Studying the Effects of "Prevent Defense" Tactic on Team's Offensive Output Across Five Major European Club Soccer Leagues

Abstract

Having looked at the full match statistics for the England-France 2022 FIFA World Cup Quarterfinal, one could come away thinking "England lost despite having played better than France": 16 to 8 shot attempts, 5 corners to France's 2, resulting in a 1-2 loss. What's disregarded is the scoring context: in the 40 minutes when the match was tied (0-0, 1-1), France actually led in those statistical categories, while consciously ceding initiative to England in the 66 minutes when up a goal in order in order to protect the lead (we'll call it "prevent defense"). We use sequenced match event data across five European club leagues over the past 15 years to study impacts of prevent defense on teams' offensive outputs when trailing, leading or tied. For that, we leverage non-linear modeling approaches tailored towards count response data, with predictors that are hypothesized to affect the likelihood of implementing defensive tactics.

Keywords. Count data, Generalized Additive Models, Negative Binomial, Poisson regression, Smoothing Splines, Sports statistics

1 Introduction

Soccer, also known as football, is one of the most popular sports in the world. According to the International Federation of Association Football (French: Fédération Internationale de Football Association), 5 billion people have engaged with the FIFA World Cup Qatar 2022 [3]. This exciting sport has gained a huge following due to its thrilling gameplay and fast-paced action. In soccer, statistics play a crucial role across various aspects of the game, from player performance evaluation to strategic analysis. Detailed statistical breakdowns of matches provide valuable insights into team dynamics, possession percentages, shot accuracy, passing accuracy, and other key performance indicators. This analysis helps teams identify areas for improvement and refine their playing style. Similarly, teams use statistical data to analyze opponents' playing styles, strengths, and weaknesses. This analysis helps in devising effective game plans and strategies to exploit the opponent's vulnerabilities while minimizing risks.

Data without context can often lead to misinterpretation. For instance, upon analyzing the match statistics of the 2022 FIFA World Cup Quarterfinal between England and France, one might erroneously conclude that "England lost despite being the *better team*" based on 16 to 8 shot attempts, 8 to 5 shots on target, 5 corners to France's 2, resulting in a 1-2 loss [2]. However, the metrics derived from this simple count of shots and corners disregards the score situation: in the 40 minutes when the match was tied (0-0, 1-1), France led in all of the mentioned statistical categories, while consciously ceding initiative to England in the 66 minutes when up a goal - a tactic we will refer to as "prevent defense" (a term borrowed from American football). We used match event sequencing data across the five major European leagues over the past 15 years, to study the impacts of prevent defense on the aforementioned statistical categories and scoring tendencies for teams if trailing or in the lead. To yield a more realistic picture of who might have been the *"better team"*, in this work we aim to study the effects of factors that could impact the likelihood of implementing the prevent defense tactics on team's offensive outputs such as the number of shot attempts and corners. These factors include things like the score and red card differential for a specific time period ($ScoreDiff$ and $RedCardDiff$), the length of said time period ($timeSpent$), the team designation as either Home or Away ($H.A$). In addition to these variables, pre-match betting coefficients are collected to gauge how evenly matched the two teams are, which we hypothesize to also contribute to the chances of a team pursuing the prevent defense approach. Inspecting the shots and corners that teams accumulate while accounting for the aforementioned factors could help get a more objective picture of the balance of power within a game, in order to dispel the notion of "Team X was better statistically and still lost" whenever it's not the reality.

Our end goal is to study the nature of the relationship between the aforementioned factors and teams' offensive production, thereby enhancing our understanding of soccer statistics.

1.1 Previous Research

There are existing metrics and approaches in soccer performance analysis that encompass a range of methodologies aimed at comprehensively understanding team dynamics and individual contributions within the game. Expected Goal (xG) models have emerged as a cornerstone in modern soccer analytics [18] [13]. These models assess the probability of a shot resulting in a goal based on various factors such as shot location, angle, distance, and type. By assigning a numerical value to the quality of scoring chances, xG models provide a quantitative measure for the quality of shots and opportunities a team generates.

Besides pure shot characteristics, some models used other factors such as proxies to quantify psychological effects, like match attendance, match importance and goal differential [13]. Mead et al. suggest that psychological pressure may influence the likelihood of scoring goals. Additionally, the study indicates that goal differential was among the most influential variables considered in the expected goals models. With that said, the exact nature of the effect wasn't studied in detail due to overly complex models with low interpretability ("black box" models).

In another study, based on feature importance analysis, every player's upcoming scoring performance is strongly associated with previous season's goals (Gls) and expected goals (xG) [8].

1.2 Research Question

This paper aims to address the limitations of traditional soccer match statistics by providing context through the analysis of match event sequencing data by studying the impacts of "prevent defense" tactics on various statistical categories and scoring tendencies, the research seeks to enhance the understanding of soccer match dynamics and outcomes. In statistical terms, the dependent variables of interest to us are shot attempts and corners, which we hypothesize to be related to several factors affecting likelihood of prevent defense tactic implementation. In particular, we will study the effects of score differential, red card differential, time spent at said score and red card differential, and weighted win probability (it is calculated based on the betting coefficients). The latter serves as a good proxy for relative strengths of the team, and our intuition for including it is that if a dominant team plays an underdog, we do not expect to see the dominant team play prevent defense no matter what the score differential is. Instead, they will keep scoring and playing aggressively against an easy opponent. We will leverage multiple regression with the aforementioned covariates as predictors, while corners or shots would act as the response variable. That way, we aim to isolate the true relationship between the independent factors (score and red card differential, betting coefficients) and dependent variables (shots, corners).

Although work has been done on modeling the goal expectations based on a variety of factors, including current score differential, our work focuses on other statistical categories such as shots and corners. Moreover, we emphasize interpretability of the results, avoiding overly complex, "black box", types of models and algorithms.

