Homework 1: Association Rules and Sports Analytics

Nakul Agrawal, Samira Arondekar, Sai Akhil Kodali, Xue Ni, Yuting Xin

9/29/2019

Contents

The Business Problem

Borussia Dortmund is one of the top teams in German Soccer League. But in some of the past seasons they have had inconsistent performances. Our objective is to help the team stay consistent with their performance and win the German League.

Our analysis is directed to help the coach of Borussia Dortmund take strategic decisions and make sure that team becomes the champion in the next season.

What is Success?

We define success as winning more games and getting as many points as possible. Soccer is a highly competitive game and a good team is the team who always wins the game. Winning more games means more reputation, more fans, more money. The nature of competitive sports means that the only definition for success is win.

Approach

A team generally has two options to maximize wins - either it can hire better players or it can improve the game play strategy. Since we do not have data regarding Dortmund's budget or player costs, we will focus on the latter.

Good coaching is essential to a soccer team and the best coach can earn more than $10 million a year. Generally, a team's coach has different plans for different games. For example, if Dortmund is going to play against a weak team in the league, the best way is to keep the ball under control and keep attacking them. However, when Dortmund is going to play against one of the best team in the history, FC Bayern Munich, who is in the same league, Dortmund needs to focus on defense and hope the counterattack would work.

We want to analyze what sets of strategies work best depending on who we are playing agaist. We hope to make a guidance manual for Dortmund's coach to help him choose the best game strategy and improve the winning rate.

Data Preparation

Loading libraries and data:

We used information from our past matches and combined it with the information from our team and player statistics.

```
setwd("~/Desktop/Homework 1")

library(plyr)
library(dplyr)
library(tidyr)
library(magrittr)
library(openxlsx)
library(xml2)
library(purrr)
library(arules)
library(reshape2)
library(data.table)
library(RSQLite)
library(arulesViz)
library(datasets)
library(sqldf)
library(stringi)
library(stringr)
library(xml2)
library(ggplot2)
library(ggrepel)
library(lemon)

con <- dbConnect(SQLite(), 'euro_soccer.sqlite')
match <- dbGetQuery(con, 'SELECT * FROM Match')
team_atts <- dbGetQuery(con, 'SELECT * FROM Team_Attributes')
player_attributes <- dbGetQuery(con, 'SELECT * FROM Player_Attributes')
team_df <- dbGetQuery(con, 'SELECT * FROM Team')
league_df <- dbGetQuery(con, 'SELECT * FROM League')
```

Filtering data:

To start our analysis, we fetched match data related only to Dortmund. We dropped all irrelevant columns for our analysis.

```
dortmund_matches <- match %>%
  filter(home_team_api_id == 9789 | away_team_api_id == 9789)
```

Adding flag for home/away matches:

A team always has 2 legs against any opponent - home and away matches. Generally a team's performance varies a lot in home and away matches due to the change in the environment and playing atmosphere. So we split the dataset into these 2 categories to see if there is any discernible patterns between their performances at home and away games.

```
dortmund_matches$home_or_away <- ifelse(dortmund_matches$home_team_api_id == 9789,
                                        'home',
                                        'away')
```

Creating columns for Goal Difference and Match Outcome:

We created new columns to calculate the value of the goal difference and infer the match Outcome (to represnt wins/loses/ties) for Dortmund based on whether this value is greater than 0 (win), equal to 0 (tie), and less than 0 (lose).

```
dortmund_matches$goal_diff <- ifelse(dortmund_matches$home_team_api_id == 9789,
                                     dortmund_matches$home_team_goal -
                                       dortmund_matches$away_team_goal,
                                     dortmund_matches$away_team_goal -
                                       dortmund_matches$home_team_goal)

dortmund_matches$match_outcome <- ifelse(dortmund_matches$goal_diff > 0,
                                         'win',
                                         ifelse(dortmund_matches$goal_diff < 0,
                                                'lose',
                                                'tie'))
```

Preparing and Merging team attributes dataset:

The team attributes dataset contains information pertaining to the team tactics employed in a season. This data is merged with the matches dataset to get all data at a match level thus facilitating our analysis match-wise.

Extracting overall rating for each player:

Each player has fifa ratings on various metrics describing the player's gameplay. These metrics are helpful in determining the overall team strength and composition. So we subset the data of Dortmund players and get their respective ratings along with their overall rating.

The player's ratings are updated every three months. We calculate a average number to represent Dortmund and its opponent scores for each match. The score is changing with every update so we will have precise scores for these teams at that time.

```
player_overall_rating <- player_attributes %>%
  subset(player_api_id %in% unique_players_ids,
         c(id, player_api_id, date, overall_rating))

player_overall_rating <- player_overall_rating %>%
  dplyr::rename(year = date)
player_overall_rating$year <- substr(player_overall_rating$year, 0, 4)

player_overall_rating_aggregate = aggregate(overall_rating ~ player_api_id + year,
                                            data = player_overall_rating,
                                            mean)
```

Merging Player scores:

We now merge these player scores to our match dataset as 'overall team score'.

```
# melting data
dortmund_matches_melt <- dortmund_matches %>%
  subset(select = c(year, match_api_id, home_player_1:away_player_11)) %>%
  melt(id = c('match_api_id', 'year'))

# merging data
```

```
dummy_merge = merge(dortmund_matches_melt,
                    player_overall_rating_aggregate,
                    by.x = c('value', 'year'),
                    by.y = c('player_api_id', 'year'),
                    all.x = TRUE)

# casting data
dortmund_matches_player_ratings <- dcast(data = dummy_merge,
                                         formula = match_api_id + year ~ variable,
                                         value.var = 'overall_rating')
```

Creating a column for opponent team strength:

We want to classify opoonent teams as strong, equal and weak on a match by match basis by their overall team rating.

strong_opponent - Overall rating of opponent team $>$ 2.5 than Dortmund

equal_opponent - Overall rating of opponent team is in between +2.5 to -2.5 of Dortmund

weak - Overall rating of opponent team $<$ 2.5 than Dortmund

We choose these intervals as Fifa generally gives ratings to the players such that teams' main playing XI have relatively similar scores. For example, the overall team rating for 2 of the top clubs in the world, Barcelona and Real Madrid, have an overall team rating of 344 and 343 respectively. So we give a buffer of 2.5 points for classifying teams as same 'level'. Source: https://www.fifaindex.com/teams/fifa19/

```
dortmund_matches$rating_diff <- ifelse(dortmund_matches$home_team_api_id == 9789,
                                       dortmund_matches$home_team_overall_rating -
                                         dortmund_matches$away_team_overall_rating,
                                       dortmund_matches$away_team_overall_rating -
                                         dortmund_matches$home_team_overall_rating)

dortmund_matches$opponent_strength <- ifelse(dortmund_matches$rating_diff > 2.5,
                                             'weak_opponent',
                                             ifelse(dortmund_matches$rating_diff < -2.5,
                                                    'strong_opponent',
                                                    'equal_opponent'))
```

Extracting data from possesion column:

We had ball possession data available for 60% of the matches in xml format. We extracted that data in order to see if ball possession in a match has any effect on the match outcome.

```
home_matches_possession <- dortmund_matches %>%
        filter(home_or_away == 'home') %>%
        select(match_unique_id, possession) %>%
        mutate(possession_value = stri_extract_first_regex(possession,"[0-9]+")) %>%
        select(-possession) %>%
        filter(!is.na(possession_value))

home_matches_possession$possession_value <-
  as.numeric(as.character(home_matches_possession$possession_value))
```

```
away_matches_possession <- dortmund_matches %>%
        filter(home_or_away == 'away') %>%
        select(match_unique_id, possession) %>%
        mutate(possession_value = stri_extract_first_regex(possession,"[0-9]+")) %>%
        select(-possession) %>%
        filter(!is.na(possession_value))

away_matches_possession$possession_value <-
  as.numeric(as.character(away_matches_possession$poss))
away_matches_possession$possession_value <-
  abs(away_matches_possession$possession_value - 100)
possession <- rbind(home_matches_possession, away_matches_possession)

dortmund_matches <- merge(x = dortmund_matches,
                          y = possession,
                          all.x = TRUE)
```

