Utrecht=0, Heracles=0,
                           Vitesse=0, Heerenveen=0,
                           Groningen=0, Twente=0, Excelsior=0, Zwolle=0)
awayteam_potential["Den Haag"] <- 0
awayteam_potential["Willem II"] <- 0
awayteam_potential["Sparta Rotterdam"] <- 0
awayteam_potential["PSV Eindhoven"] <- 0
awayteam_potential["AZ Alkmaar"] <- 0

#added on August 10th 2017
awayteam_potential["VVV Venlo"] <- 0
awayteam_potential["NAC Breda"] <- 0

# initialization of team cup participant
team_cup <- list(Feyenoord=1, Ajax=1, Roda=1, Utrecht=1, Heracles=1,
                 Vitesse=1, Heerenveen=1,
```

```
                Groningen=1, Twente=1, Excelsior=1, Zwolle=1)
team_cup["Den Haag"] <- 1
team_cup["Willem II"] <- 1
team_cup["Sparta Rotterdam"] <- 1
team_cup["PSV Eindhoven"] <- 1
team_cup["AZ Alkmaar"] <- 1

#added on August 10th 2017
team_cup["VVV Venlo"] <- 1
team_cup["NAC Breda"] <- 1

for (i in 307:612) {
  level <- ELOtraining[i, 3]
  hometeam = paste(ELOtraining[i, 1])
  awayteam = paste(ELOtraining[i, 2])
  home_rating = team_dict[[hometeam]]
  away_rating = team_dict[[awayteam]]
  home_score = team_score[[hometeam]]
  away_score = team_score[[awayteam]]
  home_round = team_round[[hometeam]]
  away_round = team_round[[awayteam]]
  home_cup = team_cup[[hometeam]]
  away_cup = team_cup[[awayteam]]
  hometeam_point = home_score+ (34 - home_round)* 3
  awayteam_point = away_score+ (34 - away_round)* 3


  # avoid relegation is worth playing for team not yet avoiding relegation
  # top 8 is worth playing for, for team not yet guarented top 8 but guarented not being relegated
  # champion is worth playing for, for team guarented top 8

  for (name in names(hometeam_potential)){
    if (name == hometeam) {
      hometeam_potential[[name]] = 0
    }else {
      team_point = team_score[[name]]+ 34 - team_round[[name]]

      #home_importance
      if (team_point < hometeam_point){
        hometeam_potential[[name]] = 1
      }else {
        hometeam_potential[[name]] = 0
      }
    }
  }

  for (name in names(awayteam_potential)){
    if (name == awayteam) {
      awayteam_potential[[name]] = 0
    }else{
      team_point = team_score[[name]]+ 34 - team_round[[name]]

      #away_importance
      if (team_point < awayteam_point){
        awayteam_potential[[name]] = 1
      } else {
        awayteam_potential[[name]] = 0
      }


    }
  }
  homecount = 0
  for(name in names(hometeam_potential)){
    homecount = homecount + hometeam_potential[[name]]
  }
  awaycount = 0
  for(name in names(awayteam_potential)){
    awaycount = awaycount + awayteam_potential[[name]]
  }

  home_importance = 0
  away_importance = 0
  if (homecount>9){
```

```
  home_importance = 1
}else if (homecount<7 & homecount>3){
  home_importance = 1
}else{
  home_importance = 0
}

if (awaycount>9){
  away_importance = 1
}else if (awaycount<7 & awaycount>3){
  away_importance = 1
}else{
  away_importance = 0
}

