Ranking System in English Premier League

Ruchit Gandhi - 001067481, Darshan Dedhia - 001054816

INFO6205 34506 Program Structure & Algorithms SEC 03 Spring 2020

Abstract

A ranking is a relationship between a set of items such that, for any two items, the first is either 'ranked higher than', 'ranked lower than' or 'ranked equal to' the second. In mathematics, this is known as a weak order or total preorder of objects. It is not necessarily a total order of objects because two different objects can have the same ranking. The rankings themselves are totally ordered.

Keywords: Colley Matrix.

Ranking System in English Premier League

This project is about

Introduction

The most important and most fundamental question when designing a computer ranking system is simply where to start. Usually, in science, one poses a hypothesis and checks it against. observation to determine its validity, but in the case of ranking college football teams, there really is no observation, there is no ranking that is an absolute truth, against which to check.

Hypothesis

Before going into depth, lets first understand why these ranking systems are required. Will a simple win-loss ratio not be enough? To answer this, let's go through an example. Consider there is a tournament with 4 teams playing in it and following are the results, and final table of the tournament standings so far.

Round 1	Round 2	Table				
Manchester United def Arsenal	Manchester United def Arsenal	Rank	Team	Played	Won	Loss
Manchester United def Chelsea	Manchester United def Chelsea	1	Manchester United	6	6	0
Manchester United def Liverpool	Manchester United def Liverpool	2	Chelsea	6	3	3
Arsenal def Chelsea	Chelsea def Arsenal	2	Arsenal	6	3	3
Arsenal def Liverpool	Arsenal def Liverpool	4	Liverpool	6	0	6
Chelsea def Liverpool	Chelsea def Liverpool					

As you can see, Manchester United has won all 6 games and sit comfortably on top, Liverpool have lost all their games and lie on bottom of table. Arsenal and Chelsea share the spoil having been defeated by Man Utd in both rounds, defeating Liverpool in both rounds and then splitting the tie between themselves with Arsenal winning in 1st round and Chelsea returning the favor in reverse fixture.

Problem with simple assignment.

Now imagine we begin 3rd round of fixtures and first couple of game's results were as below. Game:

1: Arsenal defeat Manchester United Game

2: Chelsea defeat Liverpool

With that, updated table will look like below

Table				
Rank	Team	Played	Won	Loss
1	Manchester United	7	6	1
2	Chelsea	7	4	3
2	Arsenal	7	4	3
4	Liverpool	7	0	7

Arsenal has defeated Manchester United who were dominant team in the tournament so far, whereas Chelsea defeated Liverpool who had anyways lost all their games. Arsenal should get more credit and should be ranked higher than Chelsea.

If we are using just a simple win-loss ratio, there is no way to differentiate them. Further on if you are considering goal difference and such, consider that all games have been won by the score line of 2 goals to 1, again we have a deadlock. Then how should we take into account the fact Arsenal defeated a much stronger team than Chelsea did, and hence deserves a higher ranking. This is where ranking systems come into picture.

Colley Rankings.

Mr. Wesley N. Colley suggested that instead of calculating rankings based on win-loss ratio like; ratings = [total wins] / [total games], ratings should be calculated based on following formula.

$$Rating = [1 + totalwins] / [2 + totalgames].$$

At the start of any tournament, all teams will begin with a rating of 0.5, as both 'total wins' and 'total games' will be zero. Thus, we can conclude that the average of all team's ratings in Colley's algorithm will always stay at 0.5.

We can write total wins as shown below:

$$
\begin{aligned}
total-wins &= \frac{total-wins}{2} + \frac{total-wins}{2} \\
&= \left[\frac{total-wins}{2} + \frac{total-wins}{2}\right] + \left[\frac{total-loss}{2} - \frac{total-loss}{2}\right] \\
&= \left[\frac{total-wins}{2} - \frac{total-loss}{2}\right] + \left[\frac{total-wins}{2} + \frac{total-loss}{2}\right] \\
&= \left[\frac{total-wins}{2} - \frac{total-loss}{2}\right] + \frac{1}{2}[total-wins+total-loss] \\
&= \left[\frac{total-wins}{2} - \frac{total-loss}{2}\right] + \frac{1}{2}[total-games]
\end{aligned}
$$

Looking closely into ½ * [total-games] term for a single team, we can say that ½ * [total-games] = ½ * [1+1+1…1] where '1' stands for a game played against each opponent. This in turn can be written as ½ * [total-games] = [1/2 + ½ + ½+….]. Now if you remember, as said before at the start, rating of all teams would be around 0.5 and hence average rating for all teams combined will also be 0.5. Considering this, we can say that [1/2 + ½ + ½ + ..] ~= [sum of opponent's ratings]. With this, rewriting equation for total wins.

$$total-wins \approx \left[\frac{total-wins}{2} - \frac{total-loss}{2}\right] + [sum of opponent's ratings]$$

