Lorenzo Dube

ME397

4.29.23

Final Project

For the project, I looked to analyze MLS best XI teams since 1996 to search for differences between the actual Best XI at each position, and what I model to be the top candidates for Best XI team at each position. I intended to allow for my work to be readily explored by any interested by showcasing it in a webapp what would display the true best XI winners and the top 3 candidates per the model for each position.

To complete this task, first I needed data. I leveraged the Kaggle MLS dataset at this website: https://www.kaggle.com/datasets/josephvm/major-league-soccer-dataset , and also fbref.com for the last two years of data. For the Best 11 teams, I used the table at this link: https://en.wikipedia.org/wiki/MLS_Best_XI . There were a multitude of issues in cleaning the data. There were inconsistencies in team names, team abbreviations, and player names across datasets. For the best 11 data, I had to create a script to digest the data and break it into individual players per teams as the data pasted into a csv file was quite poorly organized. To combat the different naming conventions, I created multiple dictionaries to translate the team names and team abbreviations from all data sources to consistent naming conventions such that data could be merged. For players, I was able to find a method that removed all accents on player names for consistency.

I had to be sure that statistic columns were names consistently between data from difference sources. The output of the data cleaning was a consistent dataset used for modeling

the best 11 logistic regression. Additionally, as a note, due to a small number of inconsistencies in player names, I went into the dataset and individually changed about a dozen name spellings from the data that was downloaded from Kaggle – this was much faster than writing a couple of lines of code to search through each row of the data each time the script was run considering the small number of instances where this was necessary. This was necessary to be sure that true Best 11 players were marked in the data.

The data cleaning took much longer than expected - 20+ hours of cumulative work. As a result, I did not put the data into an SQL database and perform queries on the data as indicated on my proposal. By the end of the cleaning, there would have been little value added with hosting the data in an SQL due to the size of the dataset and the time already invested in cleaning the data without any modeling or webapp building performed.

Before modeling, I looked at the box plots for available stats between Best 11 and total player population for each position, this led to outputs similar to the following where on the left is the general population of MLS forwards, and the right are Best 11 winners.

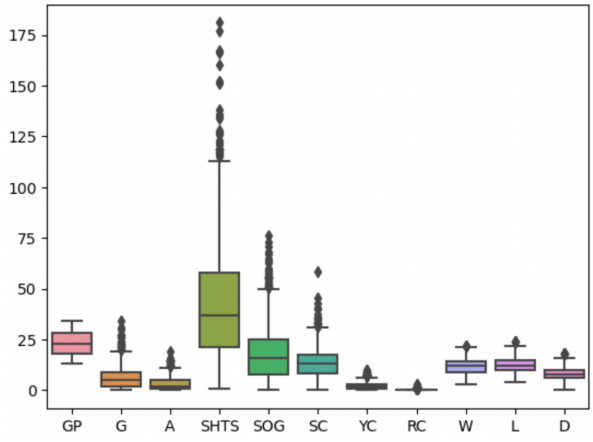

Figure 1: Population of all Forwards

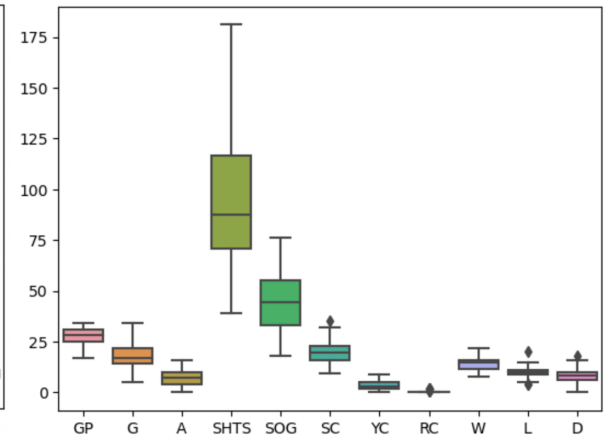

Figure 2: Best 11 Forwards

There are some clear differences between the two, which led me to think I was on the right track (at least for that position). I created a logistic regression model for each position, correlations were performed for each position to understand what variables might be most important for a model.

Table 1: Correlation Table for Goalkeeper Statistics

	GP	SHTS	SV	GA	GAA	W	L	T	Sv%	best_11
GP	1.000000	0.755510	0.725057	0.716994	-0.209016	0.670498	0.518722	0.538821	0.157465	0.174294
SHTS	0.755510	1.000000	0.974048	0.790223	0.163106	0.428763	0.654353	0.212319	0.202494	0.130636
SV	0.725057	0.974048	1.000000	0.649142	0.008225	0.472576	0.539966	0.225289	0.402201	0.184765
GA	0.716994	0.790223	0.649142	1.000000	0.489517	0.228144	0.795393	0.237463	-0.307108	-0.049967
GAA	-0.209016	0.163106	0.008225	0.489517	1.000000	-0.473246	0.455384	-0.327826	-0.643820	-0.231787
W	0.670498	0.428763	0.472576	0.228144	-0.473246	1.000000	-0.021171	0.091001	0.349760	0.283170
L	0.518722	0.654353	0.539966	0.795393	0.455384	-0.021171	1.000000	-0.052231	-0.277658	-0.059199
T	0.538821	0.212319	0.225289	0.237463	-0.327826	0.091001	-0.052231	1.000000	0.177309	0.060060
Sv%	0.157465	0.202494	0.402201	-0.307108	-0.643820	0.349760	-0.277658	0.177309	1.000000	0.241142
best_11	0.174294	0.130636	0.184765	-0.049967	-0.231787	0.283170	-0.059199	0.060060	0.241142	1.000000

From this point, I calculated the probabilities based on the log odd outputs, but the differences in the output values were subtle and so unused in further evaluation. As an output of this analysis, I created dataframes for the top three predicted Best 11 team members based on the models for each position, and the true best 11. These dataframes were transferred to a pickle file to be used for the web app output.

