1.1.1 Assignment 2: Computational Cognitive Neuroscience

Computational Modelling of Behavioural Data

Avnish Kumar | s1782889

Abstract

The drift diffusion process for decision modelling and the reinforcement learning based models are implemented on MATLAB. The online version of this document could be used to closely analyze the graphs. The main.m file reproduces all the results and graphs in the document. The computational models are analysed for performance and reliability.

1 DRIFT DIFFUSION PROCESS

1.1 Initial Simulations

The Weiner diffusion model deduces decision making as a constant accumulation of a noisy stimulus until one of the response criteria is met [1].

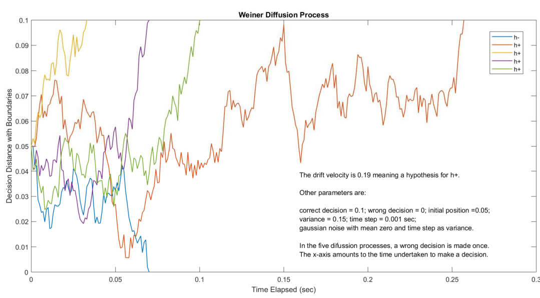

Figure 1. Simulation of the Weiner Diffusion Process.

The model is simulated ten thousand times with the parameter values stated in task (a) viz. v = 0.19, a=0.10, s=0.15, z =a/2 & dt=0.001. Figure 1 represents five of such decisions. With the stated parameter values for ten thousand runs, the mean error probability comes out to be 0.3013 or an accuracy of 69.87%. The response time distribution is separately visualized for correct and incorrect decisions in figure 2 and 3 respectively.

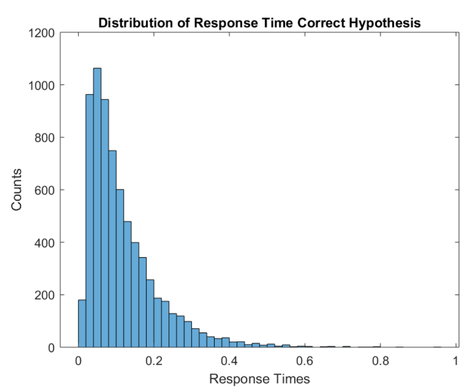

Figure 2. Response time (RT) distribution of correct responses.

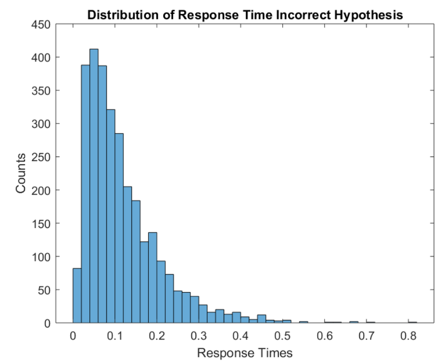

Figure 3. Response time (RT) distribution of incorrect responses.

The RT distribution represents the variability in the response time for the activity to reach the decision boundary. Both the distributions are right skewed suggesting most decisions to be reaching threshold relatively quickly.

1.2 Exploring Parameter Settings

The parameters are explored with respect to constant values of other parameters as stated in section 1.1. Accuracy and response times are measured and discussed as an effect of these variations. In figure 4 and 5, the decision separation distance is varied from 0 to 1 with a step size of 0.01 for ten thousand simulations of each of the values of parameter 'a'.

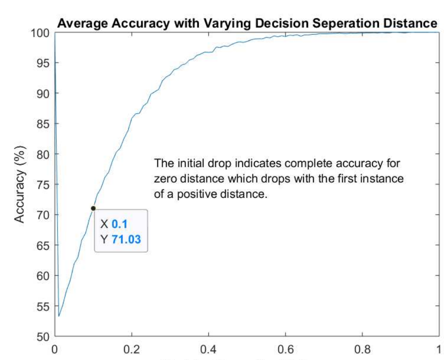

Figure 4. Average accuracy as a function of decision separation distance.

