

Ranking System for The Premier League

Northeastern University

Jing Zou, Yangyang Zhang, Mingda Ju

04/17/2020

INFO6205 Sec03

Ranking System for The Premier League

Jing Zou, Yangyang Zhang, Mingda Ju

Keywords: EPL, Java, Mathematical Expectation, Standard Deviation

1. Introduction

In this project, we will develop a ranking system to predict the ranking of the Premier League this season. Our ranking system will be designed in strict accordance with the rules mentioned below.

The Premier League is the top level of the English football league system, contested by 20 clubs. During the course of a season each club plays the others twice (a double round-robin system), once at their home stadium and once at that of their opponents', for 38 games^[1]. Teams receive three points for a win and one point for a draw. No points are awarded for a loss. Teams are ranked by total points, then goal difference, and then goals scored. Besides that, a system of promotion and relegation exists between the Premier League and the EFL Championship. The three lowest placed teams in the Premier League are relegated to the Championship. What's more, the top four teams in the Premier League qualify for the subsequent season's UEFA Champions' League group stage and the fifth team will qualify for the subsequent season's Europa League group stage. This is the basic situation of the English Premier League.

2. Objective

Due to COVID-19, the Premier League was suspended in the 29th round, and the remaining 9 games (10 games for four teams) could not be played in a long period of time. So, in this intermittent period, we want to predict the score of the remaining games and then produce a complete EPL table for this season. This table includes club, won, drawn, lost, GF, GA, GD and points.

3. Methodology

The **Expectation** is an anticipated value for an investment at some point in the future. By calculating expected values, investors can choose the scenario most likely to give the desired outcome.

$$EV = \sum P(X_i) \times X_i$$

Let X be a random variable with a finite number of finite outcomes $x_1, x_2 \dots x_n$ occurring with probabilities $p_1, p_2 \dots p_n$ respectively.

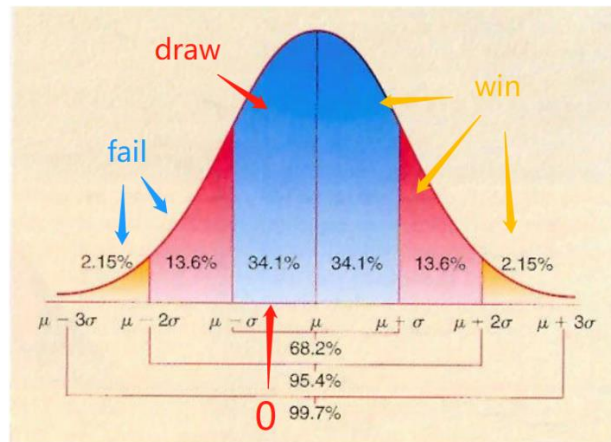
So, the points expectation should be:

$$E_{pe} = 0 \times P_1 + 1 \times P_2 + 3 \times P_3$$

P_1 represents the probability of lost, P_2 represents the probability of draw, P_3 represents the probability of winning.

In this project, we use normal distribution to calculate the expectation of each game. And we simply distribute the probability into 6 intervals by mean and standard deviation, find 0 in which interval and use this interval as P (draw), intervals less than this interval as P(fail) and intervals more than this interval as P(win).

For example, if mean – standard deviation < 0 < mean:



We calculate **Points Expectation** = $0 \times (2.15\% + 13.6\%) + 1 \times 34.1\% + 3 \times (34.1\% + 13.6\% + 2.15\%) = 0.341 + 3 \times 0.5$

```

if(count !=0) {
    if(this.goalsMean-2*this.goalsStd>0)
        this.PointsExpectation = 0.0215*1 + 0.9785*3;
    else if(this.goalsMean - 2*this.goalsStd <= 0 && this.goalsMean - this.goalsStd > 0)
        this.PointsExpectation = 0.136 + 0.8425*3 ;
    else if(this.goalsMean - this.goalsStd <= 0 && this.goalsMean > 0)
        this.PointsExpectation = 0.341 + 0.5*3 ;
    else if(this.goalsMean <=0 && this.goalsMean + this.goalsStd > 0)
        this.PointsExpectation = 0.341 + 0.1575*3 ;
    else if(this.goalsMean + this.goalsStd <= 0 && this.goalsMean + 2*this.goalsStd > 0)
        this.PointsExpectation = 0.136 + 0.0215*3 ;
    else
        this.PointsExpectation = 0.0215 ;
}

