Developing a Model to Predict Premier League Match Outcomes

Introduction

The English Premier League (EPL) is by far the most watched sports league in the world, attracting a global fanbase in the hundreds of millions every year. Football (soccer) is the world's game, and over the years data-driven analysis has permanently altered the sport's landscape. The goal of this project is to build supervised machine learning systems that predict match outcomes using rolling team performance statistics, bookmaker odds, and supplemental variables such as squad valuations.

While predicting football results is inherently uncertain, this project demonstrates how data-driven models can quantify competitive form using real-world sports data. Beyond its entertainment value, our framework illustrates practical machine learning workflows - from raw data ingestion and preprocessing, to feature engineering, to training and evaluation - in a reproducible, usable manner.

It should be noted that this project is also our final project for COE 379L: Intro to Machine Learning with Dr. Bui. The primary improvements over our baseline project for the other class will be containerization, developing an inference server, and significantly enhanced readability and coding practices.

Data Sources

We are using Football-Data.co.uk as our primary data source for Premier League match statistics. Each CSV summarizes several match statistics (match outcome, goals, shots, etc.) for all 380 matches in a Premier League season, for both the home and away teams. Datasets date as far back as the 1993/1994 season.

We will also use other data sources to supplement our main data source. For example, we will use TransferMarkt to gather information about total squad value for a given season.

Methodologies

Pre-Processing

For a given season, we exclude the first N games of each team to reserve for form. We then use the first 70% of the dataset for training, and the remaining 30% for testing. We use a temporal train-test split since a season naturally has a chronological ordering to it.

If the data for the season configured in config.py has already been loaded and processed, we will just use the processed data. Otherwise, we will have to download it and pre-process it before training our model with it.

For feature engineering, we use a rolling average approach, where over the last N games, we keep track of the averages of several statistics for a given team, including but not limited to wins, goals scored, shots on target, and fouls. For a given match, we engineer the dataset to include these stats for both the home and away teams, and train our model on the relevant features to predict the outcome using ternary classification. We can then supplement these form features with features from other datasets, such as total squad value and injuries.

Models

After pre-processing, we will train several machine learning models on our data, and then we will evaluate each model on our holdout data. We plan to generate several models, including logistic regression, XGBoost, Random Forest, Support Vector Machine, and Feedforward Neural Networks. We will then compare the accuracy of each of the models to determine which model was the best at predicting Premier League match outcomes.

Deliverables

As previously mentioned, we will implement containerization and an inference server for this project. Here's a high-level summary of the endpoints I'll implement:

- `GET /summary`: Summarizes each of the models we developed, specifically outputting the average accuracy our models had for each of the seasons.
- `GET /summary?model=<model>`: Prints the accuracy of the specified model over the given years, as well as denoting the accuracy of the model for every year it was trained on
- `POST /inference -d <data>`: Given the season, the teams, various statistics, and which model to use, predict the outcome of a match

The README will offer much more verbose explanations, but the main deliverable is a fully-functioning inference server that provides the user with information about all the models we used, as well as allowing the user to use the models they want to predict premier league match outcomes.

Here is a list of the other deliverables that will be found in the repository:

- Jupyter notebook (ipynb) file
- Report (PDF)
- Inference server (Flask API)
- Demonstration video link (YouTube) will be found in the README