Fontys Hogeschool Techniek en Logistiek

Business Informatics

Data Mining

Research report

Authors:
Marius Freyer
Niklas Schwerin

Lecturers:
J. Jacobs, M. Langenhuizen

December 15, 2017

Contents

1 Introduction

A sports bet is a bet where money is wagered on the arrival of a sports result. Nowadays many international operating companies are specialized in sports betting and provide their customers with more than 1000 types of betting in many different sports.
Today, football sports betting like matches of the European top leagues, European cup competitions and international matches dominate the betting market.
The german Bundesliga which is one of the best leagues in the world is a part of this market. With up to 10 million viewers on specific games it is very attractive to bet on. But betting is often not profitable for the customer but only for the bookmaker.
The bookmakers have access to various statistic data and have algorithms with which they create their odds. The customers often bet based on their feelings which mostly leads to a low performance. The only profitable sports betting strategy is to find so-called value. Value means in this case added value in the sense that the bookmaker's odds are set too high for a certain event.
In this research we try to find our own odds (or probabilities) for Bundesliga games with statistical and machine learning algorithms, compare each other and of course compare them with the odds of the bookmakers to find value.

2 Learning goals

1. Data analysis with RapidMiner
    - Load different data in RM
    - Apply different models and operators in RM
    - Apply machine learning algorithms in RM
    - Handle result with RM
2. Data analysis in R
    - Load data in R
    - Apply statistical algorithms in R
    - Handle results in R

3. Data preparation in Excel
    - Prepare data using filters and functions

3 Problem description

Odds in sports betting are just probabilities of different occurrences. A sports better puts a certain wager on a certain event and tries to make profit with his bet amount. Of course he wins if the events occurs.
But betting is often difficult to win. The bookmakers have a huge advantage over the customers and nearly always win over time because most people bet based on their feelings and the bookmakers use a lot of different approaches to estimate the odds.
If a sports better actually wants to make profit with sports betting he has to gather and analyze many data to get an advantage over the bookmaker which is really time-consuming without tools.

4 Research question

Our research consists of different research questions. The main focus is to find value in sports betting on Bundesliga games by using statistical. Besides we try to compare the statistical algorithm with a machine learning algorithm and try to estimate which approach is performing best based on correct predicted matches.

1. Find value in sports betting on Bundesliga games by using a statistical algorithm
2. Predict football results in sports betting on Bundesliga games by using a machine learning algorithm and a statistical algorithm
3. Compare the statistical and machine learning approaches

5 Analysis

5.1 Finding value in sports betting on Bundesliga games

5.1.1 Collection of data

The collection of data began with a internet research about Bundesliga results which are stored in a proper way. We searched for an API that saved the data in xml, csv or Excel. After some investigation we found the data set 'OpenLigaDB' on the website https://www.openligadb.de. The data format is xml which is easy readable with the help of Rapid Miner. The data is the basis of our research of finding value by using statistical algorithms.
It contains the halftime and fulltime results, goal scorers, time of the goals and basic information like the number of spectators and stadium information of many seasons of Bundesliga.

5.1.2 Data preparation

The data preparation is done in RapidMiner and Excel. We used RapidMiner because it has pre-defined operators to fastly load the data we want. Excel is used because we already worked with it and it is a fast way to prepare data. In Rapid Miner the operator "Read XML" is used to read in 10 years of Bundesliga. Therefore 10 operators are used to read the season 2007-2017. Every operator is configured the same way so that we gather the season, match day, home and away team and of course home and away goals scored in the games as shown in figure 1.
With the operator "Append" the 10 read data sets are now merged together and the operator "Write CSV" writes a new aggregate dataset with all the Bundesliga data we need (figure 12). Based on this csv file it is now possible to analyse the data.

default:Grou	default:Tean	default:Tean	default:Matcl	default:Matcl	ame[1]/text()
integer	text	text	integer	integer	text
attribute	attribute	attribute	attribute	attribute	attribute
12	Werder ...	FC Hans...	1	0	1. Fussb...
12	Bayern M...	Eintracht...	0	0	1. Fussb...
12	Bayer 04...	Arminia ...	4	0	1. Fussb...
12	1. FC Nü...	VfB Stutt...	0	1	1. Fussb...
12	Hambur...	Hertha B...	2	1	1. Fussb...
12	Hannove...	Borussia...	2	1	1. Fussb...
12	VfL Boch...	VfL Wolf...	5	3	1. Fussb...
12	Karlsruh...	MSV Dui...	1	0	1. Fussb...
13	MSV Dui...	VfL Boch...	0	2	1. Fussb...
13	VfB Stutt...	Bayern M...	3	1	1. Fussb...
13	FC Schal...	Hambur...	1	1	1. Fussb...

