Stockholms universitet

Predicting the Final Result of a Football Match

Alexander Crompton

Kandidatuppsats i matematisk statistik
Bachelor Thesis in Mathematical Statistics

Kandidatuppsats 2021:15
Matematisk statistik
Juni 2021

www.math.su.se

Matematisk statistik
Matematiska institutionen
Stockholms universitet
106 91 Stockholm

Mathematical Statistics
Stockholm University
Bachelor Thesis **2021:15**
http://www.math.su.se

Predicting the Final Result of a Football Match

Alexander Crompton[*]

June 2021

Abstract

In this paper we study how different variables effect the possible outcomes of a football match played in the British Premier League. The goal is to find the best trained model to forecast the data correctly as frequently as possible. We use machine learning techniques and logistic regression for predictions. The study is on three different outcome variables, with the first one being a binary outcome variable for a home team win and then two multinomial variables with three and five possible outcomes. The data contains information on all matches played between the 2014/15 season to the 2018/19 season. We find that the logistic regression and the SVM classifiers have the highest predictability and perform the most consistently for all three response models. The classification trees provide decent results predicting but do not perform quite at the same rate. The results regarding the importance of the match variables is that the expected goals is the most important for result predictions while the shots on goal, red cards, clearance and ranking variables also show importance in the predictions and improve the predictions significantly.

[*] Postal address: Mathematical Statistics, Stockholm University, SE-106 91, Sweden. E-mail: alexcrompton@hotmail.se. Supervisor: Dmitry Otryakhin.

Acknowledgement

This is a bachelor's thesis in Mathematical Statistics from the Department of Mathematics at Stockholm University. I would like to thank my supervisor Dmitry Otryakhin for his guidance and weekly feedback. Without his help, the work would not be the same.

Contents

1 Introduction

In this study we analyse football data using statistical methods. We build models on categorical data and use logistic regression models and machine learning techniques. Main focuses are laid on predictions on the the data and finding what impacts the outcome of a match. The work is structured starting with an introduction to football and the work, followed by a section with information on how the data was obtained and which variables we worked with. Section 3 walks through the theory used to execute the statistical analysis. From section 4 to 6 we go through the results and diagnostics for each model.

1.1 Rules & Objective

Football (also known as soccer in some parts of the world) is the world's most popular ball sport. It is estimated to be played by 250 million people world wide and is followed by more than 1.3 billion people in the world [1]. In this study we look specifically at the the Premier League which is the top tier division of Football in England (and Wales) and was founded in 1992. The League contains 20 teams who play each other twice a season. When each team has played their 38 games, the three teams that faired the worst are relegated to the English Football Leagues first division (The Championship) and replaced by 3 Championship teams. Since the the league originated in 1992 there have been 49 members and seven title winners, with Manchester United being the most successful team winning it 13 times. The League is the most viewed sports league in the world.
It is a sport with a simple objective, to get the ball into the opposition team's net more times than the opposition does in the supporting team. This should be done while using any body part except hands or arms. The simplicity of the sport is the key reason to its huge popularity around the world.

In professional football, teams of eleven face each other. One of those eleven being a goalkeeper who's objective is to save the ball and is allowed to handle the ball in the penalty area around his goal. Some of the common actions in football is to shoot, pass and dribble. A shot being to strike the ball towards the oppositions goal, pass meaning to manoeuvring the ball to a player in the same team and dribbling being to move past an opposition player by using pace or skill. A football match is played for two halves of 45 minutes and can finish with many combinations of goals for the two teams. However games played in League competition are mainly measured with 3 outcomes, with either a home team win, away team win or a draw which is when both teams finish the game with the same amount goals scored.

1.2 Purpose & Thesis

The purpose of this study is to find the model which can best predict the outcome of a match in English Association Football. We also study how the different variables effect the outcome, how significant they are and how they correlate to one another. We also compare logistic regression models with different algorithms to determine classification data such as decision trees and Support Vector Machine (SVM).

2 Data

The data studied contains 1900 rows where each row represents one match played. The data is complete with results and stats in the columns from all Premier League games between the seasons 14/15 to 18/19.

```
Rows: 1,900
Columns: 70
$ date <dttm> 2014-08-16 15:00:00, 2014-08-23 12:45:00, 2014-08-31 13:30:00
$ away_team  <chr> "AVL", "NEW", "HUL", "AVL", "ARS", "AVL", "MCI", "AVL", "AVL",
$ home_team  <chr> "STK", "AVL", "AVL", "LIV", "AVL", "CHE", "AVL", "EVE", "QPR",
$ home_goals <dbl> 0, 0, 2, 0, 0, 3, 0, 3, 2, 1, 0, 1, 1, 0, 2, 1, 1, 1, 0, 0, 1,
$ away_goals <dbl> 1, 0, 1, 1, 3, 0, 2, 0, 0, 2, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0,
$ hxG <dbl> 0.423368, 0.507525, 0.639316, 0.728097, 0.649013, 3.142180,
$ axG<dbl> 0.909774, 0.699295, 0.288880, 0.701676, 1.362240, 0.228896,
$ h_deep<dbl> 3, 4, 6, 5, 0, 6, 1, 7, 3, 3, 15, 3, 7, 6, 7, 3, 3, 6, 7, 7,
$ net_xG <dbl> -0.486406, -0.191770, 0.350436, 0.026421, -0.713227,
$ net_red <dbl> 0, -1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, -1, -1, 1, 0, 1, 0,
```

Listing 1: Data Sample

In Listing 1 a sample of the data frame is shown.

2.1 Collected

Data was collected from two kaggle datasets. The first dataset [16] contains data scraped from the Premier League website and is formatted with each row being one match. The columns contain facts about the match such as how many shots, passes, tackles and how much possession each teams that played had. Note that one row of the dataset was missing. This match was easily found by counting the total number of matches for each team, each season and finding the teams that didn't have the 19 rows of home or away matches. This match was then manually added to the dataset with data from the premier league website.

The second dataset [17] is a dataset originally scraped from understat.com. The data is structured differently with each row only representing one team's stats in a match. Therefore there are twice as many rows as the first dataset. This dataset contains more specific data for the matches such as the metrics ”expected goals” and ”deep”.

2.1.1 Restructuring of Data

The programming language R [8] is used to study and handle the data. When analysing the data we want to use the format of the first dataset and therefore need to restructure the second dataset.

```
advanced_stats$h_a<-if_else(advanced_stats$h_a=="h","home_team","away_team") #
    Home or Away variable
a_s<-advanced_stats%>%
  filter(league=="EPL" &year!="2019")%>% #Removing All other leagues and years
  mutate(hxG=if_else(h_a=="home_team",xG,xGA), # Defining variables if team
      played home/away
         axG=if_else(h_a=="away_team",xG,xGA),
         h_deep=if_else(h_a=="home_team",deep,deep_allowed),
         a_deep=if_else(h_a=="away_team",deep,deep_allowed),
         h_ppda=if_else(h_a=="home_team",ppda_coef,oppda_coef),
         a_ppda=if_else(h_a=="away_team",ppda_coef,oppda_coef),
         h_ppda_att=if_else(h_a=="home_team",ppda_att,oppda_att),
         a_ppda_att=if_else(h_a=="away_team",ppda_att,oppda_att),
         h_ppda_def=if_else(h_a=="home_team",ppda_def,oppda_def),
         a_ppda_def=if_else(h_a=="away_team",ppda_def,oppda_def),
         home_goals=if_else(h_a=="home_team",scored,missed),
         away_goals=if_else(h_a=="away_team",scored,missed)
         )%>%
  group_by(hxG,date)%>% # To find the 2 matches for the same game
  mutate(id=cur_group_id())%>% # Crating id for the match
  select(id,h_a,team,year,date,home_goals,away_goals,hxG,axG,h_deep,a_deep,h_ppda
      ,a_ppda,h_ppda_att,a_ppda_att,h_ppda_def,a_ppda_def)
a_s<-a_s%>% # New data frame with 1900 rows from 3800
  spread(h_a,team) # home and away team split to two columns, share the same row


epl_14_19<-a_s%>%left_join(epl_14_19,by=c("home_team","away_team","year"))
```

Listing 2: Restructuring Dataset

From Listing 2 we can see this is done by by grouping the data by date and the home team's expected goal stat, as there were multiple games per day we used another variable to specify which match. The data is later joined by the function ”left_join” from tidyverse [9] using the factors home team, away team and the season the match was played. This works as every team only plays the other teams once home and away every season. Teams in the first dataset were written as the three letter abbreviation of the team name.

```
epl_14_19$year<-gsub("\\/.*","",epl_14_19$season,fixed = FALSE) # variable with
     season starting year
epl_14_19$year<-as.numeric(as.character(epl_14_19$year)) # from str to int
advanced_stats$team<-str_replace_all(advanced_stats$team, c(
  "Arsenal" = "ARS","Aston Villa" = "AVL",
  "Bournemouth"="BOU", "Brighton"="BHA",
  "Burnley"="BUR","Cardiff"="CAR", "Chelsea"="CHE",
  "Crystal Palace"="CRY","Everton"="EVE",
  "Fulham"="FUL","Huddersfield"="HUD",
  "Hull"="HUL","Leicester"="LEI",
  "Liverpool"="LIV","Manchester City"="MCI",
  "Middlesbrough"="MID","Manchester United"="MUN",
  "Newcastle United"="NEW","Norwich"="NOR",
  "Queens Park Rangers"="QPR","Southampton"="SOU",
  "Stoke"="STK","Sunderland"="SUN","Swansea"="SWA",
  "Tottenham"="TOT","Watford"="WAT",
  "West Bromwich Albion"="WBA", "West Ham"="WHU",
  "Wolverhampton Wanderers"="WOL" )) # Renaming to abbreviated names
```

Listing 3: String Replacement

The teams names were shortened in the first dataset so to join the datasets we had to replace all string instances of the team names with the abbreviations using the R function "str_replace_all" from tidyverse [9] as seen in Listing 3.

```
epl_14_19<-epl_14_19%>%
  mutate(net_shots_on_target= home_shots_on_target-away_shots_on_target,
         net_possession= home_pos-away_pos,
         net_shots_off= (home_shots-home_shots_on_target)-(away_shots-away_shots_
              on_target),
         net_touch= home_touch-away_touch,
         net_pass= home_pass-away_pass,
         net_tackles= home_tackles-away_tackles,
         net_clear= home_clear-away_clear,
         net_corner= home_corner-away_corner,
         net_offside= home_off-away_off,
         net_yellow= home_yellow-away_yellow,
         net_red= home_red-away_red,
         net_fouls= home_fouls-away_fouls,
         net_xG=hxG-axG,
         net_deep=h_deep-a_deep,
         net_ppda=h_ppda-a_ppda,
         net_allowed_opposition_half=h_ppda_att-a_ppda_att,
         net_defensive_actions=h_ppda_def-a_ppda_def,
         net_goals=home_goals-away_goals)
```

Listing 4: Creating Net Variables

In Listing 4 we create variables with the net values of all the stats. This is simply done by taking the home teams value for each statistic and subtracting each corresponding statistic for the away team. For the shots off target variable we remove the number of shots that were on target from the teams total number of shots to get the corresponding variable of interest.

