“Homework 1”

Mahir Raitani, Darshika Sharma, Sidharth Sundararaman, Yuwen Yan, Siyu Zhang, Zachary Zinda

September 29, 2019

Contents

The Business Problem and Our Approach

Our Task

Over the past decade, the field of sports analytics has experienced tremendous growth, bringing change to both the fan experience as well as the strategies used by teams to achieve on-field success. Major League Baseball and the National Football League both partner with Amazon Web Services to provide analytics for fans. Nate Silver of *FiveThirtyEight* is a frequent contributor at ESPN and dedicates a large section of his popular website to data science in sports. Teams have used analytics to alter strategy and increase their chances of success which brought substantial changes to some leagues, for example, the heavy emphasis on three-point shooting in the National Basketball Association and the now ubiquitous “defensive shift” in Major League Baseball.

Our client for this project is the coach of AS Roma, an Italian soccer club. Using a dataset containing eight years of statistics from European soccer matches as well as betting odds and player information sourced from the FIFA video game series we hope to provide AS Roma with strategies that can be employed to increase on-field successes while decreasing the frequency of failure.

Our Approach

Since making drastic roster changes by signing, cutting, and trading players can take several seasons, our recommendations and analysis will focus on team strategy and formation rather than player characteristics. Strategy and formation can be most easily manipulated by the coach and can have an immediate impact.

We will first focus on the general pattern of what formations and strategies win the most games in the Italian league. We will then look at when strategy should be changed based on certain conditions.

Data Cleaning

Load necessary packages:

```
library(tidyverse)
library(RSQLite)
library(data.table)
library(arules)
library(naniar)
library(dplyr)
library(ggplot2)
library(arulesViz)
```

Load tables from database into memory:

```
con <- src_sqlite("euro_soccer.sqlite")
country_tbl <- tbl(con, "country")
league_tbl <- tbl(con, "league")
match_tbl <- tbl(con, "match")
player_tbl <- tbl(con, "player")
player_atts_tbl <- tbl(con, "player_attributes")
team_tbl <- tbl(con, "team")
team_atts_tbl <- tbl(con, "team_attributes")

country_tbl <- collect(country_tbl)
league_tbl <- collect(league_tbl)
match_tbl <- collect(match_tbl)
player_tbl <- collect(player_tbl)
player_atts_tbl <- collect(player_atts_tbl)
team_tbl <- collect(team_tbl)
team_atts_tbl <- collect(team_atts_tbl)
```

Data Transformations

Team characteristics

Since the team attributes are sourced from the FIFA video game series they are updated numerous times for each team over the eight years that the data spans. In order to have accurate team attributes for each match, it was necessary to use the "data.table" package to perform a rolling join using the match date and the date the team attribute was updated by FIFA. This allowed us to use the team attributes from the last FIFA update that occurred prior to the match. According to EA Sports, the company that produces the FIFA video game, their player and team ratings and characteristics come from a network of over 9000 coaches, professional scouts, and season ticket holders, so for the purpose of this analysis we are making the assumption that the FIFA assessments of each team are accurate portrayals of real-life performance.

```
# Joining match and team attributes table based on the team api id and date

match_tbl = data.table(match_tbl)
team_atts_tbl = data.table(team_atts_tbl)

setkey(match_tbl, home_team_api_id, date)
setkey(team_atts_tbl, team_api_id, date)

matches_and_team_atts <- team_atts_tbl[match_tbl, roll = T]
matches_and_team_atts <- matches_and_team_atts[, 1:34]

matches_and_team_atts = data.table(matches_and_team_atts)
setkey(matches_and_team_atts, away_team_api_id, date)
setkey(team_atts_tbl, team_api_id, date)
matches_and_team_atts <- team_atts_tbl[matches_and_team_atts, roll = T]
names(matches_and_team_atts) <- gsub("i\\.", "away_", names(matches_and_team_atts))
matches_and_team_atts <- matches_and_team_atts[, c(55, 4, 3, 28, 56, 57, 52:54, 51,
                                                   5:25, 29:49)]
```