2 Methods

2.1 Data Preparation

2.1.1 Data Sources

Our primary data sources are *ESPN.com* ("ESPN" stands for Entertainment and Sports Programming Network) and *Oddsportal.com* websites. We utilized JavaScript to web scrape sequential data from *ESPN.com*, collecting information from games in the major five European leagues (English Premier League, Spanish La Liga, German Bundesliga, French Ligue 1, and Italian Serie A) over the last 15 years. Web scraping is the process of extracting data from websites which involves writing code to programmatically access a webpage, retrieve its HTML content, and then parse that content to extract the desired information. Due to non-trivial URL naming conventions of *ESPN.com*, we had to use extra care when developing the coding script to systematically pull relevant games from respective leagues and seasons. In particular, we had to identify

correct ranges of game identifiers ("game IDs") by trial-and-error combined with leveraging the knowledge of total number of games in each league, as the URL didn't contain any other indicators of the league or season. Using those techniques, we gathered a massive dataset that includes detailed game commentary on minute-by-minute events, cumulative game statistics such as fouls, corners, yellow cards, red cards, shots on goal, shot attempts, saves, goal times, and final scores. For the example of game commentary webpage format from *ESPN.com*, refer to Figure 1

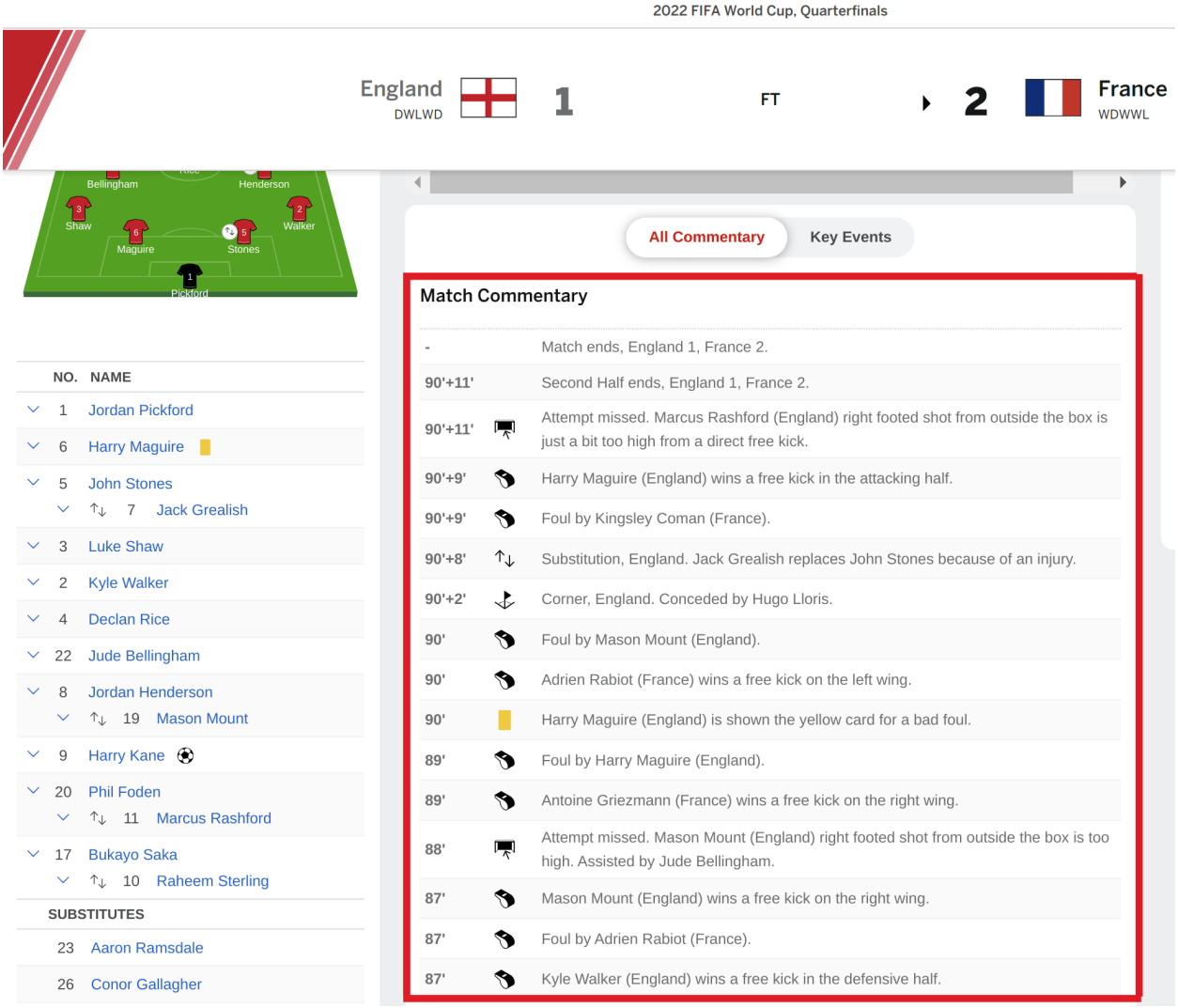

Figure 1: Example of game commentary from *ESPN.com* for the England-France Quarterfinal match of FIFA 2022 World Cup

Additionally, we scraped betting coefficients from *Oddsportal.com* for the same time frame and leagues to mitigate potential confounding effect arising from team-level differences. Betting coefficients are a good proxy for team strength as they are calculated using a combination of statistical analysis, historical data, the current outlook of a team (recent performance, injuries to key players), expert opinion, and market demand. Teams with stronger squads and a history of success are often favored over weaker opponents. Home-field advantage is a significant factor in sports, including soccer. Another important factor is that of home field advantage - teams playing at their home stadium often have better odds due to the support of their fans and familiarity

with the playing environment. The dataset of betting coefficients contains team names, the betting coefficients (for three outcomes: home team win, draw, away team win), and game dates. For the example of webpage format from *Oddsportal.com*, refer to Figure 2.