Rewriting formula for ratings,

$$ratings \approx \left[\frac{1+(total-wins)}{2+(total-games)}\right]$$

Expanding total-wins and writing down rating for individual team, we can write as.

$$r_i = \frac{1 + [w_i - l_i]/2 + \sum r_j}{2 + t_i}$$

Where r_i, w_i, l_i, and t_i represent rating, wins, loss and total game for the team of interest and Σr_j is sum of opponent's ratings. This can be further rearranged as

$$[2 + t_i] * r_i = 1 + [w_i - l_i]/2 + \sum r_j$$
$$[2 + t_i] * r_i - \sum r_j = 1 + [w_i - l_i]/2 \quad \text{—(A)}$$

Now, as with most of my other tutorials, let's use a toy example to understand mathematical equations better. Consider a case of two teams. As stated above, before beginning, they both will have a rating of 0.5 each. Now let's suppose they played one game against each other. Let r_W and r_L represent new ratings for winning and losing team respectively. If you put t_i (which will be equal to 1) for both teams, and consider their w_i and l_i values, we will get following results from equation (A)

$$\begin{aligned} 3r_W - r_L &= 3/2 \\ -r_W + 3r_L &= 1/2, \end{aligned}$$

This represents a simple two variable linear system, which when solved gives us our new ratings as rw = 5/8 and rL = 3/8. For 2 teams, equation (A) resulted in two variable linear system, for N teams, it will result in N variable linear system. In programming world, whenever we want to solve N variable linear equations, we always write them in form of a matrix. So rewriting equation (A) in matrix form as

$$C\vec{r} = \vec{b},$$

Where vector r represents column-vector of all the ratings r_i and vector b is a column-vector of the right-hand-side of equation (A). The matrix C, defined as the Colley Matrix, is just slightly more complicated. Imagine having our four favorite English premier league teams as rows and columns of a 4x4 matrix, as shown in below picture

rows / columns	Man Utd	Chelsea	Arsenal	Liverpool
Man Utd				
Chelsea				
Arsenal				
Liverpool				

The diagonal elements in Colley matrix should be filled by number = (2 + total games) for corresponding team. So 1st blue cell should have a value of 2 + number of games played by Manchester United. Similarly rest of cells to be filled for Chelsea, Arsenal and Liverpool games. The off diagonal elements represented by green cell should have negative value of the number of games played between team represented by row and column for that cell. So for e.g. cell intersecting Man Utd row and Chelsea column should have a number equal to negative value of number of Man Utd vs Chelsea games. And like that we will fill rest of the cells in Colley matrix. Once we solve this N variable linear equation Cr = b, we will get vector value for 'r', which represents ratings for each individual team.

Implementation.

Implementing Data Structure and Algorithm as follows:

Maintaining a Key Value pair for Key=Season Year, Value = Season Object

```
private static Map<Integer, Season> seasonList;
```

Maintain Colley Matrix and BMatrix for Linear equation solution.

```
//For 20 Teams per season
RealMatrix colleyMatrix = MatrixUtils.createRealMatrix(new double [20][20]);

//(Double) Matrix [20] --> Maintain Constants for N variable linear equation
RealMatrix bMatrix = MatrixUtils.createColumnRealMatrix(new double [20]);
```

For each match - > HomeTeam v/s AwayTeam

```
colleyMatrix.setEntry(TeamAIndex, TeamAIndex, (colleyMatrix.getEntry(TeamAIndex, TeamAIndex) + 1 * weight));
colleyMatrix.setEntry(TeamBIndex, TeamBIndex, (colleyMatrix.getEntry(TeamBIndex, TeamBIndex) + 1 * weight));
colleyMatrix.setEntry(TeamAIndex, TeamBIndex, (colleyMatrix.getEntry(TeamAIndex, TeamBIndex) - 1 * weight));
colleyMatrix.setEntry(TeamBIndex, TeamAIndex, (colleyMatrix.getEntry(TeamBIndex, TeamAIndex) - 1 * weight));
```

Solve Linear Equation Like :

```
DecompositionSolver solver = new LUDecomposition(colleyMatrix).getSolver();
RealVector constants = bMatrix.getColumnVector(0);
RealVector solution = solver.solve(constants);
```

Example:

$$\underset{20\text{x}20}{\begin{bmatrix} 5 & 0 & -1 & -1 & -1 \\ 0 & 4 & -1 & 0 & -1 \\ -1 & -1 & 6 & -1 & -1 \\ -1 & 0 & -1 & 4 & 0 \\ -1 & -1 & -1 & 0 & 5 \end{bmatrix}} \underset{20\text{x}1}{\begin{bmatrix} r_a \\ r_b \\ r_c \\ r_d \\ r_e \end{bmatrix}} = \underset{20\text{x}1}{\begin{bmatrix} 1/2 \\ 1 \\ 1 \\ 1 \\ 3/2 \end{bmatrix}},$$

Thus we arrive at the conclusion that, Each Row of resultant matrix represents the new ranking of the team as per the given historic data.