Obtaining player positions and formations

In order to look at how successful each formation is, it was necessary to first classify each player as a goalkeeper, defender, midfielder, or attacker for each game. It was then possible to sum the number of players by position and determine what formation the team ran that game. For example, a 4-4-2 formation consists of four defenders, four midfielders, and two attackers. After looking at the X and Y coordinates provided for each player it was determined that a Y-coordinate of 1 corresponded to the goalkeeper as there was the only player per team with that value. Y-coordinates 2-4 correspond with defenders, 5-8 are midfielders, and 9-11 are attackers.

```
con <- dbConnect(drv=RSQLite::SQLite(), dbname="euro_soccer.sqlite")
tables <- dbListTables(con)
tables <- tables[tables != "sqlite_sequence"]
df <- vector("list", length=length(tables))

for (i in seq(along=tables)) {
  df[[i]] <- dbGetQuery(conn=con, statement=paste("SELECT * FROM '", tables[[i]],
                                                  "'", sep=""))}

data2 <- df [3][[1]] ## filtering for the match dataset

data2$home_player_1_role <-   0
data2$home_player_2_role <-   0
data2$home_player_3_role <-   0
data2$home_player_4_role <-   0
data2$home_player_5_role <-   0
data2$home_player_6_role <-   0
data2$home_player_7_role <-   0
data2$home_player_8_role <-   0
data2$home_player_9_role <-   0
data2$home_player_10_role <-  0
data2$home_player_11_role <-  0
```

```r
data2$away_player_1_role <-   0
data2$away_player_2_role <-   0
data2$away_player_3_role <-   0
data2$away_player_4_role <-   0
data2$away_player_5_role <-   0
data2$away_player_6_role <-   0
data2$away_player_7_role <-   0
data2$away_player_8_role <-   0
data2$away_player_9_role <-   0
data2$away_player_10_role <-  0
data2$away_player_11_role <-  0

## Based on the player coordinates, we are tagging their position in the game field

for (i in (1: nrow(data2)))

  for (j in 116:137)

    data2[i, j] = data2[i, j - 82]


for (i in (1: nrow(data2)))

  for (j in 116:137)

    if (data2[i,j] %in% c(1)) {
      data2[i, j] <- "GK"
    } else if (data2[i,j] %in% c(2, 3, 4)) {
      data2[i, j] <- "DEF"
    } else if (data2[i,j] %in% c(5, 6, 7, 8)) {
      data2[i,j] <- "MID"
    } else if (data2[i,j] %in% c(9, 10, 11)) {
      data2[i,j] <- "ATT"
    }

## Establishing the number of Defenders, Midfielders and Strikers
## played by both the teams in a match

data2$home_defenders <- rowSums(data2[c('home_player_1_role','home_player_2_role',
                                      'home_player_3_role','home_player_4_role',
                                      'home_player_5_role','home_player_6_role',
                                      'home_player_7_role','home_player_8_role',
                                      'home_player_9_role','home_player_10_role',
                                      'home_player_11_role'
                                      )] == "DEF")

data2$away_defenders<- rowSums(data2[c('away_player_1_role','away_player_2_role',
                                       'away_player_3_role','away_player_4_role',
                                       'away_player_5_role','away_player_6_role',
                                       'away_player_7_role','away_player_8_role',
                                       'away_player_9_role','away_player_10_role',
                                       'away_player_11_role'
                                      )] == 'DEF')
```

```r
data2$home_midfield <- rowSums(data2[c('home_player_1_role','home_player_2_role',
                                       'home_player_3_role','home_player_4_role',
                                       'home_player_5_role','home_player_6_role',
                                       'home_player_7_role','home_player_8_role',
                                       'home_player_9_role','home_player_10_role',
                                       'home_player_11_role'
                                      )] == "MID")
data2$away_midfield<- rowSums(data2[c('away_player_1_role','away_player_2_role',
                                      'away_player_3_role','away_player_4_role',
                                      'away_player_5_role','away_player_6_role',
                                      'away_player_7_role','away_player_8_role',
                                      'away_player_9_role','away_player_10_role',
                                      'away_player_11_role'
                                      )] == 'MID')

data2$home_strikers <- 10 - (data2$home_defenders) - (data2$home_midfield)
data2$away_strikers <- 10 - (data2$away_defenders) - (data2$away_midfield)


## Establishing the complete Team Formation for both the teams
data2$home_formation <- paste(data2$home_defenders,"-",data2$home_midfield,
                              "-",data2$home_strikers)
data2$away_formation <- paste(data2$away_defenders,"-",data2$away_midfield,
                              "-",data2$away_strikers)

## Removing redundant columns

df_formation <- data2%>% select(c(match_api_id,home_team_api_id,
                                  away_team_api_id,home_team_goal,
                                  away_team_goal,home_defenders,
                                  home_midfield,home_strikers,
                                  home_formation,away_defenders,
                                  away_midfield,away_strikers,
                                  away_formation))
```

Subsetting the data for just Italian League games

```r
### Scope of the dataset is restricted to italian league
italia <- matches_and_team_atts[matches_and_team_atts$league_id == 10257]
italia <- italia %>% mutate(goal_diff = home_team_goal - away_team_goal)
```

Synthesis of information from the different betting agencies

Our analysis also utilizes betting odds from various sportsbooks. Oddsmakers set odds to balance wagers on both sides of a betting line. Since sports bettors are financially incentivized to closely follow the sport we believe their collective intelligence provides rich information not included elsewhere in the data. If a player is playing with an injury or weather conditions are poor, bettors will take those things into account when betting even though they are not items available in the data. To simplify our analysis we aggregated the information from the different bookies to come up with one overall opinion of whether or not the home team would win, lose, or draw.

```
df <- df [3][[1]]

## Identifying the missing values spread
df <- replace_with_na_all(data = df, condition =  ~.x %in% c("na",'N/a',
                                                  'N/A','missing','NA'))
miss_var_summary(df)
```

```
## # A tibble: 115 x 3
##    variable n_miss pct_miss
##    <chr>     <int>    <dbl>
##  1 PSH       14811     57.0
##  2 PSD       14811     57.0
##  3 PSA       14811     57.0
##  4 BSH       11818     45.5
##  5 BSD       11818     45.5
##  6 BSA       11818     45.5
##  7 GBH       11817     45.5
##  8 GBD       11817     45.5
##  9 GBA       11817     45.5
## 10 goal      11762     45.3
## # ... with 105 more rows
```

```
miss_var_summary(df[,85:103])
```