© Author: Avnish Kumar

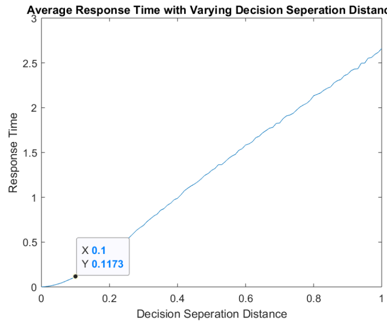

Figure 5. Mean Response Time (RT) as a function of decision separation distance.

Since the variability remains the same, an increase in the decision separation means lesser effect of the noisy stimulus on the response and thus we see the accuracy increasing with increasing decision separation distance. For a standard deviation of 0.001, the accuracy saturates at around 0.7. Similarly, the mean response time increase with an increase in parameter 'a'.

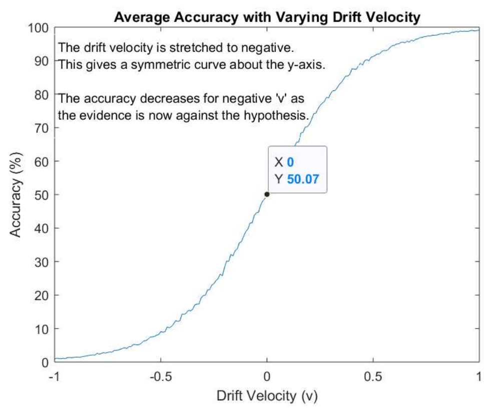

Figure 6. Average accuracy as a function of drift velocity.

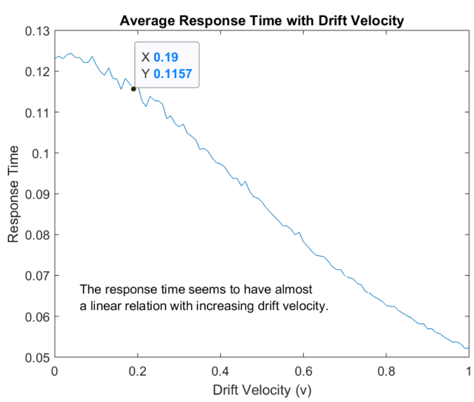

Figure 7. Mean Response Time (RT) as a function of drift velocity.

In figure 6 an 7, the drift velocity is varied from -1 to 1 keeping the other parameters fixed at the provided values. The drift velocity represents the evidence for a decision and thus dictates the quality of information. A positive value signifies evidence towards the correct hypothesis and vice-versa. 50% accuracy for 0 drift reflects random choice. For, mean RT, the value decreases with quality information as increased drift velocity.

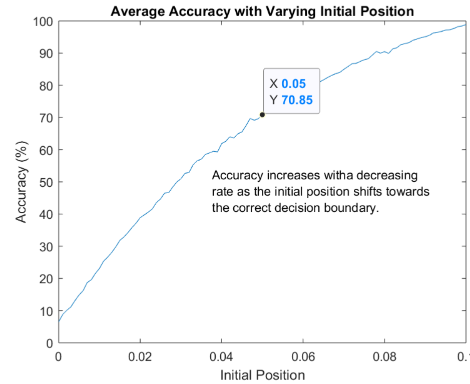

Figure 8. Average accuracy as a function of initial position 'z'.

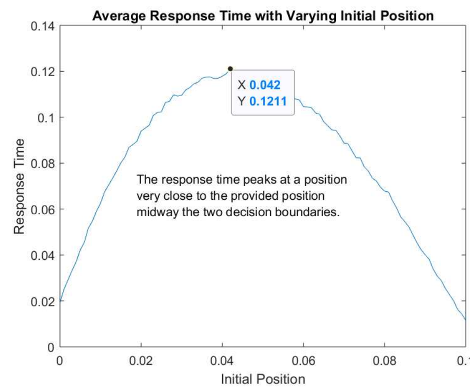

Figure 9. Mean Response Time (RT) as a function of initial position 'z'.