Below we categorize our possession information as more, less or same.

more_possession - when our ball possession is more than 55%

less_possession - when our ball possession is less than 45%

same_possession - when our ball possession is in between 45% to 55%

```
dortmund_matches$possession_category <-
  ifelse(dortmund_matches$possession_value > 55,
         'more_possession',
         ifelse(dortmund_matches$possession_value > 45,
                'less_possession',
                'same_possession'))

dortmund_possession <-
  dortmund_matches[,c('home_or_away', 'match_outcome',
                      'opponent_strength', 'possession_category')]
```

As a final step, we also created a subset of the matches dataset to represnt Dortmund and opponents team attributes and used it for running association rules.

A structure of the final matches dataset is shown below -

```
str(dortmund_matches, strict.width = "wrap")
```

```
## 'data.frame':    272 obs. of  47 variables:
## $ match_unique_id : int 7810 7824 7829 7836 7846 7862 7864 7876 7883 7891
##    ...
## $ match_api_id : int 499318 499404 499409 499416 499426 499442 499444
##    499456 499463 499471 ...
## $ away_team_api_id : int 9789 9789 9911 9789 9810 9789 8721 9789 9788 8178
##    ...
## $ year : chr "2008" "2008" "2008" "2008" ...
## $ home_team_api_id : int 8178 8722 9789 9790 9789 8295 9789 9912 9789 9789
##    ...
## $ season : chr "2008/2009" "2008/2009" "2008/2009" "2008/2009" ...
```

```
## $ stage : int 1 10 11 12 13 14 15 16 17 18 ...
## $ date : chr "2008-08-16 00:00:00" "2008-10-29 00:00:00" "2008-11-02
##    00:00:00" "2008-11-08 00:00:00" ...
## $ home_team_goal : int 2 0 1 2 4 0 0 0 2 1 ...
## $ away_team_goal : int 3 1 1 1 0 1 0 0 1 1 ...
## $ goal : chr
##    "<goal><value><comment>n</comment><stats><goals>1</goals><shoton>1</s"..
##    __truncated__
##    "<goal><value><comment>n</comment><stats><goals>1</goals><shoton>1</s"..
##    __truncated__
##    "<goal><value><comment>n</comment><stats><goals>1</goals><shoton>1</s"..
##    __truncated__
##    "<goal><value><comment>n</comment><stats><goals>1</goals><shoton>1</s"..
##    __truncated__ ...
## $ shoton : chr "<shoton />" "<shoton />"
##    "<shoton><value><stats><shoton>1</shoton></stats><event_incident_type"..
##    __truncated__
##    "<shoton><value><stats><shoton>1</shoton></stats><event_incident_type"..
##    __truncated__ ...
## $ shotoff : chr "<shotoff />" "<shotoff />"
##    "<shotoff><value><stats><shotoff>1</shotoff></stats><event_incident_t"..
##    __truncated__
##    "<shotoff><value><stats><shotoff>1</shotoff></stats><event_incident_t"..
##    __truncated__ ...
## $ foulcommit : chr "<foulcommit />" "<foulcommit />"
##    "<foulcommit><value><stats><foulscommitted>1</foulscommitted></stats>"..
##    __truncated__
##    "<foulcommit><value><stats><foulscommitted>1</foulscommitted></stats>"..
##    __truncated__ ...
## $ card : chr
##    "<card><value><comment>y</comment><stats><ycards>1</ycards></stats><e"..
##    __truncated__
##    "<card><value><comment>y</comment><stats><ycards>1</ycards></stats><e"..
##    __truncated__
##    "<card><value><comment>y</comment><stats><ycards>1</ycards></stats><e"..
##    __truncated__
##    "<card><value><comment>y</comment><stats><ycards>1</ycards></stats><e"..
##    __truncated__ ...
## $ cross : chr "<cross />" "<cross />"
##    "<cross><value><stats><corners>1</corners></stats><event_incident_typ"..
##    __truncated__
##    "<cross><value><stats><crosses>1</crosses></stats><event_incident_typ"..
##    __truncated__ ...
## $ corner : chr "<corner />" "<corner />"
##    "<corner><value><stats><corners>1</corners></stats><event_incident_ty"..
##    __truncated__
##    "<corner><value><stats><corners>1</corners></stats><event_incident_ty"..
##    __truncated__ ...
## $ possession : chr "<possession />" "<possession />"
##    "<possession><value><comment>68</comment><event_incident_typefk>352</"..
##    __truncated__
##    "<possession><value><comment>50</comment><event_incident_typefk>352</"..
##    __truncated__ ...
## $ home_or_away : chr "away" "away" "home" "away" ...
```

```
## $ goal_diff : int 1 1 0 -1 4 1 0 0 1 0 ...
## $ match_outcome : chr "win" "win" "tie" "lose" ...
## $ home_team_buildUpPlaySpeedClass : chr NA NA NA NA ...
## $ home_team_buildUpPlayPassingClass : chr NA NA NA NA ...
## $ home_team_chanceCreationPassingClass : chr NA NA NA NA ...
## $ home_team_chanceCreationCrossingClass : chr NA NA NA NA ...
## $ home_team_chanceCreationPositioningClass: chr NA NA NA NA ...
## $ home_team_defencePressureClass : chr NA NA NA NA ...
## $ home_team_defenceAggressionClass : chr NA NA NA NA ...
## $ home_team_defenceDefenderLineClass : chr NA NA NA NA ...
## $ home_team_buildUpPlayDribblingClass : chr NA NA NA NA ...
## $ home_team_chanceCreationShootingClass : chr NA NA NA NA ...
## $ away_team_buildUpPlaySpeedClass : chr NA NA NA NA ...
## $ away_team_buildUpPlayPassingClass : chr NA NA NA NA ...
## $ away_team_chanceCreationPassingClass : chr NA NA NA NA ...
## $ away_team_chanceCreationCrossingClass : chr NA NA NA NA ...
## $ away_team_chanceCreationPositioningClass: chr NA NA NA NA ...
## $ away_team_defencePressureClass : chr NA NA NA NA ...
## $ away_team_defenceAggressionClass : chr NA NA NA NA ...
## $ away_team_defenceDefenderLineClass : chr NA NA NA NA ...
## $ away_team_buildUpPlayDribblingClass : chr NA NA NA NA ...
## $ away_team_chanceCreationShootingClass : chr NA NA NA NA ...
## $ home_team_overall_rating : num 75.3 70.2 75.4 75.9 75 ...
## $ away_team_overall_rating : num 74.8 74.7 71.8 75.6 70.2 ...
## $ rating_diff : num -0.455 4.5 3.614 -0.273 4.8 ...
## $ opponent_strength : chr "equal_opponent" "weak_opponent" "weak_opponent"
##    "equal_opponent" ...
## $ possession_value : num NA NA 68 50 NA 55 NA NA NA 45 ...
## $ possession_category : chr NA NA "more_possession" "less_possession" ...
```

Exploratory Data Analysis

We start by looking at Dortmund's ranking over the past couple of years. Table below shows that Dortmund is one of the top teams in the German league. Actually, if we look at the table below, we can conclude that Dortmund is considered the second best team in German League. In order to win the championship (get rank 1) within German league, a team needs to score the highest number of points. Points are allotted to a team based on match outcomes - a win gives 3 points, tie gives 1 point, while lost match has 0 points. In case of a tie in the number of points, Goal Difference is considered to decide rank. Goal Difference is the difference between Goals scored and Goals conceded.