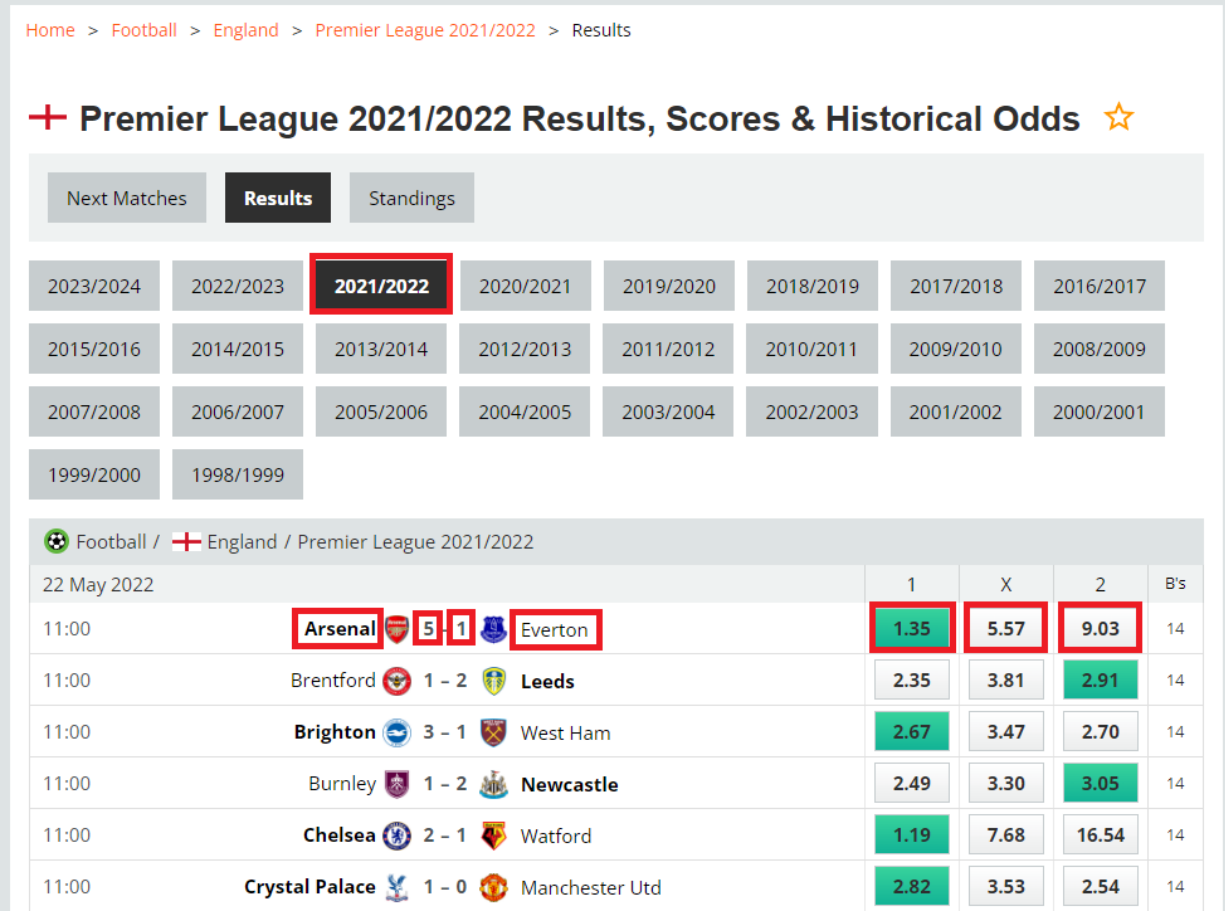

Figure 2: Example of betting coefficients data from *Oddsportal.com* for English Premier League, Season of 2021/2022. Column "1" corresponds to home team win, "X" - draw, "2" - away team win. The "B's" column contains the number of booking agencies across which the betting data was aggregated.

The data was merged from these two sources, requiring us to use game dates and the names of the home and away teams for each match to align with the odds data. However, a significant challenge was the dynamic nature of *Oddsportal.com*, rendering traditional HTTP requests ineffective for scraping purposes, as they cannot interpret JavaScript. Another challenge was that the layout of the *Oddsportal.com* web page underwent multiple changes in terms of layout of elements, thereby hindering the consistent execution of the script across all leagues. Therefore, we used the browser automation library *Selenium* [5], which provides specialized methods tailored for web scraping dynamic content from dynamic websites like *Oddsportal.com*. *Selenium* is a web automation framework that allows you to programmatically control web browsers. You can open a browser window, navigate to different web pages, interact with page elements by clicking buttons, filling forms, and extract data from web pages.

2.1.2 Data Preprocessing

We performed preprocessing for all of the leagues that consisted of 11 parts. For a thorough layout of our preprocessing pipeline see Figure 3. Preprocessing was a challenging aspect of this project since the data obtained via web scraping from two sources (*ESPN.com* and *Oddsportal*, as described in Section 2.1.1) was largely unstructured, especially the text commentary data from *ESPN.com* which initially came in JSON format with free-form data entry text fields.

Extensive coding and testing was necessary to transform the unstructured data into a cleaner format containing the quantities of interest. We heavily utilized the *Pandas* [4] and *CSV* [1] libraries in *Python* for this project. With *Pandas*, you can perform various data manipulation tasks, such as loading, cleaning, transforming, aggregating, and analyzing data, making it an essential tool for working with structured data in Python. The *CSV* library in Python allows you to easily handle CSV files by providing functions to read data from CSV files into Python data structures like lists or dictionaries, as well as functions to write data from Python data structures back to CSV files.

Below we describe the steps in detail. First, the preprocessing pipeline starts by combining the JSON files of the ESPN game commentary and game statistics for a league. The second script converts the combined JSON to CSV format to render the data in a tabular and easily accessible manner. Also, the file's encoding was changed to ASCII, resulting in accented letters being rendered as unreadable characters. We lose the special accent letters, but this will not impact our analysis. The third script segments the commentary by minute (initially, the entire game's text commentary was just one huge text entry), appending the corresponding date to each minute entry. We utilized regex to manage the stoppage time added at the end of each half. The extracted minute format includes a plus sign, such as 45+1. We split the minute string at the plus sign, converting each part to an integer, then combined them by adding the second part divided by 100. Thus, 45+1 becomes 45.01, and 45+10 becomes 45.1. Additionally, we filtered out games with missing data for game commentary and game statistics columns and addressed any repetitive game IDs (*gameID*).

The fourth script takes two files as inputs: 1) the JSON file that contains the box score (the final score of the games and the goal minutes); 2) The output of the third script which consists of *gameId*, minute, commentary, the names of home and away teams, and the date of the game. We find the event that has a goal by string matching, and subsequently comparing the minute of the event to the goal minutes from the box score. This results in a robust approach to initializing the homeScore and awayScore for each scoring event.

The fifth script is designed to filter games with more than two team names and games featuring incorrect minute stamps for given events, often stemming from erroneous data input on the *ESPN.com* website. For instance, discrepancies arise when the game title and commentary fail to align, or faulty entries within the commentary indicate "Match ends" at the start of the first half with a time stamp of 90. After resolving that problem, we observed that the starting minute of events does not consistently begin at 0. Therefore, if the first event of a game occurs at, say, minute 3, it appears to be the initial event, neglecting the preceding 3 minutes, which would affect the calculations of minutes spent at a given score and red card differential. To rectify this, we appended "First Half begins." to each game's minute 0. Additionally, we made sure to update the score differential starting from the row subsequent to the actual goal, because otherwise the shot that led to the goal would've counted towards the updated score differential, as opposed to the previous one during which it actually took place.