2.2 Description of Data

Table 1 contains descriptions for the explanatory variables:

Explanatory Variables	
Variable Name	Variable explanation
Shots on Target	Number of shots that hit within the frame of the goal
Shots off Target	Number of shots that missed the target were blocked on the way to goal or hit the frame off the goal
Possession	The percentage of time the team were in control of the ball
Passes	Number of successful passes
Touches	Number of times players in the team touched the ball
Tackles	Number of tackles
Clear	Number of times kicking the ball away from danger
Corner	Number of corners
Offside	Number of times the team was caught in an offside position
Yellow	Number of yellow cards
Red	Number of red cards
Fouls	Number of committed fouls
xG	Expected goals (xG) is the sum of the likelihood for each taken shot, being scored. The likelihood is based on position and angle from where the shot was taken
Deep	Number of passes completed within 20 yards of goal
Allowed Passes in the Opposition Half	Number of passes allowed in the opposition half
Defensive Actions	Number of defensive actions in the oppositions half. Defensive actions being tackles , interceptions, fouls and duels
ppda	A metric to measure the pressure from the defending team. Number of passes in the opposition half per defensive action
Home rank & Away rank	Variable describing previous years position in the table for the home and away team #1:#17 Placed 1 to 17 in order in the premier league #18 Promoted from the championship

Table 1: The variables used to determine outcome of our model

These variables can convincingly be used to explain the outcome of football match as they are actions from the match and may explain why the match played out the way it did. Shots on target is a variable which especially on initial thought should have impact on the match. ”You can’t score if you don’t shoot”- is a famous quote of the Footballing legend Johan Cruyff. Red

cards is a variable which should describe the outcome as playing with one man less is a huge disadvantage. Many passes, touches, high possession and deep passes are all stats that could indicate an advantage with ball control and could potentially be interesting to determine the outcome.
In Table 2 is the description for each response variable used in the different model building methods:

Response Variables	
Home Win	Binary variable with 1 if the home team wins and 0 if they lose or draw
Five Outcome	Multi level variable describing these outcomes: home team win by 2 or more, 1 goal , a draw, away win by 1 and away win by 2 or more
Multi Win	Multinomial variable with the three possible outcomes (home,draw,away)

Table 2: The variables we build our models on

In Table 3 we get the informative variables that have been used to build and work with the dataset,

Informative Variables	
Home Team	The team playing at their home stadium
Away Team	The team playing away from home
Date	The date and time the match was played
Season	The season the game was played

Table 3: Variables helping us build our dataset

2.3 Test & Trainingset

Our interest is to find the best possible model fit for the response variable. To find the most fitting model we want to avoid overfitting our model. This problem can easily occur when the model is too complex and starts to describe the given data too particularly in random errors which can lead to problems when predicting future data observation points. To lessen the risk of this problem happening and making an analysis of future data we split the data to a training and test set.

```
 train_ind <- createDataPartition(epl_14_19$win, p = .75,
                                  list = FALSE,
                                  times = 1) #Splitting the data for the 3
                                      outcome
train <- epl_14_19[train_ind, ]
test <- epl_14_19[-train_ind, ]
```

Listing 5: Creating Net Variables

The split is made by 0.75 of the data observations put into the training dataset, which is then used to build the model and then rest to the test split which is then used to test our model on. The split is mixed and divided proportionally to the number of outcomes as seen in Listing 5 by using the caret package [10].

2.4 Cross-Validation

From Bishop's book [2] we can use S-fold also commonly known as k-fold Cross-Validation to train the data. Validation is used to avoid overfitting by testing on a set of the data. K-fold cross-validation divides the training set to k-groups. The rest of the groups are then used to train the model and is tested on the remaining groups. This is done until all possible k-groups have been left out and then the average best possible model is used.

```
ctrl<-trainControl(method = "cv",number = 7) # 7 fold cross validation
model_cv<-train(multi_win ~ net_shots_on_target+net_shots_off+net_clear+net_red+
    net_fouls+net_xG+net_ppda+net_allowed_opposition_half+home_rank+away_rank,
    data = train, method="multinom", # 3 outcome multinomial model
              direction="backward",trControl=ctrl)
```

Listing 6: Creating Net Variables

To perform Cross-Validation on our data in R we use the caret package to split data. To implement the models we use the instructions from The caret Package [10]. In Listing 6 we can see how the cross validation is implemented on the multinomial 3 outcome variable model using the train function. After brief testing of 10 and 5 folds it seemed appropriate to split the data somewhere in between with 7 folds to give enough folds, yet an appropriate amount of data to validate the models on.

3 Theory

In this section we will walk through the theory of the various methods that were use to make the statistical study. Firstly we go through the general theory of logistic regression and then follow with the theory behind model selection and the predictability of the models. We also go through the theory for the machine learning methods and go through theory for collinearity.

3.1 Logistic Regression

The first model we build is a binary outcome model. To do this we perform a Logistic regression. Logistic regression is a special case of the generalised linear model and describes the relationship between the explanatory variable Intercept and a binary response variable. As the response variable is binary variable we let $\pi(x) = P(Y = 1|X = x) = 1 - P(Y = 0|X = x)$ with the logistic regression model,

$$\pi(x) = \frac{exp(\alpha + \beta x)}{1 + exp(\alpha + \beta x)}. \tag{1}$$

To find the linear relationship between the response and explanatory variables we use the log odds which is also known as the logit this gives us both a response and explanatory variable that can obtain all real values,

$$logit[\pi(x)] = log\frac{\pi(x)}{1 - \pi(x)} = \alpha + \beta x. \tag{2}$$

This now describes the change for each x value for the logit function of the response variable.

3.1.1 Multiple Logistic Regression

In our work we use multiple variables to build our model on, therefore we need to perform a multiple logistic regression. According to Agresti [3] multiple logistic regression works just like the extension for ordinary regression with multiple explanatory variables where the the model for $\pi(x)$ is defined as:

$$logit[\pi(x)] = \alpha + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_i x_i \tag{3}$$

for i explanatory variables.

3.1.2 Multinomial Logistic Regression

Football is a sport where there are multiple possible outcomes of the match. To study the multiple outcome response variables we will perform a multinomial logistic regression.

There are multiple outcomes and we get a $\pi_j(x) = P(Y = j|x)$ with j being the number of outcomes. In our 3 outcome model we define it as $j = 1, 2, 3$. From chapter 7 in Agresti [3] we treat the categories of the response variable as multinomial $\{\pi_1(x), ..., \pi_i(x)\}$. We then choose a reference variable to compare the other outcomes with, when we do this we logically get $i = 1, ..., I - 1$ logit models, where I is the total number of outcomes.

$$log\frac{\pi_i(x)}{\pi_I(x)} = \alpha_i + \beta_i' x. \tag{4}$$

Where the reference variable can be found from the comparisons by:

$$log\frac{\pi_a(x)}{\pi_b(x)} = log\frac{\pi_a(x)}{\pi_I(x)} - log\frac{\pi_b(x)}{\pi_I(x)}. \tag{5}$$

We use the package nnet [11] to execute the multinomial logistic regression in R.

3.2 Decision Trees & Machine Learning Algorithms

To compare the performance of the logistical regressions and looking for the best possible fitting model we use some different model building methods for the classification.
C5.0 is a classification tree method while CART and Random Forrest are methods that can be used for both regression data and classification data. Classification trees make binary decisions to determine the class of the observation. To execute these models in R, the packages C50[12], rpart[13] and randomForrest[14] is used.

3.2.1 C5.0

C5.0 builds its trees using information entropy as the splitting criteria. For missing data it estimates the missing values from the other values. C5.0 is derived from the C4.5 tree. C5.0 however is faster and uses smaller tress and is therefore more commonly used. According to the article Gini Impurity and Entropy in Decision Tree - ML [4] the Entropy is defined as:

$$E(S) = \sum_{i=1}^{n} -p_i \log_2 p_i. \tag{6}$$

Both entropy and Gini index essentially measure the quality of the split of the data made in the tree.

3.2.2 CART

CART trees uses GINI index as splitting criteria when building the model. The algorithm uses cost to remove redundant branches from the decision trees. Differing from the C5.0 tree CART can only give us a binary outcome. This means It can only be used properly for the binary outcome variable model. From the article Gini Impurity and Entropy in Decision Tree - ML [4] the Gini impurity is defined as:

$$GI = 1 - \sum_{i=1}^{n} (p)^2. \tag{7}$$

3.2.3 Random Forrest

From Le's article on Decision Trees in R [5] Random Forrest is an aggregate or a collection of trees. The method does a good job in treating missing and outlier values. However a problem can be that it commonly overfits noisy datasets and read in to insignificant variables. However this can hopefully be reduced as much as possible using Cross-Validation on the data. Random Forrest uses Information Gain to calculate the quality of a split. Information Gain is calculated by subtracting the weighted entropies from every branch from the first entropy. In the end the split with the maximum information gain is chosen.