```

The **standard deviation** is a measure of the amount of variation or dispersion of a set of values.^[2]

$$s_N = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$$

In this project, we used the normal distribution to calculate the standard deviation to determine whether each team's performance is stable.

```
for(CompetitionInfo competitionInfo: competitionResultRecords){
    variance = variance + (competitionInfo.getGoalDifference() - this.goalsMean) *
                    (competitionInfo.getGoalDifference() - this.goalsMean);
}

if(count != 0){
    this.goalsStd = sqrt(variance / count);
}
else{
    this.goalsStd = 0;
}
```

4. Result

- a) We counted the team's past 5 game records to obtain the most comprehensive data. We calculated the total wins, draws and losses of each team, so that we can intuitively see the performance of each team in the last 5 years, and we can have a rough level of judgment on the team.

Code for count the total wins, draws and losses:

```
public void getTeamResult(ArrayList<CompetitionInfo> competitionDirectory, int teamId){
    for(CompetitionInfo competitionInfo: competitionDirectory){
        if(competitionInfo.getHomeTeamId() == teamId){
            if(competitionInfo.getGoalDifference() > 0){
                this.winsCount = this.winsCount + 1;
            }
            else if(competitionInfo.getGoalDifference() == 0){
                this.drawCount = this.drawCount + 1;
            }
            else{
                this.failuresCount = this.failuresCount + 1;
            }
        }
    }

    for(CompetitionInfo competitionInfo: competitionDirectory){
        if(competitionInfo.getAwayTeamId() == teamId){
            if(competitionInfo.getGoalDifference() < 0){
                this.winsCount = this.winsCount + 1;
            }
            else if(competitionInfo.getGoalDifference() == 0){
                this.drawCount = this.drawCount + 1;
            }
            else{
                this.failuresCount = this.failuresCount + 1;
            }
        }
    }

    this.pointsExpectation = 3 * winsCount + 1 * drawCount;
}
```

The Ranking Of ALL Teams Based On Their Points Accumulated Over Five Years:

```
Team{Team Id : 11, Team Name : Man City, Wins = 142, Draw = 42, Failures = 63, Total Points = 468}
Team{Team Id : 10, Team Name : Liverpool, Wins = 119, Draw = 49, Failures = 36, Total Points = 406}
Team{Team Id : 17, Team Name : Tottenham, Wins = 106, Draw = 47, Failures = 50, Total Points = 365}
Team{Team Id : 6, Team Name : Chelsea, Wins = 105, Draw = 49, Failures = 60, Total Points = 364}
Team{Team Id : 12, Team Name : Man United, Wins = 102, Draw = 58, Failures = 68, Total Points = 364}
Team{Team Id : 1, Team Name : Arsenal, Wins = 101, Draw = 48, Failures = 55, Total Points = 351}
Team{Team Id : 8, Team Name : Everton, Wins = 82, Draw = 63, Failures = 88, Total Points = 309}
Team{Team Id : 16, Team Name : Southampton, Wins = 79, Draw = 71, Failures = 108, Total Points = 308}
Team{Team Id : 9, Team Name : Leicester, Wins = 81, Draw = 45, Failures = 83, Total Points = 288}
Team{Team Id : 7, Team Name : Crystal Palace, Wins = 69, Draw = 51, Failures = 103, Total Points = 258}
Team{Team Id : 18, Team Name : Watford, Wins = 68, Draw = 52, Failures = 114, Total Points = 256}
Team{Team Id : 19, Team Name : West Ham, Wins = 63, Draw = 53, Failures = 91, Total Points = 242}
Team{Team Id : 3, Team Name : Bournemouth, Wins = 61, Draw = 48, Failures = 111, Total Points = 231}
Team{Team Id : 5, Team Name : Burnley, Wins = 55, Draw = 39, Failures = 78, Total Points = 204}
Team{Team Id : 13, Team Name : Newcastle, Wins = 44, Draw = 48, Failures = 88, Total Points = 180}
Team{Team Id : 4, Team Name : Brighton, Wins = 26, Draw = 35, Failures = 52, Total Points = 113}
Team{Team Id : 20, Team Name : Wolves, Wins = 28, Draw = 23, Failures = 26, Total Points = 107}
Team{Team Id : 14, Team Name : Norwich, Wins = 21, Draw = 22, Failures = 48, Total Points = 85}
Team{Team Id : 2, Team Name : Aston Villa, Wins = 12, Draw = 13, Failures = 46, Total Points = 49}
Team{Team Id : 15, Team Name : Sheffield United, Wins = 11, Draw = 10, Failures = 7, Total Points = 43}
```

- b) Then we calculated the expected goal difference, the standard deviation and the points exception of all teams against each other for the team in this season.

