Logistic Regression Based Simulation of Major League Baseball Seasons

Dr. Richard Auer, Ms. Claire Reynolds

Loyola University Maryland, 4501 North Charles Street, Baltimore, MD 21210

Abstract

A logistic regression model, based on all 2430 Major League Baseball games from the 2010 season, is developed to simulate all of the games of that season and the ensuing playoff games. This simulation is performed 1000 times. The simulation is conducted using a 3000+ line program coded in the package, Matlab. The intent is to observe just how much variability is possible in the final divisional standings and the winner of the World Series given one set of 30 team strengths.

Keywords: Simulation, Sports Statistics, Team Strength Coefficients, World Series

1. Introduction

This research is a simulation study of the 2010 Major League Baseball (MLB) regular season and playoff game outcomes. Probabilities of game outcomes are yielded from a logistic regression model, which is fit using all of the actual game outcomes from the 2010 regular season. This fitting provides a set of 30 "team strength coefficients" and these are used to calculate estimated probabilities of any one team defeating any other team in either's home ballpark. These 900 probabilities are then used to simulate all of the original 2430 games and the outcomes to the ensuing playoffs. This simulation is easily reproducible, so the 2010 season is simulated 1000 times leading to 1000 sets of standings and playoff results.

The statistical software package, Minitab (2012), is used to fit the model for the 2010 season. But most of the computer work is conducted by over 3000 lines of Matlab (2013) code. The latter program is written so that a person conducting the simulation may input the number of times the entire season is simulated. The program, for each simulated season, also outputs the number of wins for each team in the regular season as well as results for all of the playoffs series including the World Series. The program keeps updated figures for averages over the entire set of simulations.

2. The Model

The Logistic Regression Model that is utilized takes the form:

The Probability of Team B beating Team A at B's Stadium = P (B beats A at B's Stadium) = $\frac{e^{\beta_0+\beta_B-\beta_A}}{1+e^{\beta_0+\beta_B-\beta_A}}$, where β_B is the team strength coefficient of team B, β_A is the team strength coefficient of team A, and β_0 is the home team advantage. From this equation, we also find that: P (A beats B at B's Stadium) = 1 – (the above expression) = $\frac{1}{1+e^{\beta_0+\beta_B-\beta_A}}$. Note that this model follows the form of the Bradley-Terry Model (1952) where one entity's strength is weighed against that of another yielding the probability of the first entity "winning" a paired contest.

In Minitab, the model is fit outputting estimates, the β's (team strength coefficients), for all 30 teams using all outcomes for the 2430 MLB regular season games from the 2010 season.

3. The Regular Season Simulation

To conduct the many simulations of one MLB season, the team strength coefficients of all of the teams are placed into a 30 element vector. A 30 by 30 matrix is constructed containing the number of times each team played every other team at a specific team's home field. The Matlab program is, effectively, a huge series of loops where the logistic regression model is applied to each set of opponents. The probabilities of each team beating another in each home stadium are computed. These probabilities are then compared to random and uniformally distributed values between 0 and 1 to determine the winner of each game that was played in the schedule. If the random number is less than or equal to the probability found, it is decided that the home team wins. Otherwise, the away team wins. For example: assume the model's probability that the Red Sox beat the Orioles in the Red Sox's home park is .6854. If the random number associated with one of these games is less than or equal to .6854, the game is labelled as a Red Sox win. Otherwise, the outcome is an Orioles win. This is done for all 2430 regular season games by iterating through the 30 by 30 matrix of games played and inputting the respective team strengths each time. Below is a sample of code that was part of the program that determined outcomes of regular season games.

```
for b = 1:a
    schome = sc(b,1);
    for c = 1:a
        scaway = sc(c,1);
        prhtw = (exp(hta+schome-scaway))/(1+(exp(hta+schome-scaway)));
        r = rand(1,(num(b,c)));
        if length(r) > 0
            d = length(r);
            z = zeros(1,d);
            for e = 1:d
                if (r(1,e))<=prhtw
                    z(1,e) = 1;
                    f(b,c) = (f(b,c))+1;
                    ff(b,c) = (ff(b,c))+1;
                else
                    z(1,e) = 0;
                    f(c,b) = (f(c,b))+1;
                    ff(c,b) = (ff(c,b))+1;
                end
            end
            z=z;
            homewins = sum(z);
            homelosses = d-homewins;
            hw = hw+homewins;
            hl = hl+homelosses;
        end
    end
end
```