Figure 1: Screenshot: Configuration of operator

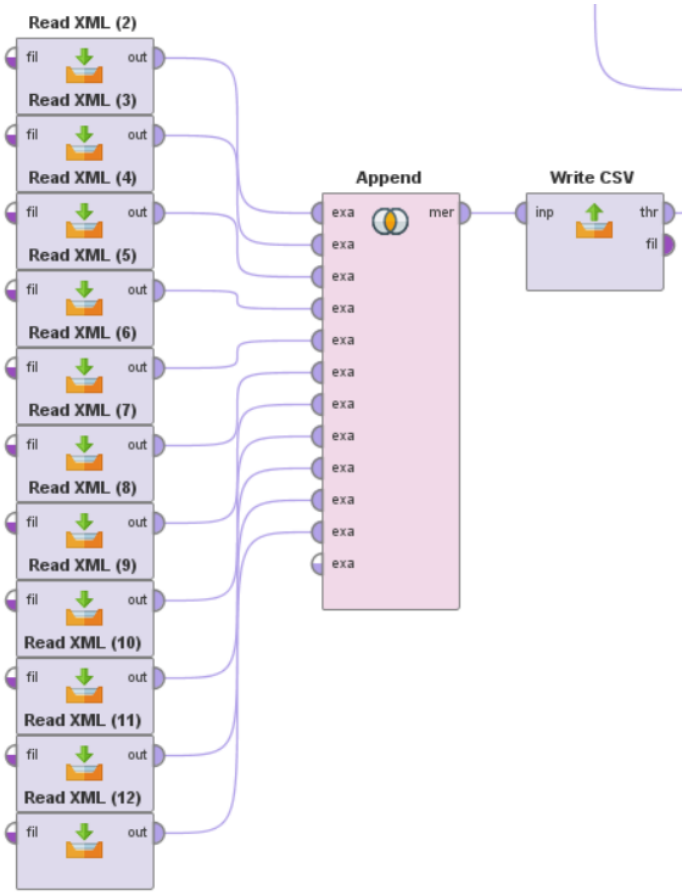

Figure 2: Screenshot: Data preparation

5.1.3 Data analysis

For the data analysis we use the R plugin in RapidMiner. R is a very powerful and flexible scripting language we already now about because of the statistic course.
For the statistical analysis we use the poisson distribution because we found some information in the internet about it as a way to predict football games i.e. on the site of the bookmaker Pinnacle (https://www.pinnacle.com/en/betting-articles/Soccer/how-to-calculate-poisson-distribution/MD62MLXUMKMXZ6A8).
To execute the R code the operator "Execute R" is used and connected with the result to show the results after we start our finished process as shown in figure 3.