3.2.4 Support Vector Machines

SVM are types of supervised learning models. It uses learning algorithms to analyse classification data. From Bishop [2] SVM makes decisions using boundaries and does not give posterior probabilities. Support vector machine decision boundary is chosen by maximising the distance or the margin between the samples. The boundaries can be set by a linear separator, essentially splitting the classification by the best line hyperplane, it can also be by a non-linear boundary or by kernel functions. Commonly used kernel functions are polynomial kernels and radial kernels.

3.3 Odds Ratio

In chapter 2.2.3 in Agresti [3] we have π which is the probability of success, the odds are then described by:

$$\Omega = \frac{\pi}{1-\pi}. \tag{8}$$

Instinctively we then get that this describes the likelihood of success compared to failure. The odds ratio (OR) is the ratio of two separate odds and is used to compare different odds with each other. To compare a logistic regression model where the x value has increased by one we get:

$$OR = \frac{\Omega(x+1)}{\Omega(x)} = \frac{\frac{\pi(x+1)}{1-\pi(x+1)}}{\frac{\pi(x)}{1-\pi(x)}} = e^{\beta}$$

This means that when x increases, it increases with the factor e^{β}. For example if $\beta = 0$ this means that the odds are multiplied by 1 which would not effect the probability, just as what would be expected from a β-value of 0.

3.4 Model Selection

From Categorical Data Analysis [3] the goal is to find a model that is simple to understand but also fits the data well.

3.4.1 AIC

The Akaike information criterion judges the models by how close the predicted values are to the real ones. The formula is described by Alan Agresti in Categorical Data Analysis [3] as:

$$AIC = -2(\text{log likelihood}) + 2p \tag{9}$$

where p is the number of parameters in the model. It is a practical measure to compare models between each other. This will be used in the variable selection process for the logistic regressions.

3.4.2 Stepwise Variable Selection

Step wise variable selection helps us pick which variables to include in our model. Instead of trying all possible combinations or trying different random combinations of variables we methodically remove or add variables to our model. In "Lineara statistiska modeller"[6] Rolf Sundberg writes that using different methods of selection may lead to varying models, and we will therefore look into both methods given in the compendium.

3.4.3 Forward selection

In Forward Selection we start with an intercept only model and add one explanatory variable at a time. There are multiple ways to pick the next variable, however the general purpose is to pick the variable that is most significant or adds the most to the model. This should be repeated until there is no more possible improvement. One example of how to select variables is the built in step function in R which uses the AIC 9 it then executes the selection based on which variable lowers the AIC the most, until it can not be further reduced.

3.4.4 Backward Elimination

In similar fashion to the Forward Selection, Backward Selection improves the model step by step. With this method we start with all explanatory variables in the model and remove them one by one until we achieve the best possible fit for the model.

3.5 Testing Predictability

Diagnosing the models is a main part of the work as we want to see how well we can predict football results. In this chapter we go over the theory of the methods used to assess the predictability of our selected models.

3.5.1 Classification Table

A classification table is used to compare the predicted value with the real observed value. The classification value is a good way to evaluate the predictive accuracy of the logistic regression. The method takes the number of correct guesses divided by the total number of number of observations. The predicted value for the logistic regression model is decided depending on a threshold value. The predicted value will be set to ($\hat{x} = 1$) when the predicted value for our observation (π_i) is over the set threshold value (π_t) and set to $\hat{x} = 0$ when $\pi_t > \pi_i$. The threshold value is commonly set at the value $\pi_t = 0.5$, but can be set to different values if there is a more appropriate or better fitting value.

		Prediction value	
		0	1
Observed Value	0	TN	FP
	1	FN	TP

From the table above we have classification table for a logistic regression with a binary variable that can take on either 0 or 1. The diagonal elements True Negative (TN) and True Positive (TP) are the correctly classified observations while False Negative (FN) and False Positive (FP) are the incorrectly predicted observations. Our classification value is calculated as below,

$$\text{Classification} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{10}$$

The same principle works for the multinomial outcome variable with the diagonal elements as the correctly predicted observations and the rest of the table elements as inaccurate predictions.

3.5.2 ROC & AUC

Receiver operating characteristic (ROC) is described in Agreti's book [3] as a plot of sensitivity, the True Positive Rate, as a function of (1- specificity), the False Positive Rate, for all the possible cutoffs. The ROC curve shows the predictability for all possible thresholds. As the classification is highly dependant on the set threshold the ROC is highly useful for us when comparing the diagnostics between the models.

$$\text{sensitivity} = TPR = \frac{TP}{TP + FN}. \tag{11}$$

$$\text{1-specificity} = FPR = \frac{FP}{FP + TP}. \tag{12}$$

The Area Under the Curve (AUC) of the ROC is used as a predictive value for the model. The value is set between 0 and 1. If all predictions are correct we get a value of $AUC = 1$ while a model with $AUC = 0.5$ indicates that we have the same proportion of correct as incorrect predictions which implicates the model is no better than random guesses. To get an idea of how to consider the AUC value Hosmer & Lemeshow [7] roughly declare a value bellow 0.7 as poor while a value of 0.8 or higher can be considered excellent.

For a multi-classification problem ROC curves are plotted with one class versus the rest. Each class will then have it's own AUC and ROC-line but the average of all the AUC values then give us the model AUC which we can use to judge the predictability of the models.

3.5.3 Kappa

When classifying our models, especially the 5 outcome variable, the amount of each outcome differ. If there is a big imbalance in the response outcome the classification rate can be a hollow measure of predictive power. For example in our 5 outcome the outcome of an away team winning by 2 or more goals only occurs 25 times in our test set. The model could then simply never predict that outcome but still get a high rate of classification. However with the Cohen's Kappa measure this is punished. Kappa is a measure of agreement with a single value. From Alan Agreti [3] kappa compares the probability of agreement which is defined as: $\sum_a \pi_{aa}$ to the expected value if the ratings were independent $\sum_a \pi_{a+}\pi_{a+}$ as

$$k = \frac{\sum_a \pi_{aa} - \sum_a \pi_{a+}\pi_{a+}}{1 - \sum_a \pi_{a+}\pi_{a+}}. \tag{13}$$

The value 1 meaning the perfect agreement has occurred. Negative kappa values can also occur this when the agreement is less than the expected under independence. Relying solely on kappa can be problematic as a single index value reduces our information about the agreement and disagreement structure, but can in our case be handy to compare our models.

3.6 Collinearity & VIF

To avoid faulty estimates on variables it is important to have a good understanding of the data. Many simple correlation problems can be avoided with simple logic and knowledge about what the variables are measuring. For example in the given data we have a variable "touches" and one for "possession", logically these variables will have high correlation as the team with the most touches of the ball will also most likely have had the ball in their possession for a higher percentage of time. From a correlation plot in R we can get a nice view of pairwise correlation between variables. However there is also a chance that a variable can have correlation explained by multiple variables in the data. This is called Multicollinearity and may be harder to detect. Therefore we use the Variance inflation factor (VIF) to avoid this problem.
According to "Lineara statistiska modeller"[6] VIF can be defined as:

$$Var(\hat{\beta}_j) = \sigma^2(S_{jj}^{-1}) = \frac{\sigma^2}{s_{jj}^2} VIF \tag{14}$$

$$VIF = \frac{1}{1 - R_j^2}. \tag{15}$$

where R_j^2 is the coefficient of determination for the variation in x_j described by the other x variables. The lowest value for the VIF is 1 which indicates there is no multicollinearity, by rule of thumb a value of 10 or more is seen as a cause for concern.

4 Binary Outcome Model

The first model we are going to look at is the home model which has a binary response variable describing if the home team won or not. The true outcome is if they won and false is if they drew or lost. The results presented in this section can be analysed as what impacts the outcome of a home victory and how well the models predict if the home team won.

4.1 Multicollinearity Check

To execute the logistic regression our explanatory variables must fulfil the condition that there is no strong correlation or multicollinearity between the variables. A correlation matrix is plotted which can be found as Figure 1.

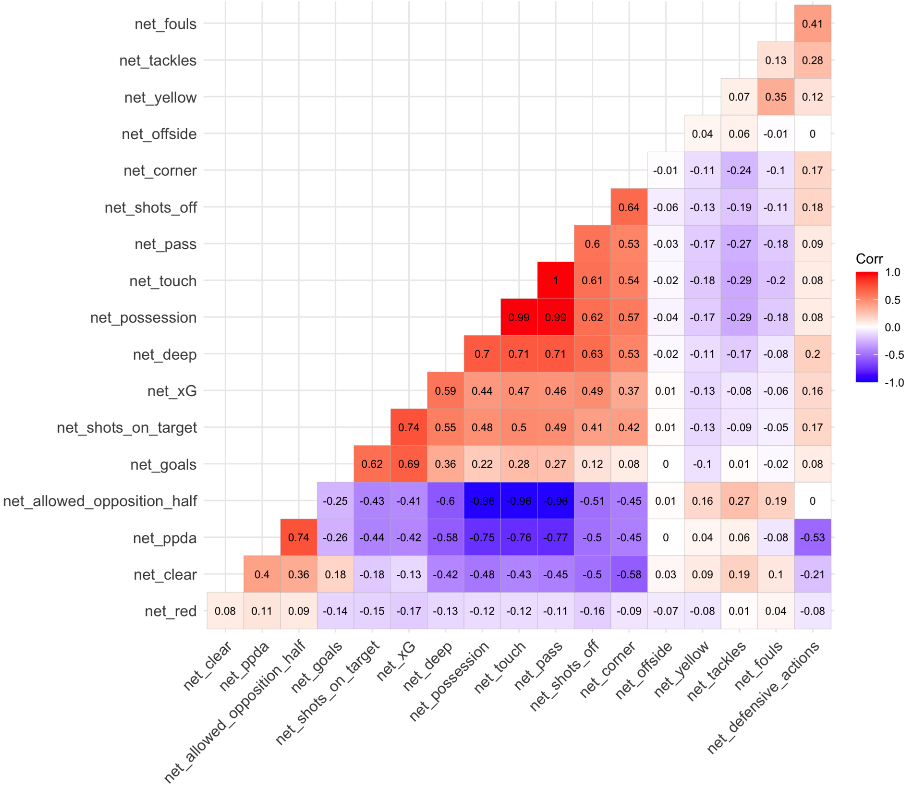


Figure 1: Correlation Matrix

From the matrix we find strong correlation between the net touches and the net number of passes and also between them two and possession. As possession measures the amount of time in control, touches the total number of touches of the ball and passes the number of completed passes between teammates the strong correlation between these three make sense as they are all different measures of control of the ball. These variables also have strong correlation to net allowed passes in the oppositions half.