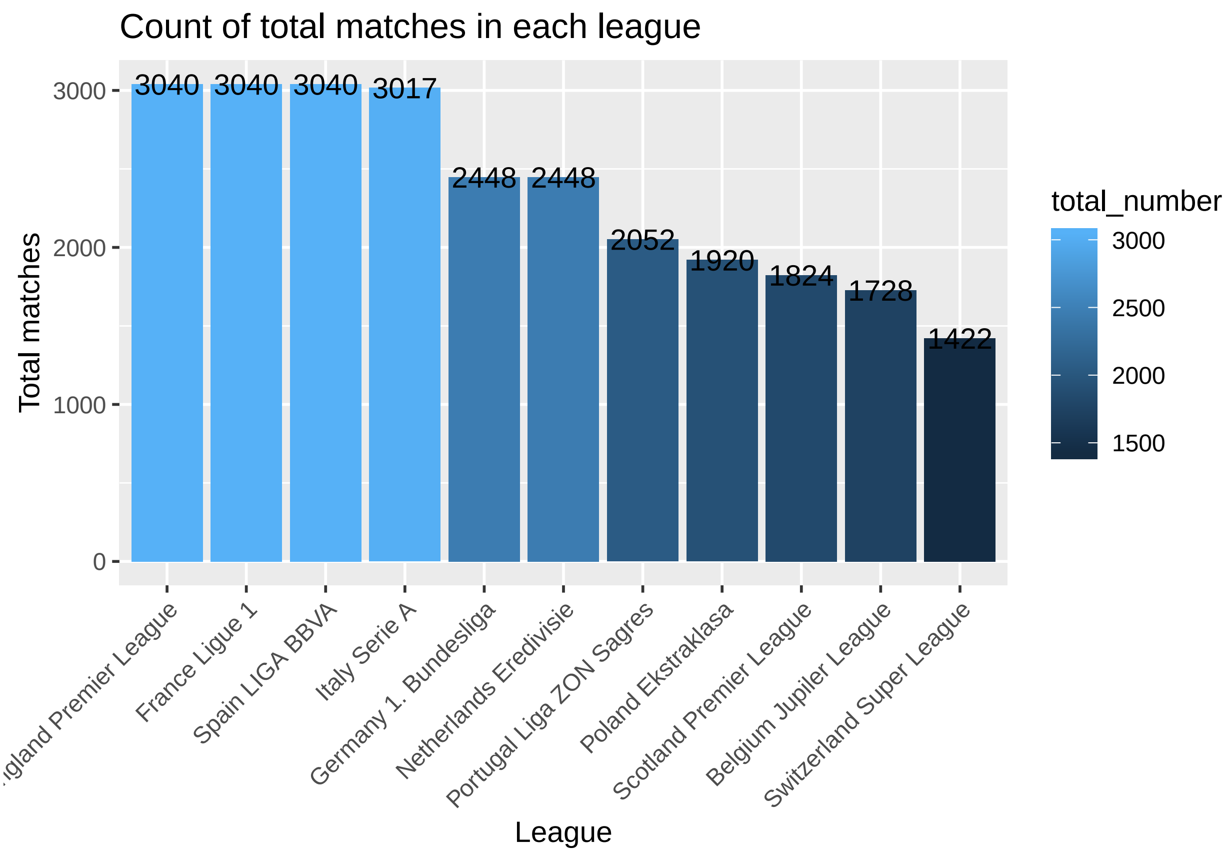
```
## # A tibble: 19 x 3
##    variable   n_miss pct_miss
##    <chr>       <int>    <dbl>
##  1 PSH         14811     57.0
##  2 PSD         14811     57.0
##  3 PSA         14811     57.0
##  4 possession  11762     45.3
##  5 IWH          3459     13.3
##  6 IWD          3459     13.3
##  7 IWA          3459     13.3
##  8 LBH          3423     13.2
##  9 LBD          3423     13.2
## 10 LBA          3423     13.2
## 11 WHH          3408     13.1
## 12 WHD          3408     13.1
## 13 WHA          3408     13.1
## 14 BWH          3404     13.1
## 15 BWD          3404     13.1
## 16 BWA          3404     13.1
## 17 B365H        3387     13.0
## 18 B365D        3387     13.0
## 19 B365A        3387     13.0
```

```
# Removing these agencies because they contain mostly null values
df<- df %>% select (-c(PSH,PSD,PSA,SJH,SJD,SJA,GBH,GBD,GBA,BSH,BSD,BSA))

# Filtering out records that contain NA's
df1 <- df %>% drop_na(c(B365H ,B365D ,B365A,
                        BWH ,BWD ,BWA,
                        IWH ,IWD ,IWA,
                        LBH ,LBD ,LBA,
                        WHH ,WHD ,WHA,
                        VCH ,VCD ,VCA))

# Converting the odds to probability

for (i in (1: nrow(df1)))
  for (j in 86:103)
    df1[i, j] = 1 / df1[i,j]

# Finding the highest probable predicted result according to the betting agencies
df1$B1_result<- colnames(df1[c('B365H','B365D','B365A')
                             ])[apply(df1[c('B365H','B365D','B365A')
                            ],MARGIN = 1, FUN = which.max)]
df1$B2_result <- colnames(df1[c('BWH','BWD','BWA')
                              ])[apply(df1[c('BWH','BWD','BWA')
                              ],MARGIN = 1, FUN = which.max)]
df1$B3_result <- colnames(df1[c('IWH','IWD','IWA')
                              ])[apply(df1[c('IWH','IWD','IWA')
                              ],MARGIN = 1, FUN = which.max)]
df1$B4_result <- colnames(df1[c('LBH','LBD','LBA')
                              ])[apply(df1[c('LBH','LBD','LBA')
                              ],MARGIN = 1, FUN = which.max)]
df1$B5_result <- colnames(df1[c('WHH','WHD','WHA')
                              ])[apply(df1[c('WHH','WHD','WHA')
                              ],MARGIN = 1, FUN = which.max)]
df1$B6_result <- colnames(df1[c('VCH','VCD','VCA')
                              ])[apply(df1[c('VCH','VCD','VCA')
                              ],MARGIN = 1, FUN = which.max)]

# Based on the results, tagging "W" / "L" from the home team perspective
df1$B1_result <- ifelse(grepl("A",df1$B1_result) == TRUE,"L"
                        ,(ifelse(grepl("D",df1$B1_result)== TRUE,"D","W")))
df1$B2_result <- ifelse(grepl("A",df1$B2_result) == TRUE,"L"
                        ,ifelse(grepl("D",df1$B2_result)== TRUE,"D","W"))
df1$B3_result <- ifelse(grepl("A",df1$B3_result) == TRUE,"L"
                        ,ifelse(grepl("D",df1$B3_result)== TRUE,"D","W"))
df1$B4_result <- ifelse(grepl("A",df1$B4_result) == TRUE,"L"
                        ,ifelse(grepl("D",df1$B4_result)== TRUE,"D","W"))
df1$B5_result <- ifelse(grepl("A",df1$B5_result) == TRUE,"L"
                        ,ifelse(grepl("D",df1$B5_result)== TRUE,"D","W"))
df1$B6_result <- ifelse(grepl("A",df1$B6_result) == TRUE,"L"
                        ,ifelse(grepl("D",df1$B6_result)== TRUE,"D","W"))

# Final result of the betting agency is decided on the results
# predicted by majority of the agencies
df1$loss_count <- rowSums(df1[c('B1_result','B2_result',
                      'B3_result','B4_result','B5_result','B6_result')] == "L")
df1$draw_count <- rowSums(df1[c('B1_result','B2_result',
                      'B3_result','B4_result','B5_result','B6_result')] == "D")
df1$win_count <-  rowSums(df1[c('B1_result','B2_result',
                      'B3_result','B4_result','B5_result','B6_result')] == "W")

df1$actual_result <- ifelse(df1['home_team_goal'] - df1['away_team_goal'] > 0 , 'W',
                      ifelse(df1['home_team_goal'] - df1['away_team_goal'] == 0, 'D','L'))

df1$bets_final_result <- ifelse(df1$loss_count >= 3,'L',
                          ifelse(df1$draw_count > 3, 'D', 'W'))

df2 <- df1%>% select(-c(B365H,B365D,B365A,BWH,BWD,BWA,IWH,IWD,IWA,LBH,LBD,LBA,
                        WHH,WHD,WHA,VCH,VCD,VCA))
df2 <- df1%>% select (-c(12:55,110:112,86:103))
df2 <- df2 %>% select(id,country_id,league_id,season,stage,date,match_api_id,
                      home_team_api_id,away_team_api_id,home_team_goal
                      ,away_team_goal,actual_result,bets_final_result
                      ,goal,shoton,shotoff,foulcommit,card,cross,corner,possession)

## Trimming the formation dataset
df3<- df_formation%>% select(match_api_id,home_defenders,home_midfield,home_strikers,
                             home_formation,away_defenders,away_midfield
                             ,away_strikers,away_formation)
df4 <- inner_join(df2,df3,by = "match_api_id")
df4 <- df4 %>% filter(league_id == 10257) #3 italian league as the dataset scope

df4$goal_diff <- df4$home_team_goal - df4$away_team_goal
df4$goal_diff <- as.factor(df4$goal_diff)

italia <- inner_join(italia, df4[, c(7, 12, 13, 25 ,29)], by = 'match_api_id')
```