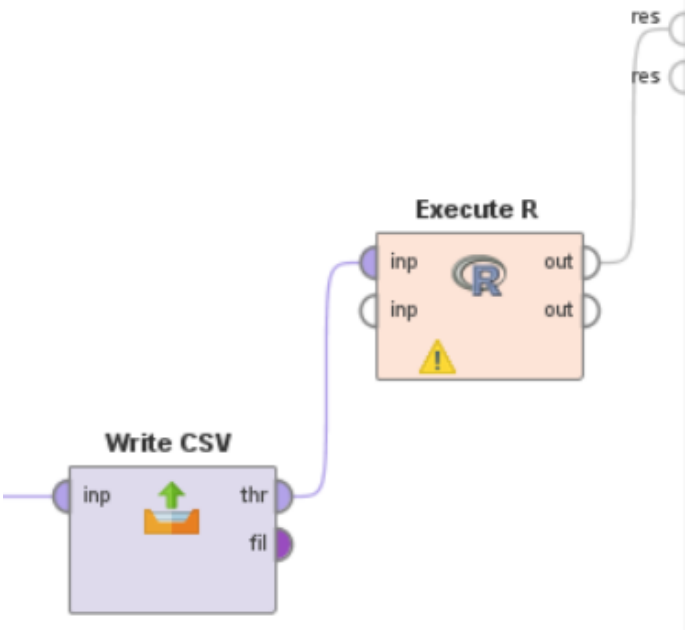

Figure 3: Screenshot: Data analysis with R operator

The data analysis in R starts with the loading of the created csv data in RapidMiner. After that we fit the Poisson model with the home and away teams and the scored goals of the home and away team. Then the summary of the model is printed.

```
#load data
buli <- read.csv("...\\buli.csv", header=TRUE)

#fit poisson model and get a summary
model <- glm(GoalsHome ~ GoalsAway + Home + Away, family=poisson(
    link=log), data=buli)
print(summary(model))
```

After that the game is picked (idc. Leverkusen vs Ingolstadt) and the average number of goals of every team are predicted with the predict function.

```
#probabilites of average goals scored
#Leverkusen
predictHome <- predict(model, data.frame(GoalsAway=1,
    Home="Bayer 04 Leverkusen", Away="FC Ingolstadt 04"),
    type="response")

#Ingolstadt
predictAway <- predict(model, data.frame(GoalsAway=0,
    Home="FC Ingolstadt 04", Away="Bayer 04 Leverkusen"),
    type="response")
```

Then we simulate 10000 rounds with the function "rpois" which generates random deviates (rpois(n, lambda)) on how many goals the teams will score based on the poisson algorithm. Then every round we subtract the home and away goals and save if the goal difference is lower(home win), equal(draw) or higher(away win) than 0. After that we divided the values by the simulated rounds to get the probabilites for a home win, draw or away win.

```
#simulation with 10.000 possible results based on the
    probabilites
  simRounds <- 10000
homeGoalsSim <- rpois(simRounds, predictHome)
awayGoalsSim <- rpois(simRounds, predictAway)
```

```
goalDiffSim <- homeGoalsSim - awayGoalsSim
#Home
homeprob <- sum(goalDiffSim > 0) / simRounds
print(homeprob)
#Draw
drawprob <- sum(goalDiffSim == 0) / simRounds
print(drawprob)
#Away
awayprob <- sum(goalDiffSim < 0) / simRounds
print(drawprob)
```

5.1.4 Evaluation of the results

For the evaluation of the results the predicted probabilities of the matchdays 14 and 15 are stored in a Excel file and transformed into the odds format (i.e. 1.75). After that, the odds of the bookmaker Tipico are also stored in this Excel file and compared with the predicted odds (figure 4 & 5).

As you can see the range of the difference is not very high which means that our estimated results are very realistic. The differences are shown with the colors green, yellow and red. Green means that the odd of the bookmaker is higher than our predicted odd which means that this odd has value and the person should bet on it.

Red means that the odd of the bookmaker is lower which means the person should avoid betting on it.

Match		Home	Draw	Away			
SC Freiburg - Hamburger SV	Prediction	2,9	3,6	2,6	0,344	0,2775	0,3788
	Tipico (Bookmaker)	2,45	3,3	2,85			
	Difference	-0,5	-0,3	0,2			
Bayern München - Hannover 96	Prediction	1,2	9,0	16,1	0,827	0,1107	0,0621
	Tipico (Bookmaker)	1,15	8	17			
	Difference	-0,1	-1,0	0,9			
TSG 1899 Hoffenheim - RB Leipzig	Prediction	4,2	3,3	2,2	0,237	0,3071	0,4564
	Tipico (Bookmaker)	3	3,7	2,2			
	Difference	-1,2	0,4	0,0			
Werder Bremen - VfB Stuttgart	Prediction	2,6	4,4	2,6	0,39	0,2281	0,382
	Tipico (Bookmaker)	2,3	3,5	3			
	Difference	-0,3	-0,9	0,4			
Bayer 04 Leverkusen - Borussia Dortmund	Prediction	2,7	3,9	2,6	0,365	0,2553	0,3798
	Tipico (Bookmaker)	2,35	3,7	2,75			
	Difference	-0,4	-0,2	0,1			
1. FSV Mainz 05 - FC Augsburg	Prediction	2,7	3,2	3,1	0,368	0,3093	0,3228
	Tipico (Bookmaker)	2,5	3,3	2,85			
	Difference	-0,2	0,1	-0,2			
FC Schalke 04 - 1. FC Köln	Prediction	1,9	3,9	5,0	0,541	0,2584	0,2011
	Tipico (Bookmaker)	1,45	4,5	7,5			
	Difference	-0,4	0,6	2,5			
Hertha BSC - Eintracht Frankfurt	Prediction	3,4	3,3	2,5	0,293	0,3011	0,406
	Tipico (Bookmaker)	2,75	3,2	2,6			
	Difference	-0,7	-0,1	0,1			
VfL Wolfsburg - Borussia Mönchengladbach	Prediction	2,3	3,9	3,2	0,427	0,2559	0,3167
	Tipico (Bookmaker)	2,55	3,5	2,65			
	Difference	0,2	-0,4	-0,5			