Conclusion on Colley matrix.

Colley's Bias Free College Football Ranking Method, based on solution of the Colley Matrix, has been developed with several salient features, desirable in any computer poll that claims to be unbiased.

1. It has no bias toward conference, tradition, history, or prognostication.
2. It is reproducible; one can check the results.
3. It uses a minimum of assumptions.
4. It uses no ad hoc adjustments.
5. It nonetheless adjusts for strength of schedule.
6. It ignores runaway scores.
7. It produces common sense results that compare well to the press polls

Shortcomings of Colley matrix as is.

If we are to process the historical data for previous season and take a conclusion for entire next season on basis of those rankings, we might not be able put in the new predicted data back into the system, which might have impacted the results based on the teams current performance. Consider these rankings for 2018 only.

Ranking for Season	**2017**		**Ranking for Season**	**2018**
Manchester City	0.855263158		Liverpool	0.84342105
Manchester United	0.713157895		Manchester City	0.83157895
Liverpool	0.689473684		Chelsea	0.65394737
Tottenham	0.689473684		Arsenal	0.63026316
Chelsea	0.630263158		Tottenham	0.61842105
Arsenal	0.571052632		Manchester United	0.60657895
Burnley	0.523684211		Wolves	0.53552632
Everton	0.476315789		Everton	0.51184211
Leicester	0.464473684		West Ham	0.48815789
Bournemouth	0.440789474		Leicester	0.48815789
Crystal Palace	0.440789474		Watford	0.47631579
West Ham	0.428947368		Crystal Palace	0.46447368
Newcastle	0.428947368		Newcastle United	0.44078947
Brighton	0.417105263		Bournemouth	0.42894737
Watford	0.405263158		Southampton	0.40526316
Southampton	0.393421053		Burnley	0.39342105
Huddersfield	0.381578947		Brighton	0.36973684
Stoke	0.357894737		Cardiff	0.33421053
West Brom	0.346052632		Fulham	0.275
Swansea	0.346052632		Huddersfield	0.20394737

If we are to predict the results of 2018 just based on results of colley matrix, we would have predicted Manchester City as rank 1, and Manchester United as rank 2 for 2018, but that's not the case.

Thus, a Feedback loop is required for the data to be pushed into the matrix after each prediction and re-calculate the rankings to be taken into consideration for next prediction.

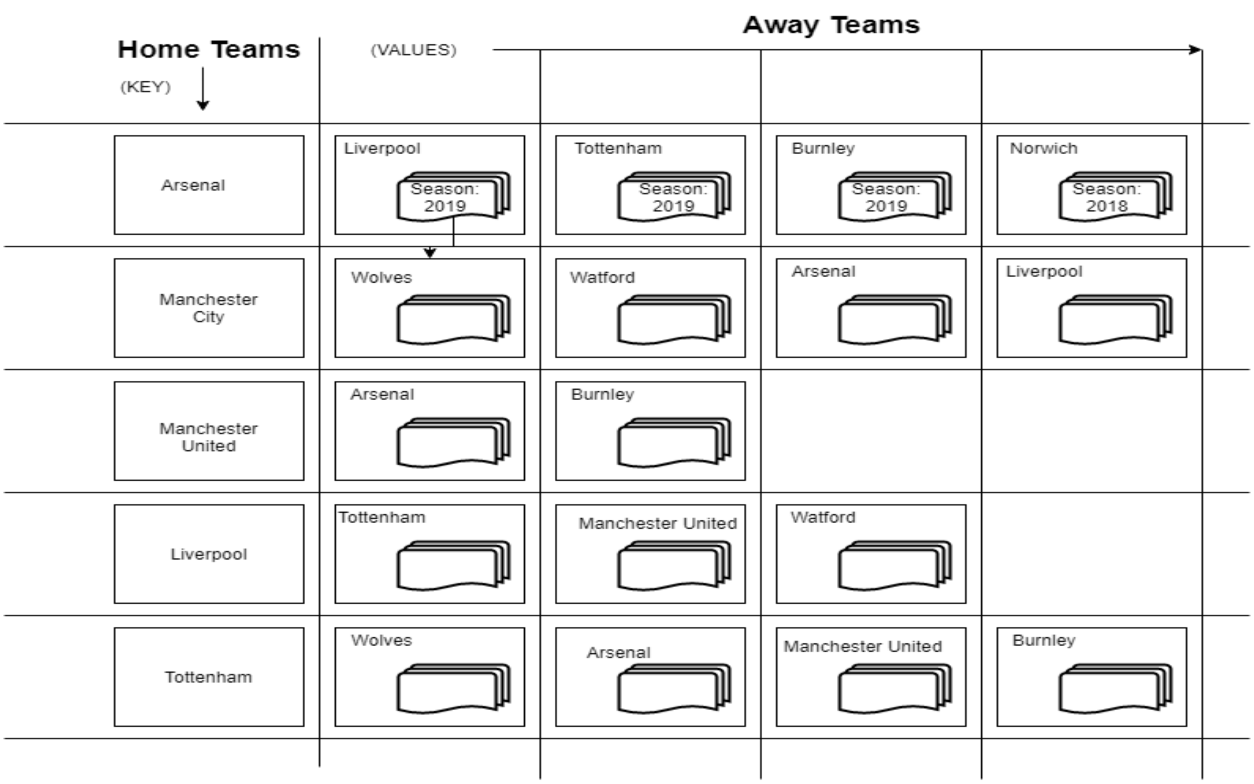
Overcoming the Shortcomings using iterative scoring.