4. Pre-World Series Playoff Simulation

The next part of the Matlab program simulates the playoffs. This is the most meticulous part of the program as there are many scenarios that can result due to ties at the end of the regular season and to the need of applying the correct tie-breaking scheme. First, all of the teams within a division are compared and the team with the highest winning percentage from each division moves into the playoffs. Then, the remaining teams are all compared and the team with the highest winning percentage in each league is chosen as that league's wild card team and these teams also move onto the playoffs. Note that, in 2010, the two-wild card team set-up had not been established.

When determining the teams that move into the playoff, ties often occur. The program deals with ties in a manner consistent with the actual rules of MLB. This may involve one game tie breakers or more complicated schemes. The program then pits the appropriate teams against each other in the first round of the playoffs. The top team in each league plays the wild card, and the 2nd and 3rd best teams play each other, unless the top team and the wild card are from the same division. In this case, the top team plays the 3rd best team and the 2nd best team plays the wild card. So two five-game series are played in each league and the two winners then play a seven game series to determine the league champion. Below is part of the program that simulates the playoffs:

```
prhtw31 = (exp(hta+(sc2(1,1))-(sc2(2,1))))/(1+(exp(hta+(sc2(1,1))-(sc2(2,1)))));
prhtw32 = (exp(hta+(sc2(2,1))-(sc2(1,1))))/(1+(exp(hta+(sc2(2,1))-(sc2(1,1)))));
r4 = rand(3,1);
r5 = rand(2,1);
results2 = zeros(3,1);

if al21>al22
    if (r4(1,1))<=prhtw31
    results2(1,1) = 1;
    end
    if (r4(2,1))<=prhtw31
        results2(2,1) = 1;
    end
    if (r4(3,1))<=prhtw31
        results2(5,1) = 1;
    if (r5(1,1))>prhtw32
        results2(3,1) = 1;
    end
    if (r5(2,1))>prhtw32
        results2(4,1) = 1;
    end
else
    if (r4(1,1))>prhtw32
    results2(1,1) = 1;
    end
    if (r4(2,1))>prhtw32
        results2(2,1) = 1;
    end
    if (r4(3,1))>prhtw32
        results2(5,1) = 1;
    end
    if (r5(1,1))<=prhtw31
        results2(3,1) = 1;
    end
    if (r5(2,1))<=prhtw31
        results2(4,1) = 1;
    end
end
results2 = results2;
sr2 = sum(results2);

if sr2 > 2
    poag3(2,1) = poag2(1,1);
    sc3(2,1) = sc2(1,1);
else
    poag3(2,1) = poag2(2,1);
    sc3(2,1) = sc2(2,1);
end
```

5. World Series Playoff Simulation

Finally, the program simulates the World Series. The two league champions are pitted against each other in a best-of-seven game series. The team that gets home team advantage, in real baseball, is based on which league wins the All Star game in that year. Hence, we assumed that the All Star game is nothing more than a flip of a coin, a 50-50 chance and based the home team advantage by assignment random: if a random number between 0 and 1 is less or equal to .5, the National League gets the home team advantage. Otherwise, the American League gets it.

The following is a portion of code used to determine the World Series winner.

```
ff = ff;
almat = ff(1:14,16:30);
nlmat = ff(16:30,1:14);

alsum = sum(sum(almat));
nlsum = sum(sum(nlmat));

r14 = rand(7,1);
wswins = zeros(7,1);

pralh=(exp(hta+(wssc(1,1))-(wssc(2,1))))/(1+(exp(hta+(wssc(1,1))-(wssc(2,1)))));
prnlh=(exp(hta+(wssc(2,1))-(wssc(1,1))))/(1+(exp(hta+(wssc(2,1))-(wssc(1,1)))));

if alsum>nlsum
    for g = 1:4
        if r14(g,1)<=pralh
            wswins(g,1) = 1;
        end
    end
    for gg = 5:7
        if r14(g,1)>prnlh
            wswins(gg,1) = 1;
        end
    end
else
    for q = 1:4
        if r14(q,1)>prnlh
            wswins(q,1)=1;
        end
    end
    for qq = 5:7
        if r14(qq,1)<=pralh
            wswins(qq,1) = 1;
        end
    end
end

wswins = wswins;
swswins = sum(wswins);

if swswins>=4
    winner = alc;
else
    winner = nlc;
end
```