Figure 4: Odd comparison matchday 14

Match		Home	Draw	Away			
VfB Stuttgart - Bayer 04 Leverkusen	Prediction	4,0	4,2	2,0	0,2517	0,2363	0,512
	Tipico (Bookmaker)	3,3	3,6	2,1			
	Difference	-0,7	-0,6	0,1			
Borussia Dortmund - Werder Bremen	Prediction	1,7	4,8	4,7	0,5809	0,2074	0,2117
	Tipico (Bookmaker)	1,4	5	7			
	Difference	-0,3	0,2	2,3			
RB Leipzig - 1. FSV Mainz 05	Prediction	2,3	2,9	4,5	0,4293	0,347	0,2237
	Tipico (Bookmaker)	1,35	5	8,5			
	Difference	-1,0	2,1	4,0			
Eintracht Frankfurt - Bayern München	Prediction	11,9	6,0	1,3	0,0837	0,1673	0,749
	Tipico (Bookmaker)	6,5	4,6	1,45			
	Difference	-5,4	-1,4	0,1			
Hamburger SV - VfL Wolfsburg	Prediction	4,1	4,0	2,0	0,2411	0,247	0,5119
	Tipico (Bookmaker)	2,45	3,4	2,8			
	Difference	-1,7	-0,6	0,8			
Borussia Mönchengladbach - FC Schalke 04	Prediction	2,9	3,8	2,5	0,3411	0,2609	0,398
	Tipico (Bookmaker)	2,1	3,5	3,3			
	Difference	-0,8	-0,3	0,8			
1. FC Köln - SC Freiburg	Prediction	2,8	3,3	2,9	0,3532	0,3005	0,3463
	Tipico (Bookmaker)	2,1	3,3	3,5			
	Difference	-0,7	0,0	0,6			
Hannover 96 - TSG 1899 Hoffenheim	Prediction	2,7	4,4	2,5	0,3676	0,2288	0,4036
	Tipico (Bookmaker)	2,75	3,3	2,55			
	Difference	0,0	-1,1	0,1			
FC Augsburg - Hertha BSC	Prediction	2,3	3,0	4,1	0,4271	0,3292	0,2437
	Tipico (Bookmaker)	1,9	3,5	4			
	Difference	-0,4	0,5	-0,1			

Figure 5: Odd comparison matchday 15

5.2 Predict football results in sports betting on Bundesliga games

In this approach we use the Naive Bayes algorithm to predict the football results and compare it with our statistical algorithm approach to determine which is performing better.

5.2.1 Collection of data

Because our dataset was not able to work with machine learning algorithms we had to research other data in the internet. We found a dataset which had the difference of the scored goals and the scored against for every team of match. The data was stored in a Excel sheet which is easy to read in Rapid Miner.

5.2.2 Data preparation

The data preparation is done in Rapid Miner. At first we used the operator "Read Excel" to read in our Excel file. Then we split the data in our training set and test set with the operator "Split Data". In our training set we select the attributes scored goals and the scored against goals which means that we delete all the attributes beside of the two to apply the Naive Bayes based on this attributes. This is done with the operator "Select Attributes" (figure 6).

