The Robinson Cano Problem

Using Bayesian Statistics to Solve Rest of Year Home Run Statistics for MLB Players

By Daniel Leong

Background

When my Seattle Mariners signed Robinson Cano to a 10 year, **240 Million** dollar contract last offseason, they made a desperate move to fix their greatest weakness: hitting. Simply put, the Mariners couldn't hit. In particular, they couldn't hit for power, with just a few players in the past five years eclipsing 25 home runs. Cano seemed like the perfect answer. Before coming to Seattle, he had amassed five seasons in a row of 25 or more home runs. He was supposed to be the remarkably consistent power hitter that they had lacked for so many years. So, it's understandable that many fans and analysts were terrified when Cano had two home runs in June, more than a third of the way through the season. They raised a simple question. After early season power outage, how many home runs was he going to end up with at the end of the season?

Method

In order to tackle a problem like this, I needed to decide how to model it. I chose to use a Poisson distribution, which assumes that a hitter has an equal likelihood at any point in time of hitting a home run. This process is commonly used with time-based games like Soccer and Hockey, were the rate (lambda) can be easily defined as Goals/Game. Implementing this in baseball is slightly more complex. Instead of using Goals/Game, I came up with my own rate, Home Runs/Season. Instead of minutes as the discrete time points measured in the rate, I used whole games. Using this assumption, I could easily use a Poisson distribution to show the likelihood of a player hitting various numbers of home runs in the season. However, finding a value for Home Runs/Season is a lot more difficult than Goals/Game. Because the number needs to account for career totals instead of season averages, there is much larger variation. Taking the average of previous year home run values will not give you an accurate picture, as it ignores inter-season trends and detrimental effect of aging on power hitters. In order to accurately predict trends, I used a weighted average of the changes in home run values between year to year, seen below.

Year	Home Runs	Change in Home Runs	Weighted Change
2010	29	N/A	N/A
2011	28	-1	-1 * (⅓) **-.66**
2012	33	+5	5 * (⅓) **= 1.66**

2013	27	-6	-6 * (½) = **-3**

I used the past four years of Cano's career for two reasons. One, he was remarkably healthy in each of the four seasons, playing all but eight games between them. Two, it gave me three change in home run values to weight, which gave me weighting that I wanted. Using this method, I found that the weighted change in Home Runs over his past four seasons was -1.5. Adding this to his 2013 home run total of 27 gave me my expectation of 25.5 Home runs per Season for 2014.

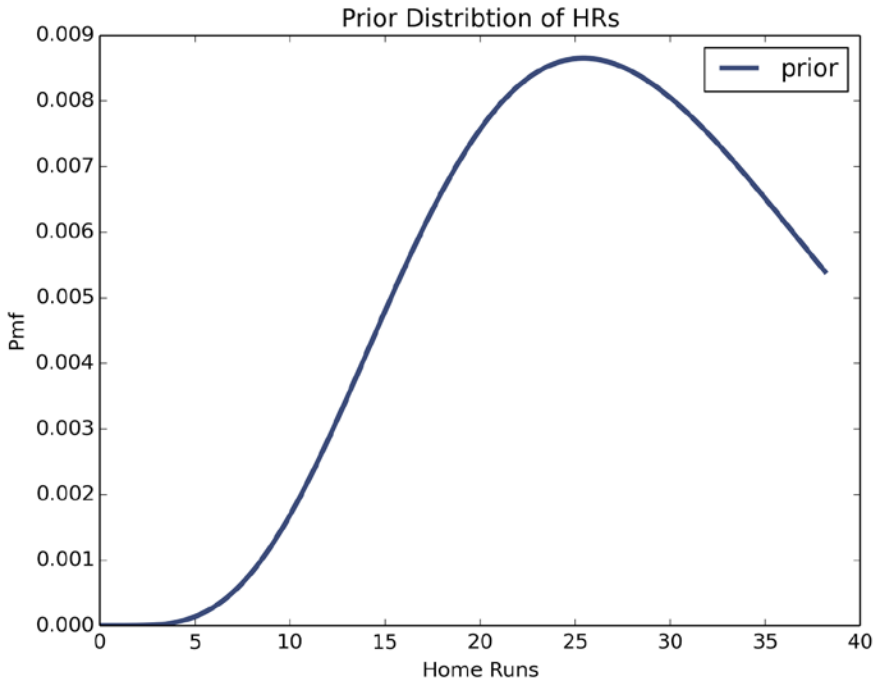
While my prediction of Cano's ability to hit 25.5 home runs per year may have been good when the season started, did it still hold true after he hit two home runs in his first 64 games? I believe it didn't. In order to more accurately assess his home run hitting ability for the last 98 games of the season. I combined the rate that he was currently hitting home runs (about 5 per season) and the rate that I expected him to hit home runs (25.5 per season) and averaged them using this equation.