In figure 8 and 9, the initial position 'z' is varied from 0 to 0.1 with a step size of 0.001. The 'z' could be assumed as a bias towards either of the responses. The value 'z' can be used to determine sensitivity towards noise or variability. The mean accuracy increases as 'z' shifts towards the correct decision boundary. The mapping of RT in this context reveals a smaller mean value whenever the bias 'z' deviates from middle distance but the accuracy depends on the direction of the shift.

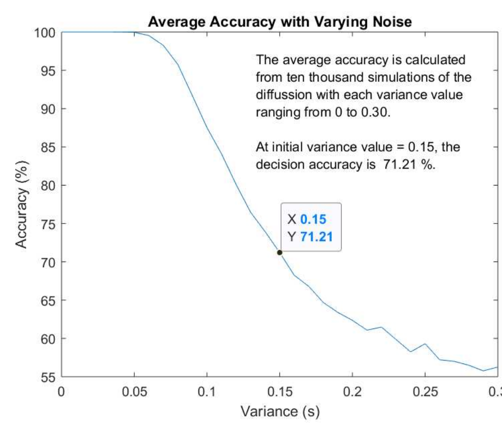

Figure 10. Average accuracy as a function of variance.

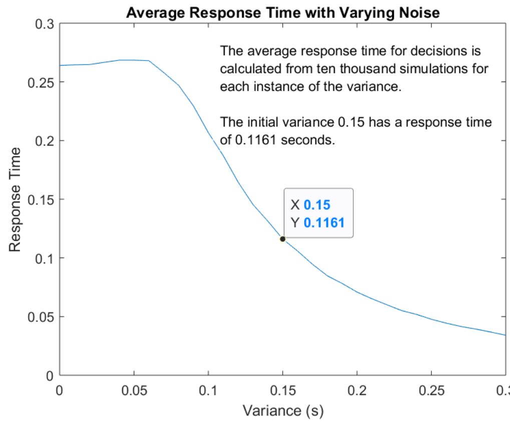

Figure 11. Mean RT as a function of variance.

The variance determines the quality of information served to the diffusion model. Here, it is the gaussian noise. The variance is varied from 0 to 0.3 with a step size of 0.01. A higher value means more erratic behaviour of the model and thus reduced accuracy. For the RT, a higher variance means larger jumps and hence a reduced RT as depicted in figure 6.

1.3 Simulating Experiments

From section 1.2, we now know the dependence of accuracy and response time on the four parameter values. The different groups constituting of major depressive disorder and healthy controls could be modelled through combination of these parameters (derived through iterative calculations of multiple parameter settings). Below graphs represents the concerned accuracy and response times models as required in the question.

For experiment 1 where the accuracy is similar across groups, but response time is different, undermentioned parameter values model aptly.

Experiment	Group	Drift Vel	Seperation	Variance	Initial Pos	Time Step	Accuracy	RT
1	Patients	0.19	0.2	0.15	0.1	0.001	85.09	0.3755
1	Healthy	0.36	0.1	0.15	0.05	0.001	84.23	0.1014
2	Patients	0.19	0.15	0.15	0.075	0.001	78.75	0.235
2	Healthy	0.19	0.1	0.085	0.05	0.001	93.74	0.2362

Table 1. Parameter values to model the experiments distinguishing groups on accuracy and response time.

The variance in response times differ across groups for the two experiments on different scales with the above parameter settings. In experiment 1, the variance for modelled patient data is very high compared to that of healthy participants which have significantly higher RT values. In the second experiment, as the RT values remain similar across groups, the variance values also remain similar. This is depicted in figure 12 and 13 for both the experiments.