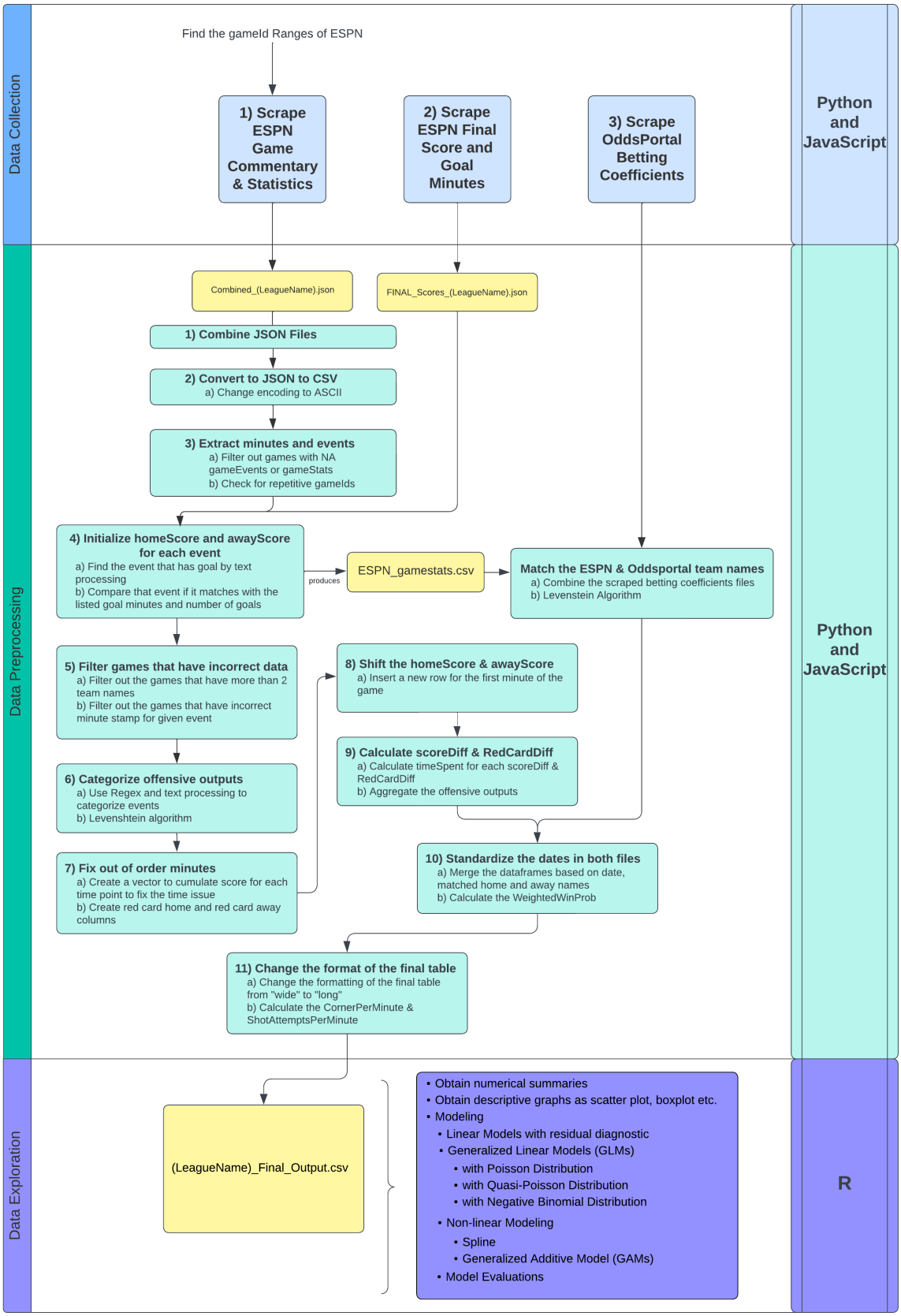

Figure 3: An image of data pipeline diagram

The sixth script employs regular expressions (Regex) and text processing techniques to classify offensive events based on game commentary, with the assistance of the Levenshtein distance algorithm. The commentary comprises a substantial portion of a text, and our task involves classifying whether the event described in a given minute falls into one of our interested categories. Regex is a sequence of characters that helps one to match and manipulate certain text arbitrary text patterns of interest. Here are a few examples of how we utilized Regex in this project: locating instances of minutes followed by events, identifying team names occurring after commas but not enclosed in parentheses, extracting integers from strings representing goals, and so forth. We identified some patterns by analyzing the commentary, which we subsequently employed in text processing. To detect goals, we searched for occurrences of the word "goal" within the commentary while excluding instances such as *"GOAL OVERTURNED"* and *"GOAL CANCELLED"*. For shot attempts, we searched for keywords "Attempt blocked", "Attempt missed", "Penalty missed", and "post". Red cards were identified through the presence of either a "red card" or a "Second yellow card". Saves were identified by the occurrence of "Attempt saved". Additionally, we identified occurrences of "own goal", "Foul", "Corner", "yellow card", "free kick", and "offside" by specifically searching for these keywords. When analyzing most of the aforementioned statistical categories, the team responsible for the event is enclosed in parentheses in the commentary (e.g. *"Foul by Raffael (Hertha Berlin)"*, or *"Attempt blocked. Marcel Schäfer (Wolfsburg) left footed shot..."*). However, in the case of own goals, corners, and offsides, the team name follows the comma rather than being enclosed in parentheses (e.g. *"Corner, Wolfsburg. Conceded by Jaroslav Drobny."*). Upon identifying such events, we extracted the team name from within the parentheses or after commas, respectively, and compared it with the listed team names using a matching algorithm according to the Levenshtein distance [14]. The Levenshtein edit distance is measure the similarity between two strings which calculates the minimum number of single character edits (insertions, deletions, or substitutions) required to change one string into the other. Within this project's scope, the Levenshtein distance serves to match a team name with the most similar counterpart from a list of scraped team names on top of ESPN for each game. This step holds significance due to variations such as letter conversion to ASCII and occasional discrepancies in team names, even when special accent characters are not present. For instance, while the commentary may mention *"1899 Hoffenheim"*, the team name in the list may appear as *"TSG Hoffenheim"*. Similarly, *"1. FC Köln"* in the commentary might correspond to *"FC Cologne"* in the team name list. The utilization of this algorithm effectively mitigates such discrepancies.

The seventh script corrects out-of-order minutes. We observed that ESPN has data input errors for some games, such as listing minute 47 after minute 50, which impacts our analysis. The eighth script shifts the home and away score by 1 as we need to adjust the indexing of goals because, despite intervals changing with each goal, we aim to incorporate the goal minute as part of the preceding interval, treating it as the interval's final event.

The ninth script computes the score differentials ($scoreDiff$), red card differentials ($RedCardDiff$), and $timeSpent$ along with offensive outputs accumulated during a given score and red card differential within the game. Each offensive output is recorded in separate columns, such as homeShots and awayShots, with the table presented in wide format.

