Econ 524 Project 5

The data for this project is in pddata.csv. The data is from a prisoner's dilemma experiment. There were twenty subjects who each played 100 rounds. In each round, they were paired with another subject chosen at random. Each round, subjects either earned \$1 or \$0. The probability of winning \$1 depended on the outcome of the prisoner's dilemma as seen in the following table.

Player's	Opponent's Choice	
Choice	A	B
A	0.105	0.005
B	0.175	0.075

The best joint outcome was if both subjects chose A (highest combined expected payoff), however the Nash equilibrium is for both subjects to choose B. B is a dominant strategy, since the probability of getting \$1 is always higher from choosing B than A, conditional on the other player's choice.

The following variables are in the data:

round: 1 to 100
id: 1 to 20 (subject identifier)
choice: the player's choice, 1 = B, 0=A
otherchoice: the player's opponent's choice, 1=B, 0=A
payoff: actual payoff earned, 0 or 1
pa: the payoff from choosing A, 0 or 1
pb: the payoff from choosing B, 0 or 1
econ: 1 for econ major, 0 otherwise

Your goal is to estimate a reinforcement learning model using this data. The following description of reinforcement learning describes a 2 parameter model of how subjects decide whether to choose A or B. Subjects have a propensity to play both strategies (A and B). Propensities are updated according to the following formula for $i > 1$:

$$A_i^j = dA_{i-1}^j + I_{i-1}(j)G_{i-1}(j) \tag{1}$$

where j is A or B and A_i^j is the propensity to play strategy j in round i, I is an indicator function equal to 1 if strategy j was played in round $i-1$ and 0 otherwise, G is the normalized payoff received from playing strategy j in round $i-1$, and $d \in [0, 1]$ is a parameter to be estimated. The probability of choosing j in round i is:

$$\frac{e^{hA_i^j}}{e^{hA_i^A} + e^{hA_i^B}} \tag{2}$$

where $h \in [0, \infty)$ is a parameter to be estimated. The initial propensities A_1^A and A_1^B are both 0.

1. Write a function called ll_rl that takes a parameter vector as input and returns the negative log likelihood for the 1 parameter reinforcement learning model (with $d = 1$). Use the nlminb fucntion to maximize the log likelihood (i.e. minimize the -LL). Save the parameter estimates as result and the log likelihood value as llresult. Compute standard errors using the bootstrap (more details next week). Save the standard errors for your parameter estimates as serror.

2. Write a function called ll_rl2 that takes a parameter vector as input and returns the negative log likelihood for the 2 parameter reinforcement learning model. Use the nlminb fucntion to maximize the log likelihood (i.e. minimize the -LL). Save the parameter estimates as result2 and the log likelihood value as llresult2. Compute standard errors using the bootstrap (more details next week). Save the standard errors for your parameter estimates as serror2.