Data structure we use to calculate and store for each pair of home team and away team's competition: (**normal distribution**)

```
private int homeTeamId;
private int awayTeamId;
private String homeTeamName;
private String awayTeamName;
private int goalsMean;
private double goalsStd;
private double expectedValue;
private int count = 0;
private int totalGoals = 0;
private double variance = 0;
private double PointsExpectation = 0;
private double PointsExpectationForAwayTeam = 0;
private String teamName;
```

Code for calculate points expectation for this game:

Because we consider 'home team A vs away team B' as a distinct set of data (not relate to 'home team B vs away team A' at all), we can also calculate the points expectation for away team inside the same class.

```

if(count !=0) {
    if(this.goalsMean-2*this.goalsStd>0)
        this.PointsExpectation = 0.0215*1 + 0.9785*3;
    else if(this.goalsMean - 2*this.goalsStd <= 0 && this.goalsMean - this.goalsStd > 0)
        this.PointsExpectation = 0.136 + 0.8425*3 ;
    else if(this.goalsMean - this.goalsStd <= 0 && this.goalsMean > 0)
        this.PointsExpectation = 0.341 + 0.5*3 ;
    else if(this.goalsMean <=0 && this.goalsMean + this.goalsStd > 0)
        this.PointsExpectation = 0.341 + 0.1575*3 ;
    else if(this.goalsMean + this.goalsStd <= 0 && this.goalsMean + 2*this.goalsStd > 0)
        this.PointsExpectation = 0.136 + 0.0215*3 ;
    else
        this.PointsExpectation = 0.0215 ;
}
else {
    this.PointsExpectation = 0;
}
//because we consider 'HomeTeam A vs AwayTeam B' as a distinct set of data
//not relate to 'HomeTeam B vs AwayTeam A' at all)
//we can also calculate the points expectation for away team for this particular game.
//the goalsMean for Away Team is -goalsMean for HomeTeam
if(count !=0) {
    if(-this.goalsMean-2*this.goalsStd>0)
        this.PointsExpectationForAwayTeam = 0.0215*1 + 0.9785*3;
    else if(-this.goalsMean - 2*this.goalsStd <= 0 && -this.goalsMean - this.goalsStd > 0)
        this.PointsExpectationForAwayTeam = 0.136 + 0.8425*3 ;
    else if(-this.goalsMean - this.goalsStd <= 0 && -this.goalsMean > 0)
        this.PointsExpectationForAwayTeam = 0.341 + 0.5*3 ;
    else if(-this.goalsMean <=0 && -this.goalsMean + this.goalsStd > 0)
        this.PointsExpectationForAwayTeam = 0.341 + 0.1575*3 ;
    else if(-this.goalsMean + this.goalsStd <= 0 && -this.goalsMean + 2*this.goalsStd > 0)
        this.PointsExpectationForAwayTeam = 0.136 + 0.0215*3 ;
    else
        this.PointsExpectation = 0.0215 ;
}