When using the different methods of variable selection if the correlating variables are then included in the final model new elections will be made without the correlating variables until the best model is found.

4.2 Model Selection

To pick the best possible models for the logistic regression we execute forward and backward selection, firstly the forward selection model:

	Estimate	Std..Error	z.value	P-value	VIF
net_xG	1.15	0.11	10.56	0.00	1.91
net_clear	0.07	0.01	11.17	0.00	1.65
net_shots_on_target	0.19	0.03	6.26	0.00	1.56
net_shots_off	-0.07	0.02	-4.68	0.00	1.95
net_red	-0.85	0.23	-3.75	0.00	1.05
home_rank	-0.06	0.01	-3.97	0.00	1.16
away_rank	0.07	0.01	4.62	0.00	1.14
net_defensive_actions	0.02	0.01	2.03	0.04	1.39
net_fouls	-0.03	0.02	-1.78	0.08	1.32

Table 4: Forward selection model

The backward selection model was firstly selected, but did however include both the pass and possession stats which lead to high VIF values. To avoid the multicollinearity both variables were tested alone, but neither were later included in the final model which turned out the same for both tests:

	Estimate	Std..Error	z.value	P-value	VIF
net_shots_on_target	0.19	0.03	6.27	0.00	1.59
net_shots_off	-0.07	0.02	-4.21	0.00	2.20
net_clear	0.07	0.01	10.98	0.00	1.70
net_red	-0.86	0.23	-3.76	0.00	1.05
net_fouls	-0.04	0.02	-2.18	0.03	1.21
net_xG	1.11	0.11	10.03	0.00	1.95
net_deep	0.02	0.02	1.47	0.14	2.34
net_ppda	-0.03	0.01	-2.34	0.02	2.89
home_rank	-0.07	0.02	-4.37	0.00	1.39
away_rank	0.08	0.02	4.97	0.00	1.37

Table 5: Backward selection model

As seen in Table 4 and 5 we can see the main differences is that the model selected through forward selection contains 9 variables while the backward selection model contains 10. Both model contain the shot on goal, off goal, clear, red, xG, fouls and the rank variables. Nearly all variables are considered significant on a five percent significance level, however the deep variable in the backward selection model has a P-value of 0.14 and the fouls variable has a P-value of 0.08 in the forward model. However when these were removed the performance of both models declined and they are kept in the models. Both stepwise functions lead to all variables having a lower VIF value than 5 which implies we have no multicollinearity.

4.3 Variable Effect

To analyse the effect of how each variable effects the model we use the odds ratio. If the odds value for the variable is close to 1 it has little effect. The more the odds differ from 1 the more the variable effects the likelihood of the specific outcome. For our net valued variables we can see the odds as the increased likelihood of the home team winning for every additional unit of net variable against the away team:

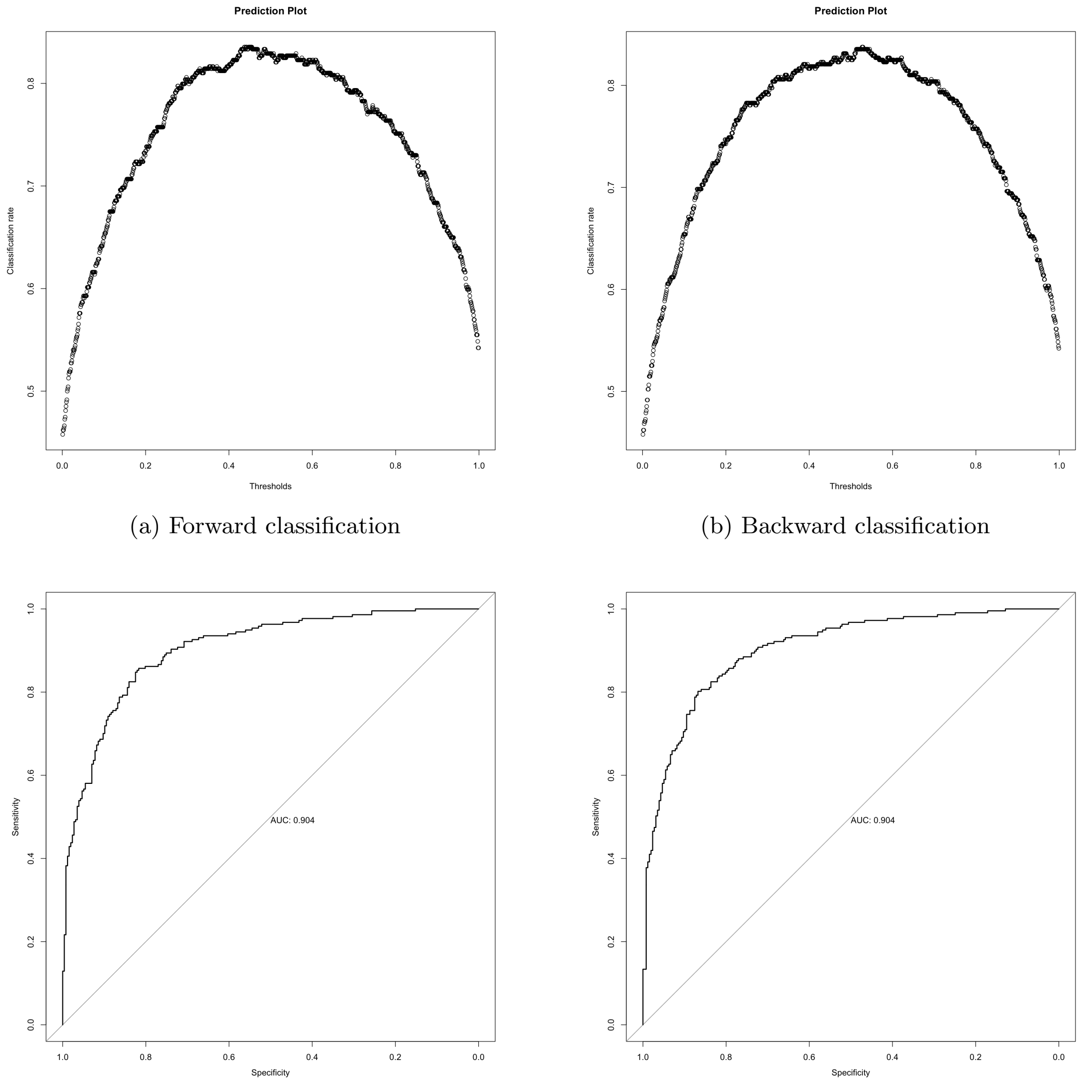
	Forward	Backward
net_xG	3.16	3.04
net_clear	1.07	1.07
net_shots_on_target	1.21	1.21
net_shots_off	0.93	0.93
net_red	0.43	0.42
home_rank	0.94	0.93
away_rank	1.07	1.08
net_defensive_actions	1.02	
net_fouls	0.97	0.96
net_deep		1.02
net_ppda		0.97

Table 6: Odds for home win

In the Table 6 we can see how the odds are formed for the forward and backward model side by side. Reasonably both models have similar odds for the variables and the variables that are only included in either of the models have values close to one which we can be interpreted as them having little to low effect on the predictions for each sample. The variables with the biggest positive effects on the home binary response variable, is the expected goals, shots on target and the clearances. This means that the more units the home team has of these compared to the away team increases the chance of a win. There are also negatively effecting variables, the two main ones being shots off target and red cards. This means that if the away team gets more shots off target or red cards this increases the chance of a home win. The two special case variables are the home and away rank. The ranks are set from 1 to 18 with the teams being set as 1 is the previous years league winner with descending values for each rank. In the odds a higher rank (lower in the table) on the home team decreases the chance of a home team win with a similar but reverse effect for the away team.

4.4 Comparing Predictability

In Figure 2 the ROC plots for the forward and backward selection are presented we also have the classification values for thresholds for the two logistic regression models.



(a) Forward classification

(b) Backward classification

(c) Forward ROC

(d) Backward ROC

Figure 2: Classification and ROC

Both models give us good predictability and high classification values, however the backward model performs slightly better and is chosen to compare with the machine learning algorithms.

Predictors	Backward	Forward
Classification	0.827	0.829
B.P Classifiaction	0.838	0.835
AUC	0.904	0.904
Kappa	0.652	0.656

Comparing the models using our predictors both perform very well and can have high predictive power on the binary outcome. We can predict if the home team wins or not 83.8% for the test set when we use the Best Possible Classification (B.P in the table) which means we select the threshold with the highest classification rate while it is correctly predicted 0.829 of the test set when the threshold was set to the standard 0.5. The AUC is outstanding and we can conclude that the linear model gives us nice predictions for the home win outcome. As the data with 1900 rows is done with a 75/25 split we get 475 matches to predict on 217 are home wins with 258 being draws. This means the data is slightly tilted, however with relatively high kappa values the models do a good job of separating the classes and not predicting too many instances of zeros (not home win). In the table bellow we let the backward model represent the best logistic model.