The tenth script takes two inputs. The first is a file containing matched betting coefficients and team names, which we match with ESPN team names to Oddsportal team names using the Levenshtein algorithm. The second input is the output from the previous script. We standardize the date in these two files and merge them based on the date, and matched home and away teams.

In addition, we calculate the $Weighted.Win.Probability$ using the betting coefficients, more details on that calculation are provided later in this section.Finally, the last script transforms the format of the final table from wide to long. For instance, we now possess an extra column designating either home or away, along with a single column for Shots.

For the final data format that was utilized during statistical modeling, see Figure 4. The first two rows represent the statistics accumulated by home and away team, respectively, whenever their game ($gameId = 252449$) was tied ($Score.Diff = 0$ for both teams) and neither team had a red card ($RedCard.Diff = 0$). One can see that teams spent 58 minutes in that setting, during which the home team took 12 shots and 3 corners, while away team got 2 shots and 2 corners. Rows 3 and 4 correspond to the statistics accumulated by those teams in that same game, but during the 10 minutes when the home team had a 1-goal lead (notice $Score.Diff = 1$ for home team, $Score.Diff = -1$ for away team). Lastly, rows 5-8 describe the time period when the home team led by 2 goals ($Score.Diff = 2$ for home team, $= -2$ for away team), but 10 minutes into this period there was a red card shown to the away team, resulting in away team playing with fewer men for the remaining 12 minutes ($RedCard.Diff = 1$ for away team, $RedCard.Diff = -1$ for home team).

gameId	Score.Diff	RedCard.Diff	Weighted.Win.Prob	HomeAway	minutes.spent	Shots	Corners
252449	0	0	0.298	Home	58	12	3
252449	0	0	-0.298	Away	58	2	2
252449	1	0	0.298	Home	10	3	1
252449	-1	0	-0.298	Away	10	0	0
252449	2	-1	0.298	Home	12	1	0
252449	2	0	0.298	Home	10	1	0
252449	-2	1	-0.298	Away	12	2	1
252449	-2	0	-0.298	Away	10	4	1

Figure 4: Finalized data format used for statistical modeling

One variable we are yet to thoroughly describe is the weighted win probability ($Weighted.Win.Prob$). It was calculated based on prematch betting coefficients via converting those to probabilities of various game outcomes (home win, draw, away win) and assigning respective weights (+1 for probability of the team winning, 0 for draw, -1 for probability of a loss). For example, the betting coefficients for the game 252449 from Figure 4 - which took place between Hoffenheim and Koln in 2009 Bundesliga, with Hoffenheim as the home team - the betting coefficients were 1.87 for a home team win (meaning that one would win 1.87 times the amount they bet on that outcome), 3.5 for a draw, 4.22 for an away team win. The formula for betting coefficient for outcome A is $\frac{1}{P(A)}$, hence the actual outcome probability is simply the reverse of that, making the probability of home team win $\frac{1}{1.87} = 0.53$, draw - $\frac{1}{3.5} = 0.29$, away team win $\frac{1}{4.22} = 0.24$. Note that those don't add up to 1.0 because the betting coefficient reported for each outcome is a result of finding the best odds for said outcome across several bookmaking companies, which can differ to a certain degree. To obtain the weighted win probability of home team (in our case, Hoffenheim), we proceed to calculate a weighted sum of those probabilities as follows: $(+1) \times P(\text{home team win}) + 0 \times P(\text{draw}) + (-1) \times P(\text{away team win}) = (+1) \times 0.53 + 0 \times 0.29 + (-1) \times 0.24 = 0.29$. For the away team weighted win probability, given that the weights of outcome probabilities simply flip signs (it becomes "-1" for home team win, "+1" for away team win), we simply take the negative of that: -0.29. This way, weighted win probabilities close to 0 indicate a game between teams that are relatively closely matched in level, and a lopsided balance of power other-

wise (being strongly positive for the favorite and strongly negative for the underdog, respectively)

All-in-all, instead of simply analyzing the game totals (e.g. total shots in a game, total corners, etc), we break matches down into multiple intervals based on score and red card differential at the time, and examine the team's offensive output during each of these segments. Moreover, we account for teams' balance of power that helps gauge the likelihood of implementing a defensive tactic. The aforementioned aspects result in our approach acknowledging the fluidity of play and the evolving tactics employed by teams and individual players over the course of the match.

2.2 Statistical Modeling

2.2.1 Generalized Linear Models

Generalized Linear Models (GLMs) present an extension to classic linear models, where a special treatment might be developed for cases of non-continuous response variables. Each generalized linear model consists of three components: the random component, the systematic component, and the link function. Additionally, each generalized linear model uses a predictor X or vector of predictors $\vec{X}$ to predict a response variable Y. The random component defines the conditional probability distribution of the response variable Y given a predictor X or vector of predictors $\vec{X}$. The systematic component describes the relationship between the predictor and the expected value of the response variable. The systematic component is often written as $\eta = \mathbf{X}\beta$ where η is the linear predictor, $\mathbf{X}$ is the design matrix of predictors, and β is the vector of coefficients, resulting in a linear function of predictors with β acting as weights. The link function, known as g(.), connects the systematic and random components as follows: $g(E[Y]) = \eta$, where $E[Y]$ denotes the expected response value of the response ($E[Y]$) [16, p. 170].

2.2.2 Poisson GLM

Poisson regression provides a more suitable framework for count data as it explicitly accounts for key characteristics of the response, such as its discreteness and non-negativity. Poisson distribution is a discrete probability distribution, intended for random variables that describe counts of a certain event occurring either over a fixed interval of time, making it perfect for our setting where we model the number of shots or corners during a fixed period of time. Suppose that we have a random variable Y that we know takes on non-negative integer values. If Y follows the Poisson distribution with rate μ, denoted as $Y \sim Pois(\mu)$, then

$$P(Y = k) = \frac{e^{-\mu}\mu^k}{k!} \quad \text{for k = 0,1,2...}$$

The initial full GLM Poisson regression model equation considered in this work is:

$$\begin{cases} Y_i \sim_{ind.} \text{Pois}(\mu_i) \\ \log(\mu_i) = \alpha + log(\text{timeSpent}_i + 1) + \beta_1 \text{ScoreDiff}_i + \beta_2 \text{RedCardDiff}_i + \beta_3 \text{Weighted.Win.Prob.} + \beta_4 \text{H.A}_i \end{cases}$$

The $\log(\text{timeSpent}_i + 1)$ component represents the offset variable, which directly translated into modeling a per-minute event rate, e.g. shots or corners per minute. It is shifted by 1 to avoid the cases of log(0) when the period lasts 0 minutes (although such cases are really rare).