```

Let me show you the analysis results of the three teams, the team level is high, medium and poor.

```

Competition Probability Distribution:
Competition(NO.1 Arsenal VS No.2 Aston Villa, Expected Goal Difference = 2.0, Std = 1.5811, Points Expectation = 2.6635)
Competition(NO.1 Arsenal VS No.3 Bournemouth, Expected Goal Difference = 2.0, Std = 1.0954, Points Expectation = 2.6635)
Competition(NO.1 Arsenal VS No.4 Brighton, Expected Goal Difference = 0.0, Std = 1.5000, Points Expectation = 0.8135)
Competition(NO.1 Arsenal VS No.5 Burnley, Expected Goal Difference = 2.0, Std = 1.6583, Points Expectation = 2.6635)
Competition(NO.1 Arsenal VS No.6 Chelsea, Expected Goal Difference = 0.0, Std = 1.6903, Points Expectation = 0.8135)
Competition(NO.1 Arsenal VS No.7 Crystal Palace, Expected Goal Difference = 0.0, Std = 1.5811, Points Expectation = 0.8135)
Competition(NO.1 Arsenal VS No.8 Everton, Expected Goal Difference = 1.0, Std = 1.5811, Points Expectation = 1.8410)
Competition(NO.1 Arsenal VS No.9 Leicester, Expected Goal Difference = 1.0, Std = 0.5000, Points Expectation = 0.0215)
Competition(NO.1 Arsenal VS No.10 Liverpool, Expected Goal Difference = 0.0, Std = 1.0000, Points Expectation = 0.8135)
Competition(NO.1 Arsenal VS No.11 Man City, Expected Goal Difference = 0.0, Std = 1.9437, Points Expectation = 0.8135)
Competition(NO.1 Arsenal VS No.12 Man United, Expected Goal Difference = 1.0, Std = 1.4907, Points Expectation = 1.8410)
Competition(NO.1 Arsenal VS No.13 Newcastle, Expected Goal Difference = 2.0, Std = 1.5811, Points Expectation = 2.6635)
Competition(NO.1 Arsenal VS No.14 Norwich, Expected Goal Difference = 1.0, Std = 0.0000, Points Expectation = 0.0215)
Competition(NO.1 Arsenal VS No.15 Sheffield United, Expected Goal Difference = 0.0, Std = 0.0000, Points Expectation = 0.0215)
Competition(NO.1 Arsenal VS No.16 Southampton, Expected Goal Difference = 1.0, Std = 1.0000, Points Expectation = 1.8410)
Competition(NO.1 Arsenal VS No.17 Tottenham, Expected Goal Difference = 0.0, Std = 1.6833, Points Expectation = 0.8135)
Competition(NO.1 Arsenal VS No.18 Watford, Expected Goal Difference = 2.0, Std = 1.5275, Points Expectation = 2.6635)
Competition(NO.1 Arsenal VS No.19 West Ham, Expected Goal Difference = 1.0, Std = 1.8974, Points Expectation = 1.8410)
Competition(NO.1 Arsenal VS No.20 Wolves, Expected Goal Difference = 0.0, Std = 0.0000, Points Expectation = 0.0215)

```

This is Arsenal, it is a very high-level team. It can be seen that when playing against teams such as Man United or Man City, Std will be relatively large, saying that the life of the team is not very stable, and the goal difference and score are also very small.

```

Competition{NO.5 Burnley VS No.1 Arsenal, Expected Goal Difference = 0.0, Std = 1.1547, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.2 Aston Villa, Expected Goal Difference = -1.0, Std = 0.0000, Points Expectation = 0.0215}
Competition{NO.5 Burnley VS No.3 Bournemouth, Expected Goal Difference = 1.0, Std = 1.7321, Points Expectation = 1.8410}
Competition{NO.5 Burnley VS No.4 Brighton, Expected Goal Difference = 0.0, Std = 0.7071, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.6 Chelsea, Expected Goal Difference = -1.0, Std = 1.6583, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.7 Crystal Palace, Expected Goal Difference = 0.0, Std = 1.4142, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.8 Everton, Expected Goal Difference = 0.0, Std = 2.1213, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.9 Leicester, Expected Goal Difference = 0.0, Std = 1.2649, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.10 Liverpool, Expected Goal Difference = 0.0, Std = 1.6903, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.11 Man City, Expected Goal Difference = -1.0, Std = 1.8898, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.12 Man United, Expected Goal Difference = -1.0, Std = 0.8660, Points Expectation = 0.2005}
Competition{NO.5 Burnley VS No.13 Newcastle, Expected Goal Difference = 0.0, Std = 1.0000, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.14 Norwich, Expected Goal Difference = 2.0, Std = 0.0000, Points Expectation = 0.0215}
Competition{NO.5 Burnley VS No.15 Sheffield United, Expected Goal Difference = 0.0, Std = 0.0000, Points Expectation = 0.0000}
Competition{NO.5 Burnley VS No.16 Southampton, Expected Goal Difference = 0.0, Std = 1.3540, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.17 Tottenham, Expected Goal Difference = 0.0, Std = 1.7728, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.18 Watford, Expected Goal Difference = 0.0, Std = 1.5811, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.19 West Ham, Expected Goal Difference = 0.0, Std = 1.6733, Points Expectation = 0.8135}
Competition{NO.5 Burnley VS No.20 Wolves, Expected Goal Difference = 1.0, Std = 0.7071, Points Expectation = 2.6635}