Since for calculating ranks, Probability and scores we need to continuously access Home Team, Away Teams score across seasons and hence efficient data structure is required to access such data. Hence, we planned on moving forward with Directed graph. Since the graph is dense as every team plays with every other team, Adjacency matrix would be the ideal choice, but the challenge is that every season bottom three teams are replaced with lower tier's highest ranked team and hence we planned on moving forward with adjacency table implementation. We have implemented adjacency table using combinations of maps for which structuring details is given below:

DATA STRUCTURE:

Map(**key:** Home Team, **value:** Map(**key:** Away Team, **value:** Map(**Key:** Season, **value:** Match)))

Below is the pictorial structure of our data structure:



Benefit of using Maps is its size can scale easily unlike arrays, which is of utter importance as every season new teams come in. Also, another advantage is gaining constant time access to all possible combinations.

Scoring System:

While designing our scoring system, we rewarded teams with certain scores based on their game outcomes. We planned on EPL scoring system as our rewarding criteria, i.e. every winning team's scores increase by 3 points, losing team gets 0 points and on draw both teams get 1 point. So, if Arsenal goes Head to Head with Chelsea and Arsenal wins than it gets three points, if Chelsea get zero and if match draws then both get one point.

While Considering scores we have 4 different components for scoring team know as Scoring Criteria's:

(Let's consider that Home Team is Team of interest and Away Team is Opponent Team.)

1. Scoring Same fixture of Previous Seasons (Team of Interest vs Opponent Team)
2. Scoring Opposite fixture of same/previous season. (Opponent Team vs Team of Interest)
3. Scoring Team of Interest against all other team's v/s All other teams against Opponent Team. This is done to find performance of how team of interest has performed at Home against every other team in same/ previous season and how opponent team has performed against all when they were playing away.
4. And finally scoring opposite combination of point 3 i.e All v/s team of interest when playing away and Opponent Team v/s All when playing at home against all. This gives a basic benchmark of team's performance across seasons.

But now since we are considering team's performance across seasons it might be possible that in earlier season team might have performed really well but is unable to perform in current season. In such scenarios if we use basic scoring system we might predict wrong results and hence to overcome such false predictions we have introduced damping factor.

So what does damping factor mean?

While score we give highest weightage to current season scoring, with less weightage to previous season and even lesser weightage to all earlier seasons and so on. This helps us in actually predict based on current trends and current team performances.

Damping effect for current season (2019-2020):

Total Score = $\frac{1}{(2020-2019)} * (Scores\ of\ 2019-2020) + \frac{1}{(2020-2018)} * (Scores\ of\ 2018-2019) + \frac{1}{(2020-2017)} * (Scores\ of\ 2017-2018) + \cdots$

Hence, by every passing season score damp by a factor of difference of year and hence complete score of current seasons is assumed, half score of previous season is assumed and one third score of earlier season is considered and so on.

Now let's take an example of how We calculate scores (We are using data from 2017 onwards): Let's consider, Team of Interest as: Liverpool and Opponent Team as Manchester City

Matches data of Liverpool (at Home) vs Manchester City (Away):

Season	**Home**	**Away**	**Draw**
2019-2020	1	0	0
2018-2019	0	0	1
2017-2018	1	0	0
Total	2	0	1

Team of Interest Score = $\frac{1}{1} * (1 * 3\ points) + \frac{1}{2} * (1 * 1\ point) + \frac{1}{3}(1 * 3\ points)$

Opponent Team Score = $\frac{1}{2} * (1 * 1\ point)$

The above scenario suffices 1st point of scoring criteria and gives use understanding of team's performance of the same fixture that we want to predict.

Secondly, we check opposite fixtures that is Manchester City at Home vs Liverpool away:

Season	Home	Away	Draw
2019-2020			
2018-2019	1	0	0
2017-2018	1	0	0
Total	2	0	0

Team of Interest Score = 0;

Opponent Team Score = $\frac{1}{2} * (1 * 3\ points) + \frac{1}{3} * (1 * 3points)$

The above fixture suffices 2nd point of scoring criteria and gives us the overall performance and strength of two team playing against each other.