# cup participant
if (i>351){
  team_cup["Vitesse"] <- 0
  team_cup["Sparta Rotterdam"] <- 0
  team_cup["NAC Breda"] <- 0
  team_cup["Excelsior"] <- 0
  team_cup["Den Haag"] <- 0
}
if (i>387){
  team_cup["Utrecht"] <- 0
  team_cup["Heerenveen"] <- 0
  team_cup["Groningen"] <- 0
}
if(i>457){
  team_cup["VVV Venlo"] <- 0
  team_cup["Ajax"] <- 0
  team_cup["Heracles"] <- 0
}
if(i>486){
  team_cup["Zwolle"] <- 0
  team_cup["PSV Eindhoven"] <- 0
  team_cup["Roda"] <- 0
}
if(i>531){
  team_cup["Twente"] <- 0
  team_cup["Willem II"] <- 1
}
if(i>594){
  team_cup["AZ Alkmaar"] <- 0
}



ELOtraining$home_rating[i] <- home_rating
ELOtraining$away_rating[i] <- away_rating
ELOtraining$ratingdiff[i] <- home_rating - away_rating
#ELOtraining$home_score[i] <- home_score
#ELOtraining$away_score[i] <- away_score
#ELOtraining$home_round[i] <- home_round
#ELOtraining$away_round[i] <- away_round
ELOtraining$home_importance[i] <- home_importance
ELOtraining$away_importance[i] <- away_importance
ELOtraining$home_cup[i] <- home_cup
ELOtraining$away_cup[i] <- away_cup

newRatings = eloRating(team_dict[[hometeam]], team_dict[[awayteam]])
team_round[[hometeam]] <- team_round[[hometeam]] + 1
team_round[[awayteam]] <- team_round[[awayteam]] + 1
if (level == "H") {
  team_dict[[hometeam]] <- newRatings[1, 2]
  team_dict[[awayteam]] <- newRatings[2, 2]
  team_score[[hometeam]] <- team_score[[hometeam]] + 3
  team_score[[awayteam]] <-  team_score[[awayteam]]
} else if (level == "A") {
  team_dict[[hometeam]] <- newRatings[1, 4]
  team_dict[[awayteam]] <- newRatings[2, 4]
  team_score[[hometeam]] <- team_score[[hometeam]]
  team_score[[awayteam]] <-  team_score[[awayteam]] + 3
```

```
  } else if (level == "D") {
    team_dict[[hometeam]] <- newRatings[1, 3]
    team_dict[[awayteam]] <- newRatings[2, 3]
    team_score[[hometeam]] <- team_score[[hometeam]] + 1
    team_score[[awayteam]] <-  team_score[[awayteam]] + 1
  }
}



FTRordered <- factor(ELOtraining$FTR, levels = c("A", "D", "H"), ordered = TRUE)
ratingdiffordered <- factor(ELOtraining$ratingdiff, ordered = TRUE)
home_importanceordered <- factor(ELOtraining$home_importance, levels = c(0,1), ordered = TRUE)
away_importanceordered <- factor(ELOtraining$away_importance, levels = c(0,1), ordered = TRUE)
home_cupordered <- factor(ELOtraining$home_cup, levels = c(0,1), ordered =TRUE)
away_cupordered <- factor(ELOtraining$away_cup, levels = c(0,1), ordered =TRUE)
Yellowdiffordered <- factor(ELOtraining$Yellowdiff, ordered = TRUE)
Reddiffordered <- factor(ELOtraining$Reddiff , ordered = TRUE)



#ordered probit regression
newmodel <- polr(FTRordered ~ ratingdiff, data = ELOtraining, Hess = TRUE)
model <- polr(FTRordered ~ ratingdiff + home_importanceordered + away_importanceordered + home_cupordered + away_cup
summary(model)
(ctable <- coef(summary(model)))
p <- pnorm(abs(ctable[, "t value"]), lower.tail = FALSE) * 2
(ctable <- cbind(ctable, "p value" = p))

modelwithsignificance <- polr(FTRordered ~ ratingdiff + Reddiff, data = ELOtraining, Hess = TRUE)
myModel.step<-stepAIC(modelwithsignificance,trace=1)
myModel.step$anova

require("xtable")
xtable(ctable, digit = c(0,9,10,7,2))

model1 <- polr(FTRordered ~ ratingdiff, data = ELOtraining, Hess = TRUE)
model2 <- polr(FTRordered ~ Reddiff, data = ELOtraining, Hess = TRUE)

BIC(modelwithsignificance)
BIC(model1)
BIC(model2)
```

Appendix B

Model testing

In this section the code for prediction is presented. First the variables are initialized and then the final model obtained from the previous section is applied to predict the football outcome. After that some descriptive statistics and plots are created to visualize the result.