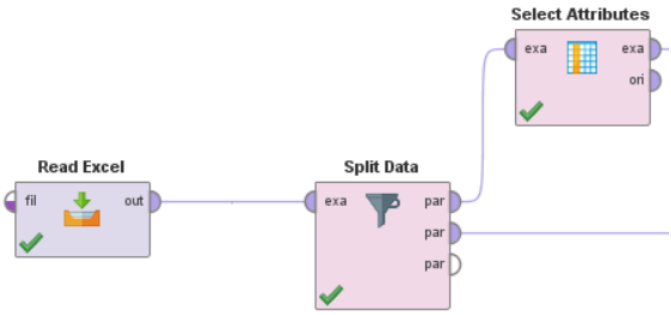

Figure 6: Screenshot: Data preparation

5.2.3 Data analysis

The data analysis is also done with Rapid Miner. At first the operator "Naive Bayes" is applied on the training set with the selected attributes. The operator creates a Naive Bayes classification model.

Then we apply the Naive Bayes model with the operator "Apply Model" with the training set

on the data of the test set. We connect the test set with the result to show the created data table with the predictions in our output.
The created data is also connected with the operator "Performance" to evaluate the performance of the output and get the accuracy of the prediction as shown in figure 7

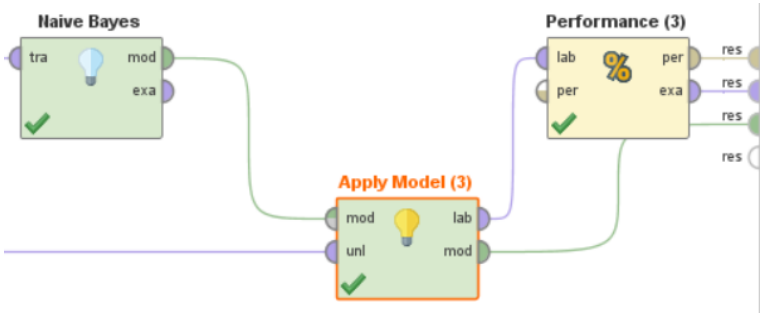

Figure 7: Screenshot: Data analysis

5.2.4 Evaluation of the results

Our result test set shows us the different Bundesliga matches with the winning team. The column "prediction" shows us the predicted winner of the Naive Bayes algorithm and the other columns "confidence" show which percentage the predicted outcome home team wins, away team wins and draw has(figure 8).
In figure 9 the accuracy of the prediction is shown. It is 50.42% that our algorithm predicted the right result. The table shows that a 1 is predicted well in 76.19% of the games, a draw is predicted well in just 0.82% of the games and a 2 is predicted correct in 50.11% of the matches.

Row No.	Saison	Heim	Gast	Gewinner	prediction(G...	confidence(1)	confidence(X)	confidence(2)
1	2010	Bayern	Wolfsburg	1	1	0.690	0.238	0.072
2	2010	Gladbach	Nuernberg	X	1	0.535	0.235	0.230
3	2010	Hannover	Frankfurt	1	2	0.342	0.236	0.421
4	2010	Hamburg	Schalke	1	1	0.390	0.248	0.362
5	2010	Mainz	Stuttgart	1	2	0.324	0.279	0.397
6	2010	Dortmund	Leverkusen	2	1	0.438	0.259	0.303
7	2010	Schalke	Hannover	2	1	0.689	0.269	0.043
8	2010	Nuernberg	Freiburg	2	1	0.414	0.253	0.332
9	2010	Wolfsburg	Mainz	2	1	0.688	0.163	0.149
10	2010	Frankfurt	Hamburg	2	2	0.349	0.228	0.423
11	2010	Leverkusen	Gladbach	2	1	0.756	0.190	0.055
12	2010	Stuttgart	Dortmund	2	1	0.414	0.253	0.332
13	2010	Hoffenheim	Schalke	1	2	0.309	0.247	0.444
14	2010	Dortmund	Wolfsburg	1	1	0.527	0.305	0.168

Figure 8: Prediction test set

accuracy: 50.42%

	true 1	true X	true 2	class precision
pred. 1	544	247	233	53.12%
pred. X	2	3	0	60.00%
pred. 2	168	118	234	45.00%
class recall	76.19%	0.82%	50.11%	

Figure 9: Performance Naive Bayes

In the next step the predictions of the Naive Bayes algorithm is compared with the predictions of our statistical approach with the Poisson algorithm. Therefore an Excel sheet is created with 50 matches of 2016 and the predictions of both algorithms. The Poisson algorithm is also used with Bundesliga data from 2010 to 2016 to have similar data to analyse.