Now,

Team of interest i.e. Liverpool at Home vs All:

Season	Home	Away	Draw
2019-2020	15	0	0
2018-2019	17	0	2
2017-2018	12	0	7
Total	44	0	9

Team of Interest Score = $\frac{1}{1} * (12 * 3\ points + 7 * 1points) + \frac{1}{2} * (17 * 3\ points + 2 * 1point) + \frac{1}{3} * (15\ *\ 3\ \text{points})$

The above scoring helps us understand our team of interest's strength against all and helps us benchmark it against all.

And for finding opponent's score i.e. All team's v/s Manchester City (away):

Season	Home	Away	Draw
2017-2018	1	16	2
2018-2019	3	14	2
2019-2020	5	9	1
Total	44	0	9

Opponent Teams Score = $\frac{1}{1} * (9 * 3points + 1 * 1point - 5 * 1point) + \frac{1}{2} * (14 * 3points + 2 * 1point - 3 * 1point) + \frac{1}{3} * (16 * 3points + 2 * 1point - 1 * 1point)$

The above scoring helps us identify how opponent team has performed every-time when they have played away. Finally, here we calculated 3rd point of Scoring criteria and helps us identify Team of interest against all vs All against opponent Team. Similarly, we calculate 4th point of scoring criteria i.e. All v/s Team of interest and Opponent Team v/s All which helps us identify overall team performances when team of interest is playing away, and opponent team is playing home. Addition of scores of all criteria will give us basic understanding of how strong each team are against each other which gives us fair idea of results.

But in this calculation, we have one issue i.e. if lower ranking team wins several matches against another lower ranking, this scoring system actually considers team a strong contender and pushes team up in scoring matrix. Hence to avoid such false predictions we use ranked scores for estimation of wins and losses. To find ranked score we multiply team's score with opponent's Colley rank which helps us identify actual scores that needs to add up to each team.

Example of Colley Ranked based Scoring is given below:

Team Name	Team Score	Rank	Ranked Score
Team A	5	0.7	5*0.9 = 4.5
Team B	8	0.9	8*0.7 = 5.6

This amalgamation of Scoring and ranking technique helps us quickly identify if any lower rank team is strong and beating previously title holding strong contenders.

Our Currently Predicted Results:

Home Team	Away Team	Predicted Result
Manchester City	Newcastle United	HOME
Aston Villa	Crystal Palace	AWAY
Aston Villa	Chelsea	AWAY
Liverpool	Aston Villa	HOME
West Ham	Chelsea	DRAW
Aston Villa	Sheffield United	AWAY
Wolves	Arsenal	DRAW
Liverpool	Crystal Palace	HOME
Crystal Palace	Manchester United	AWAY
Southampton	Sheffield United	HOME
Manchester City	Bournemouth	HOME
Everton	Bournemouth	AWAY
Manchester City	Arsenal	HOME
Everton	Southampton	HOME
Norwich	West Ham	AWAY
Norwich	Everton	HOME
Bournemouth	Leicester	AWAY
Brighton	Manchester United	AWAY
Tottenham	Manchester United	AWAY
Norwich	Brighton	AWAY
Burnley	Watford	HOME
Bournemouth	Crystal Palace	AWAY
Norwich	Burnley	AWAY
Everton	Aston Villa	AWAY
Manchester City	Burnley	HOME
Crystal Palace	Chelsea	AWAY
Newcastle United	Tottenham	AWAY
Southampton	Brighton	HOME
West Ham	Aston Villa	AWAY
Aston Villa	Manchester United	AWAY
Sheffield United	Wolves	AWAY
Bournemouth	Tottenham	AWAY
Chelsea	Norwich	HOME
Watford	Southampton	AWAY
Manchester United	Bournemouth	HOME
Wolves	Everton	AWAY
Southampton	Manchester City	AWAY
Watford	Leicester	AWAY
Liverpool	Chelsea	HOME
Leicester	Brighton	HOME
Manchester United	Sheffield United	AWAY
Tottenham	Everton	HOME
Chelsea	Watford	HOME
Brighton	Liverpool	AWAY
Wolves	Crystal Palace	AWAY
Watford	Norwich	HOME
Manchester United	West Ham	AWAY
Wolves	Bournemouth	HOME
West Ham	Wolves	AWAY

