Project Proposal

1. Data

My dataset is called: European Soccer Database 25k+ matches, players & teams attributes for European Professional Football. The dataset is from Kaggle.com, link is below:

https://www.kaggle.com/datasets/hugomathien/soccer

This is the EA sport FIFA soccer database for data analysis for machine learning. They offer 25,000 matches, over 10,000 real players and top 11 European Countries with their lead championship league. Within the league, they offer starting line-up, record of matches events, even betting odds. They started collecting data from 2008 until 2016 (most recent update is 16th Oct 2016).

I believe this is the best soccer data set systematically collected in Kaggle (They were give golden with 3000+ upvotes). Unfortunately, they don't have recent 2022 update, but 2016 is a close timeline where the peak of current football generation was

2. Questions

My initial idea is to base on the line-up, league, match events and betting odds from 2008 to 2016, make a prediction of who is the most likely to be the champion in English Premier League in 2017, then compare it with the real 2017 record. I would like to use the same data to make predict who will be wining this year 2020 English Premier League, but the result will likely be inaccurate due to a gap in data.

Some hypotheses:

- The player stats will most likely affect the team performance (goal/match, assist/match, hour played, number of time that they're in the starting line-up)
- The team history will also affect the team performance (league title, standings last few years, ..)

Some questions:

- Which of the quality in a player that affect the team performance the most? Example: Central Back will be number of tackles, number of hours played, but Central Forward will be goal scored, goal assisted. This will be different across positions. We need to rank the factor based on the importance, the significance of the contribution in the model.
- How do I address the team history with the team performance? Will that be the case where the winning team last year should also be wining this year?
- Other factors, like betting odd (this one is new). Do these factors also contribute to the winning chance of a team? Higher bet is favorable to win but sometimes the underdog can make miracle happen. Like 2016 with Leicester City winning the Premier League out of nowhere. How can we investigate this scientifically?

4. Models

This problem that I investigate is a predicting win/lose odd for a soccer team in a league. This will be categorized as supervised machine learning model on the training data and classify the probability as either Won or Lost. On top of my head, I think I can use:

- Naïve Bayes: a classification algorithm based on Bayes Theorem, that determines the probability of winning or losing an Opportunity given that each predictor variable has taken on a certain value.
- Logistic Regression: A generalized linear model but with a binary outcome, which is transformed into a probability using the sigmoid function. The 'weight' for each predictor variable is determined by the model in order to reduce the error between the actual and predicted values.
- Extreme Gradient Boost: Builds an ensemble of Decision Trees in a sequential manner, where the residuals of each model are fit in the subsequent model.

This will be susceptible to change during working with the data. Naïve Bayes might be simple but can work best out of the three. There might also be more models that I can test in the future that is not included here.

Conclusion:

My passion is soccer (European football). I would like to use data science to investigate the matches, teams, results systematically to give me a guess of who will be winning the next league title. Right now, my goal is to develop a model that works for year 2017-2018 since we only have data from 2008-2016. This will be a supervised learning task because we have history and real data in comparison. Later, if possible, I would like to predict this year's 2022 wining champion but lack of 2017-2021 data might impact the result.