The following figure 10 shows 50 games in a Excel sheet which are analysed with the two algorithms. The red columns are the predictions of the Naive Bayes algorithm and the blue columns of the Poisson algorithms. As shown in figure 11 the Naive Bayes algorithm performs better with a correct prediction performance of 50% against the prediction of the Poisson algorithm with a performance of 42%.

Really remarkable is the fact that the Naive Bayes algorithm almost never predicts a draw.

The predictions with the Poisson algorithm are more likely a draw and therefore the outcome of it has higher probabilities.

Home	Away	Home	Draw	Away	Prediction	Home	Draw	Away	Prediction	Result
Bayern	Bremen	0,9	0,1	0,0	1	0,9	0,1	0,0	1	1
Augsburg	Wolfsburg	0,4	0,2	0,4	1	0,3	0,3	0,4	2	2
Dortmund	Mainz	0,7	0,1	0,2	1	0,6	0,2	0,2	1	1
Koeln	Darmstadt	0,5	0,2	0,3	1	0,2	0,5	0,3	X	1
Hamburg	Ingolstadt	0,5	0,2	0,3	1	0,2	0,4	0,3	X	X
Frankfurt	Schalke	0,3	0,2	0,5	2	0,3	0,3	0,4	2	1
Gladbach	Leverkusen	0,5	0,2	0,3	1	0,4	0,3	0,4	1	1
Schalke	Bayern	0,3	0,2	0,5	2	0,1	0,2	0,7	2	2
Leverkusen	Hamburg	0,6	0,2	0,2	1	0,6	0,2	0,2	1	1
Wolfsburg	Koeln	0,5	0,2	0,3	1	0,5	0,3	0,2	1	X
Darmstadt	Frankfurt	0,5	0,2	0,3	1	0,3	0,5	0,2	X	1
Ingolstadt	Berlin	0,4	0,2	0,4	2	0,4	0,5	0,2	X	2
Bremen	Augsburg	0,5	0,2	0,3	1	0,3	0,3	0,5	2	2
Mainz	Hoffenheim	0,5	0,2	0,3	1	0,4	0,3	0,3	1	X
Bayern	Ingolstadt	0,8	0,1	0,1	1	0,6	0,3	0,1	1	1
Dortmund	Darmstadt	0,7	0,1	0,1	1	0,4	0,5	0,2	X	1
Hoffenheim	Wolfsburg	0,4	0,2	0,4	1	0,3	0,3	0,4	2	X
Frankfurt	Leverkusen	0,3	0,2	0,5	2	0,3	0,3	0,4	2	1
Gladbach	Bremen	0,7	0,1	0,2	1	0,6	0,2	0,2	1	1
Augsburg	Mainz	0,4	0,2	0,4	1	0,3	0,3	0,4	2	2
Berlin	Schalke	0,5	0,2	0,3	1	0,2	0,3	0,5	2	1
Wolfsburg	Dortmund	0,3	0,2	0,5	2	0,2	0,3	0,5	2	2
Darmstadt	Hoffenheim	0,4	0,2	0,4	1	0,3	0,5	0,2	X	X
Ingolstadt	Frankfurt	0,5	0,2	0,3	1	0,3	0,4	0,3	X	2

Figure 10: Extract: Comparison Naive Bayes & Poisson

Algorithm	Prediction Naive	Prediction Poisson
Games correct	25	21
Percentage	50,0%	42,0%

Figure 11: Result comparison

6 Verifying prediction results

6.1 Teams strength

During the process of finding suitable algorithms for predicting soccer match results, a basis for evaluation was needed. Therefore an excel sheet with the average attacking and defensive strength of the soccer teams in relation to the league average was created. The attacking strength is defined as the average goals shot of a team divided by the average goals shot per team in league average. The same applies for the defensive strength except the goals against the teams are used for the calculation. So a value above 1 in attacking strength indicates a high likelihood of scoring goals, while a value above 1 in defensive strength indicates a weak defense and a high likelihood of goals against the team.