Season	Rank	Club	Points	Goal Difference
2015-16	1	Bayern	88	63
	2	**Dortmund**	78	48
	3	Bayer	60	16
	4	Monchengladbach	55	17
	5	Schalke 04	52	2
	6	Mainz	50	4
	7	Hertha	50	0
2014-15	1	Bayern	79	62
	2	Wolfsburg	69	34
	3	Monchengladbach	66	27
	4	Bayer	61	25
	5	Augsburg	49	0
	6	Schalke 04	48	2
	7	**Dortmund**	46	5
2013-14	1	Bayern	71	90
	2	**Dortmund**	71	42
	3	Schalke 04	64	20
	4	Bayer	61	19
	5	Wolfsburg	60	13
	6	Monchengladbach	55	16
	7	Mainz	53	-2
2012-13	1	Bayern	91	80
	2	**Dortmund**	66	39
	3	Bayer	65	26
	4	Schalke 04	55	8
	5	Freiburg	51	5
	6	Eintracht	51	3
	7	Hamburg	48	-11
2011-12	**1**	**Dortmund**	81	55
	2	Bayern	73	55
	3	Schalke 04	64	30
	4	Monchengladbach	60	25
	5	Bayer	54	34
	6	VfB Stuttgart	53	17
	7	Hannover	48	-4
2010-11	**1**	**Dortmund**	75	45
	2	Bayer	68	20
	3	Bayern	65	41
	4	Hannover	60	4
	5	Mainz	58	12
	6	Nurnberg	47	3
	7	Kaiserslautern	46	-3
2009-10	1	Bayern	70	41

Season	Rank	Club	Points	Goal Difference
	2	Schalke 04	65	22
	3	Werder Bremen	61	31
	4	Bayer	59	21
	5	**Dortmund**	57	12
	6	VfB Stuttgart	55	10
	7	Hamburg	52	15

Rank Source: https://www.espn.com/soccer/standings/_/league/ger.1/season/2009/german-bundesliga

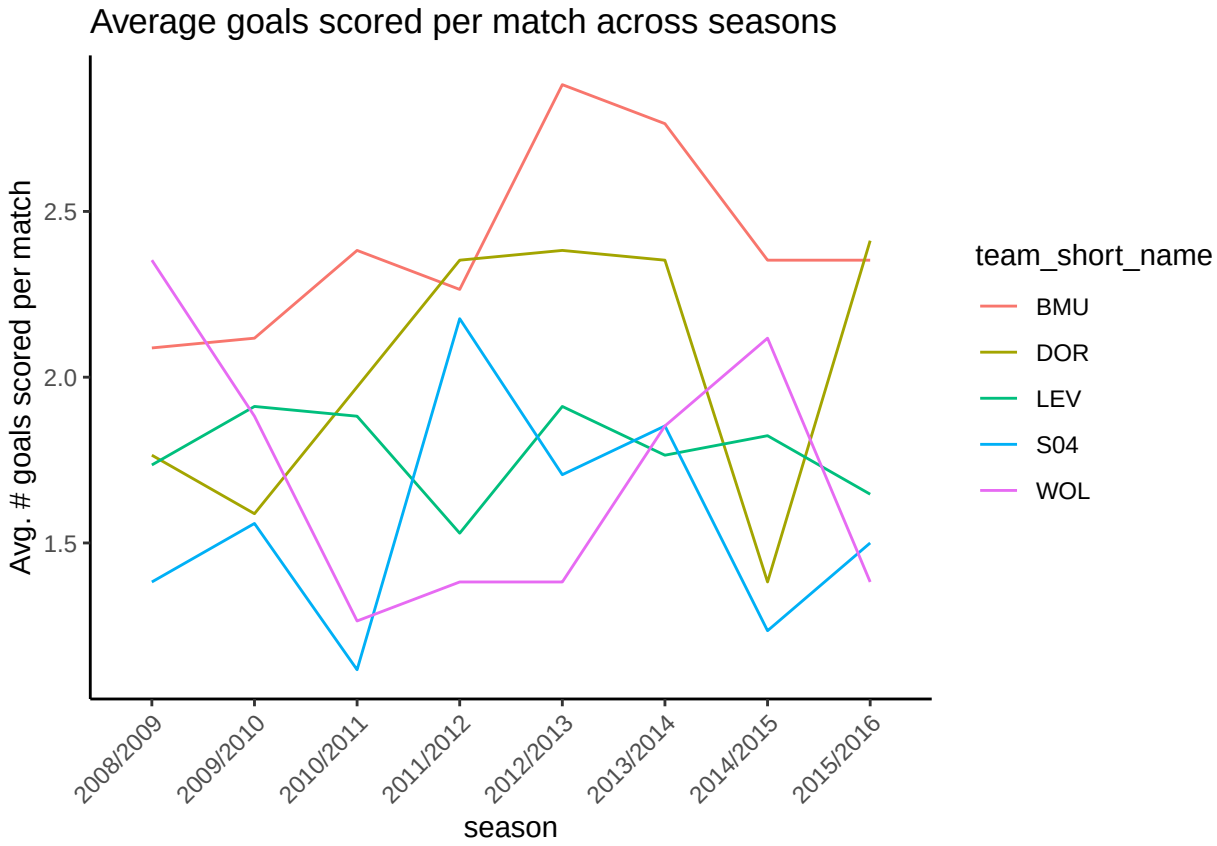
Interpretation:

We observe that although Dortmund is able to maintain rank in top 2 in most years, they have not won the rank 1 since past 4 years. To get rank 1 and win the championship, Dortmund would have to score points higher than FC Bayern (which has been the champion in most seasons). The difference in points in the last 2 seasons has been of 10 points. So winning about even 4 more games would help Dortmund in winning the German league championship or at least end up in the top 3.

To win more games, Dortmund either needs to score more goals (have a good attack) or concede less goals (have a good defence). Let's have a look at their goals statistics:

