Project problem statement

The main goal of the project is to measure football players' performance. In order to do that I built the model to predict the number of goals scored for a season. Interpretation of the model's results allowed me to understand how different parameters influence the number of goals scored.

The result of this project can be useful for administration staff: managers, club administration, scouts and agents. For example, managers can understand what parameters payers should improve (number of passes, tackles, fouls, etc.) to score more goals. Club administration can understand what they should improve (venue surface, venue capacity, etc.). Scouts can search for players of a certain nationality, age or current team.

Background on the subject matter area

Data science will work for this project as there are a lot of ways how to measure players, games and other related information. All that information can be quantified. Furthermore, there are a lot of data sources for that kind of information.

There are a lot of similar systems which gather lots of statistical information about players and games (Transfermarkt, Wyscout, Sofascore). Those apps are mostly focused on gathering statistical information not predictions. Not every football club requires advanced analytics, like predictions. That's why apps that can predict or interpret the results are developed on demand from more successful clubs.

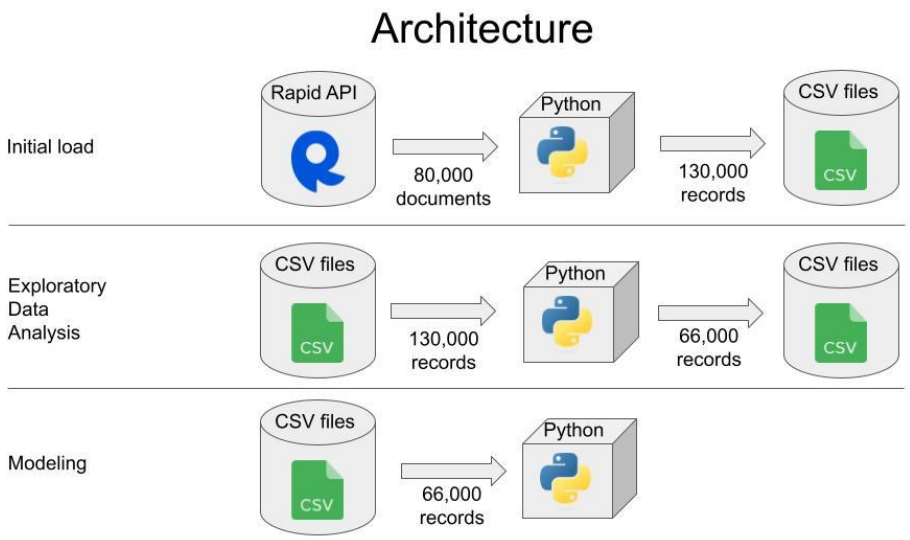
Details on dataset

I got the data from the Rapid API platform using an API interface. Rapid API is a platform that hosts various data sources. I used API-Fottball API developed by API-Sports.

It contains all kinds of information. I was interested in: seasons, leagues, teams, players, players' statistics. The data was collected in a semi-structured format (JSON documents). Then it was transformed into a structured format that was saved into .csv files.

Summary of cleaning and preprocessing

The architecture of the data processing is described in the image below.



I uploaded 80,000 JSON documents using the API (leagues, teams, players, players statistics). Then all data was converted from semi-structured data into a structured one. The result of that action was 130,000 records in total. Then all information was joined together. The resulting dataset was 66,000 records.

When I got one data set I performed EDA which includes cleaning and transformation. As a result of that action, I reduced the number of columns by 59% (from 79 to 33) and was able to retain 99.9% of rows (which means the quality of the data was really good).

Drop columns:

- All values are the same
- A lot of distinct values (categorical)
- With a very weak correlation
- A correlation was not statistically significant
- Multicollinear features
- Not relevant for ML (pictures, names, IDs, duplicated info)

Drop rows:

- Key values are null
- Duplicates
- Negative numeric numbers

Remove values:

- Special characters
- Measures (cm, kg)

Update values:

- Fill nulls with 0 for numeric
- Fill nulls with text for categorical

Insights, modeling, and results

I started the process of modeling with feature engineering. I created 2 features: the year when teams were founded binned based on historical events and if the player is a foreign one.

I created a preprocessor to manage different types of features: numeric (applied scaling), categorical (applied one hot encoding), ordinal (applied ordinal encoding) and removed the rest of the features (features that were helpful for analytics but not modeling).

I compared several algorithms: Linear Regression, KNN (K Nearest Neighbours), Decision Tree, SVR (Support Vector Machines Regression). The best one was the Decision Tree and the worst one was the Linear Regression.

I used Linear Regression (analyzed the coefficients) and Decision Tree (analyzed the rules) to interpret the results.

Findings and conclusions

Results:

- I was able to get the list of the most important features using 2 methods: KBest and PCA.
- Linear Regression interpretation: to score 1 goal a player needs to make 3 shots; to score 1 goal a player needs to make 13 assists; Colombians score more goals than other nations; the Spanish league is the most scoring one; to score more goals a player needs to be Barcelona player.
- Decision Tree interpretation: a player will score less than 5 goals if he plays less than 50% of games; a player could score 9 more goals without injuries during the season; players of the starting lineup could score 14 goals more; a player could score 8 more goals if his weight is between 65 than 83 kg or age less than 33; playing in Manchester United a player could score 9 more goals, in PSG 8, in Ajax or Bayern Munich 5.

Future improvements:

- Add more engineered features: number of foreigners in the team, number of games.
- Try more complex algorithms, like Neural Networks. It can help to get more accuracy but won't help with the interpretation.
- Try to create a model based on match results, not season results.
- Get more expensive data from statistical portals. It will improve the quality and complexity of the data.
- Get anatomic data, like foot size. Get genetic data, like bone structure and ligament strength.