This excel sheet was then used for reviewing the predicted match result by the different algorithms.

	Home		Away	
Team	Attacking Strength	Defensive Strength	Attacking Strength	Defensive Strength
1. FC Kaiserslautern	0.852	1.119	1.099	1.144
1. FC Koeln	0.818	0.952	0.933	0.981
1. FC Nuernberg	0.931	1.011	0.994	1.198
1. FSV Mainz 05	1.018	0.777	1.099	0.893
Arminia Bielefeld	0.876	1.094	0.848	1.446
Bayer 04 Leverkusen	1.311	0.785	1.413	0.771
Borussia Dortmund	1.434	0.651	1.639	0.831
Borussia Moenchengladbach	1.162	0.817	1.040	0.874
Eintracht Braunschweig	0.852	1.143	0.628	1.365
Eintracht Frankfurt	0.994	1.066	0.900	0.941
Energie Cottbus	1.041	1.168	0.659	1.325
FC Augsburg	0.868	0.862	0.764	0.810
FC Bayern Muenchen	1.917	0.522	1.740	0.502
FC Hansa Rostock	0.805	1.044	0.816	1.245
FC Ingolstadt 04	0.473	0.423	0.377	0.442
FC Schalke 04	1.273	0.671	1.187	0.863
FC St. Pauli	0.947	1.641	0.879	1.325
Fortuna Duesseldorf	0.947	1.392	0.816	1.124
Hamburger SV	0.961	0.885	1.074	1.028
Hannover 96	1.252	1.127	1.019	1.236
Hertha BSC	0.757	0.858	1.036	0.964
Karlsruher SC	0.970	1.069	0.848	1.265
MSV Duisburg	0.899	1.442	1.068	1.044
RB Leipzig	0.852	0.348	0.628	0.442
SC Freiburg	0.913	1.001	1.032	1.056
SC Paderborn 07	0.947	1.491	0.628	1.365
SpVgg Greuther Fuerth	0.426	1.591	1.005	0.924
SV Darmstadt 98	0.379	0.522	0.565	0.301
TSG 1899 Hoffenheim	1.178	0.978	1.235	1.075
VfB Stuttgart	1.231	1.061	1.382	1.124
VfL Bochum	1.152	1.508	0.984	1.071
VfL Wolfsburg	1.326	0.945	1.300	0.867
Werder Bremen	1.236	1.039	1.363	1.156

Figure 12: Screenshot: Teams strength

6.2 SOM map

Another approach for verifying the predicted results was using the SOM map to classify the different matches of the teams. As it turned out the SOM map approach did not give us the results we expected. The matches of the different teams were rather mixed and could not be used to classify the teams by strength and their likelihood of relegation or staying in the league. So we discontinued this approach quite quickly.

7 Reflection on the learning goals

In this chapter a reflection on our learning goals is done.
We defined three main learning goals which were the data analysis with RapidMiner and in R such as the data preparation in Excel. For every main learning goals activities were defined which are reflected in this chapter.
The first learning goal is "Data analysis with RapidMiner". We successfully loaded different data in RapidMiner such as xml files in our analysis "Finding value in football betting" and an Excel file to predict the result of football matches with the Naive Bayes algorithm.
Furthermore we successfully applied different models and operators in RapidMiner like the operators "Append", "Write CSV", "Split Data", "Select Attributes", "Apply Model" and "Performance". We also applied the Naive Bayes algorithm as a machine learning algorithm to predict the result of Bundesliga matches and got a predicted data set which we handled with RapidMiner as stated in our learning goals.
Furthermore we conducted a data analysis in R. Therefore we load csv data in R, applied the Poisson algorithm in R which is a statistical algorithms which we stated in our learning goals. Then we handled the results in R and printed them successfully out.
The last learning goals was the data preparation in Excel using filters and functions. We created a Excel sheet with defensive and offensive strength of the teams and also an Excel sheet where we stored our results of our two compared algorithms.