An Overview of Roma's Performance

Before diving into Roma's performance, we first want to have an overview of European soccer and what all leagues look like. Thus, we create a graph showing the total number of matches played in each league to better understand our dataset.

```
# Total matches by League
match_by_league <- match_tbl %>%
  group_by(league_id) %>%
  summarize(n()) %>%
  rename(id = league_id, total_number = `n()`) %>%
  left_join(league_tbl, by = 'id') %>%
  select(name, total_number) %>%
  arrange(desc(total_number))
p <- ggplot(match_by_league, aes(x = reorder(name, -total_number),
                                 y = total_number, fill = total_number)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = total_number))
p <- p + labs(x="League",
              y="Total matches",
              title="Count of total matches in each league")
p + theme(axis.text.x=element_text(angle=45, hjust=1))
```



From the above graph, we can see the total number of games played in Italy Series A League is only a small portion of our data and differs from other leagues.

Since our client is AS Roma, we would like to know which league it is playing in and we would focus our analysis on that league only since there are differen ces between leagues.

```
# Choose only Roma's data
long_team_name <- 'Roma'
roma_record <- team_tbl %>%
  filter(grepl(long_team_name, team_long_name))

home_matches <- filter(match_tbl, home_team_api_id == roma_record$team_api_id)
away_matches <- filter(match_tbl, away_team_api_id ==  roma_record$team_api_id)
ROMA_matches <- rbind(home_matches, away_matches)
league_tbl %>% filter(id %in% unique(ROMA_matches$league_id))
```

```
## # A tibble: 1 x 3
##      id country_id name
##   <int>      <int> <chr>
## 1 10257      10257 Italy Serie A
```

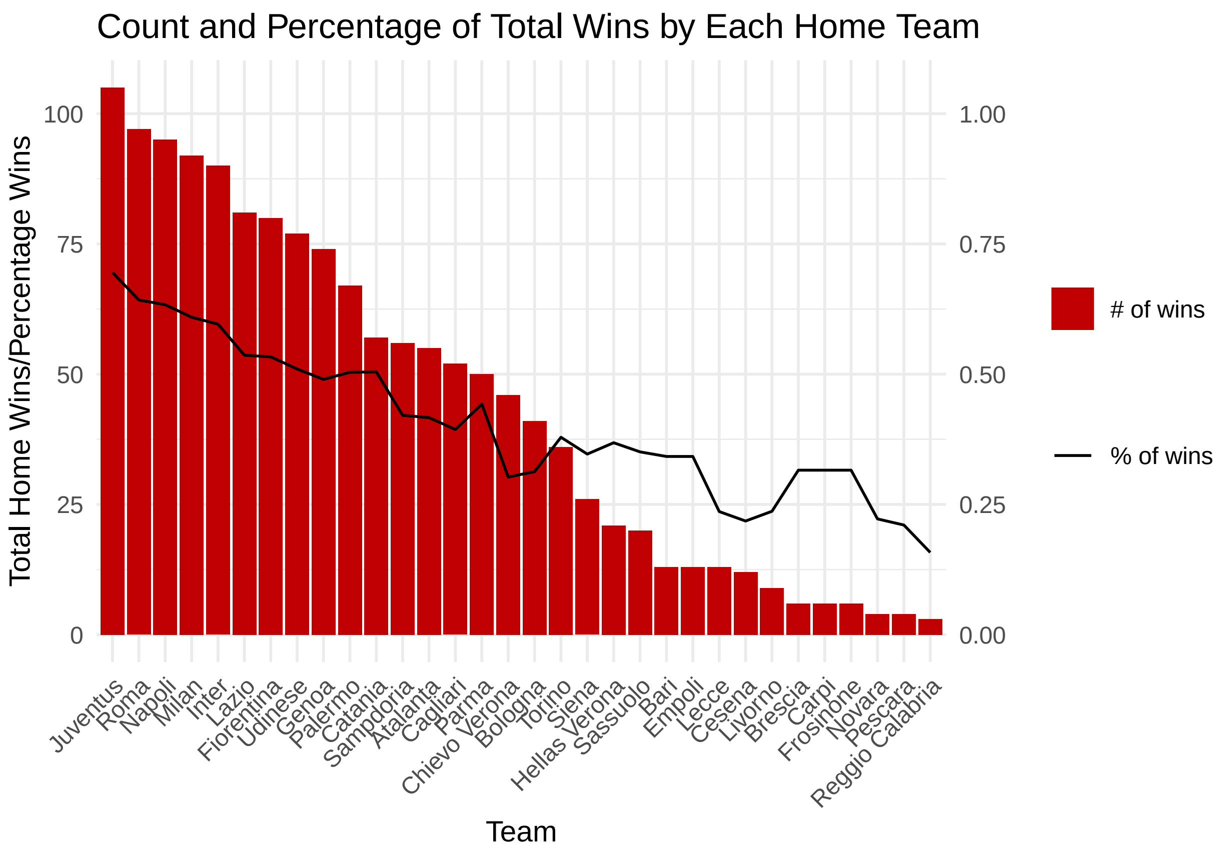
From the above code, we found that AS Roma only plays in the matches under Italian league and competes with other teams under Italian league. Thus, in our further analysis, we only focus on Italian league games.