Manchester City	Norwich	AWAY
Burnley	Sheffield United	AWAY
Manchester City	Liverpool	AWAY
Sheffield United	Chelsea	AWAY
Chelsea	Wolves	HOME
West Ham	Burnley	AWAY
West Ham	Watford	HOME
Tottenham	Leicester	DRAW
Leicester	Sheffield United	HOME
Southampton	Arsenal	AWAY
Burnley	Wolves	AWAY
Brighton	Newcastle United	DRAW
Sheffield United	Tottenham	AWAY
Newcastle United	Liverpool	AWAY
Brighton	Arsenal	DRAW
Arsenal	Liverpool	AWAY
Bournemouth	Southampton	HOME
Aston Villa	Arsenal	AWAY
Newcastle United	West Ham	AWAY
Crystal Palace	Tottenham	AWAY
Watford	Manchester City	AWAY
Everton	Leicester	AWAY
Burnley	Brighton	HOME
Watford	Newcastle United	AWAY
Tottenham	West Ham	HOME
Aston Villa	Wolves	AWAY
Arsenal	Watford	HOME
Brighton	Manchester City	AWAY
Arsenal	Leicester	AWAY
Leicester	Manchester United	AWAY
Arsenal	Norwich	AWAY
Sheffield United	Everton	HOME
Newcastle United	Sheffield United	HOME
Newcastle United	Aston Villa	AWAY
Crystal Palace	Burnley	HOME
Bournemouth	Newcastle United	AWAY
Leicester	Crystal Palace	AWAY
Chelsea	Manchester City	AWAY
Norwich	Southampton	AWAY
Manchester United	Southampton	HOME
Tottenham	Arsenal	AWAY
Liverpool	Burnley	HOME
Everton	Liverpool	AWAY

*********** Ranking for Season: 2019 ***********

Team Name	Rank
Liverpool	0.9454085080378126
Manchester City	0.7048602215322984
Leicester	0.6694106480021411
Chelsea	0.5930182261454575
Manchester United	0.5906559970322878
Tottenham	0.5862909751676493
Wolves	0.5571177156225713
Sheffield United	0.520550199975902
Arsenal	0.5169361939616116
Crystal Palace	0.4842393741345027
Burnley	0.4653402544343726
Southampton	0.44455699910669816
Newcastle United	0.43895968937938135
Everton	0.43114591004461067
West Ham	0.38764273283299333
Brighton	0.37094287363287537
Aston Villa	0.3479952794917741
Bournemouth	0.3341815781938895
Watford	0.30826420704578006
Norwich	0.30248241622548516

Bibliography

1. Wesley N. Colley (2002).The Colley Matrix Method. *Journal Title*, Pages From - To.

2. W. N. Colley. Colley's bias free college football ranking method: The Colley Matrix explained. Manuscript, 2002. Available at http://www.colleyrankings.com/.

3. Bowl Championship Series Official Website. http://www.bcsfootball.org/. Accessed October 4, 2011

4. Zaem Nalla (2019). EPL Full Set Data. https://www.kaggle.com/zaeemnalla/premier-league

5. Premier League Official website. https://www.premierleague.com/results

6. Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P., 1992, Numerical Recipes in FORTRAN, Cambridge University Press, Cambridge, UK, pp. 89-91

Tables

Table 1

Convergence of ratings to final, stable values

Iteration	Winning Team's Rating	Losing Team's Rating
0	0.666667	0.333333
1	0.611111	0.388889
2	0.62963	0.37037
3	0.623457	0.376543
4	0.625514	0.374486
5	0.624829	0.375171
6	0.625057	0.374943
7	0.624981	0.375019
8	0.625006	0.374994
9	0.624998	0.375002
10	0.625001	0.374999
11	0.625	0.375

Note: Table 1: Convergence of ratings to final, stable values, after 11 iterations, for the simple two team, one game case. The initial ratings are 2/3 for the winner and 1/3 for the loser, before any adjustment for schedule strength. Moving down the table are successive adjustments for strength of schedule. Because the winning team beat a below average (0-1) team, while the losing team lost to an above average (1-0) team, the final ratings are lower for the winning team, and greater for the losing team than were the initial ratings.

Tables

Table 2

Comparison of Final Rankings with Press Polls

press rank	Colley Ranking for Teams with Given Press Rank								
	1998	1999		2000		2001		2002	
	AP	AP	Coaches	AP	Coaches	AP	Coaches	AP	Coaches
1	1	1	1	1	1	1	1	1	1
2	2	5	2	3	3	3	3	3	3
3	3	2	5	2	2	4	4	2	2
4	6	11	11	5	6	2	2	4	4
5	8	3	3	6	5	6	6	5	5
6	4	7	7	4	4	10	10	6	11
7	10	4	4	7	8	8	5	11	6
8	5	6	6	8	11	5	8	8	8
9	11	10	10	11	7	7	7	7	7
10	9	8	8	9	13	11	13	12	12
11	7	9	9	13	9	13	11	10	14
12	13	13	12	22	22	9	9	14	17
13	12	12	15	19	19	15	15	13	13
14	17	15	13	17	12	12	12	19	20
15	16	14	14	10	17	16	16	17	18
16	22	24	24	12	10	14	17	18	19
17	14	28	23	15	30	17	14	9	9
18	19	23	17	20	23	31	31	20	24
19	20	17	28	29	15	18	18	24	32
20	23	18	22	30	20	20	20	21	23
21	24	21	18	23	29	30	21	15	21
22	15	27	27	28	27	24	28	25	28
23	27	22	21	14	18	28	30	28	15
24	21	16	29	27	14	33	19	32	26
25	26	26	16	18	32	19	24	23	25
η Colley vs. poll	1.224	1.309	1.281	1.287	1.331	1.262	1.232	1.200	1.253
AP vs. Coaches	n/a	1.071		1.098		1.037		1.082	