```
# initialization of ELO ratings on August 9th 2018 for new teams promoted
team_dict["For Sittard"] <- team_dict["Roda"]
team_dict["Graafschap"] <- team_dict["Sparta Rotterdam"]
team_dict["FC Emmen"] <- team_dict["Twente"]

# initialization of team scores
team_score <- list(Feyenoord=0, Ajax=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Excelsior=0,Zwolle=0)
team_score["Den Haag"] <- 0
team_score["Willem II"] <- 0
team_score["PSV Eindhoven"] <- 0
team_score["AZ Alkmaar"] <- 0
team_score["VVV Venlo"] <- 0
team_score["NAC Breda"] <- 0

#added on August 9th 2018
team_score["For Sittard"] <- 0
team_score["Graafschap"] <- 0
team_score["FC Emmen"] <- 0

# initialization of team rounds
team_round <- list(Feyenoord=0, Ajax=0, Utrecht=0, Heracles=0,
                   Vitesse=0, Heerenveen=0,
                   Groningen=0, Excelsior=0, Zwolle=0)
team_round["Den Haag"] <- 0
team_round["Willem II"] <- 0
team_round["PSV Eindhoven"] <- 0
team_round["AZ Alkmaar"] <- 0
team_round["VVV Venlo"] <- 0
team_round["NAC Breda"] <- 0

#added on August 9th 2018
team_round["For Sittard"] <- 0
team_round["Graafschap"] <- 0
team_round["FC Emmen"] <- 0


#initialization of hometeam_potential
hometeam_potential <- list(Feyenoord=0, Ajax=0, Utrecht=0, Heracles=0,
                           Vitesse=0, Heerenveen=0,
                           Groningen=0, Excelsior=0, Zwolle=0)
hometeam_potential["Den Haag"] <- 0
hometeam_potential["Willem II"] <- 0
```

```
hometeam_potential["PSV Eindhoven"] <- 0
hometeam_potential["AZ Alkmaar"] <- 0
hometeam_potential["VVV Venlo"] <- 0
hometeam_potential["NAC Breda"] <- 0

#added on August 9th 2018
hometeam_potential["For Sittard"] <- 0
hometeam_potential["Graafschap"] <- 0
hometeam_potential["FC Emmen"] <- 0


#initialization of awayteam_potential
awayteam_potential <- list(Feyenoord=0, Ajax=0, Utrecht=0, Heracles=0,
                           Vitesse=0, Heerenveen=0,
                           Groningen=0, Excelsior=0, Zwolle=0)
awayteam_potential["Den Haag"] <- 0
awayteam_potential["Willem II"] <- 0
awayteam_potential["PSV Eindhoven"] <- 0
awayteam_potential["AZ Alkmaar"] <- 0
awayteam_potential["VVV Venlo"] <- 0
awayteam_potential["NAC Breda"] <- 0

#added on August 9th 2018
awayteam_potential["For Sittard"] <- 0
awayteam_potential["Graafschap"] <- 0
awayteam_potential["FC Emmen"] <- 0


# initialization of team cup participant
team_cup <- list(Feyenoord=1, Ajax=1, Utrecht=1, Heracles=1,
                 Vitesse=1, Heerenveen=1,
                 Groningen=1, Excelsior=1, Zwolle=1)
team_cup["Den Haag"] <- 1
team_cup["Willem II"] <- 1
team_cup["PSV Eindhoven"] <- 1
team_cup["AZ Alkmaar"] <- 1
team_cup["VVV Venlo"] <- 1
team_cup["NAC Breda"] <- 1