6. Summarizing the Results of 1000 Simulated Seasons

After the program completes 1000 simulated seasons, there are many descriptive statistics and graphs that can used to summarize the findings. The tables, below, list all 30 teams, by League and Division. Table 1 summarizes the American League East. Column 1 and 2 specify the team and the number of actual team wins in 2010. Column 3 lists the average wins over the 1000 seasons and the standard deviations of the 1000 win totals. Colum 4 shows the difference between the actual team wins and the average team wins over the simulations. The largest difference between these two values is just .389 for

the Chicago Cubs. So the most the simulation varies from an actual team win total is under a half of a game. The logistic model does a wonderful job of accounting for teams' wins over the long run. These differences are computed simply to show how well-calibrated the model is. Note that it is not the intent of this research to develop a model that, in one simulation, closely resembles the outcomes of one MLB season.

Column 5 lists the team strength coefficients as found by fitting the logistic model. Column 6 shows the number of times the team wins the World Series in 1000 simulated seasons, Column 7 is the number of times the team comes in first place in its division in the 1000 simulated seasons, and the rest of the columns show the number of times the team finishes in the remaining levels of the standings.

Note that Tables 2-6 cover the other five divisions in MLB and note that the content of Columns 3-5 of Table 1 is omitted in the latter five tables.

Table 1: Simulation Results for Teams in the American League East

A.L. East Team	Real Wins	Mean (SD) of Simulated Wins	Abs (Real-Sim)	Team Strength	WS Wins	Div Wins	2nd in Div	3rd in Div	4th in Div	5th in Div
Tampa Bay	96	96.28 (5.96)	.28	.444599	137	479	318	153	50	0
New York	95	94.81 (6.25)	.19	.415369	117	395	341	198	65	1
Boston	89	88.81 (6.12)	.19	.293867	25	92	245	406	254	3
Toronto	85	84.72 (6.26)	.28	.217333	8	35	99	242	599	25
Baltimore	66	65.96 (6.00)	.04	-.240282	0	0	0	12	27	961

Table 2: Simulation Results for Teams in the American League Central

A.L. Central Team	Real Wins	Team Strength	WS Wins	Div Wins	2nd in Div	3rd in Div	4th in Div	5th in Div
Minnesota	96	.336433	137	690	253	55	4	1
Chicago WS	95	.174200	117	255	547	185	12	1
Detroit	89	.017258	4	55	182	619	119	24
Cleveland	85	-.231634	8	0	8	98	525	368
Kansas City	66	-.282936	0	0	10	43	340	606

Table 3: Simulation Results for Teams in the American League West

A.L. West Team	Real Wins	Team Strength	WS Wins	Div Wins	2nd in Div	3rd in Div	4th in Div	5th in Div
Texas	90	.221232	75	724	217	59	0	0
Oakland	81	.035340	4	154	397	437	12	0
L.A. Angels	80	.009019	8	122	386	480	12	0
Seattle	61	-.416812	0	0	0	24	976	0

Table 4: Simulation Results for Teams in the National League East

N.L. East Team	Real Wins	Team Strength	WS Wins	Div Wins	2nd in Div	3rd in Div	4th in Div	5th in Div
Philadelphia	97	.371493	153	704	250	38	5	1
Atlanta	91	.206189	60	266	550	144	29	5
Miami	80	-.031651	2	16	16	433	371	80
New York Mets	79	-.061216	5	12	99	346	430	106
Washington	69	-.314314	0	2	4	39	165	808

Table 5: Simulation Results for Teams in the National League Central

N.L. Central Team	Real Wins	Team Strength	WS Wins	Div Wins	2nd in Div	3rd in Div	4th in Div	5th in Div	6th in Div
Cincinnati	91	.117587	77	672	252	58	14	1	0
St. Louis	86	.021133	33	264	444	198	64	41	10
Milwaukee	77	-.186540	2	28	113	290	302	254	7
Houston	76	-.192913	0	29	98	247	291	318	13
Chicago	75	-.229070	0	7	93	205	314	352	21
Pittsburgh	57	-.658183	0	0	0	2	15	34	949

