Squad Analysis Tech Write-up

1 Introduction

This document is a supplement to the Wigan Squad analysis write-up, that paper focuses on the footballing aspects of the project whereas this will focus on the technical specifics. This project's codebase was created almost entirely from scratch with the exception of the stealth_scraper() function, which I developed for a previous project. The Player Stats and Team Data/Performance datasets (which were all borrowed from previous projects I've worked on) were used in feature selection to understand the correlation between the features and Goals Scored (GS) and Goals Against (GA) for the top 5 leagues. The rest of the datasets were built during the project to aid analysis. In this supplement I will give more detail on the feature selection process and the data collection & formatting, with the aim of explaining the motivations for some of the technical decisions made.

2 Feature Selection

The first stage of this project was interestingly not Data Collection as usual, instead I already had all the data I needed for the feature selection process I was pursuing. The final dataset I would need would have the stats for each team in the top 5 leagues between 2017 and 2022 and each team's GS and GA. I already had all the player stats for this period (Player Stats 17-22.csv) and the GS/GA for each team in this period (Team Performance.csv) from previous projects. To create the dataset needed I grouped the player stats by season and team, and merged that grouped data with the goals conceded for that team in that season. This grouped data will unfortunately contain some inaccuracies as there will be a small number of players missing from the player dataset. This is an unfortunate reality and is what motivated me to use goals scored from the grouped players rather than the team performance, as if a player is missing so will their goals in the final dataset. This means that there might be a slight misalignment between stats and goals conceded. However, as this is an MVP and because accessing in-depth stats for all teams in the top 5 leagues over the past 5 years is not the simplest task, I felt it was an acceptable trade-off.

When building the dataset, the most common problem I run into when working with football data once again presented itself. Whether you're working with team or player data, two sources will often use the different names for the same player/team i.e. Manchester Utd, Man U, Manchester United etc. To get around this problem I use fuzzy matching, when presented with a string this algorithm assigns similarity scores between that string and those in the set provided. I extract the most similar string to the one presented and if the string extracted is over a confidence score of 75 then it is added to a dictionary of names to be changed.

The feature selection process was motivated by a desire to understand some of the less obvious correlations between a team's stats and their GS/GA. To do this I first normalised the data, then used KNN feature permutation to extract the feature importance. This process permutes through each feature and measures the difference in negative MSE when the feature value is changed. Features that cause a large reduction in negative MSE when changed are assigned high importance values and vice versa. Negative MSE was chosen over Negative RMSE as I wanted to heavily penalise large errors and therefore emphasise those features that cause the largest changes in accuracy. This process is repeated 5 times for each feature and 5 times overall, with the scores for the process repetitions averaged.

The interpretation of these feature importance scores combined the raw numbers with my intuition as to what stats should be used in the final model. The top feature importances can be seen in the original paper in table 1. The top features were selected as the top 4/5 features before a significant drop-off in importance. I allowed progressive passes in for offensive stats even though it it rated as lower than passes into the penalty area as I feel it is an important stat for analysing attack quality and volume. In the same vein I decided to remove crosses into the penalty area from the defensive stats. This is because the number of crosses a player gets into the opposition box doesn't feel particularly relevant for how many goals their team concedes.

3 Data Collection & Formatting

All the data scraped for this project is sourced from FBref.com, however in the original design I was utilising data from WhoScored.com aswell. Due to this fact I had to use a Selenium based scraper to get around the CloudFlare protection that WhoScored use. This form of scraping allows you to get past most anti-scraping proxys like Cloudflare but slows the runtime and makes the code less scalable. As this product is an MVP on a small scale I felt it was an appropriate trade-off to make. Scraping the data from FBref involved isolating the data I wanted using BeautifulSoup before adding it to a Pandas Dataframe.

From FBref I scraped and built two datasets, the first contains data from the Championship goal teams described in the main document. The second contains all the players from the leagues targeted that match Wigan's wage structure.

By far the biggest drawback for the data scraping package is its scalability, this isn't an issue due to the small size of task taken on in this project however if I wanted to scrape a larger volume of data the package would certainly have to be reworked. An inefficiency in the code that I couldn't find a way to work around is the extraction of the text from a column of soup. Currently, I use list comprehension to extract the text from the soup and convert it to the necessary type, this can be seen in the extract_stat() function in the championship_data_collection file, but iterating over each column does not feel like the most efficient method.

4 Conclusion

This very brief supplement outlines and explains some of the justifications for the technical decisions made in the creation of this project. It also shows that the biggest issue with the codebase is scalability. This being said I feel that for a task which involves scraping all relevant player stats from three leagues a runtime of $\sim$300 seconds is somewhat acceptable. This project was really interesting from a technical perspective and gave me more insight into working efficiently with football data to create a product under time-constrained conditions.