Note: Table 2: Comparison of final rankings to AP Poll for 1998–2002, and to the Coaches' Poll for 1999–2002. At bottom is a statistic η, described in the text. Essentially, it is the typical ratio of the Colley ranking to the poll ranking, or vice versa, so that the larger of the two always in the numerator, (specifically η is the exponent of the mean of the absolute values of the logs of the ratios), so η = 1.25 means the rankings would differ by typically one place at #4, and 5 places at #20.

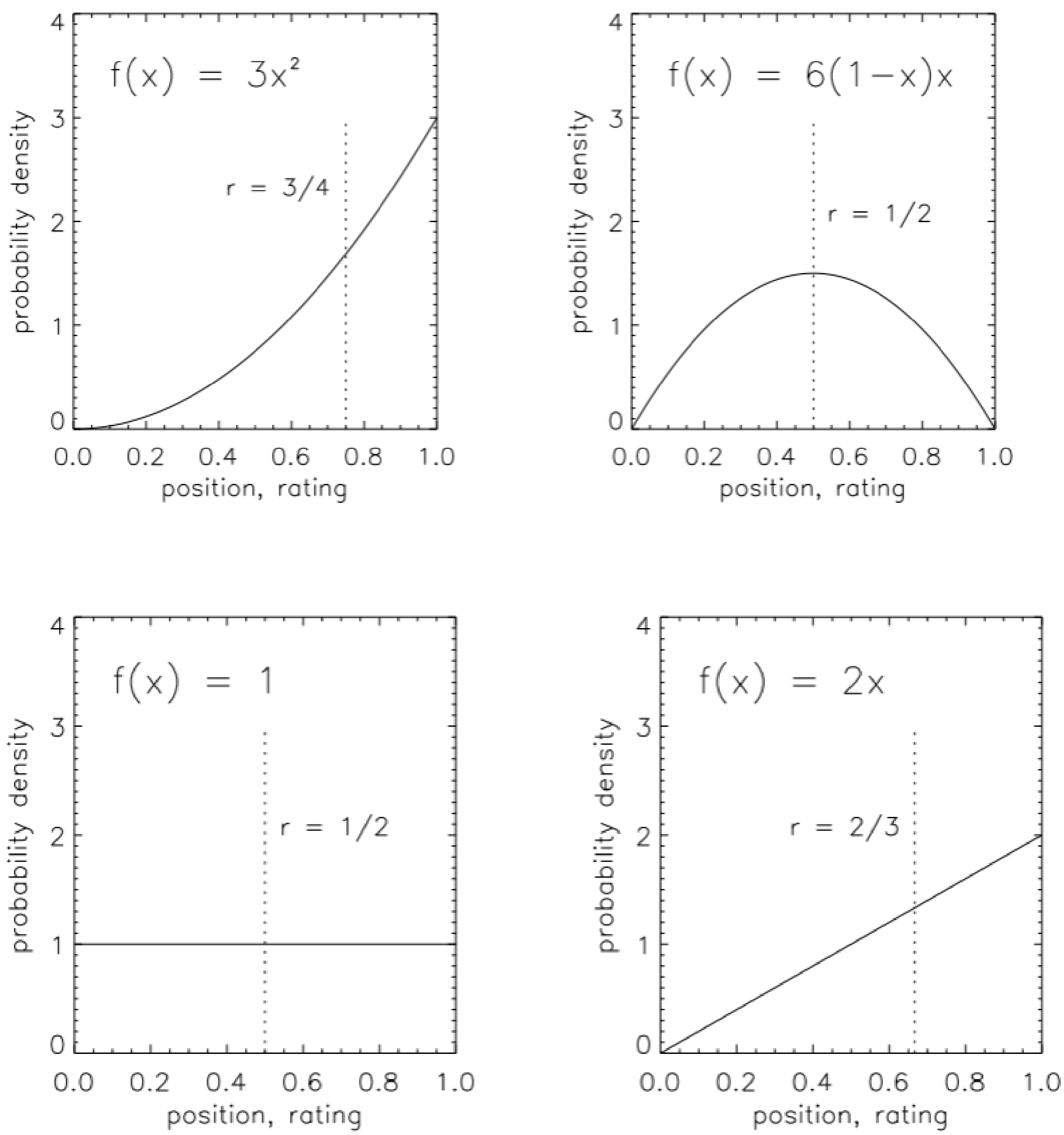
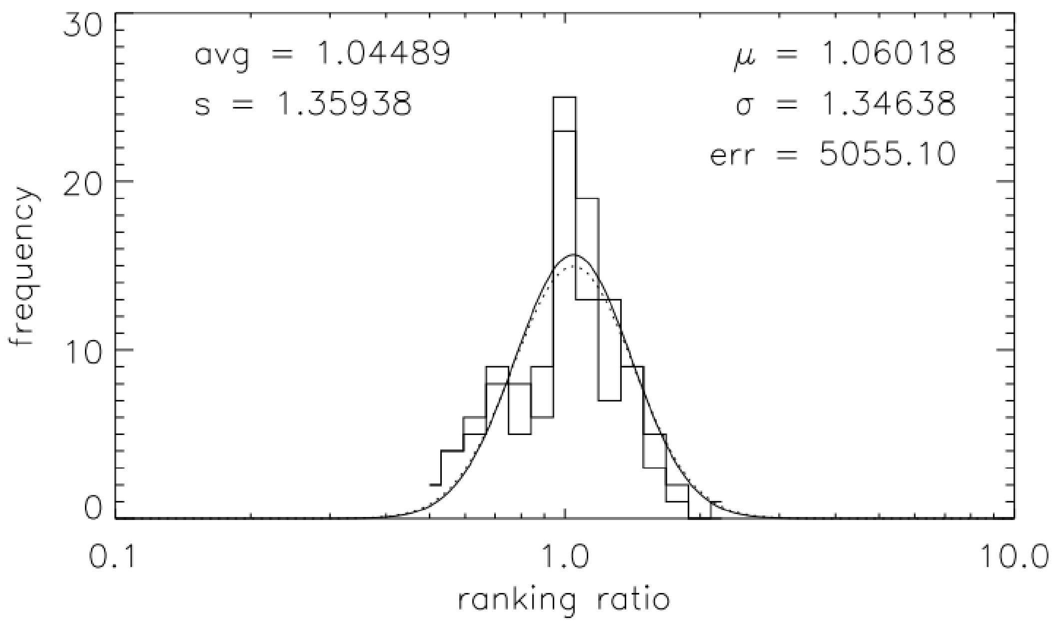