The webapp was created using streamlit and pulls the aforementioned pickle files and opens them as dataframes. The user of the webapp is allowed to select a position and a given year for the data. Based on these inputs, the dataframes are altered to show the true and top three model predicted Best XI players.

All scripts and data for the project were git version controlled and at the completion of the project locally, pushed to a GitHub repository.

For results, the models created worked well for Forwards, less so for Midfielders and Goalkeepers, and poorly for defenders. The models we're typically good at avoiding false positives (predicting players were Best 11 winners that weren't) but had very many false negatives relative to the number of true positives. Having data more specifically tailored to the positions in the future would likely help to solve this problem. With the amount of time invested in data cleaning, it did not make sense to try to reset and search for new data given the project time constraints. The model outputs can be seen in the confusion matrices:

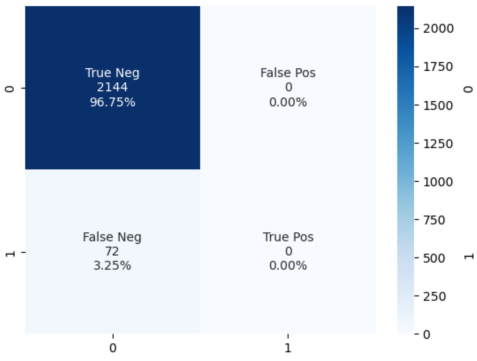

Figure 1: Defense Model

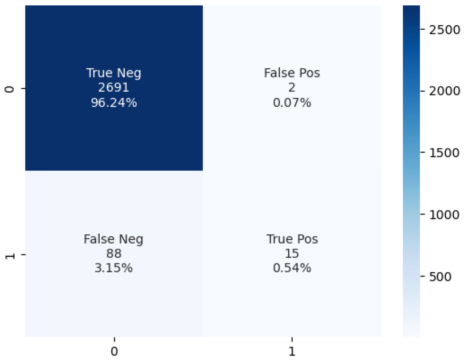

Figure 2: Midfield Model

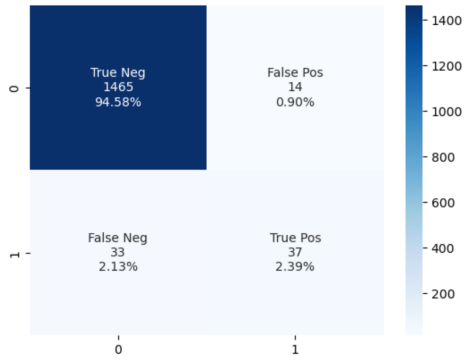

Figure 3: Forward Model

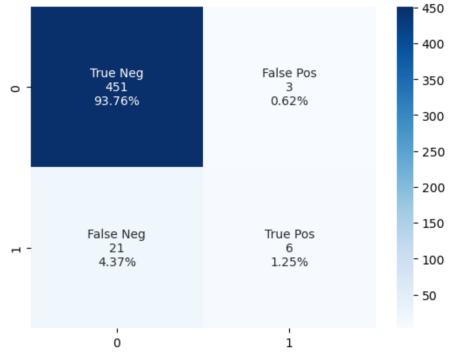

Figure 4: Goalie Model

The web app output was successful, and is accessible at this link: https://lmdube21-dube-me397-final-app-6ucqf7.streamlit.app/

For future work, I would likely use all data from the same source to allow for substantially less time cleaning the data. This could be done using fbref.com for all data. Statistics available on this site that would be more relevant for Midfield and Defense that would likely improve he models as a more accurate reflection of position responsibilities (ex. Tackles is likely a better metric for defenders than goals scored). This would allow for additional insights.

Output	Filename (with folder where relevant
GitHub Repository (all data and files can be found here)	https://github.com/lmdube21/DUBE_ME397_FINAL
Raw Data	See raw_data folder in repository
Data Cleaning Script	data_cleaning.py
Cleaned Goalkeeper Data (output of data_cleaning.py)	final_data/goalkeeper_data.csv
Cleaned Field Player Data (output of data_cleaning.py)	final_data/field_player_data.csv
Analysis/Modeling Script	analysis.py
Top 3 Field Players for Best 11 as Predicted by Model for Each Year (output of analysis.py)	final_data/model_results_field.pkl
True Best 11 for Field Players (output of analysis.py)	final_data/best_11_results_field.pkl
Top 3 Goalkeepers for Best 11 as Predicted by Model for Each Year (output of analysis.py)	final_data/model_results_gk.pkl
True Best 11 for Goalkeepers (output of analysis.py)	final_data/best_11_results_gk.pkl
App Script	app.py
WebApp Site	https://lmdube21-dube-me397-final-app-6ucqf7.streamlit.app/