Problem set 4

(First submission due on 24th November, midnight IST)

In this problem set, you will run simulations of four of the five models that you implemented in problem set 3. In particular, you will look at how changing the parameters of these models changes their behavior. The goal of this problem set is to help you learn how to explore the inner workings of a model. Given that a model represents a hypothesis, exploring the parameter space of a model is equivalent to exploring the scope of the hypothesis. Please read the section "Simulate, simulate, simulate" (including Box 2) in Wilson & Collins (2013) before you solve this problem set.

Problem 1: Take your corrected scripts for models 1,2,3 and 4, which you should have from solving problem set 3. Your first task is to turn these *scripts* into *functions*. In particular, you want to create 4 new files (model1_func.m, model2_func.m…etc) based on your existing simulator scripts. To learn what the difference between a script and a function is, consult this page from MathWorks:

https://www.mathworks.com/help/matlab/matlab_prog/scripts-and-functions.html

- The functions should take as inputs:
  - The *free parameters* of the model
    - only *bias* **b** for model 1,
    - only **epsilon** for model 2
    - both *learning rate* **alpha** and *inverse temperature* **beta** for model 3
    - both *learning rate* **alpha_ck** and *inverse temperature* **beta_ck** for model 3
  - The number of trials to simulate (**T** = 1000)
  - The number of choices (**K** = 2)
  - The reward probabilities (**u** = [0.2,0.8])
- The function should return as outputs:
  - A variable called **choices** which contains the actual choices made the model on each of the T trials.
  - A variable called **rewards** which contains the actual rewards observed by the model on each of the T trials.
  - A variable called **choice_probabilities** which contains the choice probabilities for each option on each of the T trials.

The solution for model 1 is provided as an example for you (model1_func.m). Note that the functions should not plot any figures. They should only return the required outputs. Once you complete problem 1, you can use these functions for problem 2.

Problem 2: After solving problem 1, you should have functions for models 1 thru 4. Your second task is to write new scripts to simulate each model for 150 iterations (an iteration is one run of 1000 trials with specified parameters, 900 iterations means that you repeat that simulation 150 times) at a range of values of its free parameters and save their outputs. The script for doing this for model 1 is provided (model1_sims_ps4.m) as an example. The ranges you should use for each parameter in each model are provided below:

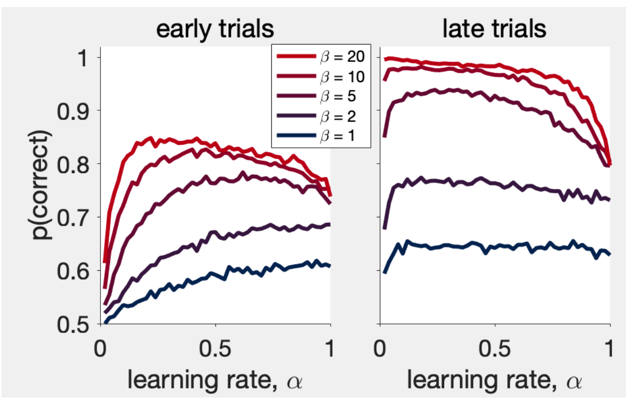
- Model 1: bias **b** = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
- Model 2: **epsilon** = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
- Model 3: learning rate **alpha** = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]

Inverse temperature **beta** = [1, 2, 5, 11, 21]

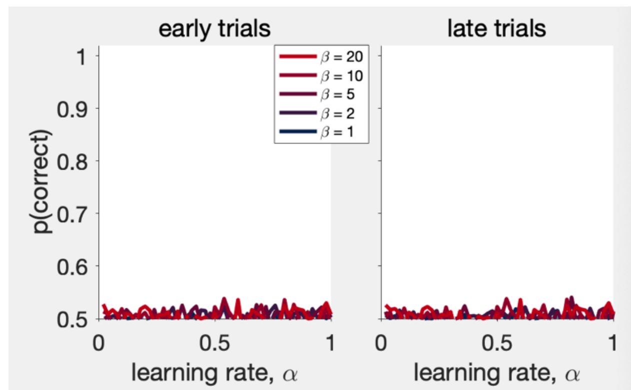
- Model 4: learning rate **alpha_ck** = [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]
  Inverse temperature **beta_ck** = [1, 2, 5, 11, 21]

In total you will run 150 x 130 = 18,000 simulations.

Problem 3: Your third task is to plot some figures and interpret them. You will take the simulated data saved from problem 2, and use it to plot the mean probability (i.e. mean taken over the 150 iterations) of choosing the correct option (i.e. option 2) as a function of parameter values on *early trials* (defined as the first 10 trials) and (late trials defined as the last 10 trials). Examples of these plots are shown below for two of the models. Note that for model 1 and 2, there will online be one curve in each of the two panels (early trials, late trials) as there is only one free parameter in those models and the x-axis will be that free parameter (**b** or **epsilon**).



Model 3



Model 4

To make this easier for you, I've included a script called plot_figs_ps4.m. Once you have run all the simulations and saved the data, you can run this script for each model by changing the model_number variable on line 7 to plot the relevant figures.

In your submission, please plot the figures like the one above for each of the four model (don't just copy paste these figures from above – I will know ☺) and provide your *interpretation* of those figures. In interpreting the figures, I'd like you to think about what the figure tells you about the model and its underlying hypothesis. Note that the plot you see above on the left is also presented in Wilson & Collins (2013) as Box 2 – Figure 1B. Please read the figure legend of this figure in the paper to get a sense of what kind of interpretation I'm looking for. Please also provide all your code in a zip file.

(Optional) Problem 4: For those interested, you can also all of the above for model 5. Note that model 5 has 4 parameters and so there are six plots of all the various combinations of parameters.

References:

Wilson, R. C., & Collins, A. G. (2019). Ten simple rules for the computational modeling of behavioral data. Elife, 8, e49547.