A Method for Determining Relative Offensive and Defensive Strengths of Football Teams

Technical Memo
19 OCT 2016

Ed Ramsden
www.edscave.com
edr@sensorlytics.com

Introduction

People may want to be able to compare the relative strengths of sports teams for various purposes, such as for attempting to set up matches in which the two teams have comparable skill levels. Simple ranking methods can be based on the number (percentage) of games won, the average 'margin-of-victory', or by linear regression techniques which attempt to derive a 'skill rating' for a given team, when considered over a history of matches.

In two-team (player) games in which each team (player) accumulates a separate score and for which there are both offensive and defensive components, we propose a ratingg method based on score data only that estimates separate relative rankings for each team's offensive and defensive capabilities. We will use data from the 2014-15 NFL season to perform an example ranking.

The Model

For our approach, we chose to use a model that considers the two scores (home and visitor) as two separate 'games' within a given match. The individual scores can each be interpreted as the outcome of a game in which one team is trying to drive the score up (Offense), while the opposing team is trying to drive the score down (Defense). A complete game (both home and visitor scores) can be represented by the following pair of equations:

$$Score_{HOME} = Mean\_Score * (Offensive\_Rating_{HOME} / Defensive\_Rating_{VISITOR}) + Home\_Advantage / 2$$

$$Score_{VISITOR} = Mean\_Score * (Offensive\_Rating_{VISITOR} / Defensive\_Rating_{HOME}) - Home\_Advantage / 2$$

Where

- ***Score*** are the final game scores, both home and visitor for each game in the data set to be used, for example, a season's worth of games.
- ***Mean_Score*** is the average score achieved by all teams in all games, including both home and visitor
- ***Offensive_Rating*** is a factor that describes the relative offensive strength of the team trying to increase the *Score* value. For the Home Score, this is the Home Team and for the Visitor Score this is the Visitor Team. The *Offensive_Rating* is initially unknown and is must be estimated from the score results of many individual games.
- ***Defensive_Rating*** is a factor that describes the relative defensive strength of the team trying to decrease the score value. Like the *Offensive_Rating* value, the *Defensive_Rating* value must also be estimated.
- ***Home_Advantage*** is a factor that is the difference between the average of home and visitor scores over all games. Because 'home team advantage, is well known, this factor is included in an attempt to reduce bias introduced by which team is playing home as opposed to visitor.

An iterative algorithm (pseudocode) to execute this model is shown below. This algorithm assumes that a list of games is available, and that for each game that home and visitor team identifier and numerical scores are available for each game. The result of this procedure are arrays or lists of offensive and defensive ratings.

Figure 1 – Pseudocode for Solution Algorithm

```
 //** Calculate Mean Score for all games

Mean_Score = Average(Home_Scores[], Visitor_Scores[])

//** Initialize All Team Factors to '1'

Offensive_Rating[] = 1
Defensive_Rating[] = 1


//** Iterate some number of times to get factors to converge

for Iteration = 1 to MAX_ITERATIONS

    //** Run algorithm over all played games.

    for Game = 1 to TOTAL_GAMES

      //** Get home and visitor scores for current game
      hact = Home_Score[Game]
      vact = Visitor_Score[Game]

      //** Get Home and Visitor Teams for current game
      hteam = Home_Team[Game]
      vteam = Visiting_Team[Game]

      //** Calculate  estimates of home & visitor scores from offense, defense
      //** and home advantage factors

      hest = Mean_Score * Offensive_Rating[hteam] / Defensive_Rating[vteam] + Home_Advantage/2
      vest = Mean_Score * Offensive_Rating[vteam] / Defensive_Rating[hteam] - Home_Advantage/2

      //** Calculate errors between actual game scores and estimates

      herr = hact-hest
      verr = vact-vest

      //** Use error terms to update all of the factors.

      Offensive_Rating[hteam] =  Offensive_Rating[hteam] + eta * herr
      Defensive_Rating[vteam] =  Defensive_Rating[vteam] - eta * herr
      Offensive_Rating[vTeam] =  Offensive_Rating[vTeam] + eta * verr
      Defensive_Rating[hteam] =  Defensive_Rating[hteam] - eta * verr
      Home_Advantage = Home_Advantage + 0.25 *eta*(herr-verr)

      //** Limit rating values to a range around 1 (0.25...4) to prevent uncontrolled drift

      Offensive_Rating[hteam] =  LIMIT( Offensive_Rating[hteam], 0.25, 4.0 )
      Defensive_Rating[vteam] =  LIMIT( Defensive_Rating[vteam], 0.25, 4.0 )
      Offensive_Rating[vTeam] =  LIMIT( Offensive_Rating[vTeam], 0.25, 4.0 )
      Defensive_Rating[hteam] =  LIMIT( Defensive_Rating[hteam], 0.25, 4.0 )

    next Game

next iteration
```

This algorithm iteratively converges on the rating factors, typically requiring less than a few hundred iterations. The convergence method is similar to that used in a linear perceptron, where the current 'guesses' for the various factors are used to estimate scores for a given game, and the difference between the estimates and the actual scores is used to incrementally adjust the values of the factors. The rate of convergence is controlled by the variable eta, which will typically need to be set in the range of 0.001 to 0.01. Higher eta values result in faster convergence, but also increase the likelihood of numerical instability. To control drift in in the offensive and defensive ratings,

their values are limited to an arbitrary range (0.25...4.0) on each iteration.

Model Credibility

One question that should be asked of any rating system is that of *why it should be believed*. The best argument to be made for the credibility of this model is that in addition to being useful as a team rating method, the factors are also useful from a predictive standpoint. When the model is run using data from weeks 1...N, the score estimate values that can be made for games played in week N+1 (the following week), can be used to predict wins and losses with a typical accuracy in the range of 60-65%. While this is far from perfect, it is comparable to next-week predictions made by human experts. Because the factors (Offensive_Rating, Defensive_Rating) have predictive value, it is likely that they have some meaning and are not just arbitrary assignments.

Experimental Results

We ran the above algorithm using game data from the 2014 NFL season – a total of 267 games. This resulted in offensive and defensive ratings for each team, which are plotted out in the Figure below.