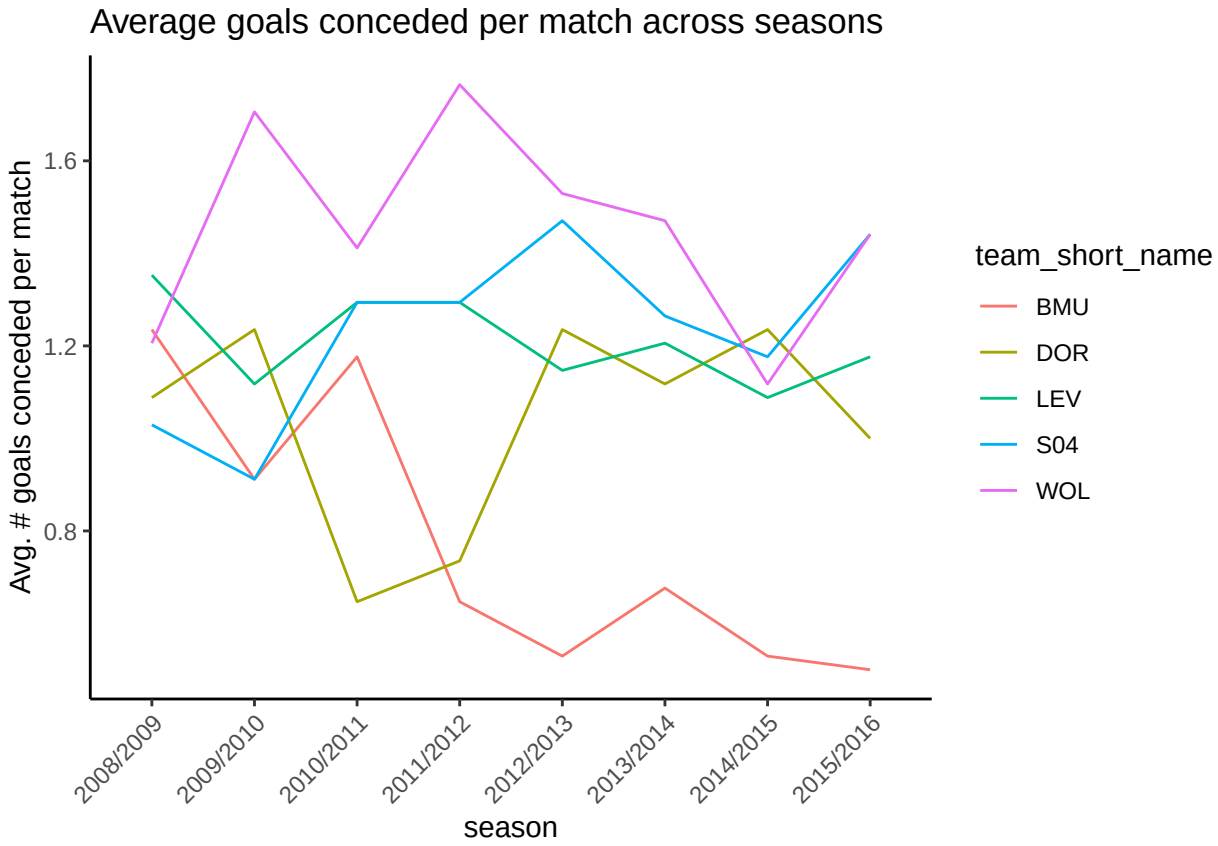
```
plot_ger_team_sgoal <- ggplot(top5_ger_season_stats,
                              aes(x = season,
                                  y = tot_scored / matches,
                                  colour = team_short_name,
                                  group = team_short_name)) +
  geom_line() +
  ylab("Avg. # goals scored per match") +
  ggtitle('Average goals scored per match across seasons') +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

plot_ger_team_sgoal
```




```
plot_ger_team_cgoal <- ggplot(top5_ger_season_stats,
                              aes(x = season,
                                  y = tot_conceded / matches,
                                  colour = team_short_name,
                                  group = team_short_name)) +
  geom_line() +
  ylab("Avg. # goals conceded per match") +
  ggtitle('Average goals conceded per match across seasons') +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

plot_ger_team_cgoal
```



Historically, all German teams have been improving, by both scoring more goals and conceding fewer goals than previous seasons.

Over time, *Dortmund* has improved their goal scoring capabilities from ~1.75 to ~2.4, but their defence seems to be fairly constant with an average of about 1. The graph above shows that conceded goals have increased. On the other hand, their most popular rival, *Bayern Munich* significantly improved their defense while maintaining strong attacking capabilities. *Bayern's* average goals scored only improved marginally from 2.19 to 2.35, but the average number of goals conceded has reduced significantly from 1.24 to 0.5.

A quick look at Dortmund's past strategy tells us that this constant goals concede might be because of no change in their defense strategy since 5 seasons.

```
test <- team_atts %>% filter(team_api_id == 9789) %>% subset(select = c(year, defencePressureClass, def
test
```

```
##   year defencePressureClass defenceAggressionClass
## 1 2010               Medium                 Double
## 2 2011               Medium                 Double
## 3 2012               Medium                 Double
## 4 2013               Medium                 Double
## 5 2014               Medium                 Double
## 6 2015               Medium                 Double
##   defenceDefenderLineClass
## 1              Offside Trap
## 2                     Cover
## 3                     Cover
## 4                     Cover
## 5                     Cover
```

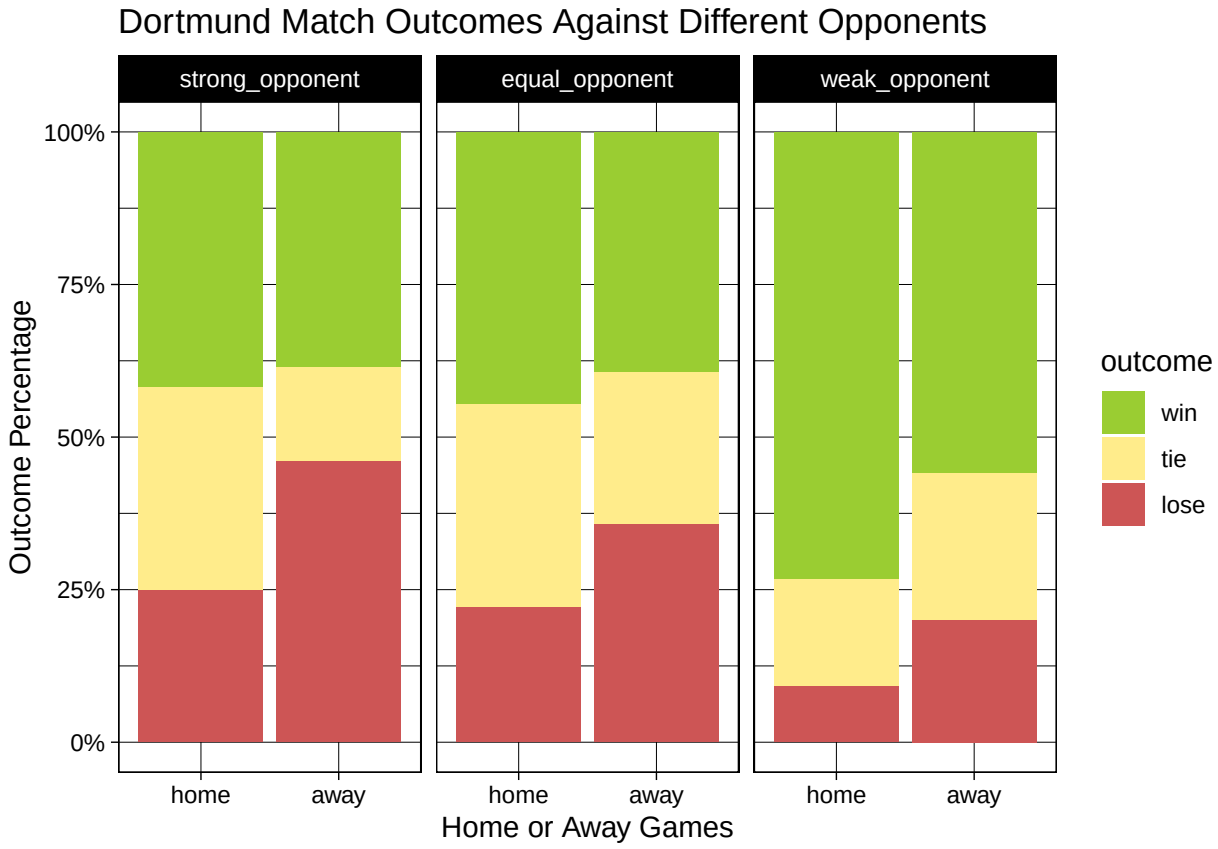
```
## 6                    Cover
```

Interpretation:

This is a clear indication that in order to improve their defense performance, Dortmund should try different strategies against different set of opponents to conceded minimum goals possible.