	Model						
Predictors	Logistic	RF	C5.0	CART	SVM L	SVM NL	SVM P
Classification	0.827	0.806	0.776	0.732	0.833	0.825	0.819
B.P Classif.	0.838	0.821	0.795	0.732	0.833	0.825	0.819
AUC	0.904	0.893	0.878	0.759			
Kappa	0.656	0.608	0.550	0.457	0.665	0.647	0.634

Interpreting from the table the best classification is made by the linear SVM while the non-linear and polynomial SVM also provide correct predictions over 0.8 of the time. The CART and C5.0 classification have the lowest predictive power while random forrest does a better job predicting. Note that the SVM does not provide likelihoods for each predictions which means no AUC value can be used to compare the methods predictability using that method. However when comparing AUC and classification the linear models both out perform the tree methods, the SVM methods also performs better when comparing the classification rate than the classification tree. The linear SVM shows the best agreement level with the highest kappa while the CART and C5.0 predicts too many instances of 0 which can also be seen in Figure 4 for the CART and C5.0 prediction histogram.

In the Figure 5 in the appendix histograms for each models predictions of the likelihood of a sample being a home win. A model that provides more predictions close to zero and one shows the model is more decisive and is more certain in its predictions.

5 Three Outcome Model

The multinomial outcome logistic regression describes the result of the match with the 3 possible outcomes. The outcomes chosen are home win, draw and away win. This might be the most interesting outcome model as it describes the 3 outcomes most commonly used to describe the outcome of the match (which team won). To execute this for the multinomial logistic regression we have to set a reference level for one of the outcomes. I decided on the draw outcome for the reference as this is the ”middle ground” and makes sense to compare with.

5.1 Model Selection

	Coef Away	Coef Home	P Away	P Home
net_xG	-0.92	0.86	0.00	0.00
net_clear	-0.04	0.05	0.00	0.00
net_shots_on_target	-0.15	0.15	0.00	0.00
net_shots_off	0.05	-0.06	0.01	0.00
home_rank	0.05	-0.04	0.00	0.01
away_rank	-0.05	0.05	0.00	0.00
net_red	0.69	-0.63	0.01	0.01
net_defensive_actions	0.00	0.02	0.97	0.05
net_fouls	0.01	-0.03	0.63	0.13

Table 7: Forward selection model for 3 outcome model

	Coef Away	Coef Home	P Away	P Home
net_shots_on_target	-0.14	0.16	0.00	0.00
net_shots_off	0.05	-0.05	0.01	0.00
net_clear	-0.04	0.05	0.00	0.00
net_red	0.68	-0.66	0.01	0.01
net_fouls	0.01	-0.03	0.65	0.06
net_xG	-0.92	0.85	0.00	0.00
net_ppda	-0.00	-0.03	0.99	0.02
home_rank	0.05	-0.06	0.01	0.00
away_rank	-0.05	0.06	0.00	0.00

Table 8: Backward selection model for 3 outcome model

With the multiclass outcome we now have as many variables in the forward selection as the backward selection. The same variables are picked for the forward selected model as for the binary response variable, however the backward selection removed net_deep from the equation. It is important to note that the P-value and estimates in the table above are against the reference variable meaning that there are 2 values for each. This leads to some variables being significant for the home but not the away outcome and vice versa. We avoid multicollinearity with low VIF values for both models.

5.2 Variable Effect

	Backward		Forward	
	Away	Home	Away	Home
net_shots_on_target	0.87	1.17	0.86	1.16
net_shots_off	1.05	0.95	1.05	0.94
net_clear	0.96	1.05	0.96	1.06
net_red	1.98	0.52	2.00	0.53
net_fouls	1.01	0.97	1.01	0.97
net_xG	0.40	2.34	0.40	2.37
net_ppda	1.00	0.97		
net_defensive_actions			1.00	1.02
home_rank	1.05	0.94	1.05	0.96
away_rank	0.95	1.07	0.95	1.05

Table 9: Odds estimates for home and away

In the Multinomial outcome we again get similar odds values for the 2 different logistic models. As expected the draw outcome seems to take the middle ground for nearly all variables which means that the closer the difference of the values for the home and away team is to zero the model will more likely predict a draw as the most likely outcome. Expected goals describe a high positive value for home win and a negative value gives the away team a more likely outcome. The same goes for the net shots on, the net clearances and the away rank. While more net red cards, net shots off goal, net fouls and a higher home rank lead to the outcome away win being more likely and home win as less likely

5.3 Comparing Predictability

For the multi class outcomes we use the function multiclass.roc from the pROC package [15] in R to determine the AUC for the model. The function takes the average AUC from all the pairwise ROC plots for the different outcomes. The classifier for the logistic and decision tree chooses the most probable outcome chosen by the model.

Predictors	Backward	Forward
Classification	0.722	0.722
AUC	0.843	0.844
Kappa	0.552	0.551

The forward model just edges the backward model in predictability for the match outcome classifier with a higher AUC, if ever so slightly. However both get their predictions right 72.2 percent of the time and provide an excellent average AUC of values rounded to 0.84.

	Model					
Predictors	Logistic	RF	C5.0	SVM L	SVM NL	SVM P
Classification	0.722	0.673	0.681	0.726	0.690	0.709
AUC	0.844	0.812	0.796			
Kappa	0.551	0.465	0.4837	0.558	0.500	0.525

For the classification the C5.0 is the worst performing, while linear SVM performs the best with the logistic regression being a narrow second. One interesting observation is that the Random Forrest performs better than the C5.0 tree according to the AUC but actually gets a lower classification rate. It also has a worse kappa value and seems to predict draws way too seldom only predicting it 11 times when there are in fact 113 instances of the draw occurring. When studying the classification tables the hardest outcome to correctly predict is for the draw outcome with the best model predicting draws being the forward selected multinomial logistic regression doing it 31 times.

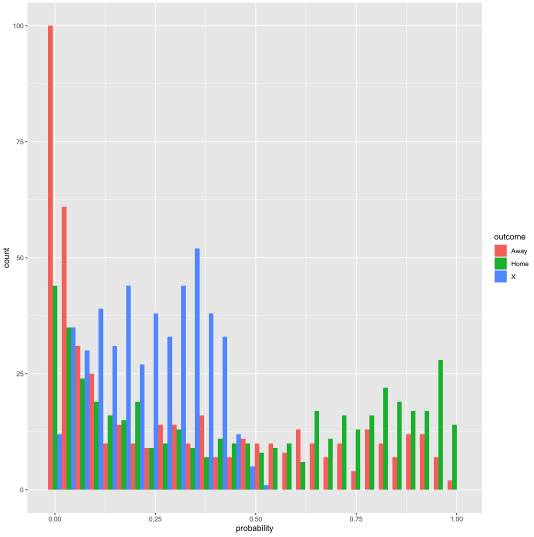


Figure 3: Probabilities for Outcomes in Forward model

In Figure 3 we can see the quantities of the predictions of each class for different probabilities. The desired outcome is for many of the predictions to be close to 1 and 0 which would suggest a model is more certain in its predictions. The figure shows that the majority of draw predictions mostly are in the range between 0 to 0.45 while the predictions for home and away wins are more spread out and show a slightly more desired result. The likely reason for this is that a draw is the outcome between the home and away outcomes which results in it often being the second most likely outcome for the sample. In the appendix we can see the rest of the predictions for the other models in Figure 6.

6 Five Outcome Model

With the 5 outcome model we build models for a multinomial response variable with 5 outcomes. The first outcome is a home win by 2 or more goals margin, secondly a home win by a 1 goal margin, then a draw outcome, fourthly an away win by 1 goal and lastly an outcome for an away win by 2 or more goals. Our interest in this response is to find if there are any variables or methods that can describe the difference in goals between the home and away team or if anything can describe how "close" a game is. Similarly to the 3 outcome response, we pick the draw outcome as the reference level for the response variable in the 5 outcome response variable.

6.1 Model Selection

The forward selection model:

	P Away 1	P Away 2+	P Home 1	P Home 2+
net_shots_on_target	0.00	0.00	0.00	0.00
net_shots_off	0.06	0.00	0.06	0.00
net_clear	0.00	0.00	0.00	0.00
net_corner	0.80	0.03	0.45	0.10
net_red	0.01	0.27	0.01	0.07
net_fouls	0.55	0.44	0.06	0.59
net_xG	0.00	0.00	0.00	0.00
net_deep	0.76	0.25	0.26	0.08
net_defensive_actions	0.89	0.39	0.12	0.83
home_rank	0.01	0.16	0.00	0.04
away_rank	0.02	0.01	0.00	0.17

Table 10: Forward selection model for 5 outcome model

The backward selection model:

	P Away 1	P Away 2+	P Home 1	P Home 2+
net_shots_on_target	0.00	0.00	0.00	0.00
net_shots_off	0.02	0.00	0.02	0.00
net_pass	0.03	0.00	0.01	0.17
net_clear	0.00	0.00	0.00	0.00
net_corner	0.77	0.03	0.47	0.11
net_red	0.01	0.41	0.01	0.07
net_fouls	0.59	0.54	0.22	0.43
net_xG	0.00	0.00	0.00	0.00
home_rank	0.02	0.24	0.00	0.03
away_rank	0.03	0.02	0.00	0.18

Table 11: Backward selection model for 5 outcome model

The 5 outcome variable provides a forward model with a higher amount of selected variables than in the other response variables with 11 variables, while the backward model has 10 variables. In the forward selection the defensive actions variable does not provide significance for a 10 percent level for any of the outcomes and neither does the fouls in the backward model.

To summarise the logistic models from all three models, we have 8 variables that are included in all selected logistic models, they are listed bellow:

- net_shots_on_target
- net_shots_off
- net_clear
- net_red
- net_xG
- home_rank
- away_rank
- net_fouls

All of these variables are significant on a 0.05 level in all models except the net_fouls variable.

