Problem Set 3 - Sristi Bafna

1. Answers:
    A. We first generate a random number between 0 and 1. We then check first whether this number is less than 0.55 (which is the bias value for our model). If this condition is satisfied then we choose option 1 otherwise we choose option 2 (*i.e.*, when the random number generated is less than 0.45 which is 1-b). This method works because the probability of a number being less than 0.55 is 55% which is the bias towards option 1/choice probability for that option. Likewise, the probability of a number being less than 0.45 is 45% which is our bias value/choice probability for option 2.
    For 3 options, let's assume the following data:
    Bias (b) for Option 1 - 50% or 0.5
    Bias (b) for Option 2 - 30% or 0.3
    Bias (b) for Option 3 - 20% or 0.2
    Choice probability array = [0.5 0.3 0.2]
    Based on this, we would use the following pseudocode:
    ```
    generate a random number between 0 and 1 (rand)
    if rand < 0.5
            choose Option 1
    elseif rand > 0.8
            choose Option 3
    else
            choose Option 2
    ```

    The logic is the same as for 2 options but the implementation would be slightly different.

    B. Figures attached in the PSet Zip file.

2. Answers:
    A. I shifted the conditions checking for the choice on the previous trials and the reward for the same to the part within the loop. This is because the variable t which is our undefined loop variable had not been defined and thus the code was not running. Then I corrected the choice probabilities for the 4 sub-conditions of the model. Also, for the third condition (reward on the previous trial is 0 and the choice is 1), the second statement said choice_probabilities(t,1) again instead of (t,2) which I corrected.

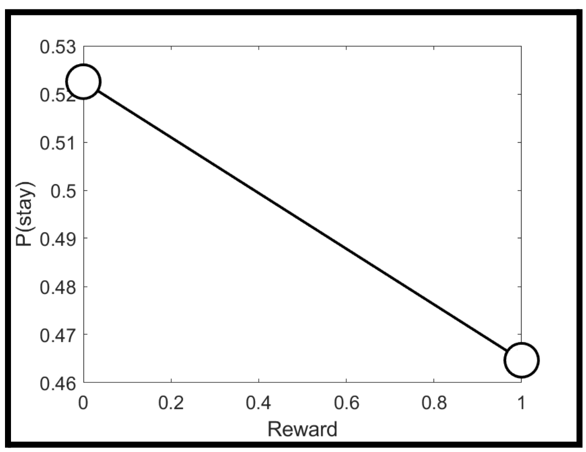
    B. Figures attached in the PSet Zip file.

3. Answers:
   A. The 2 errors in the code were:
      a. Softmax function code - I fixed this by adding this statement = `exp(beta*Q(t,:)) / sum(exp(beta*Q(t,:)))`
         It is important to write (t,:) as Q is an array of value estimates for all trials and for a specific trial, we only need the value estimate of that trial.
      b. Code for updating the value estimates - We only had to change the `beta` to `alpha` here as the value estimate for the current trial is the value estimate for the previous one added to the prediction error which is multiplied by the learning rate `alpha` and not the inverse temperature `beta`.
   B. Figures attached in the PSet Zip file for all given code as well as figure 4 for value estimates for both options as a function of trial number.

4. Code file attached.

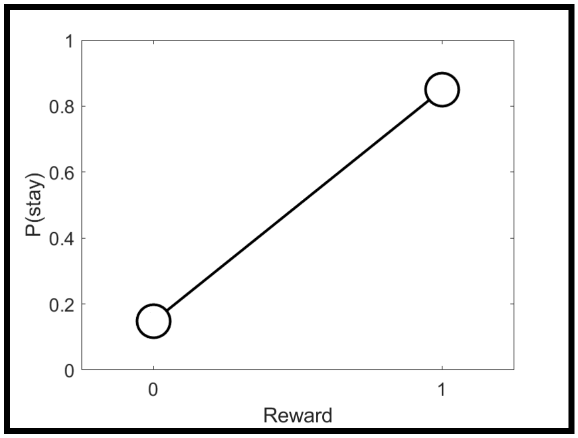
5. Answers:

   a. Model 1 :