Now we will deep dive to understand how is Dortmund's overall match performance with respect to its different types of opponents (strong, equal and weak).

```
ggplot(data = Dor_outcome, aes(x = home_or_away, y = pct, fill = match_outcome)) +
  geom_bar(stat = "identity", position = 'fill') +
  facet_grid(. ~ opponent_strength) +
  ggtitle("Dortmund Match Outcomes Against Different Opponents") +
  theme_linedraw() +
  ylab('Outcome Percentage') +
  scale_y_continuous(labels = scales::percent_format(accuracy = 1)) +
  xlab('Home or Away Games') +
  scale_fill_manual('outcome', values = c("olivedrab3","lightgoldenrod1","indianred3")) +
  # scale_color_manual(values = c(NA,"yellow")) +
  guides(colour = FALSE)
```



Interpretation:

From the above plot we see that Dortmund's performance varies depending on level of opponent and whether they are playing at home or as an away team. Over 8 years, Dortmund played 272 games in total. Since Dortmund is a strong team, in most of the matches (192 matches) the opponent team is categorized as 'weak_opponent'. In 55 matches, their opponent team is categorized as 'equal_opponent' and in 25 matches Dortmund played against 'strong_opponent'.

We can observe two things in the plot - first, as expected, Dortmund has the highest winning rate agaist 'weak_opponents', followed by 'equal_opponents', and lastly 'strong_opponents'. Among 192 games against 'weak' teams, Dortmund won 124 (65% winning rate). However, Dortmund only won 10 out of 25 games(40% winning rate) when playing against 'strong team'. Second, Dortmund's performance is better when they play home game compared to away games. For instance, they have a 73% winning rate when played against 'weak_opponent' at home and only 56% winning rate when playing in away matches.

Conclusions:

Since Dortmund's performance depends on their opponents' level and whether they are playing at home or not, coach should adjust the team's strategy according to these circumstances. So, we seperate Dortmund's matches into six categories:

1. home games -> against weak opponent
2. away games -> against weak opponent
3. home games -> against equal opponent
4. away games -> against equal opponent
5. home games -> against strong opponent
6. away games -> against strong opponent

Then we analyze what strategy works best in each category by using association rules.

Association Rules

Effect of Ball Possession on match outcome

We will first analyze ball possession and its impact on match outcome.

Explanation

Possession is a crucial factor in a soccer game. Possession means the percentage of time a team has a ball under its control. For instance, if Dortmund has a possession of 60 in a game, it means 60% of time the ball is under the control of Dortmund's players. Possession can reflect the strategy of a team. If a team has a high possession in a game, it means the team's strategy is to keep attacking opponent. However, having the ball under control means the player will lose more energy and the defense will be weak since most players need to participate in offense part and ignore the defense task. On the other end, giving up the ball control means a team is trying to save the energy and utilize the counterattack chance to destroy the opponent at once.

In short, possession reflects a team's basic strategy in a game. We categorize our possession value so that if we have more than 55% possession, then we conclude we have more possession than the opponent. If our possession is between 45% and 55%, we conclude that both teams have equal possession. If our possession is under 45%, we conclude that our possession is less than opponent.

We have calculated our winning rates according to above mentioned 6 categories i.e. if we are playing against strong/equal/weak team at home/away. We will analyze each category individually. For example, our winning rate for weak team/home game is 0.73, we want to see the rule with confidence over 0.73 with 'win' at the right hand side because this can show us only those useful strategy which can improve our performance.

Loading the data

Out of 272 games, 152 games have the data for possession. We can see the count of each label and sample of the data.

```
setwd("~/Desktop/Homework 1")

soccer = read.transactions("dortmund_possession_arules.csv",
                           format = "basket",
                           sep = ",",
                           rm.duplicates = TRUE)

itemFrequency(soccer, type = "absolute")
```

```
##            away  equal_opponent            home less_possession
##              75              37              77              59
##            lose more_possession same_possession strong_opponent
##              33              69              24              17
##             tie   weak_opponent             win
##              33              98              86
```

```
temp <- read.csv("dortmund_possession_arules.csv",
                 header = FALSE)
head(temp)
```

```
##     V1   V2             V3             V4
## 1 home  tie   weak_opponent more_possession
```

```
## 2 away lose  equal_opponent less_possession
## 3 away  win   weak_opponent less_possession
## 4 home  tie strong_opponent same_possession
## 5 away lose strong_opponent same_possession
## 6 home  tie strong_opponent same_possession
```

Analyzing home games against 'weak_opponents'

Since the winning rate right now is 0.73, we want to generate rules that give us a winning rate greater than 0.73. From the result, we can see that if we keep our possession the same as our opponent, we can actually increase our winning chances by 43%.

```
rules <- apriori(soccer, parameter = list(supp = 0.03, conf = 0.1))
```

```
rules <- sort(rules, by = "confidence", decreasing = TRUE)

rules %>%
  subset(subset = (rhs %pin% "win")) %>%
  subset(subset = confidence > 0.72) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'weak_opponent')) %>%
  inspect()
```

```
##     lhs                                      rhs   support    confidence
## [1] {home,less_possession,weak_opponent} => {win} 0.08552632 0.8125
##     lift     count
## [1] 1.436047 13
```

Analyzing away games against 'weak_opponents'

We can see that right now our winning rate is 0.56. If we play with same possession, we will improve our winning chances by 6%. However, the count is only 3 in this case.

```
rules <- apriori(soccer, parameter = list(supp = 0.01, conf = 0.1))
```

```
rules <- sort(rules, by = "confidence", decreasing = TRUE)

rules %>%
  subset(subset = (rhs %pin% "win")) %>%
  subset(subset = confidence > 0.56) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'weak_opponent')) %>%
  inspect()
```

```
##     lhs                                      rhs   support    confidence
## [1] {away,same_possession,weak_opponent} => {win} 0.01973684 0.6
##     lift     count
## [1] 1.060465 3
```

Analyzing home games against 'strong_opponents'

We can see that right now our winning rate is 0.42. From our dataset we can see that if we play with less possesion, our winning rate can go up by 6%.

```
rules <- apriori(soccer, parameter = list(supp = 0.01, conf = 0.1))
```

```
rules <- sort(rules, by = "confidence", decreasing = TRUE)

rules %>%
  subset(subset = (rhs %pin% "win")) %>%
  subset(subset = confidence > 0.42) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'strong_opponent')) %>%
  inspect()
```

```
##     lhs                                          rhs   support    confidence
## [1] {home,less_possession,strong_opponent} => {win} 0.01973684 0.6
## [2] {home,strong_opponent}                 => {win} 0.02631579 0.5
##     lift      count
## [1] 1.0604651 3
## [2] 0.8837209 4
```

Analyzing away games against 'strong_opponents'

We could not find any patterns to improve our win rates for away matches against strong teams as we don't have enough past data to refer to. However, interestingly, we find that when we play with less possession, our losing rate will increase three folds. Hence, we should avoid playing with less possession.

```
rules <- apriori(soccer, parameter = list(supp = 0.01, conf = 0.1))
```

```
rules <- sort(rules, by = "confidence", decreasing = TRUE)

rules %>%
  subset(subset = (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = confidence > 0.38) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'strong_opponent')) %>%
  inspect()
```

```
##     lhs                                          rhs    support    confidence
## [1] {away,less_possession,strong_opponent} => {lose} 0.01315789 0.6666667
## [2] {away,strong_opponent}                 => {lose} 0.03289474 0.5555556
## [3] {away,same_possession,strong_opponent} => {lose} 0.01973684 0.5000000
##     lift     count
## [1] 3.070707 2
## [2] 2.558923 5
## [3] 2.303030 3
```