Table 6: Simulation Results for Teams in the National League West

N.L. West Team	Real Wins	Team Strength	WS Wins	Div Wins	2nd in Div	3rd in Div	4th in Div	5th in Div
San Francisco	92	.219214	86	520	306	127	51	5
San Diego	90	.161869	56	366	371	193	68	0
Colorado	83	.014026	5	78	200	398	305	16
Los Angeles	80	-.045916	5	36	123	270	516	53
Arizona	65	-.384695	0	0	0	12	60	9261

Looking at the East Division of the American League, note that 2010's last place Orioles never win their division though all 1000 simulations. And they, therefore, never win the World Series. The Phillies (National League Central), with the highest number of regular season wins the World Series, not surprisingly, the most times of all 30 teams. The actual World Series Winner from 2010, the San Francisco Giants (National League West), only wins 86 of the 1000 simulated Series. But note that four teams had more regular season wins than the Giants in 2010 and six other teams finished within four of the Giants. One may argue that the actual 2010 season was effectively just one simulated

trial of the 2010 MLB season. Our analysis demonstrates just how variable the outcomes of that season could have been.

From these tables, it is also easy to see the impact of the division a team plays in. For example, the Yankees and the Rays (both in the American League East), won 95 and 96 actual season games, respectively. The Phillies won 97 actual season games, just one more than the Rays and two more than the Yankees, so one may claim they are nearly equally strong teams. However, when you look at the number of times each team won the World Series, there is a large difference. This is because the Yankees and the Rays play in the American League East and the two teams tend to split the number of times each wins the A.L. East giving them less opportunities to win the World Series. Meanwhile, the Phillies do not have another really strong team within their division, so they are constantly winning their division (704 times compared to the Yankees and the Rays 390 and 478 times).

Another interesting observation is that both the Yankees and the Rays actually hold a larger strength coefficient than the Phillies (.415369 and .444599 versus the Phillies .371493), even though the Phillies won more regular season games. This once again confirms how important a role a team's assigned division plays. The Yankees and Rays are "stronger" teams, yet, since they play stronger competition in the regular season games, their overall win totals are small compared to Philadelphia.

Another observation from these tables, which is one of the main themes of this paper, is the role randomness plays in Major League Baseball. People are often quick to claim that the team that wins the World Series in a particular season is one of the best if not the best team in the MLB. However, one could consider that a Major League Baseball season is nothing more than one simulation of 2430+ games. For example, throughout the 1000 simulated seasons the White Sox (American League Central) won the World Series only 27 times. In the actual 2010 season, they went 88 and 74, so they were a strong team while not a powerhouse. Our simulation suggests that there is a small chance (27/1000) that a team of that caliber could "luck" themselves into the World Series title. But, had the White Sox actually won the title in 2010, they would have been automatically considered an elite team. The fact is that there is a degree of randomness that exists in each game played in MLB. And, hence, a degree of randomness in the final divisional standings and World Series result as well.

From the 1000 simulated seasons, it is also very interesting to see how many times each team finishes in each "place" within their final divisional standings. From the tables, one can see that there are certain teams that never win their division, such as the Orioles, and there are certain teams that never finish last, such as the Reds. So, again, there is evidence of a great deal of variation in how teams finish within their divisions.

Below, note histograms for the total number of wins over the 1000 simulated seasons for three teams. The horizontal axis is the number of wins in the regular season and the vertical axis is the frequency for the number of times in 1000 simulated seasons those number of wins resulted.