#added on August 9th 2018
team_cup["For Sittard"] <- 1
team_cup["Graafschap"] <- 1
team_cup["FC Emmen"] <- 1

data1819 <- read.table(file = "~/Desktop/data1819.csv", header=TRUE, sep=",",stringsAsFactors = FALSE)
Testing <- data1819[c("HomeTeam", "AwayTeam", "FTR")]
Testing$Yellowdiff <- data1819$HY - data1819$AY
Testing$Reddiff <- data1819$HR - data1819$AR

count_home = 0
count_away = 0
count_draw = 0

for (i in 1: 153) {
  hometeam = paste(Testing[i, 1])
  awayteam = paste(Testing[i, 2])
  home_rating = team_dict[[hometeam]]
  away_rating = team_dict[[awayteam]]
  home_score = team_score[[hometeam]]
  away_score = team_score[[awayteam]]
  home_round = team_round[[hometeam]]
  away_round = team_round[[awayteam]]
  home_cup = team_cup[[hometeam]]
  away_cup = team_cup[[awayteam]]
  hometeam_point = home_score+ (34 - home_round)* 3
  awayteam_point = away_score+ (34 - away_round)* 3


  # avoid relegation is worth playing for team not yet avoiding relegation
  # top 8 is worth playing for, for team not yet guarented top 8 but guarented not being relegated
  # champion is worth playing for, for team guarented top 8

  for (name in names(hometeam_potential)){
    if (name == hometeam) {
```

```
    hometeam_potential[[name]] = 0
  }else {
    team_point = team_score[[name]]+ 34 - team_round[[name]]

    #home_importance
    if (team_point < hometeam_point){
      hometeam_potential[[name]] = 1
    }else {
      hometeam_potential[[name]] = 0
    }
  }
}

for (name in names(awayteam_potential)){
  if (name == awayteam) {
    awayteam_potential[[name]] = 0
  }else{
    team_point = team_score[[name]]+ 34 - team_round[[name]]

    #away_importance
    if (team_point < awayteam_point){
      awayteam_potential[[name]] = 1
    } else {
      awayteam_potential[[name]] = 0
    }


  }
}
homecount = 0
for(name in names(hometeam_potential)){
  homecount = homecount + hometeam_potential[[name]]
}
awaycount = 0
for(name in names(awayteam_potential)){
  awaycount = awaycount + awayteam_potential[[name]]
}

home_importance = 0
away_importance = 0
if (homecount>9){
  home_importance = 1
}else if (homecount<7 & homecount>3){
  home_importance = 1
}else{
  home_importance = 0
}

if (awaycount>9){
  away_importance = 1
}else if (awaycount<7 & awaycount>3){
  away_importance = 1
}else{
  away_importance = 0
}


# cup participant
if (i>54){
  team_cup["NAC Breda"] <- 0
  team_cup["Excelsior"] <- 0
  team_cup["Groningen"] <- 0

}
if (i>90){
  team_cup["PSV Eindhoven"] <- 0
  team_cup["Graafschap"] <- 0
  team_cup["Heracles"] <- 0
  team_cup["Den Haag"] <- 0
  team_cup["FC Emmen"] <- 0
}
if(i>144){

  team_cup["Zwolle"] <- 0
```

```
  team_cup["Utrecht"] <- 0

}
if(i>162){
  team_cup["Vitesse"] <- 0
  team_cup["Heerenveen"] <- 0
  team_cup["For Sittard"] <- 0
  team_cup["Heerenveen"] <- 0
}
if(i>207){
  team_cup["AZ Alkmaar"] <- 0
  team_cup["Willem II"] <- 0