Analyzing home games against 'equal_opponents'

We can see that right now our winning rate is 0.44. From our dataset we can see that if we play with more possesion, our winning rate can go up to 0.625. When we play with same possession, winning rate is 0.5, slightly higher than 0.44. We should never play with less possession, since this will give us a losing rate of 0.5, higher than original losing rate of 0.22.

```
rules <- apriori(soccer, parameter = list(supp = 0.01, conf = 0.1))
```

```
rules <- sort(rules, by = "confidence", decreasing = TRUE)

rules %>%
  subset(subset = (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = confidence > 0.44) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'equal_opponent')) %>%
  inspect()
```

```
##     lhs                                          rhs    support    confidence
## [1] {equal_opponent,home,more_possession} => {win}  0.03289474 0.6250
## [2] {equal_opponent,home}                 => {win}  0.05921053 0.5625
## [3] {equal_opponent,home,same_possession} => {win}  0.01315789 0.5000
## [4] {equal_opponent,home,less_possession} => {lose} 0.01315789 0.5000
## [5] {equal_opponent,home,less_possession} => {win}  0.01315789 0.5000
##     lift      count
## [1] 1.1046512 5
## [2] 0.9941860 9
## [3] 0.8837209 2
## [4] 2.3030303 2
## [5] 0.8837209 2
```

Analyzing away games against 'equal_opponents'

We can see that right now our winning rate is 0.39. From our dataset we can see that if we play with more possesion, our winning rate can go up to 1. If we play with same possession, winning rate will also increase to 0.5. If we play with less possession, losing rate will increase from 0.36 to 0.4.

```
rules <- apriori(soccer, parameter = list(supp = 0.01, conf = 0.1))
```

```
rules <- sort(rules, by = "confidence", decreasing = TRUE)

rules %>%
  subset(subset = (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = confidence > 0.39) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'equal_opponent')) %>%
  inspect()
```

```
##     lhs                                          rhs    support    confidence
## [1] {away,equal_opponent,more_possession} => {win}  0.01315789 1.0000000
## [2] {away,equal_opponent,same_possession} => {lose} 0.01315789 0.5000000
## [3] {away,equal_opponent}                 => {win}  0.05921053 0.4285714
## [4] {away,equal_opponent,less_possession} => {lose} 0.03947368 0.4000000
## [5] {away,equal_opponent,less_possession} => {win}  0.03947368 0.4000000
##     lift      count
## [1] 1.7674419 2
## [2] 2.3030303 2
## [3] 0.7574751 9
## [4] 1.8424242 6
## [5] 0.7069767 6
```

Conclusion

Based on above analysis, we can provide several rules for our team's coach:

1. home games and against weak opponent - have less possession
2. away games and against weak opponent - have same possession
3. home games and against equal opponent - 1st choice: more possession. 2nd choice: same possession, do not play with less possession.
4. away games and against equal opponent - more possession, do not play with same possession
5. home games and against strong opponent - haveless possession
6. away games and against strong opponent - do not play with less possession

Effect of Team Attributes on match outcome

Next we analyze team attributes and their impact on match outcome.

Explanation

Now we will use team attributes data to generate rules to develop strategies for our team's coach. Team attribution table contains the strategy which a team uses and it is updated every season. We combined eight features into every game for both Dortmund and opponent's team. Some of the features are buildup passing class, chance crossing class, and defense pressure class. For example, if the passing class is 'long', it means Dortmund is focusing on long passing in the game. If the passing class is 'mixed', it means Dortmund will use both short passes and long passes in a game. If chance creation crossing class is 'normal', it means Dortmund is going to play normally. But when chance creation crossing class is 'risky', it means Dortmund will try to make some risky crosses even though it might lead to lost of possession or even conceding a goal. We will explain what each term mean in the following section.

Terms

1. Play Speed

Slow: Team plays a slow pace game

Balanced: Team plays with a balanced pace game

Fast: Team plays with a fast pace game

2. Play Passing

Short: Team focuses on short passing

Mixed: Team does both short passing and long passing

Long: Team focuses on long passing

3. Chance creation passing

Safe: Teams plays safe when there is a chance

Normal: Team plays normally when there is a chance

Risky: Team plays with risks when there is a chance

4. Chance creation crossing

Lots: Team tends to try lots of cross passings

Little: Team tends to try little cross passings

5. Chance creation positioning

Organised: Team asks players to play by plan

Free Form: Team allows player to play with improvise

6. Defense Pressure

Deep: Team focuses less on defense

Medium: Team is average on defense

High: Team focuses a lot on defense

7. Defense Aggression

Contain: Team is conservative and does not want to commit fouls on defense

Press: Team give pressure on defense

Double: Team will double the offensive player on defense

8. Defender line class

Cover: Team plays with normal defense strategy

Offside Trap: Team sets up off side traps on defense

We added 'Dortmund' and 'Opponent' before each term to distinguish between them.

Approaches

We have 19 columns in our table and we want our rule as specific as possible, so that we need to adjust parameter accordingly for each of six analysis. For example, when we analyze playing against weak team at home, if we set our minimum length to 16, we will generate 8341 rules. That many rules are unnecessary. If we set our minimum length to 17, we will have 826 rules this time. After filtering the condition, we will have a total of 16 applicable rules, which are adequate. Also, we need to adjust our support each time. Because Dortmund plays a lot of home game against weak team, we can set up our support high for that category. However, since Dortmund only plays a few home games against equal team (12 games), we need to lower our support. Also, we need to filter by confidence each time to make sure we have only those rules that can increase our winning rate.

Loading the data

Out of 272 games, 204 games have the data for team attribution.

```
setwd("~/Desktop/Homework 1")
team = read.transactions("dortmund_dataset_arule.csv",
                         format = "basket",
                         sep = ",",
                         rm.duplicates = TRUE)
```

Analyzing home games against 'weak_opponents' for win

Since the winning rate right now is 0.73, we want to generate rules that gives us a winning rate greater than 0.74.

```
rules1 <- apriori(team,parameter = list(supp = 0.03, conf = 0.33, maxlen = 20, minlen = 7))
```

```
rules1 <- sort(rules1,by = "confidence",decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "win")) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'weak_opponent')) %>%
  subset(subset = confidence > 0.73)
inspect(rules1[1:3])
```

```
##     lhs                                rhs      support confidence     lift count
## [1] {Dortmund_Normal_Shooting,
##      Dortmund_Risky_Passing,
##      home,
##      Opponent_Normal_Passing,
##      Opponent_Normal_Shooting,
##      weak_opponent}                 => {win} 0.06862745  0.9333333 1.511111    14
## [2] {Dortmund_Free Form_Positioning,
##      Dortmund_Normal_Shooting,
##      home,
##      Opponent_Normal_Passing,
##      Opponent_Normal_Shooting,
##      weak_opponent}                 => {win} 0.06862745  0.9333333 1.511111    14
## [3] {Dortmund_Free Form_Positioning,
##      Dortmund_Normal_Shooting,
##      Dortmund_Risky_Passing,
##      home,
##      Opponent_Normal_Passing,
##      Opponent_Normal_Shooting,
##      weak_opponent}                 => {win} 0.06862745  0.9333333 1.511111    14
```

We see that in our top 3 rules which have confidence greater than 80% and a high lift, Dortmund wins most of their matches against weak opponents at home primarily based on 3 characteristics:

1. The positioning of their players is free form while creating chances. This could mean that when Dortmund creates chances based on the flow of the game (eg: match situation) and not their structured passing routines, they have a better chance of winning matches against the weaker opposition.
2. The 'chanceCreationPassingColumn' is 'Risky' which implies that they create chances that might be risky through passing. So this could inclue long passes, lob passes, and short passes in a tight space. This style of play provides Dortmund with more wins.
3. The 'chanceCreationShootingColumn' is normal indicating their shooting has been normal and that is is good enough to win the matches.