2.2.3 Overdispersion and Underdispersion

If $Y \sim \text{Pois}(\mu)$, the following properties hold: $E[X] = \mu$ and $V[Y] = \mu$. The $V[Y] = \mu \equiv E[Y]$ property shows how restrictive the Poisson distribution is as opposed to, for example, normal distribution (where $V[Y]$ is defined via a separate parameter σ^2, unrelated to the mean $E[Y] = \mu$). This restrictive nature of Poisson may lead to issues of overdispersion, which occurs when the variance of the response variable is greater than its mean ($V[Y] > \mu$) , and, less frequently, underdispersion ($V[X] < \mu$), which occurs when the variance of the response variable is less than its mean [11].

2.2.4 Negative Binomial GLM

The Negative Binomial distribution is commonly used to model count data with overdispersion. We suppose that Y_i follows the Poisson distribution and that its expected count μ_i^* is a Gamma-distributed and unobservable random variable with mean μ_i and a constant scale parameter ω. The setup for such negative binomial model is as follows [7, p. 432-433]:

$$\begin{cases} Y_i \sim \text{Pois}(\mu_i^*), i = 1, 2, ..., n \\ \mu_i^* \sim \text{Gamma}(\mu_i, \omega) \\ \log(\mu_i) = \alpha + \beta_1 x_{i,1} + ... + \beta_k x_{i,k} \end{cases}$$

We can write the negative binomial random component in shorthand as $Y_i \sim \text{NegBin}(\mu_i, \omega)$ where $\text{NegBin}(\mu_i, \omega)$ is the marginal distribution of Y_i that we receive from the specification of distribution in the above equation.

The negative binomial model specified as above has expected value $E[Y_i] = \mu_i$ and variance $V[Y_i] = \mu_i + \frac{\mu_i^2}{\omega}$. Since ω is restricted to values greater than 0, looking at the variance formula reveals that the negative binomial model type accounts for overdispersion. The negative binomial assumes that the variance is greater than the mean therefore it is only appropriate for modeling overdispersion and not for underdispersion.

2.2.5 Nonlinear Modeling Techniques

Unlike linear models, which assume a linear relationship between the independent and dependent variables, non-linear models allow for more complex relationships to be captured. Linear models assume that changes in the independent variable lead to proportional changes in the dependent variable, while non-linear models allow for more flexibility in capturing the relationship between variables. Non-linear models can take various forms, such as exponential, logarithmic, polynomial, sigmoidal, or other non-linear functions. The key difference between linear and non-linear models lies in the nature of the relationship they describe. In this work, we will focus on generalized additive models, which in their turn leverage smoothing splines [9].

2.2.6 Smoothing Splines

Splines are a flexible class of functions used for modeling potentially non-linear relationships between variables. Splines are made up of piecewise polynomial functions that are joined together at specific points called knots. These knots act as breakpoints where the polynomial segments are connected. Splines come in various forms, such as cubic splines, natural splines, and B-splines. For instance, in the context of regression splines, it's necessary to define a series of

knots, generate a sequence of basis functions, and employ least squares to compute the spline coefficients. Alternatively, a distinct methodology can be adopted with smoothing splines. For our research, we will use smoothing splines, where the main goal is to find the function g that minimizes

$$\sum_{i=1}^{n}(y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt, \tag{1}$$

where y_i are the true response values. The $\sum_{i=1}^{n}(y_i - g(x_i))^2$ part represents the classic sum of squared error, while $\lambda \int g''(t)^2$ is the penalty being imposed on overly wiggly fits, thereby avoiding overfitting and encouraging smoothness of the resulting function (hence the name "smoothing" splines). A smoothing spline ends up being simply a natural cubic spline with knots at every unique value of x_i.

2.2.7 Generalized Additive Models

Generalized Additive Models (GAMs) are a framework for extending multiple linear regression such that we allow nonlinear relationships between each predictor and the response, summing together the contributions from each predictor. A generalized additive model is written as follows: $y_i = \beta_0 + f_1(x_{i,1}) + \ldots + f_k(x_{i,k}) + e_i$ where each f_j for $j = 1, 2, \ldots, k$ is a smooth nonlinear function [16, p. 309-310]. Any smooth nonlinear function technique can be used to fit a function f_j, including the smoothing splines we discussed above, which is what we are going to use.

Generalized additive models offer several advantages over traditional linear models, especially when dealing with complex relationships and non-linearities in the data. First, they fit a nonlinear f_j to each X_j, they will automatically model nonlinear relationships that would not be included in standard multiple linear regression. Additionally, these nonlinear fits can potentially generate predictions that are more accurate than those made by an analogous multiple linear regression model. Finally, the additive nature of the model allows us to examine the partial effect of each covariate on the response. While the description of generalized additive models above directly extends from multiple linear regression, we can apply GAM techniques to any generalized linear model or hierarchical generalized linear model. Furthermore, we can restrict the application of GAM techniques to any subset of covariates in our model.

2.2.8 Criteria for Model Comparison

Initially, we employed both the Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC) for model selection among a set of candidate models. These criteria consider two main components in their calculations: (1) the number of parameters or explanatory variables in the model and (2) the quality of fit, measured using the likelihood approach [12]. However, we later decided to use only BIC instead of AIC because BIC imposes a more substantial penalty for the number of parameters, favoring simpler models more aggressively. Additionally, BIC is asymptotically consistent, meaning it tends to select the true model as the sample size increases, provided the true model is among the candidates. BIC balances the trade-off between goodness of fit and model complexity by penalizing the addition of parameters to the model. A lower BIC value indicates a better balance between these factors, leading us to select the model with the lowest BIC value as the most appropriate [17].

3 Results

3.1 Model comparison

Table 1 below illustrates the BIC values across all considered modeling approaches and all five soccer leagues. One can see that Negative Binomial approach outperforms regular Poisson model, while GAM does considerably better than linear approaches. That results in Negative Binomial GAM being the best model across all five leagues. It involves the most amount of parameters and represents the highest complexity out of all the models considered (trying to accommodate both the overdispersion and non-linearity), but nonetheless provides such an improvement to the quality of the fit that even the harsh penalty the BIC imposes on model complexity doesn't drop it down in the model hierarchy. This aspect bodes well for the model's capability to generalize to new data and avoid overfitting.