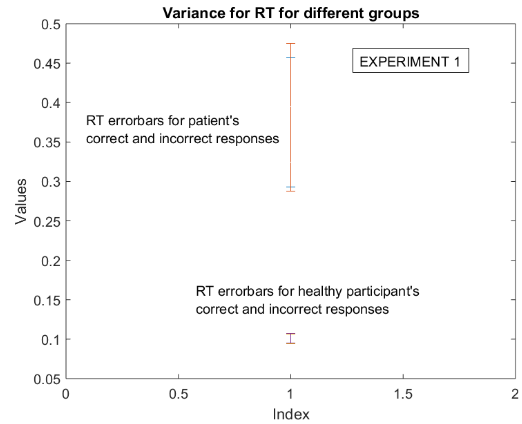

Figure 12. Error bars are plotted for experiment 1with model having similar accuracy but different RT.

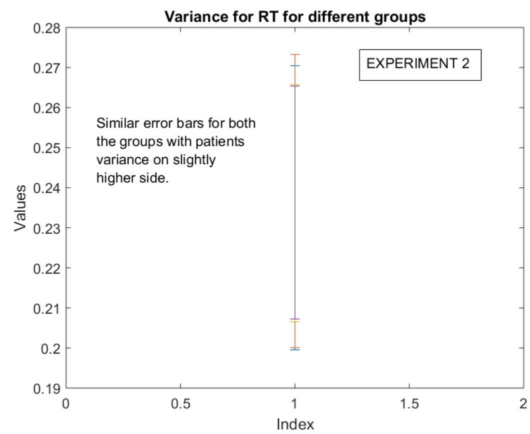

Figure 13. Error bars are plotted for experiment 2 (right) with model having similar RT but different accuracies.

2 MODEL FITTING

2.1 Simulations

250 choices with parameter settings $\varepsilon = 0.35$ and $\beta = 5.5$ are simulated and the resulting values for stimulus A & B are visualized in figure 14 and 15 respectively.

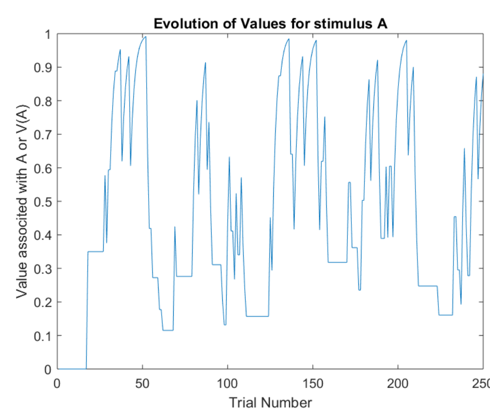

Figure 14. Evolution of Values of stimulus A.

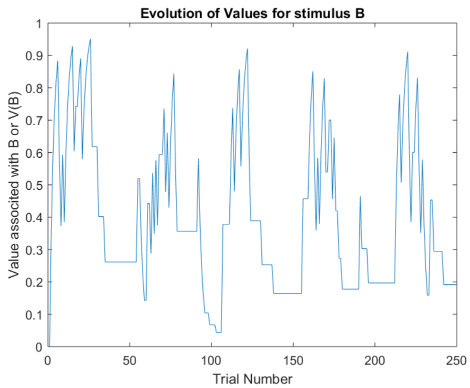

Figure 15. Evolution of Values of stimulus B.

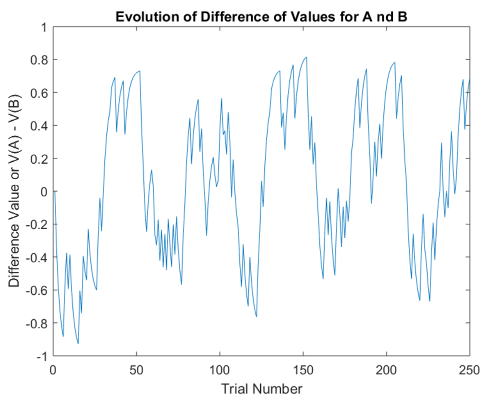

Figure 16. Evolution of difference of Values of stimulus A & B, i.e. V(A) – V(B).