6.2 Variable Effect

Forward	Away 2+	Away 1	Home 1	Home 2+
net_shots_on_target	0.70	0.88	1.16	1.32
net_shots_off	1.17	1.04	0.97	0.87
net_clear	0.96	0.96	1.05	1.06
net_corner	1.12	1.01	0.98	0.94
net_red	1.63	2.02	0.52	0.52
net_fouls	0.97	1.01	0.97	1.02
net_xG	0.14	0.45	2.00	7.08
net_deep	0.96	0.99	1.02	1.05
net_defensive_actions	1.02	1.00	1.02	1.00
home_rank	1.05	1.05	0.94	0.94
away_rank	0.91	0.96	1.07	1.04

Table 12: Odds for the Forward selection model

Backward	Away 2+	Away 1	Home 1	Home 2+
net_shots_on_target	0.71	0.89	1.15	1.32
net_shots_off	1.20	1.05	0.96	0.87
net_pass	0.99	1.00	1.00	1.00
net_clear	0.95	0.95	1.05	1.06
net_corner	1.12	1.01	0.98	0.94
net_red	1.43	1.97	0.51	0.52
net_fouls	0.98	1.01	0.98	1.02
net_xG	0.13	0.44	2.08	7.53
home_rank	1.04	1.04	0.95	0.94
away_rank	0.92	0.96	1.06	1.04

Table 13: Odds for the Backward selection model

Naturally it would be expected that the estimated odds values increase or decrease from left to right for the outcomes. For example we can see from Table 12 that the shots on target variable increases with every outcome as more shots hitting the target increases the odds of a win by more goals. However this is not what happens with the away rank variable for both logistic regressions where the home +1 has higher odds than the home 2+ or the net red card variable which shows higher odds for away +1 than the away+2. The natural increase does however apply to the net shots on goal and net expected goals, but also in the net deep passes for the forward model. Decrease occurs in net shots off and net corners for both models.

6.3 Comparing Predictability

Predictors	Backward	Forward
Classification	0.601	0.606
AUC	0.872	0.869
Kappa	0.446	0.452

Both the multinomial logistic regressions classify correctly just over 60% of the time and have good AUC values of close to 0.87

	Model					
Predictors	Logistic	RF	C5.0	SVM L	SVM NL	SVM P
Classification	0.606	0.551	0.468	0.618	0.555	0.551
AUC	0.869	0.838	0.795			
Kappa	0.452	0.377	0.281	0.471	0.375	0.369

Compared to the previous responses the 5 outcome response provides more inconsistent classification rates between the prediction models. The linear SVM performs well again together with the multinomial logistic regression while the C5.0 tree easily has the lowest rate with the rest performing about the same. There are only 25 instances of the away win by 2 goals in the test set. This is best classified by the linear SVM that correctly classifies 13 of the of these instances , while all 12 incorrectly classified instances all are made on the away win by 1 goal, which shows the model does a great job in classifying this compared to the other models. It also does the best job in predicting draws getting the highest classification rate while not predicting any home or away wins by a 2 goal margin. In other words it does a neat job of predicting that games are close. In Figure 7 in the appendix we find the probabilities for the 5 outcome variables. From these we can see that away 2+ and home +2 is quickly ruled out in many cases, we can also see that many fitted draw values are in the same range as for the 3 outcome variable while few values other than home win by 1 goal and away win by one goal has higher fitted values than 0.5.

7 Conclusion

So which model performed best? And which variables had the biggest effect on the outcome of the matches? Firstly we can conclude that the multiple and multinomial logistic regression performed very well and consistently outperformed the classification trees and the random forrest when it came to receiver operating characteristics and classification. However when it came to classification the Linear SVM actually did a better job classifying the data and had a higher classification rate than the logistic regressions in the multi-class responses. To conclude the SVM and Logistic regressions were the most consistent and best performers and did a good job classifying.

I was slightly disappointed with the performance of the decision trees and was especially holding out for it do a better job classifying the multinomial outputs, maybe by finding interesting patterns in other variables not selected in the logistic models.

When it came to importance of variables nothing beat the expected goals. It shows that the difference in "dangerousity" of the shots and attempts of the home and away team has high importance to the outcome of a match. It makes sense for it to be very significant and revealing to how the outcome of the match turned out as a team has to create chances and shoot to win. However as goals can be scored on low chance efforts and high chance shots can go missed or be saved by a good goalkeeper the expected goals variable doesn't tell the whole story. An interesting variable that showed high significance and was regularly included in all models was the clearance variable. While I was not expecting for it to have such high and positive significance, clearing the ball turned out to be important preventing the opposition from getting in to the real danger areas. Even if my original thought was that many clearances could show that a team was being pressed back and needed to clear the ball from danger, more occasions than the opposition, which didn't really sound like a good thing. However from the results clearing the ball is an important aspect which is needed and shows that the team does a good job stopping danger and increases the chance of the result going their way.

A very significant variable was the red card variable. This was very expected as getting a man sent off and having to play with less players than the opposition naturally should effect the outcome and gives the team with one or more players than the opposition a big advantage to win or score more goals. Although it was unexpected seeing the away +1 outcome having a bigger odds effect than the away +2 for the red card. While not reading too much into this, it could be down to the fact that red cards are a pretty rare event and certainly does not occur in every match. It could also be a sign that the team plays more defensively and tries to minimise the amount of chances while not trying to score after taking the lead. Another theory could be the away teams simply being comfortable in a 1 goal lead not risking conceding a goal and simply "seeing" out the match.

The rank variables performed pretty much as expected with better ranked teams based on the previous seasons performance meaning higher likelihood of the responses going that way. Fouls were included in all models but only provided a slight negative effect on the odds for the home win in the binary and three outcome variable. In the 5 outcome the fouls odds went down and up but overall had a pretty low effect on the fitted outcomes.

Shots off goal leading to a negative effect was a bit surprising to me as a chance, although missed could be a sign of a team creating many opportunities to score which sounds like a positive thing. However on the flip side this demonstrates an opportunity missed and also potentially some bad shooting ability. Combining this with a variable showing expected goals this could be a natural reaction showing that if teams with many high danger scoring chances also missed the target a lot results in the outcome not going their way.

When I approached this task I was expecting some of the variables that I expected to be relevant to the models were excluded by the stepwise selection. One of those was the possession stats. As a frequent viewer of the sport these statistics are commonly shown during and after matches. Although different styles of play alter the amount of ball possession a team has, with some teams actually letting the opposition contain the ball, looking to hit the opposition on a counter attack. Despite of this, my belief was that it would carry some kind of significance nonetheless, as it is so often displayed in games and signals how the game is carrying out. The conclusion from our result is generally not to focus too much on ball possession, making high volumes of passes or passing it close to the oppositions goal as this won't be beneficial. Other variables that were insignificant and not included in any logistic model were tackles and corners. This was also kind of a surprise as tackles are a key part of the game to win the ball back and corners not only being a good goal scoring opportunity but also often an indicator that the team has been on the attack.

8 Discussion

Firstly I would like to deliberate about the ranking variables that I created and added to the data. The idea was to add a kind of quality ranking to each home and away team with my thought being that better quality and more talented teams could help describe the outcome of a match. I do believe that this could be implemented in a better way than purely on previous years finish. This as many factors carry in on quality of a team with team play and individual talent being main factors, but they can be difficult to measure. Another factor that could have been added is the current form of team (how well they've been performing recently), as the momentum of a team could lead to better results than stats show. Some stats that I couldn't get a hold of, that I thought definitely could carry some interest and importance to models was running stats. Football is a sport that demands good conditioning, stamina, pace and some physicality. Some of the things that I thought could have been interesting is the amount of fast runs, total distance covered by the teams and time since last match. The running stats were unavailable and time since last match was difficult to implement as teams often play matches outside the league games with 2 different domestic cups sometimes played midweek and also European Cup games for the top teams.

I am happy with the result and believe it has some interesting pointers. Often heard about football when discussed is that there are many tactics and that some are better than others. However the main purpose is to score goals and the best way of doing this is by creating high danger chances. No matter if a team's tactic is to keep possession, play with high pressure on the opposition or to sit back and counter attack the most important factor of a game is to get to ball to high danger areas to shoot the ball. The fact that clearing the ball provided high importance also could be a sign that less risky play is rewarding.

We should also note that this work is on one specific football league. It would be interesting to see if and how variable effect differs between leagues and countries. The Premier League is traditionally known for its physicality and this could be a reason why variables such as possession and passes aren't significant as the play could encourage more crosses and long balls which more commonly leads to loss of possession than short passes. In countries such as Spain possession is glorified and many teams carry a tradition of holding the ball. For this reason I think it would be interesting to compare if there was significance for some other variables in other leagues across Europe.

When it came to different ways to classify the data such as using different regressions and machine learning techniques I decided to try these specific one's because they are prominently used and do a great job for classification data. Another method that I thought of but chose not to use was neural networking. I decided against this as it is commonly done for data with much larger sample sizes.

The analysis was based on the net value of nearly all the variables. I note that this at times can be impractical as some information is lost about each team's real amount of each variable. For example when analysing the multinomial outcomes the extra information about each team's amount of expected goals could come in useful. Imagining a scenario with extremely low total amount of chances in a game, this information is then lost and instead shown as a low difference in expected goals between teams. This information could have been useful as the likelihood of a team winning or winning by many goals margin is very unlikely when no or very few chances are created. To motivate my choice on the net valued variables it is practical when showing which variables do effect the match and making it more intuitional to understand how the different stats actually change the outcome of a match. By stating that the home team has this many more instances of this stat then the away team did we get a more tangible result of which variables are important and can actually effect the outcome of a match, rather than multiple variables for each statistic making it more difficult to compare what is actually important for a team to focus on.

An important fact is that different statistics carry different amounts. For example teams does a lot more passes in a game than shots meaning that the net amount between these variables are more likely to vary than some others. So a final analysis about the variable effect is that the high odds on difference in expected goals carry high importance to the decision the logistic regression makes, but it must be noted that this stat doesn't usually vary a great deal between the home and away team which may contribute to why this variable has such high odds. Interestingly this makes the odds on the net clearance and shots on goals stats even more noteworthy as they occur more often and may vary more.

