Lab 12: International Chess: Hot Or Not?

One of the earliest examples of a convergent, adaptive Markov process was the rating system devised by Arpad Elo to rank chess players. It has endured for so long as a simple system for so long that it is used as a primary ranking system in many other scenarios, including the venerable NFL football (Nate Silver), the cruciverbalist Scrabble (NASPA), and the gross FaceMatch (The Social Network).

The gist of it: two players have ratings R_A and R_B. The estimated probability that player A will win is modeled by a logistic curve,

$$P(A) = \frac{1}{1 + \exp(R_B - R_A)}$$

and once a game is finished, a player's rating is updated based on whether they won the game:

$$R_A(new) = R_A(old) + K * (1 - P(A))$$

or lost it:

$$R_A(new) = R_A(old) - K * P(A)$$

for some factor K. (Note that both player ratings change.)

Today we will simulate a repetitive tournament with 10,000 games to see if it converges on the true values.

1. Create a "true" vector of ratings for 13 players whose ratings range from -2 to 2 in even intervals. Create another vector with the current ratings which will be updated on a game-by-game basis, and a matrix with 13 rows and 10,000 columns into which we will deposit the ratings over time.
2. Write a function that simulates a game between players i and j given their true underlying ratings. This should be a simple draw from `rbinom(1,1,p)` with the appropriate probability.
3. Write a function that, given a value of K, replaces the ratings for the two players who just played a game with their updated ratings given the result from the previous question.
4. Write a function that selects two players at random from the 13, makes them play a game according to their true ratings, and updates their observed ratings.
5. Finally, write a function that simulates a tournament as prescribed above: 10,000 games should be played between randomly chosen opponents, and the updated ratings should be saved in your rating matrix by iteration.
6. Run this tournament with $K = 0.01$. Plot the rating for the best player over time using `plot(..., ty="l")`; add the rating for the worst player using `lines(...)`. Do they appear to converge to the true ratings?
7. Repeat the previous step with K equal to 0.03, 0.06, 0.1, 0.3, 0.6 and 1. Which appears to give the most reliable rating results?