Fig. 1.— Probability distribution functions for the Laplace dice problem, analogous to rating by wins and losses. At top left is the initial condition {no dies thrown; no games played}. At top

right is {one die left; one game won}: the probability density must be zero at the left. At bottom left is {two dies left; two games won}: the probability densities multiply. At bottom right is {one die left, one die right; one game won, one game lost}: the probability density must be zero at the left and at the right. The functions here have been normalized to have an integral of one, which is irrelevant in section 3 of the text, because the normalizations explicitly divide out.



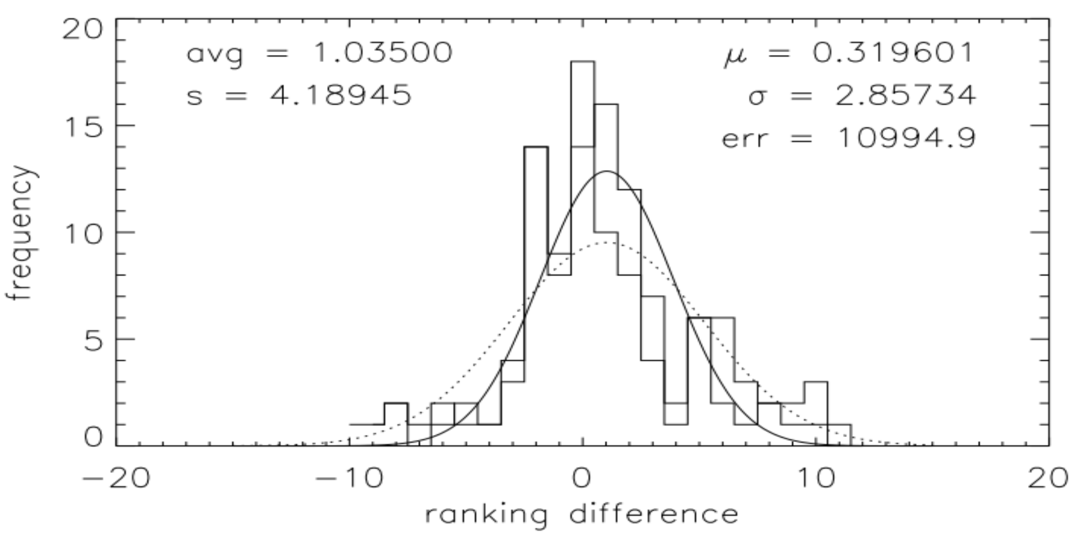


Fig. 2. — Two different ways for comparing the Colley Rankings with the Press Polls. (at top) Arithmetic differences, (Colley − press), between the final rankings for 1999–2002 for both the AP and Coaches' polls. Over-plotted are the normal curves from direct measurement of mean (= avg) and standard deviation (= s), and from fitting for the mean (= μ) and standard deviation (= σ). (at bottom) Same plots, but for ratios (Colley ÷ press) in logarithmic space.