Roma's performance in Italian league We then calculated the number and percentage of wins for each team under Italian league in their home match and away match respectively to get an overview of Roma's performance.

```r
italian <- match_tbl %>% filter(league_id == 10257)
italian_matches <- match_tbl %>% filter(league_id == 10257) %>%
  mutate(goal_diff = home_team_goal - away_team_goal)
italian_matches$home_win <- ifelse(italian_matches$goal_diff > 0, 1, 0)
italian_matches$away_win <- ifelse(italian_matches$goal_diff < 0, 1, 0)
home_wins_percentage <- italian_matches %>% select(home_team_api_id, home_win) %>%
  group_by(home_team_api_id) %>%
  summarise(home_wins = sum(home_win), win_percent = sum(home_win)/n()) %>%
  rename(team_api_id = home_team_api_id) %>%
  left_join(team_tbl, on = team_api_id) %>%
  select(team_long_name, home_wins, win_percent) %>%
  arrange(desc(home_wins))
```

```
## Joining, by = "team_api_id"
```

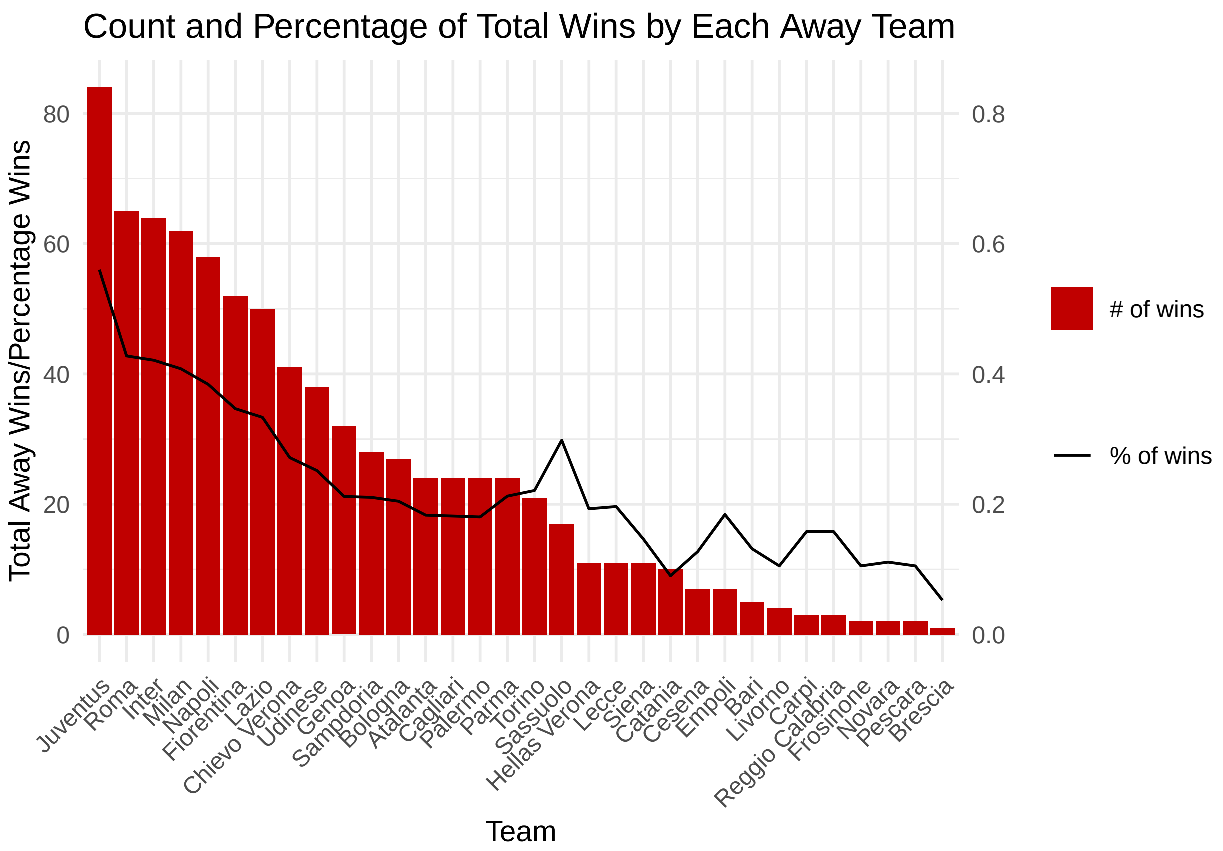
```r
q <- ggplot(home_wins_percentage, aes(x = reorder(team_long_name, -home_wins))) +
  geom_col(aes(y = home_wins, fill="redfill")) +
  geom_line(aes(y = win_percent*100, group = 1, color = 'blackline')) +
  scale_y_continuous(sec.axis = sec_axis(trans = ~ . / 100)) +
  scale_fill_manual('', labels = '# of wins', values = "#C00000") +
  scale_color_manual('', labels = '% of wins', values = 'black') +
  theme_minimal()
q <- q + labs(x="Team",
            y="Total Home Wins/Percentage Wins",
            title="Count and Percentage of Total Wins by Each Home Team")
q + theme(axis.text.x=element_text(angle=45, hjust=1))
```



```r
away_wins_percentage <- italian_matches %>% select(away_team_api_id, away_win) %>%
  group_by(away_team_api_id) %>%
  summarise(away_wins = sum(away_win), win_percent = sum(away_win)/n()) %>%
  rename(team_api_id = away_team_api_id) %>%
  left_join(team_tbl, on = team_api_id) %>%
  select(team_long_name, away_wins, win_percent) %>%
  arrange(desc(away_wins))
```

```
## Joining, by = "team_api_id"
```

```r
a <- ggplot(away_wins_percentage, aes(x = reorder(team_long_name, -away_wins))) +
  geom_col(aes(y = away_wins, fill="redfill")) +
  geom_line(aes(y = win_percent*100, group = 1, color = 'blackline')) +
  scale_y_continuous(sec.axis = sec_axis(trans = ~ . / 100)) +
  scale_fill_manual('', labels = '# of wins', values = "#C00000") +
  scale_color_manual('', labels = '% of wins', values = 'black') +
  theme_minimal()
a <- a + labs(x="Team",
              y="Total Away Wins/Percentage Wins",
              title="Count and Percentage of Total Wins by Each Away Team")
a + theme(axis.text.x=element_text(angle=45, hjust=1))
```



Interpretation From the above graph, we can see that Roma is within the top teams level. Its number of wins in both home matches and away matches ranks second among the teams in Italian league. The main competitor for Roma is Juventus, which is the top team in the Italian League regarding both the number of wins and percentage of wins. Napoli, Milan and Inter rank slightly lower than Roma, but they are also strong competitors especially regarding the percentage of win.

Analysis of What Leads to Success

In order to find the correct combination of team attributes, we utilized association rules. We began our analysis by looking at what general strategy leads to the most success in the Italian league before exploring more specific situations.