```

This is Burnley, a mid-level team. When he played against Liverpool and other strong teams, his goal difference was almost 0, indicating that he is almost impossible to win. When he played against Newcastle, which is also a mid-stream team, his std was almost zero, indicating that he played relatively stable against a team of his own level.

```

Competition{NO.3 Bournemouth VS No.1 Arsenal, Expected Goal Difference = 0.0, Std = 1.8127, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.2 Aston Villa, Expected Goal Difference = 0.0, Std = 1.0000, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.4 Brighton, Expected Goal Difference = 1.0, Std = 0.8165, Points Expectation = 2.6635}
Competition{NO.3 Bournemouth VS No.5 Burnley, Expected Goal Difference = 0.0, Std = 1.4832, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.6 Chelsea, Expected Goal Difference = 0.0, Std = 2.3905, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.7 Crystal Palace, Expected Goal Difference = 0.0, Std = 1.4142, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.8 Everton, Expected Goal Difference = 0.0, Std = 1.0000, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.9 Leicester, Expected Goal Difference = 0.0, Std = 1.6475, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.10 Liverpool, Expected Goal Difference = -2.0, Std = 1.9494, Points Expectation = 0.2005}
Competition{NO.3 Bournemouth VS No.11 Man City, Expected Goal Difference = 0.0, Std = 2.6458, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.12 Man United, Expected Goal Difference = 0.0, Std = 1.8439, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.13 Newcastle, Expected Goal Difference = 0.0, Std = 1.0000, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.14 Norwich, Expected Goal Difference = 1.0, Std = 1.5811, Points Expectation = 1.8410}
Competition{NO.3 Bournemouth VS No.15 Sheffield United, Expected Goal Difference = 0.0, Std = 0.0000, Points Expectation = 0.0215}
Competition{NO.3 Bournemouth VS No.16 Southampton, Expected Goal Difference = 0.0, Std = 1.4720, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.17 Tottenham, Expected Goal Difference = 0.0, Std = 2.6458, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.18 Watford, Expected Goal Difference = 0.0, Std = 1.6583, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.19 West Ham, Expected Goal Difference = 0.0, Std = 1.3416, Points Expectation = 0.8135}
Competition{NO.3 Bournemouth VS No.20 Wolves, Expected Goal Difference = 0.0, Std = 0.7071, Points Expectation = 0.8135}

```

This is Bournemouth. It is a low-level team. We can see that he plays against teams of almost any level, and the goal difference is almost negative, indicating that it is difficult to win. At the same time, his std is very small, indicating that his strength is indeed very weak, and the result of the game has almost no suspense.

- c) Finally, we set two maps, one is for calculating total points for each team, the other one is for distinguishing finished game and future game. If this game is finished, we will judge the team's winning or losing by judging the positive or negative of the goal difference. The winning team scores 3 points, the flat team scores 1 point, and the losing team does not score. If the game is in the future, we add our expectation of this game in Home Team and Away Team respectively. Then we plus these two parts exceptions together, we can get the final predicted standings.

```
public ForecastForThisSeason(CompetitionDirectory cd, TeamCompetitionResultDirectory td, TeamDirectory teamDirectory) {
    this.cd = cd;
    this.td = td;
    HashMap<Integer, Double> resultMap = new HashMap<Integer, Double>();
    //use resultMap to calculate total points for each team, key is teamID, value is points
    HashMap<Integer, Boolean> RecordMap = new HashMap<Integer, Boolean>();
    //use RecordMap to distinguish finished game and future game, key is GameID.
    //if this game is finished, then value is true, else value is false
    //GameID = HomeTeamID*20 + AwayTeamID, there is only one game ID for a pair of specific HomeTeam and AwayTeam
    //HomeTeamID = GameID/20, AwayTeamID = GameID%20
    for (int i = 1; i <= 20; i++) {
        for (int j = 1; j <= 20; j++) {
            if (i != j) {
                RecordMap.put(i * 20 + j, false);
            }
        }
    }
    for (int i = 1; i <= 20; i++) {
        resultMap.put(i, 0.00);
    }
    for (CompetitionInfo c : cd.getCompetitionDirectory()) {
        if (c.getCompetiYear() == 2019) {
            int gameID = c.getHomeTeamId() * 20 + c.getAwayTeamId();
            RecordMap.put(gameID, true); //update recordMap
            if (c.getGoalDifference() > 0) // HomeTeam Win
            {
                resultMap.put(c.getHomeTeamId(), 3.00 + resultMap.get(c.getHomeTeamId()));
            } else if (c.getGoalDifference() == 0) // Draw
            {
                resultMap.put(c.getHomeTeamId(), 1.00 + resultMap.get(c.getHomeTeamId()));
                resultMap.put(c.getAwayTeamId(), 1.00 + resultMap.get(c.getAwayTeamId()));
            } else //HomeTeam Fail
            {
                resultMap.put(c.getAwayTeamId(), 3.00 + resultMap.get(c.getAwayTeamId()));
            }
        }
    }
    for (TeamCompetitionResult t : td.getTeamCompetitionResultDirectory()) { // input the expectation of future games
        if (!RecordMap.get(t.getHomeTeamId() * 20 + t.getAwayTeamId())) { // if the game is in future
            resultMap.put(t.getHomeTeamId(), t.getPointsExpectation() + resultMap.get(t.getHomeTeamId()));
            //add our expectation of this game in HomeTeam's total points
            resultMap.put(t.getAwayTeamId(), t.getPointsExpectationForAwayTeam() + resultMap.get(t.getAwayTeamId()));
        }
    }
}
```