(Home Runs Hit) * (GamesPlayed/162) + (Home Runs Predicted) * (1-GamesPlayed/162)

Using this equation, I found that Cano's Home run/ Season rate for the rest of 2014 was about 17. This means that for the rest of the season, the model assumes that he will hit home runs at a rate of 17 per season. With this number solved, I could implement it in code.

Code Implementation

In order to implement this, I created a "season" class. This class used an exponential evaluation of a Probability Density Function which takes in a hypothetical value and a rate and outputs a probability density proportional to the probability of the data. The rate input into this function is the rate of Home Runs per game, or (Home runs per season) / 162. Using this likelihood function, I created a linspace vector in the season class, then updated it with Cano's Home Runs per year for his previous three years and the composite value calculated in the last section. This gave a prior for my study, seen below.



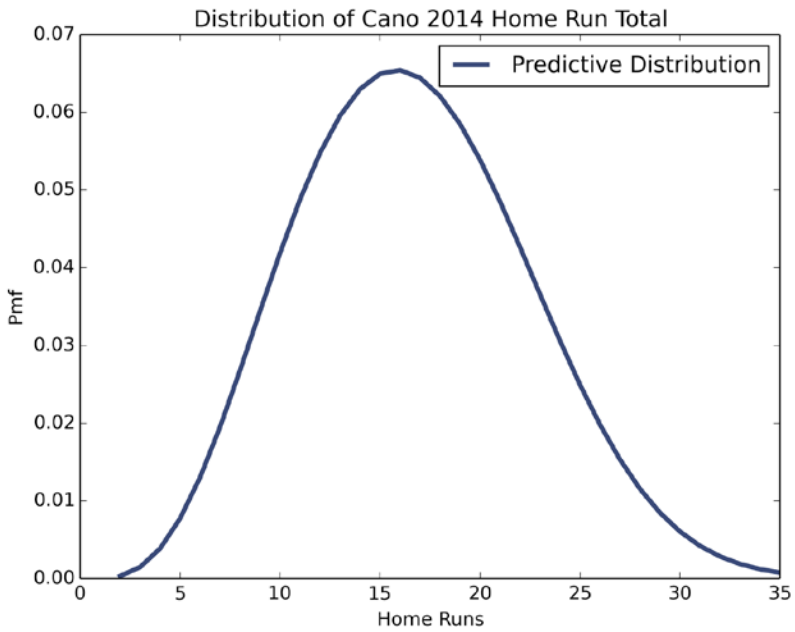
After creating a prior, I wrote a function within the class to predict the number of home runs that Cano would hit for the rest of the season. It takes in his current home run total, the prior, and the number of games remaining and constructs a predictive distribution. The distribution of home runs for the rest of the season is Poisson and based on the home run rate lambda multiplied by the amount of time left in the season. We don't actually know lambda as a discrete term, so instead we iterate through the prior to test a distribution of lambdas. After constructing the predictive distribution, we can find the most likely value, or the most likely number of home runs that Cano will hit during the season.

Limitations

This model has a couple main limitations. First and foremost, it assumes a 162 game season where a player is able to play every game. In 2014, only five players played every game, with a good fraction missing major time due to injuries. Robinson Cano is an easy player to analyze because he misses so few games, but it would require an adaptation to the model to account for injury disturbed seasons. Secondly, the model doesn't take into account park factor, or how hard it is to hit home runs in different stadiums. In between 2013 and 2014, Cano moved from one of the easiest places to hit home runs (Yankee Stadium) to one of the hardest (Safeco Field). A more in depth model would seek to address this problem in its predictions.

Results and Analysis

Plotted below is the predictive distribution of Robinson Cano's home run total for 2014:



In the plot above, the posterior distribution is both narrower and lower than the prior shown in the previous section. This takes into account his lack of power in the first third of the season, but also reflects his consistent history as a home run hitter. Although the model does not predict that Cano will reach anywhere close to where he was for the past four years, it predicts that he will break out of his slump and hit somewhere in the 10 to 20 home run range. According to the Bayesian analysis of the system, Cano should have gone on to hit **15** home runs.

One advantage of analyzing a season that has already concluded is that I can validate my model against the actual results of the season. In reality, Cano finished with a respectable 14 home runs, just one shy of my model's prediction. Because this model did not take into account park factors and many other changes that could have affected his power, this value is well within the margin of error.

Conclusion and Future Work

Overall, I was very pleased with how the model turned out. Given current and past home run data, it is very capable of predicting home run totals for the rest of season. In the future, it could be adapted to include park factors, which affect home run totals, or be used to predict breakout and career years. Because the center framework of the model is so simple, it could be easily expanded to fit a variety of uses.