References

Articles & Books

[1] Peter Christopher Alegi. Football - brittanica. https://www.britannica.com/sports/football-soccer.

[2] Christopher M. Bishop. *Pattern Recognition And Machine Learning*. Springer, 2006.

[3] Alan Agresti. *Categorical Data Analysis, Third Edition*. Wiley, 2012.

[4] abhinav43. Gini impurity and entropy in decision tree – ml. https://www.geeksforgeeks.org/gini-impurity-and-entropy-in-decision-tree-ml/, 2020.

[5] James Le. Decision trees in r. https://www.datacamp.com/community/tutorials/decision-trees-R.

[6] Rolf Sundberg. Lineara statistiska modeller, August 2020.

[7] David W. Hosmer, Stanley Lemeshow, and Rodney X. Sturdivant. *Applied Logistic Regression Third Edition*. Wiley, 2013.

R Packages

[8] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020.

[9] Hadley Wickham, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, Alex Hayes, Lionel Henry, Jim Hester, Max Kuhn, Thomas Lin Pedersen, Evan Miller, Stephan Milton Bache, Kirill Müller, Jeroen Ooms, David Robinson, Dana Paige Seidel, Vitalie Spinu, Kohske Takahashi, Davis Vaughan, Claus Wilke, Kara Woo, and Hiroaki Yutani. Welcome to the tidyverse. *Journal of Open Source Software*, 4(43):1686, 2019.

[10] Max Kuhn. *caret: Classification and Regression Training*, 2021. R package version 6.0-87.

[11] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. ISBN 0-387-95457-0.

[12] Max Kuhn and Ross Quinlan. *C50: C5.0 Decision Trees and Rule-Based Models*, 2020. R package version 0.1.3.1.

[13] Terry Therneau and Beth Atkinson. *rpart: Recursive Partitioning and Regression Trees*, 2019. R package version 4.1-15.

[14] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

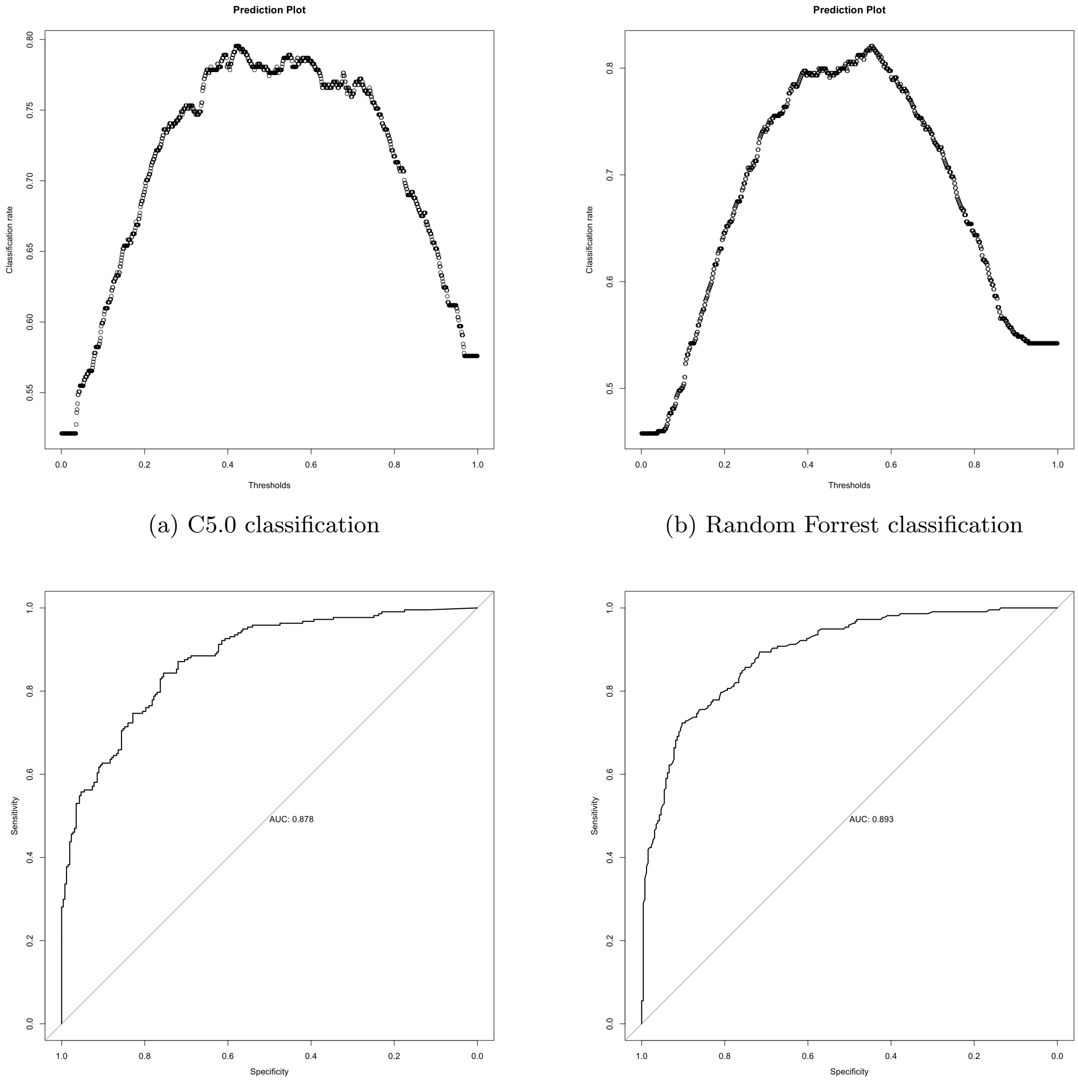
[15] Xavier Robin, Natacha Turck, Alexandre Hainard, Natalia Tiberti, Frédérique Lisacek, Jean-Charles Sanchez, and Markus Müller. proc: an open-source package for r and s+ to analyze and compare roc curves. *BMC Bioinformatics*, 12:77, 2011.

Data Sources

[16] Epldataset. https://www.kaggle.com/englader/epldatase.

[17] Expected goals and other metrics. https://www.kaggle.com/englader/epldataset.