	Linear		GAM	
Distributions / Leagues	Poisson	Negative Binomial	Poisson	Negative Binomial
Bundesliga	115241.4	114067.0	114747.3	**113704.7**
La Liga	136271.8	134899.9	135648.0	**134411.2**
Ligue 1	130402.5	129032.6	129996.8	**128706.0**
Premier League	120879.4	117778.4	120346.4	**117469.3**
Serie A	142586.0	140536.6	142053.1	**140139.2**

Table 1: BIC values comparing all considered models across all leagues

3.2 Covariate Effect Estimates

Having determined Negative Binomial GAM as the best model, Tables 2 and 3 demonstrate the statistical significance for each covariate of interest, while Figure 5 illustrates the nature of their effects. In particular, Tables 2 and 3 shows that each numerical covariate (score differential, red card differential, betting coefficients) had a statistically significant effect on shots and corners, with score differential requiring the most "effective degrees of freedom" (edf) - meaning that it had the most non-linear nature out of the three. To confirm it, one could look at Figure 5, where score differential exhibits much higher level of wiggliness compared to red cards and betting coefficients. These results are quite consistent across all five leagues. For more detailed discussion of the intuition behind the nature of the effects being observed, see Section 4.1.

Approximate significance of smooth terms				
	edf	Chi.sq	p-value	Signif.
s(ScoreDiff)	7.820	1067.4	<2e-16	***
s(RedCardDiff)	1.849	736.5	<2e-16	***
s(WeightedWinProb)	5.161	3131.4	<2e-16	***
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				

Table 2: Significance of smooth terms in Negative Binomial GAM with shots as the response variable, Bundesliga.

Approximate significance of smooth terms				
	edf	Chi.sq	p-value	Signif.
s(ScoreDiff)	6.902	839.4	<2e-16	***
s(RedCardDiff)	1.906	242.5	<2e-16	***
s(WeightedWinProb)	4.772	1613.0	<2e-16	***
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1				

Table 3: Significance of smooth terms in Negative Binomial GAM with corners as the response variable, Bundesliga.

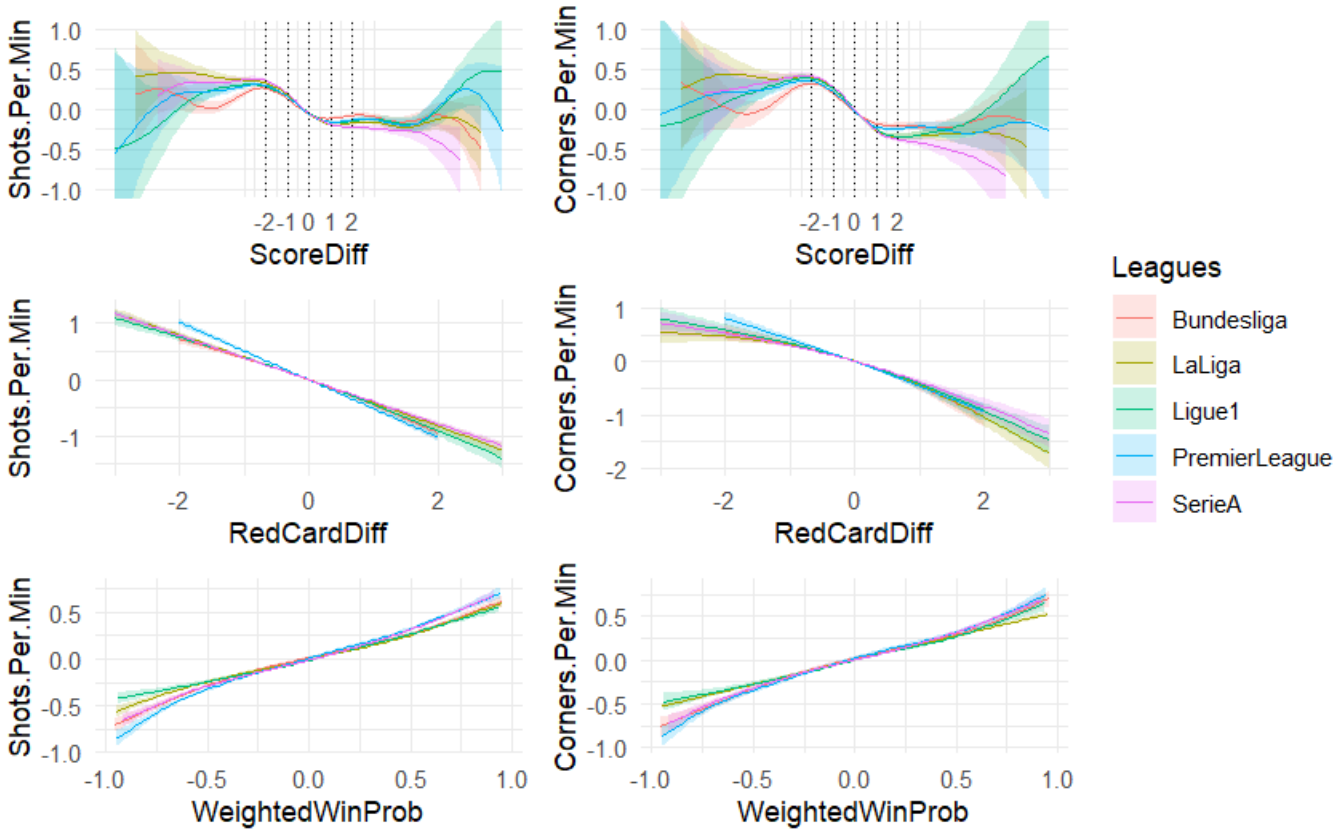

Figure 5: Plots for effects of hypothesized "prevent defense" covariates (score differential, red card differential, weighted win probability) on shots and corners per minute, across five major European soccer leagues.

4 Discussion & Future Work

4.1 Discussion of Results

Having conducted an array of modeling techniques which were tailored to the count data we used as response variables (number of shots and corners per minute), we found the Negative Binomial GAM to perform better its Poisson and linear counterparts. Negative binomial allowed to account for the overdispersion in the data, while the smoothing splines turned out to be more appropriate for the non-linear effect of the scoring differential (as could be seen on Figure ..). The effects of red cards and betting coefficients, on the other hand, looked relatively linear and likely could've been modeled via respective linear terms.

As for the nature of the effects, of utmost interest was the impact of score differential. Given that large score differentials (e.g. ± 3 and larger) aren't as prevalent as the smaller ones, it isn't surprising to see the most narrow confidence bands in that -2-to-2 range, where the results are to be trusted the most (as opposed to the large score differentials, where the estimates are the least stable). Across all five leagues, and for both shots and corners, one can notice a clear negative trend in offensive production from a team that's trailing in score (e.g. -2 or -1) compared to team that's leading ($+1$, $+2$). That confirms the hunch of leading teams being more likely to implement the prevent defense strategy, while the trailing teams become more aggressive as they try to catch up in score.