The evolution of values for stimulus A & B are almost regular patterns of increasing and decreasing values. The model learns for each of the 25 trials and then fails due to a switch in the reward mechanism (probabilities). The values for A & B fluctuate in tandem where the value for one stimulus is high and other low at each trial as suggested by the difference V(A) – V(B) graph. The evolution depicts the switch of reward values as local learning. The average reward from the model comes to approximately 146.79. Average number of received **rewards** for **randomly** responding participants is **134.1710**.

2.2 Exploring Parameter Settings

In figure 17, the learning rate and the inverse temperature are iterated over a range of values and mapped to the average reward for each of the combinations with 250 trials. Each combination of parameters is iterated ten thousand times to compute the average reward. The range of learning rate is taken 0 to 1 with step size of 0.1 and inverse rate as 0 to 10 with step size 0.25.

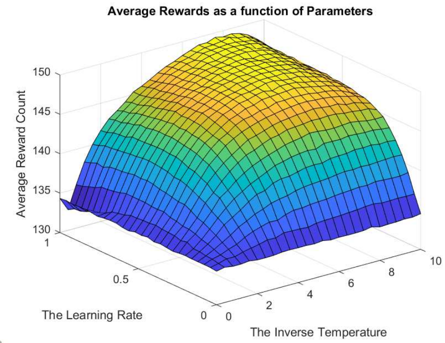

Figure 17. Average reward as a function of learning rate and inverse temperature.

The approach of mapping reward as function f(ε, β), elucidates the dependence of each of the parameter on the performance of the model. As evident from figure 11, the learning rate and the inverse temperature are directly correlated with reward value and an increase in any of these increase the rewards earned. So, a higher learning rate and inverse temperature means a better performance from the model. To analyze the behavior of the parameter settings, the range is further extended in figure 12.

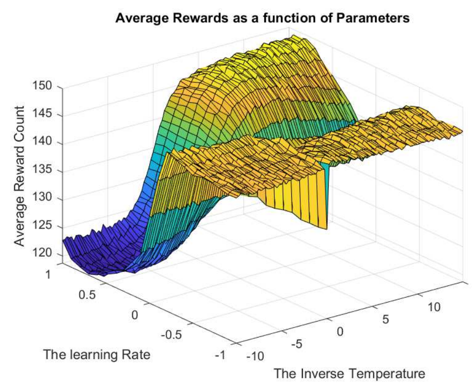

Figure 18. Average reward as a function of learning rate and inverse temperature for extended ranges.

Though negative values could be considered as absurd inputs, one significant inference is for the saturation of reward minimum and maximum which are approximately 120 and 150 correspondingly.

2.3 Likelihood Function

The function ‘nll.m’ is available in the code zip. The negative log likelihood is computed as the sum of the log of the choice probabilities.

The NLL computed for the second patient is 127.2260.

2.4 Model Fitting

The provided data is passed through the nll function. Another function fminunc is used to optimize the input parameters [ε β] by feeding a set of initial values. The fitted values are saved as a table in the code zip. The initial values of theta[0 0] lead to the same fitted parameter values as reported using values [0.3 4], [0.9 5], [0.5 6]. This indicates that the optimized values are invariant to the initial guesses of the parameters. But, for absurd values like [3 20], the optimizer does not converge to the same values.

The parameters estimated using sampled data are closest to the unknown population parameters when the negative log likelihood is minimized through the model. It's invariance to sensible initial guesses means that the model reaches the same minima in the region of the values fed as initial guesses.

In figure 19, the best estimated parameters from the fitting are visualized distinctly for different groups. The learning rate is similar across the groups, but the inverse temperature is larger on average for the healthy controls relative to the patients. In figure 13, green represents healthy participants and red represents the patients.