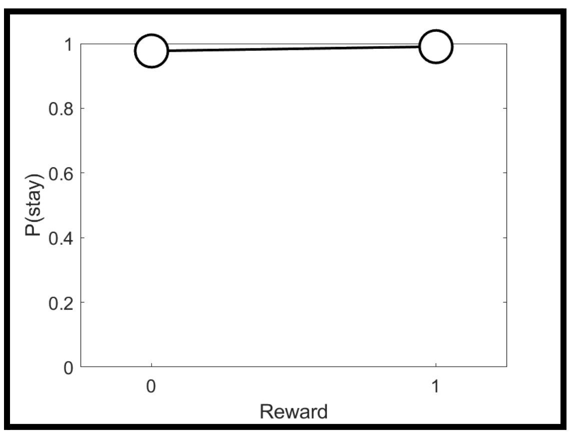
   Insights : As is visible, the probability of staying with the correct option does not depend on the outcome of the previous trial considering this is a random responding model. The probability of the model staying with a particular option shows no dependence on whether it is getting a reward or not and in fact, here, it is less for the correct option (i.e., the option that gives it a positive reward).

   b. Model 2 :



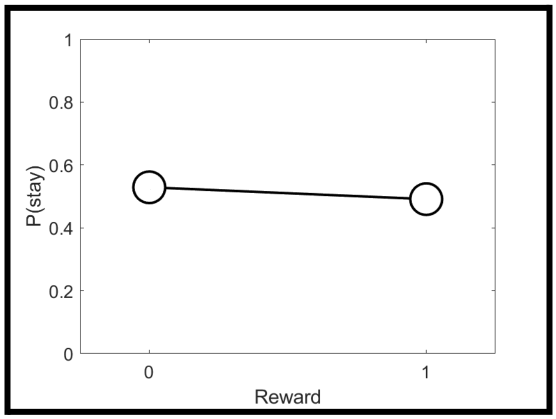
Insights : As is visible from the graph given, Model 2 or the win-stay lose-shift model shows heavy dependence on the outcome from the previous trials. On getting a reward for selecting a particular option, the model shows a strong increase in probability of staying with it. This is because of the model's dependence on the outcome of the immediately prior trial. There is some degree of randomness introduced into the model from epsilon (here, 0.3) which is the only free parameter of this model.

c. Model 3 :



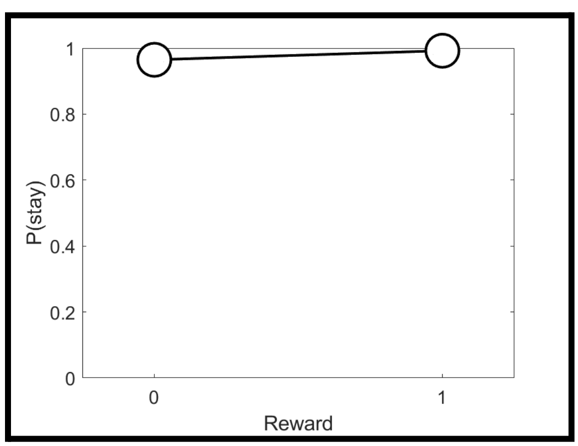
Insights : Model 3 or the Rescorla-Wagner model shows higher probability of staying with the correct option but does not show much dependence on the outcome of the previous trials. This is because the Rescorla-Wagner model does take into account the outcome of the previous trials but does not completely depend on it. It has 2 parameters - alpha and beta. Alpha or the learning rate controls the extent to which it takes into account the outcome from the previous trials and here we had a very low alpha (0.1) as a result of which there was not much dependence on the previous outcomes. However, it was sufficient enough for the model to show higher probability for staying with the correct option (Option 2). Beta which is inverse temperature controls the random choices made by the model implying the extent to which the model considers the value estimate of the trial depending on the previous outcomes. Our beta here was 2 which is quite low which implies a higher chance of the model being driven by noise and having more randomness in selection.

d. Model 4 :



Insights : Model 4 or the Choice Kernel model shows low dependence on the outcome from the previous trials as well. The stay probability seems insensitive to the reward as the line is nearly parallel to the x-axis. Every trial is partially independent and there is some degree of stay in the model's choices.

e. Model 5 :



Insights : Model 5 shows a strong dependence of stay behavior on the outcome of the previous trials due to its Choice Kernel part which takes into account the familiarity component. However, it also has a high tendency to stay with the correct option because of the value estimate component due to the Rescorla-Wagner part of the model.