Now that we have insight into our strengths against weak opponents, let's see what factors cost us matches at home against these type of opponents. We generate rules to see which items on the left hand side co-occur together to give us losses which comes up on the right hand side of the rule.

Analyzing home games against 'weak_opponents' for lose

```
rules1 <- apriori(team,parameter = list(supp = 0.03, conf = 0.093, maxlen = 20, minlen = 1))
```

```
rules1 <- sort(rules1,by = "confidence",decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "lose")) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'weak_opponent')) %>%
  subset(subset = confidence > 0.11)
inspect(rules1[1:3])
```

```
##     lhs                                rhs       support confidence      lift count
## [1] {Dortmund_Lots_Shooting,
##      home,
```

```
##      Opponent_Normal_Shooting,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {lose} 0.03431373  0.2000000 1.0461538     7
## [2] {Dortmund_Lots_Shooting,
##      home,
##      Opponent_Normal_Passing,
##      weak_opponent}                  => {lose} 0.03431373  0.1842105 0.9635628     7
## [3] {Dortmund_Lots_Shooting,
##      home,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {lose} 0.03431373  0.1842105 0.9635628     7
```

It is evident from the sample size of 19 matches for loses over 6 seasons at home that Dortmund performs really well against weak teams. They should capitalize on maximizing this home advantage to convert these losses into wins to get those extra points and climb up in the league standings.

The rule with the most lift informs us that when the opponents' chance creation positioning class is organized and the chance creation passing class is normal, Dortmund tends to play with a lot of crosses. This could mean that Dortmund creates opportunities by taking a lot of shots but they do not get converted to goals. The accuracy of shots might not be good and ensuring conversion shots to goals is essential to avoid losses.

We move our attention to the ties games at home.

Analyzing home games against 'weak_opponents' for tie

```
rules1 <- apriori(team,parameter = list(supp = 0.03, conf = 0.15, maxlen = 20, minlen = 3))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "tie")) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'weak_opponent')) %>%
  subset(subset = confidence > 0.15)
inspect(rules1[1:3])
```

```
##     lhs                                rhs      support confidence      lift count
## [1] {Dortmund_Free Form_Positioning,
##      home,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {tie} 0.03921569  0.1666667 0.8717949     8
## [2] {home,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {tie} 0.05882353  0.1643836 0.8598525    12
## [3] {Dortmund_Risky_Passing,
##      home,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {tie} 0.03921569  0.1632653 0.8540031     8
```

Since the lift is not very high for any of these rules, we do not have any conclusive evidence. So we cannot tell what exactly contributed to the losses for the tied games.

Analyzing away games against 'weak_opponents'

So far, we have looked at matches played at home with weaker opposition. We need to analyze how Dortmund fares in away matches.

We can see that right now our winning rate is 0.55. We want to have every rule that has a winning rate over 0.55.

```
rules1 <- apriori(team,parameter = list(supp = 0.03, conf = 0.55, maxlen = 20, minlen = 7))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)

rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "win")) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'weak_opponent')) %>%
  subset(subset = confidence > 0.56)
inspect(rules1[1:3])
```

```
##     lhs                                  rhs      support confidence     lift count
## [1] {away,
##      Dortmund_Normal_Passing,
##      Dortmund_Normal_Shooting,
##      Dortmund_Organised_Positioning,
##      Opponent_Normal_Shooting,
##      weak_opponent}                  => {win} 0.03921569  0.7272727 1.177489     8
## [2] {away,
##      Dortmund_Normal_Passing,
##      Dortmund_Organised_Positioning,
##      Opponent_Normal_Passing,
##      Opponent_Normal_Shooting,
##      weak_opponent}                  => {win} 0.03921569  0.7272727 1.177489     8
## [3] {away,
##      Dortmund_Normal_Passing,
##      Dortmund_Organised_Positioning,
##      Opponent_Normal_Shooting,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {win} 0.03921569  0.7272727 1.177489     8
```

Interpretation

We can inspect the top three rules we generate here. We can see a huge increase in winning rate for the first two rules. The winning rate increases from 0.55 to 0.78. Effectively, if Dortmund's opposition plays plays with an organized ball positioning, Dortmund's best chances of winning are when they play with risky passes where they throw the long balls and ensure quick short passes in tight spaces.

Analyzing away games against 'weak_opponents' for lose

```
rules1 <- apriori(team,parameter = list(supp = 0.03, conf = 0.15, maxlen = 20, minlen = 3))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "lose")) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'weak_opponent')) %>%
  subset(subset = confidence > 0.20)
inspect(rules1[1:3])
```

```
##     lhs                                  rhs       support confidence     lift count
## [1] {away,
##      Dortmund_Free Form_Positioning,
##      Dortmund_Lots_Shooting,
```

```
##      Opponent_Normal_Passing,
##      weak_opponent}                  => {lose} 0.03921569  0.3076923 1.609467     8
## [2] {away,
##      Dortmund_Free Form_Positioning,
##      Dortmund_Lots_Shooting,
##      Opponent_Normal_Passing,
##      Opponent_Normal_Shooting,
##      weak_opponent}                  => {lose} 0.03921569  0.3076923 1.609467     8
## [3] {away,
##      Dortmund_Normal_Passing,
##      Opponent_Normal_Passing,
##      Opponent_Normal_Shooting,
##      weak_opponent}                  => {lose} 0.03431373  0.2916667 1.525641     7
```

Here, what we see common amongst all the rules is that whenever Dortmund plays with lot of crosses in the game, they tend to lose. This was the case even with weaker opponents in the home games which led to their loss. Although the confidence isn't very high, it is slightly better than the base percentage and we can assume that the items that occur on the left hand side of the rule lead to losses.

Analyzing away games against 'weak_opponents' for tie

Dortmund ends up tieing a lot of matches which earn them just 1 point instead of 3 which can be scored with a win. They have 24 tied games in away matches across 6 seasons. There is scope for capitalizing on these matches to secure wins since the number of tied matches are very high. Converting even a few of these matches into wins can drastically alter the chances of getting the first spot in the league.

```
rules1 <- apriori(team,parameter = list(supp = 0.03, conf = 0.15, maxlen = 20, minlen = 3))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "tie")) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'weak_opponent')) %>%
  subset(subset = confidence > 0.24)
inspect(rules1[1:3])
```

```
##     lhs                                rhs      support confidence     lift count
## [1] {away,
##      Dortmund_Lots_Shooting,
##      Dortmund_Risky_Passing,
##      Opponent_Normal_Passing,
##      Opponent_Organised_Positioning,
##      weak_opponent}                  => {tie} 0.03431373  0.3043478 1.591973     7
## [2] {away,
##      Dortmund_Free Form_Positioning,
##      Dortmund_Lots_Shooting,
##      weak_opponent}                  => {tie} 0.04411765  0.3000000 1.569231     9
## [3] {away,
##      Dortmund_Free Form_Positioning,
##      Dortmund_Lots_Shooting,
##      Opponent_Normal_Shooting,
##      weak_opponent}                  => {tie} 0.04411765  0.3000000 1.569231     9
```

It is obvious from the rules that Dortmund takes a lot of shots at the goal and plays with the flow of the game rather than relying to much on pre-planned tactices. The passing between players is normal and not risky. Somehow they are failing to convert those shot to goals.