Ranking of The Forecast for This Season:

```
Team{Team Id : 10, Team Name : Liverpool, Total Points Forecast : 92.2295}
Team{Team Id : 11, Team Name : Man City, Total Points Forecast : 68.4835}
Team{Team Id : 9, Team Name : Leicester, Total Points Forecast : 60.5445}
Team{Team Id : 6, Team Name : Chelsea, Total Points Forecast : 56.0950}
Team{Team Id : 12, Team Name : Man United, Total Points Forecast : 54.8930}
Team{Team Id : 17, Team Name : Tottenham, Total Points Forecast : 52.6455}
Team{Team Id : 1, Team Name : Arsenal, Total Points Forecast : 50.5445}
Team{Team Id : 7, Team Name : Crystal Palace, Total Points Forecast : 47.6515}
Team{Team Id : 20, Team Name : Wolves, Total Points Forecast : 47.0355}
Team{Team Id : 5, Team Name : Burnley, Total Points Forecast : 45.9315}
Team{Team Id : 8, Team Name : Everton, Total Points Forecast : 45.5720}
Team{Team Id : 15, Team Name : Sheffield United, Total Points Forecast : 43.0000}
Team{Team Id : 13, Team Name : Newcastle, Total Points Forecast : 41.9225}
Team{Team Id : 16, Team Name : Southampton, Total Points Forecast : 41.5355}
Team{Team Id : 18, Team Name : Watford, Total Points Forecast : 34.7360}
Team{Team Id : 4, Team Name : Brighton, Total Points Forecast : 34.5175}
Team{Team Id : 3, Team Name : Bournemouth, Total Points Forecast : 32.6945}
Team{Team Id : 19, Team Name : West Ham, Total Points Forecast : 31.9240}
Team{Team Id : 2, Team Name : Aston Villa, Total Points Forecast : 26.1000}
Team{Team Id : 14, Team Name : Norwich, Total Points Forecast : 24.4895}
```


This is the final EPL standings for EPL (2019-2020)

Position	Club	Played	Points
1	Liverpool	38	92.2295
2	Man City	38	68.4835
3	Man United	38	60.5445
4	Tottenham	38	56.0950
5	Leicester	38	54.8930
6	Arsenal	38	52.6455
7	Everton	38	50.5445
8	Chelsea	38	47.6515
9	Crystal	38	47.0355
10	Burnley	38	45.9315
11	Newcastle	38	45.5720
12	Wolves	38	43.0000
13	Watford	38	41.9225
14	Sheffield	38	41.5355
15	Brighton	38	34.7360
16	Bournemouth	38	34.5175
17	Southampton	38	32.6945
18	West Ham	38	31.9240
19	Norwich	38	26.1000
20	Aston Villa	38	24.4895

We can see that Liverpool will undoubtedly win the championship. At the same time, the top four will enter the Champions League next year, the fifth place will enter the UEFA Cup, and the last three will unfortunately be relegated.

References

- [1] When will goal-line technology be introduced? Archived 9 July 2013 at the Wayback Machine
The total number of matches can be calculated using the formula $n*(n-1)$ where n is the total number of teams.
- [2] Bland, J.M.; Altman, D.G. (1996). "Statistics notes: measurement error". *BMJ*. 312 (7047): 1654. doi:10.1136/bmj.312.7047.1654. PMC 2351401. PMID 8664723.