9 Appendix



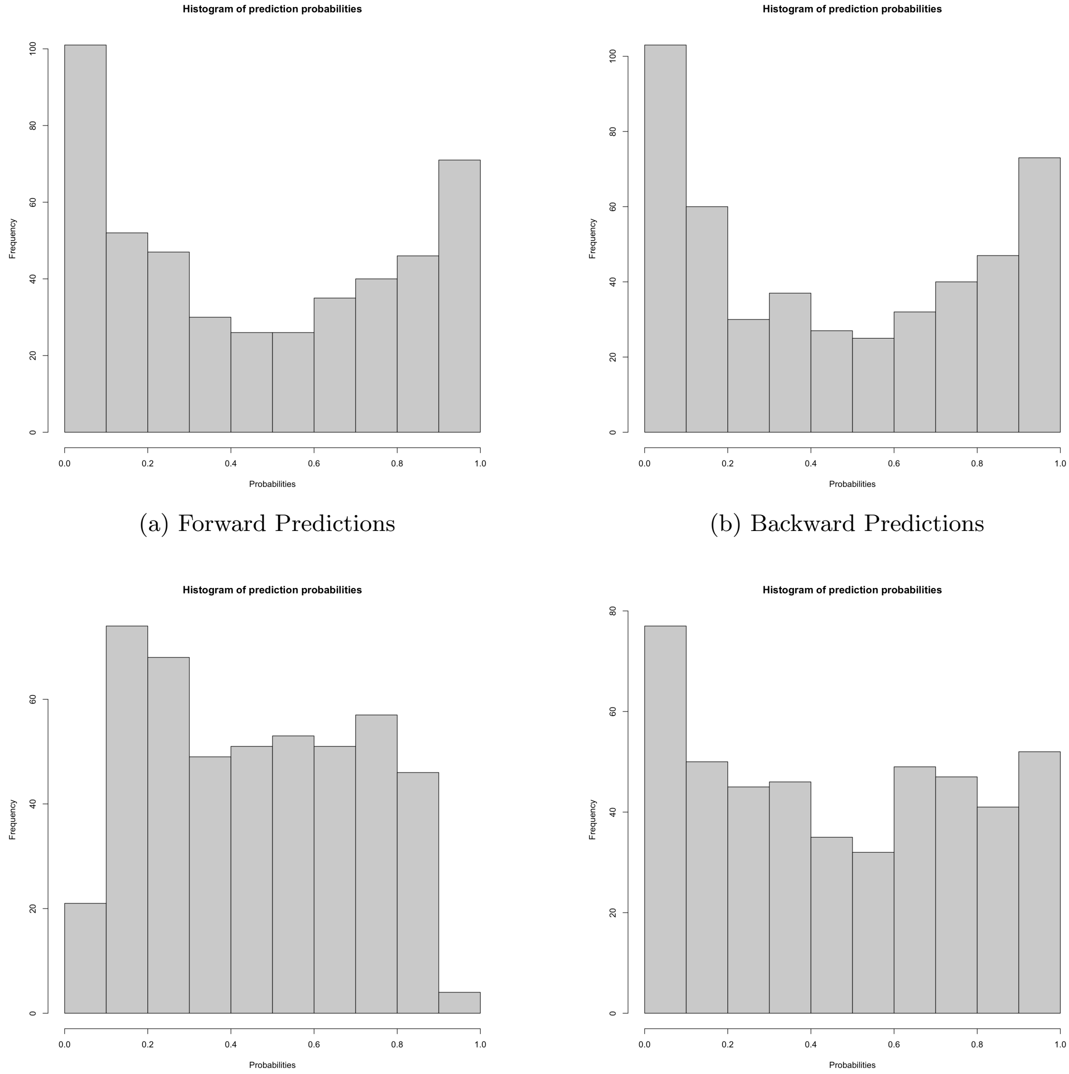
(a) C5.0 classification

(b) Random Forrest classification

(c) C5.0 ROC

(d) Random Forrest ROC

Figure 4: Classification and ROC



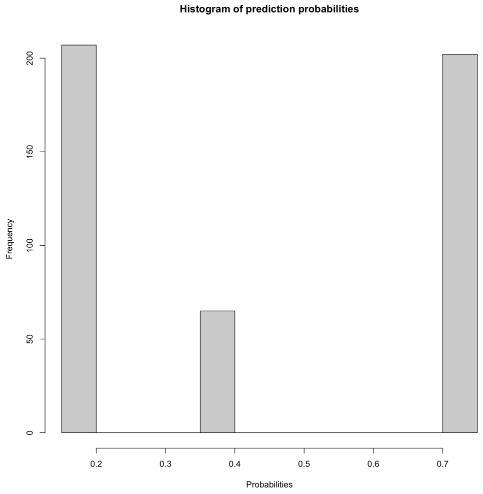
(a) Forward Predictions

(b) Backward Predictions

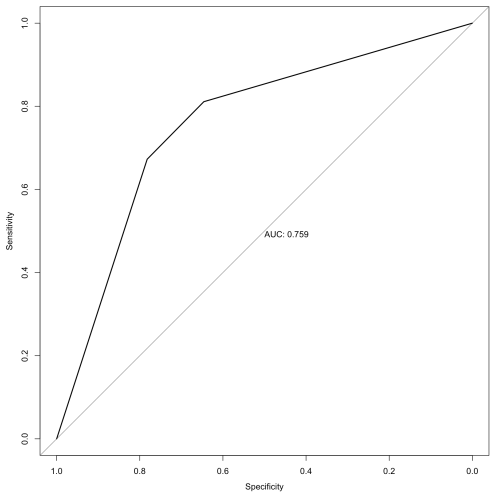
(c) Random Forrest Predictions

(d) C5.0 Predictions

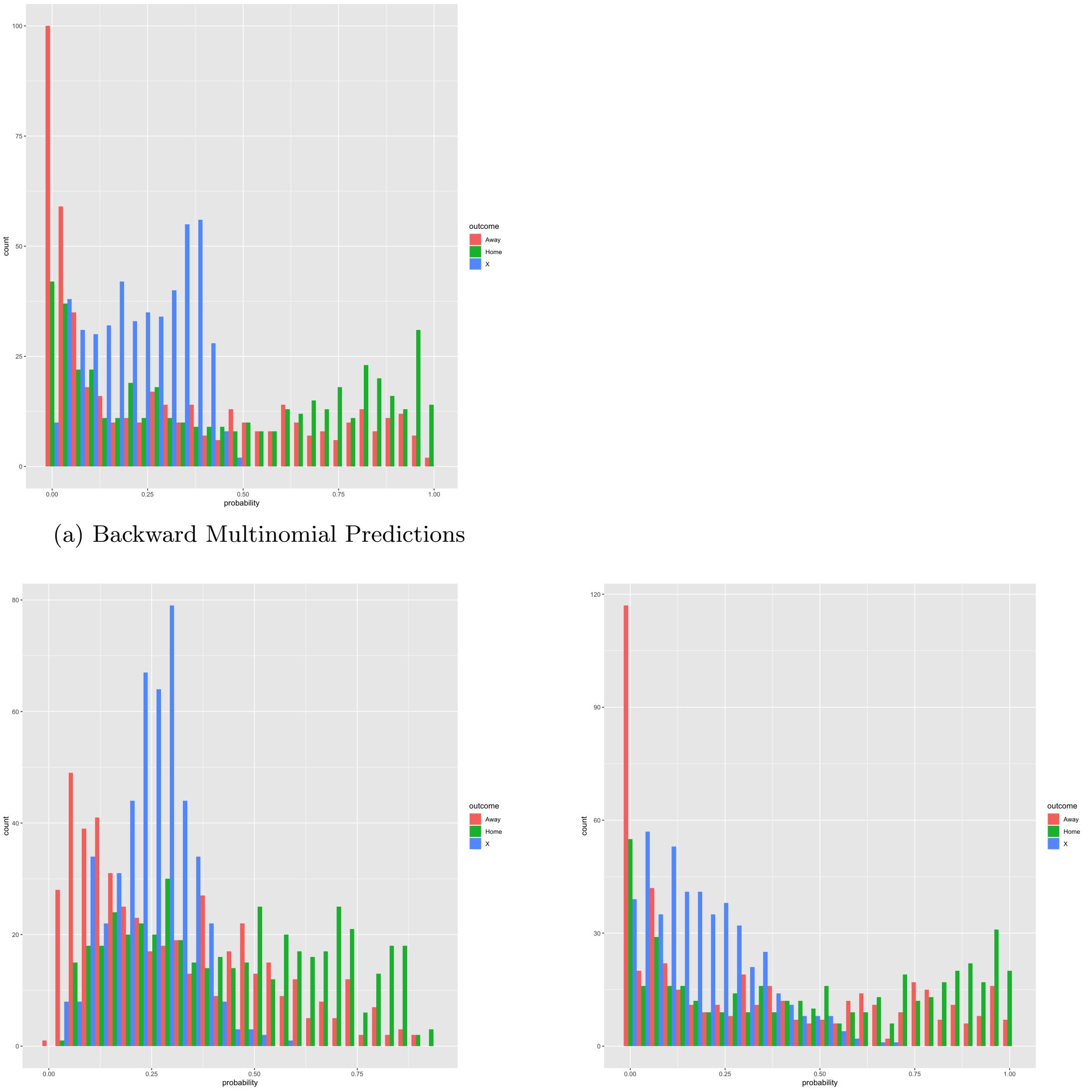
Figure 5: Histogram of each sample predictions



(a) CART Predictions



(b) CART ROC

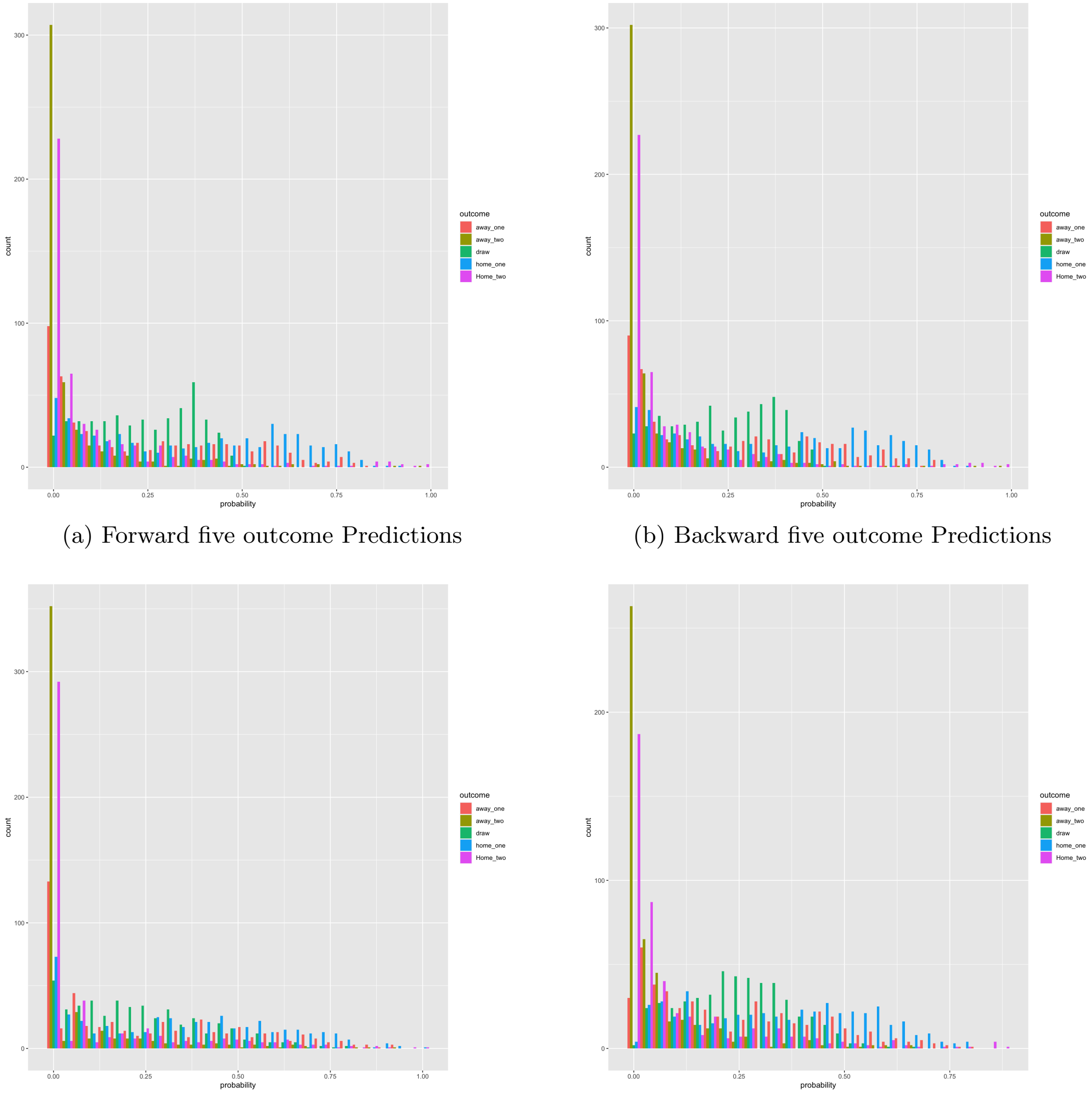


(a) Backward Multinomial Predictions

(b) Random Forrest Multinomial Predictions

(c) C5.0 Multinomial Predictions

Figure 6: Histogram of total number of predictions in likelihoods for each outcome



(a) Forward five outcome Predictions

(b) Backward five outcome Predictions

(c) Random Forrest five outcome Predictions

(d) C5.0 five outcome Predictions

Figure 7: Histogram of total number of predictions in likelihoods for each outcome