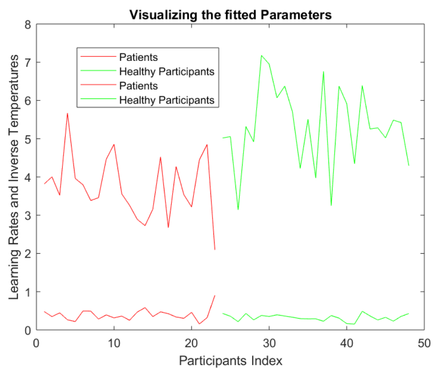

Figure 19. Best fitted parameters visualized for different groups of participants.

The correlation is represented using the Pearson coefficient which measures the linear dependence between random variables. The Pearson correlation coefficient between the estimated parameters within the patients comes out to be **-0.5916**. For the healthy controls, it is **0.1815**. The correlation across the groups turns out to be **-0.35**. This means that the patients parameters have somewhat negative correlation as compared to healthy controls which seem to be highly uncorrelated. Overall, there isn't substantial evidence to conclude total correlation between the estimated parameters.

2.5 Group Comparison

Two-sample t-test comparing the location parameter of two independent data samples, is used to measure the difference between the estimated parameters across groups. The null hypothesis for the learning rate is zero, indicating no significant difference between the two groups for the concerned parameter. Contrastingly, the null hypothesis is rejected in case of inverse temperature suggesting a significant difference across the group. P-values, degrees of freedom and t-statistics are provided in the table below.

Parameter	p-value	Dof	t-stats	sd
Learning Rate	0.0524	46	1.9916	0.5978
Inverse Temperature	7.4375e-07	46	-5.726	0.9579

Table 2. Two-sample t-test statistics across groups.

The results indicate that the if at all, the inverse temperature could be used to model depression, i.e. some change in decision making. The reward learning is not significantly distinct across MDD and healthy controls. Though a very small p-value for inverse temperature casts doubt on the validity of the null hypothesis.

If the data were to be real, this would mean that depressed populations have trouble properly executing their decision or in other words, have a changed decision-making mechanism.

2.6 Parameter Recovery

Table three reports the mean and variance of the distribution constructed by the fitted parameter values which are used to subsequently sample parameter values.

Parameter	Mean	Variance
Learning Rate	-0.6041	0.3799
Inverse Temperature	4.5700	1.5382

Table 3. Mean and Variance of estimated parameter values

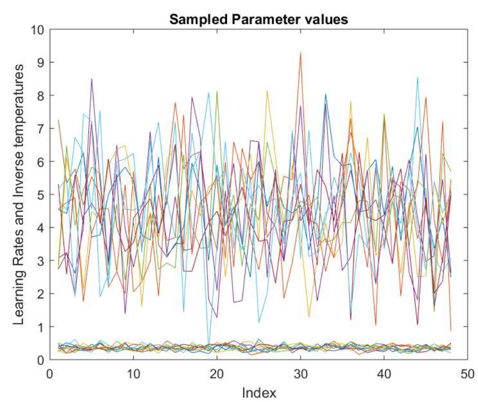

Figure 20. Visualizing ten sets of sampled parameter values form the distribution of best estimated parameters.

Ten sets of sampled parameter values are visualized in figure 20. The learning rate is passed through a sigmoid to output values between 0 to 1.

Forty-eight sets of parameter values are sampled from the normal distribution of previously fitted parameters and again fitted using the fminunc optimization to check for parameter recovery. The Pearson correlation coefficient for the learning rate is -0.23, suggesting a negative correlation between the new fitted parameter values and the ones sample from the distribution. For the inverse temperature, the value is -0.07, suggesting that the data is almost uncorrelated. The correlation is visualized in figure 21 and 22.

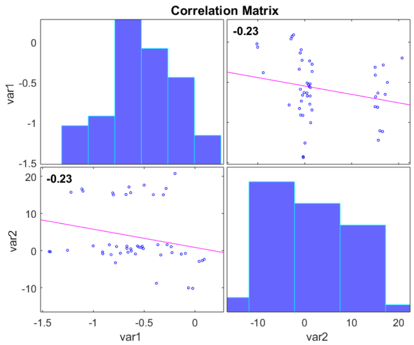

Figure 21. Visualizing the correlation between the fitted and the sampled learning rates.