}
if(i>297){
  team_cup["Feyenoord"] <- 0
}



Testing$home_rating[i] <- home_rating
Testing$away_rating[i] <- away_rating
Testing$ratingdiff[i] <- home_rating - away_rating
#Testing$home_score[i] <- home_score
#Testing$away_score[i] <- away_score
#Testing$home_round[i] <- home_round
#Testing$away_round[i] <- away_round
Testing$home_importance[i] <- home_importance
Testing$away_importance[i] <- away_importance
Testing$home_cup[i] <- home_cup
Testing$away_cup[i] <- away_cup

#model6 <- polr(FTRordered ~ ratingdiff + away_importanceordered + home_cupordered +
                 #Reddiff, data = ELOtraining, Hess = TRUE)


#predict<-basepredict(model,c(Testing$ratingdiff[i], Testing$home_importance[i] ,Testing$away_importance[i] , Test
predict<-basepredict(modelwithsignificance,c(Testing$ratingdiff[i],Testing$Reddiff[i]))
#predict <-basepredict(model6,c(Testing$ratingdiff[i],Testing$away_importance[i],Testing$home_cup[i],Testing$Reddi
count_home = count_home + predict[3,1]
count_away = count_away + predict[1,1]
count_draw = count_draw + predict[2,1]

Testing$home[i] = predict[3,1]
Testing$draw[i] = predict[2,1]
Testing$away[i] = predict[1,1]
if (max(predict[,1]) == predict[1,1]){
  Testing$prediction[i] <- "A"
}
else if (max(predict[,1]) == predict[2,1]){
  Testing$prediction[i] <- "D"
}
else if(max(predict[,1]) == predict[3,1]){
  Testing$prediction[i] <- "H"
}

level <- Testing$prediction[i]
#trueR <- Testing$FTR
#if (trueR[i] =="D"){
#  print(predict)
#}
newRatings = eloRating(team_dict[[hometeam]], team_dict[[awayteam]])
team_round[[hometeam]] <- team_round[[hometeam]] + 1
team_round[[awayteam]] <- team_round[[awayteam]] + 1
if (level == "H") {
  team_dict[[hometeam]] <- newRatings[1, 2]
  team_dict[[awayteam]] <- newRatings[2, 2]
  team_score[[hometeam]] <- team_score[[hometeam]] + 3
  team_score[[awayteam]] <-  team_score[[awayteam]]
} else if (level == "A") {
  team_dict[[hometeam]] <- newRatings[1, 4]
  team_dict[[awayteam]] <- newRatings[2, 4]
  team_score[[hometeam]] <- team_score[[hometeam]]
  team_score[[awayteam]] <-  team_score[[awayteam]] + 3
```

```
  } else if (level == "D") {
    team_dict[[hometeam]] <- newRatings[1, 3]
    team_dict[[awayteam]] <- newRatings[2, 3]
    team_score[[hometeam]] <- team_score[[hometeam]] + 1
    team_score[[awayteam]] <-  team_score[[awayteam]] + 1
  }


}


count <- 0
for (i in 1: length(Testing[,3])) {
  if (Testing$FTR[i] == Testing$prediction[i]){
    count = count + 1
  }
}
summary(as.factor(Testing$FTR))
summary(as.factor(Testing$prediction[1:153]))
confusionMatrix(factor(Testing$prediction[1:153]), factor(Testing$FTR[1:153]))


plot(Testing$home[1:153],col="red",ylim=range(0:1),xlab='Outcome of the nth game', ylab='Probabilities of obtaining
par(new=TRUE)
plot(Testing$away[1:153],col="blue",ylim=range(0:1),xlab='', ylab='', axes=F)
par(new=TRUE)
plot(Testing$draw[1:153],col="green",ylim=range(0:1),xlab='', ylab='', axes=F)
legend("topleft", c("homewin","draw","awaywin"), fill = c("red","green","blue"))
abline(h=0.33, col="black")
text(140, 0.35, "P=0.33", col = "black")
```