What's also curious is the fact that the drop-off in production from a team that leads by 1 goal ($ScoreDiff = 1$) as opposed to a team leading by 2 ($ScoreDiff = 2$) is rather negligible, compared to incremental drop-offs observed all the way from -2 to 1. Given that a win is worth 3 points (2 more than a draw), whenever one goes up in the score - that lead becomes extremely valuable to protect, regardless of how big it might be (e.g. be it 1 goal or 2 goals). That typically results in a binary (rather than incremental) switch in team's mentality to start playing a more careful, defensively responsible, style of soccer. Therefore, after the notable drop-off in offensive output from a tied game to a team that leads by 1 goal, the offensive output doesn't change much between teams leading by 1 and 2 goals. If you're the trailing team, on the other hand, although you technically have no lead to protect in either case (whether you're down 1 or 2 goals), typically teams don't abandon defense instantly upon going down 1 goal so as to keep the game within reach at least until the final minutes (when they could make their final push for a draw). If the team goes down 2 goals though, they tend to start playing much more desperately right away, strongly shifting towards offense while for the most part abandoning their defensive responsibilities. Reason for that is because, at that point, game starts getting out of reach, and the risk of allowing another goal (to go down 3 instead of 2) isn't that much worse than the current, already dire, situation. So the upside of potentially scoring a goal by increasing your offensive output outweighs the downside of allowing another goal.

As for the effects of red cards and betting coefficients, their nature was quite intuitive. A team with higher number of red cards, hence fewer men on the field, tends to produce less offensive output than their opponent. A team that's favored to win the game according to the bookmakers typically outperforms their opponent on offense. Despite a slight non-linear bend in the GAM estimates of relationships between red cards and offense in some of the leagues, the aforementioned effects looked rather linear for the vast majority of the settings. Therefore, unlike scoring differential, they likely could be modeled linearly, without the need to resort to smoothing splines.

4.2 Limitations

One limitation of this work is the potential violation of independence assumption when conducting inference and model estimation. Each time period spent within a game at the same score and red card differential results in two observations - one for home team and one for away team. We treat those independently, although a strong argument can be made for dependence of offensive outputs between two teams competing against one another for those outputs in the same game. Methods to potentially address it will be discussed as part of future work.

4.3 Future Work

One approach to address the dependence issue is to introduce random effects to represent the same game or same time period within the game, which is conveyed by hierarchical/mixed-modeling approach [15]. Moreover, given that we could treat each time period as giving rise simultaneously to multiple outputs (e.g. shots for home team, away team; corners for home, away team), multivariate regression approaches could be leveraged [10], with multivariate Poisson log-normal approach being more tailored toward count response data [6].

Moreover, given some of the "switch" tendencies observed in team's choices of playing tactics based on score differential, we will attempt treating the score difference as a categorical predictor. In addition, it could also allow us to conduct variable selection in determining the score differentials at which there's true deviations from typical offensive output during a tied game (so we will treat score difference of "0" as the reference category).

Another avenue would be to account for the exact minute in the game when the event (shot or corner) occurred, rather than just accumulating the total across a time period. That could help studying the tendencies in offensive outputs as the game nears the end, which could be different based on the increased sense of urgency with the clock running out.

Most importantly, our final goal is to create a statistical metric, or adjustment, that provides a more objective picture of teams' performances in a game given the contextual information. As opposed to treating each shot or corner equally, it would involve accounting for the score and red card differential at which the offensive output was accumulated, therefore adjusting for the prevent defense factor.

Lastly, with the current work mostly focusing on national club leagues, where there's no concept of playoffs or elimination games, and there's clear point denomination for each outcome (3 points for a win, 1 for a draw, 0 for a loss). A great extension would be to study the dynamics of other competitions such as, for example, Football Association Challenge Cup (FA Cup), World Cup or Champions League playoffs, where the format is notably different, with more reliance of score differential - not points - as a way to determine who proceeds to the next round. It would be curious to investigate whether the nature of the covariate effects on the offensive outputs remain unchanged, especially that of the scoring differential.

References

[1] Csv python documentation.

[2] Espn: France 2-1 england (dec 10, 2022) commentary.

[3] One month on: 5 billion engaged with the FIFA world cup qatar 2022™.

[4] Pandas 2.2.2 documentation.

[5] The selenium browser automation documentation.

[6] Aitchison, J. and C. Ho (1989). The multivariate poisson-log normal distribution. *Biometrika 76*(4), 643–653.

[7] Fox, J. *Applied regression analysis and generalized linear models* (Third edition ed.). Sage.

[8] Giannakoulas, N., G. Papageorgiou, and C. Tjortjis. Forecasting goal performance for top league football players: A comparative study. In I. Maglogiannis, L. Iliadis, J. MacIntyre, and M. Dominguez (Eds.), *Artificial Intelligence Applications and Innovations*, pp. 304–315. Springer Nature Switzerland.

[9] James, G., D. Witten, T. Hastie, R. Tibshirani, et al. (2013). *An introduction to statistical learning*, Volume 112. Springer.

[10] Mardia, K. and J. Kent (1979). Bibby jm. multivariate analysis. *New York: AcademicPress*.

[11] McCullagh, P., J. A. N. (1989). Generalized linear models (chapman & hall/CRC monographs on statistics and applied probability): 9780412317606: McCullagh, p.: Books.

[12] McElreath, R. *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. CRC Press. Google-Books-ID: T3FQDwAAQBAJ.

[13] Mead, J., A. O'Hare, and P. McMenemy. Expected goals in football: Improving model performance and demonstrating value. *18*(4), e0282295. Publisher: Public Library of Science.

[14] Navarro, G. A guided tour to approximate string matching (2001). *33*(1), 31–88.

[15] Raudenbush, S. W. and A. S. Bryk (2002). *Hierarchical linear models: Applications and data analysis methods*, Volume 1. sage.

[16] Sohil, F., M. U. Sohali, and J. Shabbir. An introduction to statistical learning with applications in r: by gareth james, daniela witten, trevor hastie, and robert tibshirani, new york, springer science and business media, 2013, $41.98, eISBN: 978-1-4614-7137-7. *6*(1), 71–72.

[17] Stoica, P. and Y. Selen. Model-order selection: a review of information criterion rules. *21*(4), 36–47. Conference Name: IEEE Signal Processing Magazine.

[18] Whitmore, J. What are expected goals on target (xGOT)?