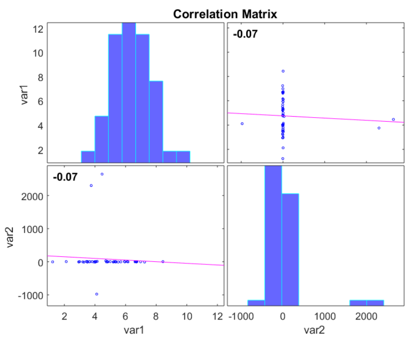

Figure 22. Visualizing the correlation between the fitted and the sampled inverse temperatures.

A higher negative correlation value for the learning rate means a relatively better parameter recovery by the model for the learning rate. The same is not the case for the inverse temperature where the data seems to be almost uncorrelated , indicating a relatively bad recovery of the estimated parameters. This could be attributed to the fact that the inverse temperature is distinct across groups. Multiple simulations of recovery results in varied correlation values for both the parameters are volatile and thus it is also very hard comment about the sampled distribution. The number 48 was chosen as the provided data has 48 samples, this would allow us to judge if the number is sufficient to represent the distribution and thus be recreated.

The performance of parameter recovery with varying sample size X and number of simulation trials is carried out for a range of 50 to 500 for sample X and 100 to 1000 for number of simulation trials respectively. The performance are measured by conducting a two-sample t-test and the Pearson correlation coefficient of the estimated parameters against the initially fitted values. The tstat value of the two-sample t-test and the correlation coefficients are plotted as a measure of performance. Figure 23, 24, 25 and 26 are the performance graphs.

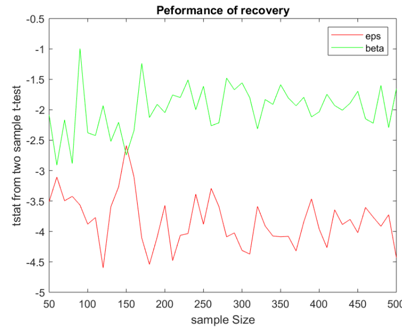

Figure 23. Visualizing the performance of parameter recovery with varying sample X for learning rate and the inverse temperature.

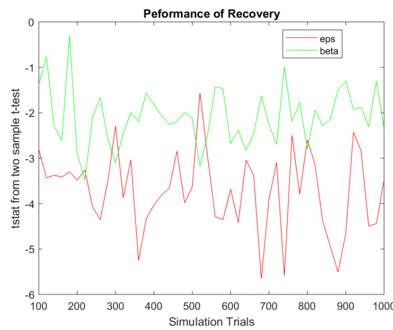

Figure 24. Visualizing the performance of parameter recovery with varying number of simulation trials for learning rate and the inverse temperature.

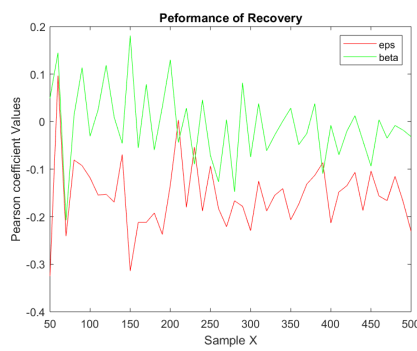

Figure 25. Visualizing the performance of parameter recovery with varying sample X for learning rate and the inverse temperature.

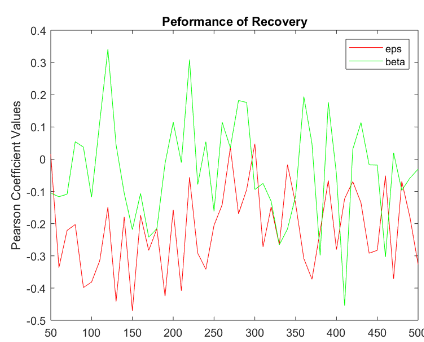

Figure 26. Visualizing the performance of parameter recovery with varying number of simulation trials for learning rate and the inverse temperature.