Before we dive into the analysis, we want to explain some of the diferent features in the dataset for better consumption of the results. Build-up play is when a team is attempting to move the ball upfield while the opponent is in an organized defense, so the different build-up play class attributes for speed, dribbling, passing, and positioning provide us information on the team's standard attacking strategy. The chance creation classes take into account the strategy employed on the third of the field closest to the opponent's goal. The different defense classes account for the team's strategy when trying to regain possession of the ball.

Association rules for team characteristics

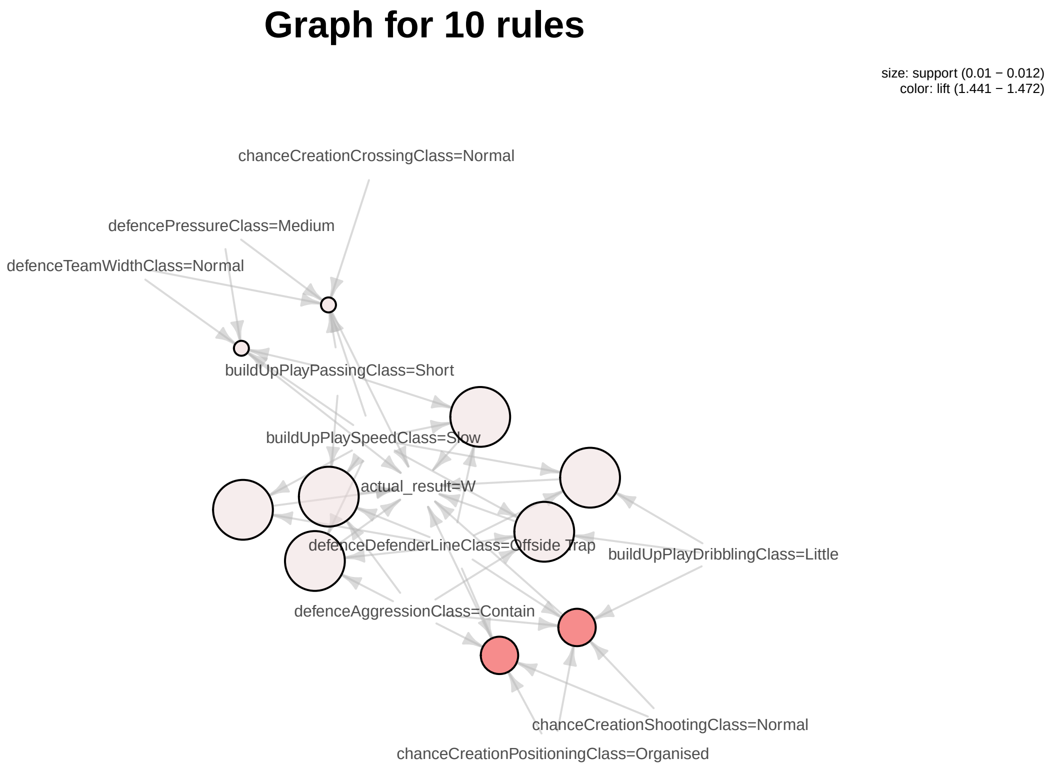
We began our analysis by treating win result as the right-hand side of the association rule and looking for what combinations of team characteristics provided the greatest lift. There were a number of team characteristics we saw repeatedly in our top rules. A build-up play speed of "slow" was present in eight of our top ten rules. A defender line class of "Offside Trap" was present in seven. We also found build-up play dribbling class of “little” was present often, indicating dribbling should be minimized during build-up play. We also found a passing class of “short” to be present in four of the rules that provided the most lift as well as defensive aggression of “contain”. These team characteristics indicate an overall trend that a balanced, conservative playstyle leads to the most wins in the Italian league.

```
basket <- as(italia[, c(12, 14, 16, 17, 19, 21, 23, 24, 26, 28, 30,
                        31, 54)], 'transactions')
rules <- apriori(basket, parameter = list(supp = 0.01, conf = 0.1, minlen = 2))
win_rules_no_odds <- subset(rules, subset =  rhs %pin% "actual_result=W")
win_rules_no_odds <- sort(win_rules_no_odds, by='lift', decreasing = TRUE)
```

```
inspect(win_rules_no_odds[0:10])
```

```
##      lhs                                            rhs                  support confidence     lift
## [1]  {chanceCreationShootingClass=Normal,
##       chanceCreationPositioningClass=Organised,
##       defenceAggressionClass=Contain,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01114488  0.6875000 1.471936
## [2]  {buildUpPlayDribblingClass=Little,
##       chanceCreationShootingClass=Normal,
##       chanceCreationPositioningClass=Organised,
##       defenceAggressionClass=Contain,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01114488  0.6875000 1.471936
## [3]  {buildUpPlaySpeedClass=Slow,
##       buildUpPlayPassingClass=Short,
##       defencePressureClass=Medium,
##       defenceTeamWidthClass=Normal}             => {actual_result=W} 0.01046944  0.6739130 1.442846
## [4]  {buildUpPlaySpeedClass=Slow,
##       buildUpPlayPassingClass=Short,
##       chanceCreationCrossingClass=Normal,
##       defencePressureClass=Medium,
##       defenceTeamWidthClass=Normal}             => {actual_result=W} 0.01046944  0.6739130 1.442846
## [5]  {buildUpPlaySpeedClass=Slow,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01182033  0.6730769 1.441056
## [6]  {buildUpPlaySpeedClass=Slow,
##       defenceAggressionClass=Contain,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01182033  0.6730769 1.441056
## [7]  {buildUpPlaySpeedClass=Slow,
##       buildUpPlayPassingClass=Short,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01182033  0.6730769 1.441056
## [8]  {buildUpPlaySpeedClass=Slow,
##       buildUpPlayDribblingClass=Little,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01182033  0.6730769 1.441056
## [9]  {buildUpPlaySpeedClass=Slow,
##       buildUpPlayPassingClass=Short,
##       defenceAggressionClass=Contain,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01182033  0.6730769 1.441056
## [10] {buildUpPlaySpeedClass=Slow,
##       buildUpPlayDribblingClass=Little,
##       defenceAggressionClass=Contain,
##       defenceDefenderLineClass=Offside Trap}    => {actual_result=W} 0.01182033  0.6730769 1.441056
```

```
sub_win_rules <- head(win_rules_no_odds, n = 10, by = "lift")
plot(sub_win_rules, method = "graph", cex = 0.5)
```