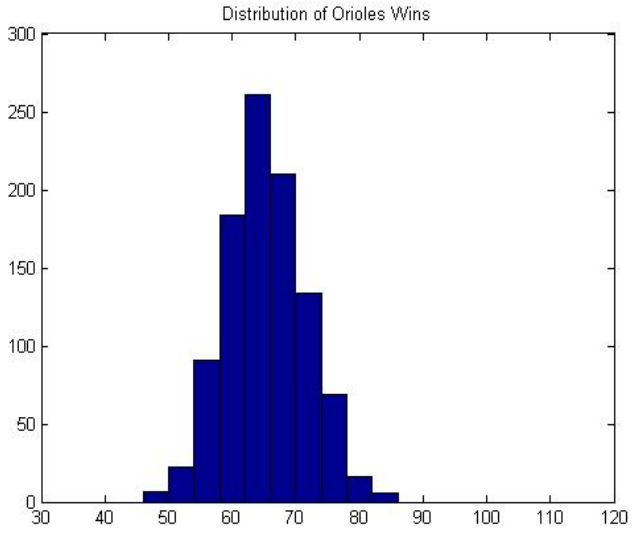

Figure 1: The Distribution of Total Wins Over 1000 Simulations for the 66-96 Baltimore Orioles

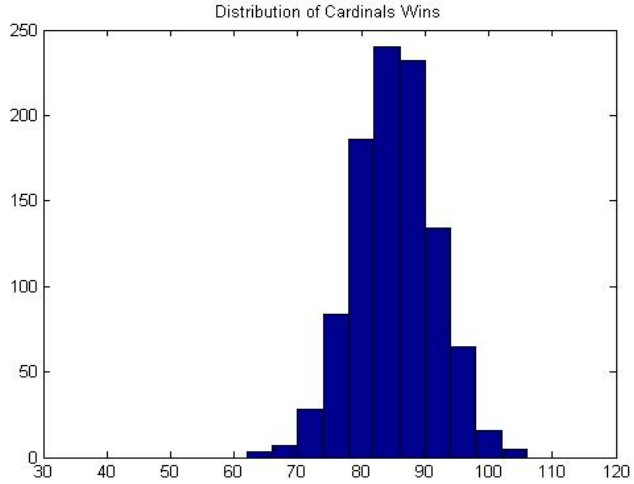

Figure 2: The Distribution of Total Wins Over 1000 Simulations for the 86-76 St. Louis Cardinals

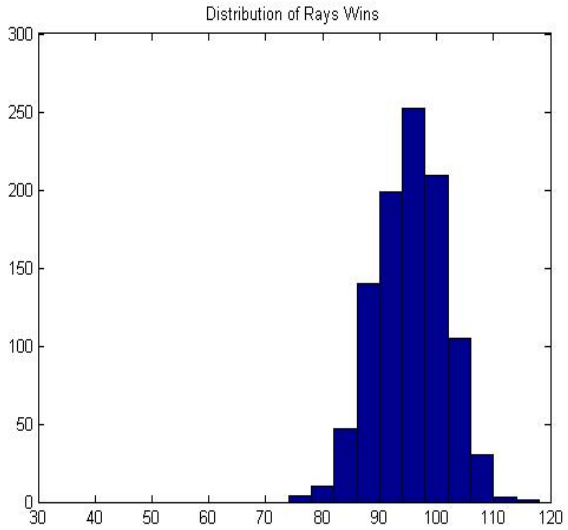

Figure 3: The Distribution of Total Wins Over 1000 Simulations for the 96-66 Tampa Bay Rays

The shapes of the three histograms appear fairly similar and bell curved. Note how the three graphs, of similar shape and similar spread, seem to be simply "moving up the horizontal axis." Looking at Table 1, note that the SD of wins for the American League East runs from 5.96 to 6.26. Over all 30 teams, the SDs only vary from 5.96 to 6.55. A test of the correlation between SD and average wins over the 1000 simulations suggest no linear relationship between the two measures. So neither the better teams nor the lesser teams tend to show larger variation over the simulations.

7. Summary

In summary, the Logistic Regression Model, over many simulations yields average wins very close to the actual 2010 season values. And the model also shows how easily the final standings and champion could have deviated from those seen in real life. There is much randomness in baseball (and in all sports) that is often under-appreciated.

References

Attaway, S. 2013. *Matlab, third edition: a practical introduction to programming and problem solving*. Butterworth-Heinemann, Oxford, UK.

Bradley R.A. and Terry, M.E. 1952. Rank analysis of incomplete block designs I: the method of paired comparisons. *Biometrika,* 39, 324-345.

Ryan, B.F., Ryan, T.A., and Joiner, B.L. 2012. *MINITAB handbook: update for release 16.* Stamford, CT.