Predicting English Premier League Results

Capstone Project Proposal

The English Premier League is an English professional league for football and it is English's primary football competition. Contested by 20 clubs, it operates on a system of promotion and relegation with the lower leagues. Being the most watched sports league in the world, it boasts of unpredictability as seen in the previous season when Leicester City beat odds of 5000/1 to win the league.

Problem

To try and build a model that would try predicting the result of a match based on previous meetings, current form and other features.

Openly available historic data contains stats for each match that includes the participating teams, number of shots taken by each team, number of corners taken by each team, number of offsides, number of bookings, half time score, final score and a lot of data on the odds calculated by different bookies before the game.

y

Such an analysis would be extremely helpful to fantasy football league players, bettors and teams. If a team knows that it scores more whenever it takes more number of corners, then they would obviously try to gain more corners in a game to maximize the chances of them scoring.

To fantasy football league players and bettors, this might give a fair idea of what to expect in a match in the upcoming week's fixtures.

Data can be obtained from two sources
1. http://www.football-data.co.uk/
2. https://www.kaggle.com/hugomathien/soccer

More interesting big data including number of passes completed by each player, each player's position etc. is being tracked by a company called Opta (and only this company is doing it) but Opta has made it proprietary. They said they'd provide data for research purposes for free but that data isn't of much use.

1. Extract the data for only English Premier League from the two datasets
2. Check for missing values and clean the data so it could be used for further analyses.
3. Analyse and create new features like form, results from previous meetings (To think of more).
4. Build models (initially hunch : Random Forests or Gradient Boosted Trees) using these new features to guess the outcome of a match.
5. Get features on their level of importance.

1. Code - iPynb on github.
2. A report
3. Slide deck
4. Blog post on how I went about doing the analyses