Graph for 10 rules
size: support (0.01 − 0.012)
color: lift (1.441 − 1.472)
chanceCreationCrossingClass=Normal
defencePressureClass=Medium
defenceTeamWidthClass=Normal
buildUpPlayPassingClass=Short
buildUpPlaySpeedClass=Slow
actual_result=W
defenceDefenderLineClass=Offside Trap
buildUpPlayDribblingClass=Little
defenceAggressionClass=Contain
chanceCreationShootingClass=Normal
chanceCreationPositioningClass=Organised

Association rules for formations

Team formation is often representes team's idea and style for the game. Formations are often suited to best optimize the team's strength and drive an effective wedge into the opponent's strategy. Rightly deciding this factor often goes in great length in deciding the outcome of the match. Though 4-4-2 is the most popular formation for both home and away squads, 4-5-1 appears to provide a slight advantage for home teams.

```
formations_no_odds <- italia[,c(54, 56)]
formation_basket <- as(formations_no_odds, 'transactions')
rules <- apriori(formation_basket, parameter = list(supp = 0.05, conf = 0.1,
         minlen = 2), appearance = list(default = "lhs", rhs = "actual_result=W"))
rules<-  sort(rules, by='lift', decreasing=TRUE)
inspect(rules)
formations_no_odds <- italia[,c(54, 57)]
formation_basket <- as(formations_no_odds, 'transactions')
rules <- apriori(formation_basket, parameter = list(supp = 0.05, conf = 0.1,
         minlen =2), appearance = list(default = "lhs", rhs = "actual_result=W"))

rules<- sort(rules, by='lift', decreasing=TRUE)
```

```
inspect(rules)
```

```
##     lhs                           rhs               support    confidence
## [1] {away_formation=4 - 4 - 2} => {actual_result=W} 0.16176967 0.4877800
## [2] {away_formation=4 - 5 - 1} => {actual_result=W} 0.07193516 0.4754464
## [3] {away_formation=3 - 5 - 2} => {actual_result=W} 0.08713273 0.4542254
## [4] {away_formation=4 - 3 - 3} => {actual_result=W} 0.09422492 0.4325581
##     lift      count
## [1] 1.0443360 479
## [2] 1.0179298 213
## [3] 0.9724955 258
## [4] 0.9261060 279
```

Exceptions to the General Trends

While we previously identified the strategy and formation that are generally the most effective in the Italian league, we further investigated the hypothesis that the best strategy and formation might change in circumstances where a team is outmatched by their opponent versus situations where they are already likely to win. For this analysis we utilized the betting information that we detailed in the data preparation section.

Increasing your likelihood of winning when the team is picked to lose

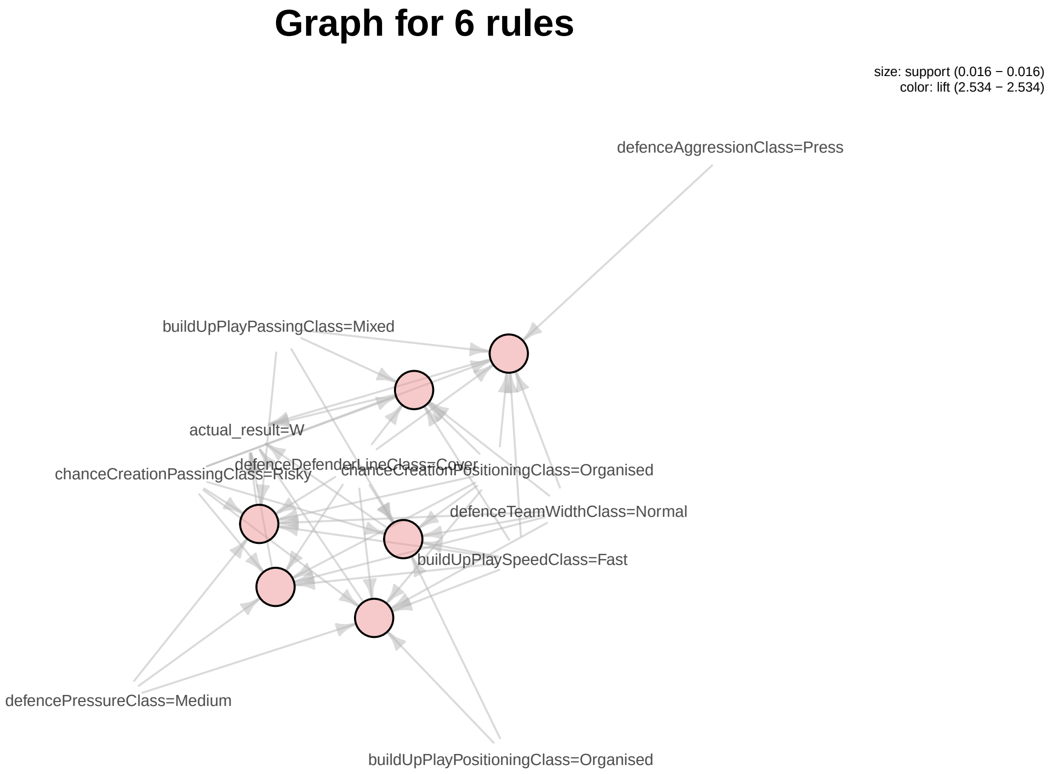
When a team is picked by bettors to lose we found that the team characteristics most likely to produce an upset victory are very different from the general trend found in the previous analysis. Defender line class of “cover,” which means organizing the defense so there is always a second “covering” defender near the opponent with the ball, and fast pace in build-up play as well as risky chance creation passing provide the greatest lift. Consistent with the findings that a more aggressive playstyle can increase a team’s chances of pulling off an upset, adding an extra attacker and running a 4-4-3 formation provides the greatest lift in this situation.

```
basket2 <- as(italia[italia$bets_final_result == 'L', c(12, 14, 16, 17, 19, 21, 23,
                                           24, 26, 28, 30, 31, 54)], 'transactions')

 rules2 <- apriori(basket2, parameter = list(supp = 0.01, conf = 0.1, minlen = 2))

 win_rules_odds <- subset(rules2, subset =  rhs %pin% "actual_result=W" & lift > 2.2)
```

```
inspect(head(sort(win_rules_odds, by='lift', decreasing = TRUE)))
```

```
##     lhs                                           rhs                 support confidence     lift co
## [1] {buildUpPlaySpeedClass=Fast,
##      buildUpPlayPassingClass=Mixed,
##      chanceCreationPassingClass=Risky,
##      chanceCreationPositioningClass=Organised,
##      defenceTeamWidthClass=Normal,
##      defenceDefenderLineClass=Cover}           => {actual_result=W} 0.0164557  0.6190476 2.533925
## [2] {buildUpPlaySpeedClass=Fast,
##      chanceCreationPassingClass=Risky,
##      chanceCreationPositioningClass=Organised,
##      defencePressureClass=Medium,
##      defenceTeamWidthClass=Normal,
##      defenceDefenderLineClass=Cover}           => {actual_result=W} 0.0164557  0.6190476 2.533925
## [3] {buildUpPlaySpeedClass=Fast,
##      buildUpPlayPassingClass=Mixed,
##      buildUpPlayPositioningClass=Organised,
##      chanceCreationPassingClass=Risky,
##      chanceCreationPositioningClass=Organised,
##      defenceTeamWidthClass=Normal,
##      defenceDefenderLineClass=Cover}           => {actual_result=W} 0.0164557  0.6190476 2.533925
## [4] {buildUpPlaySpeedClass=Fast,
##      buildUpPlayPassingClass=Mixed,
##      chanceCreationPassingClass=Risky,
##      chanceCreationPositioningClass=Organised,
##      defenceAggressionClass=Press,
##      defenceTeamWidthClass=Normal,
##      defenceDefenderLineClass=Cover}           => {actual_result=W} 0.0164557  0.6190476 2.533925
## [5] {buildUpPlaySpeedClass=Fast,
##      buildUpPlayPassingClass=Mixed,
##      chanceCreationPassingClass=Risky,
##      chanceCreationPositioningClass=Organised,
##      defencePressureClass=Medium,
##      defenceTeamWidthClass=Normal,
##      defenceDefenderLineClass=Cover}           => {actual_result=W} 0.0164557  0.6190476 2.533925
## [6] {buildUpPlaySpeedClass=Fast,
##      buildUpPlayPositioningClass=Organised,
##      chanceCreationPassingClass=Risky,
##      chanceCreationPositioningClass=Organised,
##      defencePressureClass=Medium,
##      defenceTeamWidthClass=Normal,
##      defenceDefenderLineClass=Cover}           => {actual_result=W} 0.0164557  0.6190476 2.533925
```

```r
subrules2 <- head(sort(win_rules_odds, by = 'lift', decreasing = TRUE))
plot(subrules2, method = 'graph', cex = 0.5)
```