*Analyzing home games against 'equal_opponents' for win / lose**

We can see that right now our winning rate is 0.48. Any rule with a confidence over 0.48 could increase our performance. Since we have smaller sample size(15 matches over 6 seasons) this time, we decrease our support from 0.03 to 0.01 this time.

```
rules1 <- apriori(team,parameter = list(supp = 0.01, conf = 0.33,maxlen = 20,minlen = 6))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'equal_opponent')) %>%
  subset(subset = confidence > 0.47)
inspect(rules1[1:3])
```

```
##     lhs                                rhs      support confidence     lift count
## [1] {Dortmund_Normal_Shooting,
##      Dortmund_Risky_Passing,
##      equal_opponent,
##      home,
##      Opponent_Risky_Passing}         => {win} 0.01470588          1 1.619048     3
## [2] {Dortmund_Free Form_Positioning,
##      Dortmund_Normal_Shooting,
##      equal_opponent,
##      home,
##      Opponent_Risky_Passing}         => {win} 0.01470588          1 1.619048     3
## [3] {Dortmund_Normal_Shooting,
##      equal_opponent,
##      home,
##      Opponent_Organised_Positioning,
##      Opponent_Risky_Passing}         => {win} 0.01470588          1 1.619048     3
```

Interpretation

We can inspect the top three rules we generate here. We can see that with the right strategy, we will have a 100% winning rate. However, we know that in real life, it is impossible to guarantee that we can win game every time. 100% winning rate appears because our sample size is relatively small so that we can have a unexpected higher confidence value here. But these rules still indicate an increase. Hence, we should collect more information for these games to generate a more supportive conclusion.

Analyzing away games against 'equal_opponents' for win / lose

```
rules1 <- apriori(team,parameter = list(supp = 0.01, conf = 0.33,maxlen = 20,minlen = 6))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'equal_opponent')) %>%
  subset(subset = confidence > 0.37)
inspect(rules1[1:3])
```

```
##     lhs                                rhs      support confidence     lift count
## [1] {away,
```

```
##      Dortmund_Normal_Shooting,
##      equal_opponent,
##      Opponent_Lots_Shooting,
##      Opponent_Normal_Passing}        => {win} 0.01470588          1 1.619048     3
## [2] {away,
##      Dortmund_Risky_Passing,
##      equal_opponent,
##      Opponent_Lots_Shooting,
##      Opponent_Normal_Passing}        => {win} 0.01470588          1 1.619048     3
## [3] {away,
##      Dortmund_Free Form_Positioning,
##      equal_opponent,
##      Opponent_Lots_Shooting,
##      Opponent_Normal_Passing}        => {win} 0.01470588          1 1.619048     3
```

Similar to home games against equal_opponents, we have high confidence of 100% here with just a sample size of 3. We cannot rely on these rules and proceed further with our analysis.

Analyze home games against 'strong teams' for win / lose

```
rules1 <- apriori(team,parameter = list(supp = 0.01, conf = 0.33,maxlen = 20,minlen = 6))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = (lhs %pin% 'home' & lhs %pin% 'strong_opponent')) %>%
  subset(subset = confidence > 0.33)
# inspect(rules1[1:3])
```

Could not find any rules here

Analyze away games against 'strong teams' for win / lose

```
rules1 <- apriori(team,parameter = list(supp = 0.01, conf = 0.33,maxlen = 20,minlen = 6))
```

```
rules1 <- sort(rules1, by = "confidence", decreasing = TRUE)
rules1 <- rules1 %>%
  subset(subset =  (rhs %pin% "win" | rhs %pin% "lose")) %>%
  subset(subset = (lhs %pin% 'away' & lhs %pin% 'strong_opponent')) %>%
  subset(subset = confidence > 0.45)
# inspect(rules1[1:3])
```

Could not find any rules here

Conclusion

Findings

From our analysis above, based on team attribution table, we can conclude that this is a good approach to generate rules for each categories. However, since we do not have a large sample size for games agaist equal teams and strong teams, the rules work the best when Dortmund plays against weak teams because we have enough counts to generate a detailed plan.

The association rules provide us with insights for the strategies that are effective and not so effective for Dortmund against weak teams in both home and away games. Dortmund generally performs very well at home which is well known. The results slightly change when they play away games where their wins reduce and the number of ties and losses go up significantly as well.

These are areas that elude them of vital points that give them a shot at the top 3 spots of the league. One prominent feature resulting most of their games which end up in ties and losses in both home and away games are that even though they take a lot of shots, they fail to net the ball into the goal. It could be due to the extra pressure in the away conditions and poor finishing. This is a huge area for improvement. If Dortmund can convert even one of the many shots that they have at the goal, it could be the differentiating factor in converting a loss to a tie and changing a tie to a win.

The most import finding from our analysis is that strategies do impact game outcomes in a big way. When Dortmund plays with correct strategies, they can increase their winning rates significantly. By studying the ratings and features of opponents, Dortmund can adjust its plan accordingly and this will provide team with an advantage and a higher chance to win.

Recommendations

We believe that the only way for a club to get the top spot in the league is to score as many points as possible either through wins or ties. Given the fact that we have a dominant team FC Bayern Munich who always has much more budget than Dortmund and can easily sign the best players, it is difficult to beat them by forming a team with better skills than FC Bayern Munich. Instead, the best approach to achieve success is to pick the best strategy each game especially against weaker opponents since that is where we can get most points.

An area for significant improvement is the shot accuracy and finishing. Dortmund should give more intense drills and practice on finishing touches and shots.

Our analysis has shown the huge potentianl benefits that can be brought by correct strategies. Based on our analysis, we suggest that Dortmund should look into both Dortmund's and opponent's team status such as possession, defense tasks, passing styles and other features and find how each feature can affect the game outcome.

In short term, we suggest that Dortmund should start to apply the rules we found through our analysis immediately. These rules have no cost or minimal costs to implement. We believe the coach of one of the best teams in the German League would already know how to employ various strategies and we wanted to show what are the factors and strategies that lead to Dortmund's wins and losses. This way, we provide rules or effective strategies that would work against various types of oppositions and the coach would have to effectively communicate these tactics to the players and employ player formations suitable for such strategies.

We are very confident that our rules toward weak teams will be effective. Due to the low counts and small sample size of our rules toward equal team and strong teams, we hope that coach can investigate and verify them before applying.

In long term, we suggest that Dortmund should establish a database which focuses on collecting information on match-level and team-level data for all its opponents. With more information, Dortmund can construct a more precise model that can generate detailed suggestions for team's coach before each game. Buying good

players can boost Dortmund's performance for one or two years but establishing a database and constructing a data-driven model will benefit the club in the ong run.

Limitations

1. We have specific strategies that should be executed for different scenarios. However, we are not sure whether the team can successfuly apply them. For instance, for some games rules suggest that Dortmund will have a higher winning chance by doubling the defense. Coach might not be able to apply this if the team does not have enough defensive players at the moment. We need to keep in touch with the coach in order to adjust strategies based on current status.

2. Some confidence values from our rules can be inaccurate because of the lack of a big sample size. For example, when we analyze rules against strong teams, a certain strategy gives us a confidence value of 1. This means if we apply this strategy, our winning rate can be 100%. This cannot be true since the count is only 3, which means the winning rate is overestimated. Even though these rules are useful, we need to be cautious and it is difficult to know what the real confidence value is.

3. Setting hyper-parameters for association rules can be tricky. If we want to have a very specific rule, we need to set our minimum length high, and this will lead to less rules. However, if we set our minimum length low, we will have some rules with really high confidence, lift but smaller length. It is up to coach team's choice what kind of rules they would like to have. As a future consideration, we will combine our suggestions with real life outcomes to find out the best hyper-parameters for our association rules.