Graph for 6 rules
size: support (0.016 − 0.016)
color: lift (2.534 − 2.534)
defenceAggressionClass=Press
buildUpPlayPassingClass=Mixed
actual_result=W
defenceDefenderLineClass=Cover
chanceCreationPassingClass=Risky
chanceCreationPositioningClass=Organised
defenceTeamWidthClass=Normal
buildUpPlaySpeedClass=Fast
defencePressureClass=Medium
buildUpPlayPositioningClass=Organised

```r
formations_win_odds <- italia[italia$bets_final_result == 'L', c(54,56)]
formation_basket2 <- as(formations_win_odds, 'transactions')

rules5 <- apriori(formation_basket2, parameter = list(supp = 0.05,
    conf = 0.1, minlen = 2),appearance = list(default = "lhs", rhs = "actual_result=W"))

rules5<-  sort(rules5, by='lift', decreasing=TRUE)
```

```r
inspect(rules5)
```

```
##     lhs                              rhs                support    confidence
## [1] {home_formation=4 - 3 - 3} => {actual_result=W} 0.06202532 0.2737430
## [2] {home_formation=4 - 4 - 2} => {actual_result=W} 0.08860759 0.2545455
##     lift     count
## [1] 1.120503 49
## [2] 1.041922 70
```

Increasing your likelihood to draw when the team is picked to lose

In some contexts, particularly late in the season, it may be advantageous for a team to play for a draw since it still earns them one point in the standings and goal differential is often used as a tie-breaker when two teams have the same number of points. When bettors pick a team to lose, the likelihood of playing to a draw can be increased by adding a midfielder and switching to a 3-5-2 formation.

```r
formations_draw_odds <- italia[italia$bets_final_result == 'L', c(54,56)]
formation_basket3 <- as(formations_draw_odds, 'transactions')

rules6 <- apriori(formation_basket3, parameter = list(supp = 0.05,
  conf = 0.1, minlen = 2), appearance = list(default = "lhs", rhs = "actual_result=D"))
rules6<-  sort(rules6, by='lift', decreasing=TRUE)
```

```r
inspect(rules6)
```

```
##     lhs                              rhs                support    confidence
## [1] {home_formation=3 - 5 - 2} => {actual_result=D} 0.06329114 0.3125000
## [2] {home_formation=4 - 3 - 3} => {actual_result=D} 0.05443038 0.2402235
## [3] {home_formation=4 - 4 - 2} => {actual_result=D} 0.07848101 0.2254545
##     lift      count
## [1] 1.2101716 50
## [2] 0.9302771 43
## [3] 0.8730838 62
```

Takeaways and Recommendations

Based on our analysis we recommend that the coach should utilize different strategies and formation based on the opponent Roma is playing and what outcome they are seeking. In the Italian league, a team is awarded three points in the standings for a victory, one point for a draw, and nothing for a loss. Italian league teams that the bettors feel will lose see the greatest improvement in their chances of winning by utilizing a strategy of fast-paced build-up play, riskier chance creation passing when near the goal, using a "covering" defender, and running a 4-4-3 formation. Prior to the game, if the coach sees that Roma is not picked by bettors to win and the team badly needs the three points for a victory, he should adjust the formation and strategy as outlined above. If a team is an underdog and is satisfied with playing to a draw and earning one point the team should run a 3-5-2 formation. The coach needs to assess where Roma is positioned in the league standings and decide if playing a riskier strategy to increase the likelihood of earning three points is warranted or if the team should play more conservatively and settle for earning one point from a draw.

If the team is not picked by betters to lose or the opponents is relatively evenly matched to Roma the coach should use the strategies that have the greatest overall success in the Italian league, that is a 4-5-1 or 4-4-2 formation with little dribbling in build-up play, short passing, and a defensive strategy of "offside-trap."