2.7 Alternative Models

The alternative models 1 and 2 are created as two separate functions for each simulation and negative log likelihood calculation. Figure 25 and 26 represent the evolution of values for stimulus A and B, for model 2 and 3 respectively.

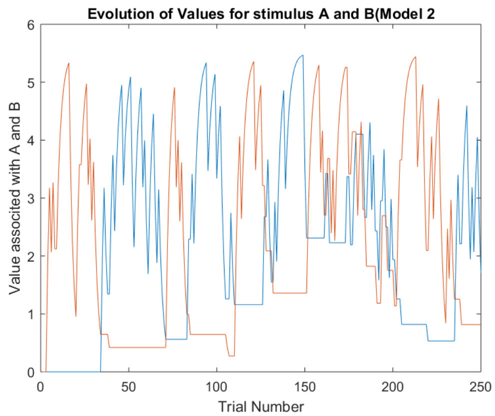

Figure 27. Visualizing evolutions of values of stimulus A & B for model 2.

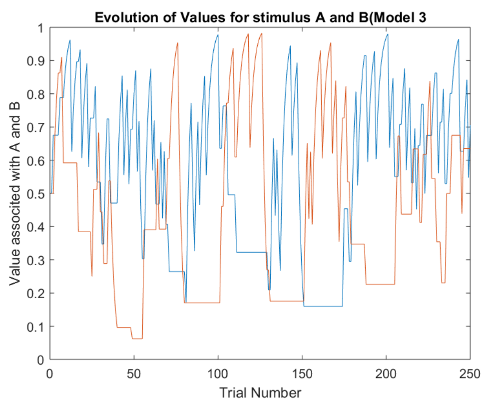

Figure 28. Visualizing evolutions of values of stimulus A & B for model 3.

Figure 29 and 30 represent the fitted parameter estimates for model 2 and model 3 respectively.

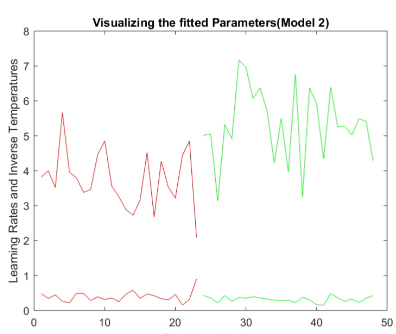

Figure 29. Visualizing the fitted parameter estimates for model 2.

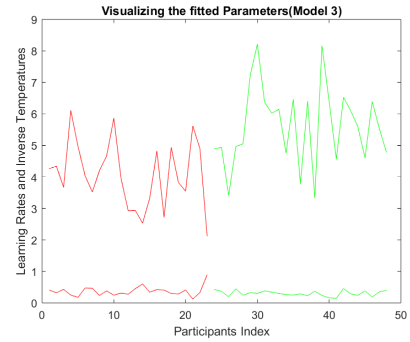

Figure 30. Visualizing the fitted parameter estimates for model 3.

The fitted parameter values for model 1 and model 2 are almost same. Model 3 has values which are different compared to the other two models but has a larger negative log likelihood corresponding to these values suggesting a worse relative fit to the simulation data. This suggests that model 1 and model 2 perform better on the provided data as compared to 3. The different value initiation in model 3 does not map the model to the actual unknown parameter values that well (relative to model 1 & 2). Exactly same result for model 1 and model 2 is a surprising result suggesting rho to be very similar to beta.

2.8 Model Comparison

Table 4 reports the AIC and BIC values for the three different models.

Model	AIC	BIC
1	10744	11082
2	10744	11082
3	10883	11221

Table 4. AIC and BIC values for the three models

The AIC and BIC values are the same for model 1 and model 2 which is less than that of the third model. Thus, either of the first two models can serve as the best model for the provided data. The striking similarity in model 1 and 2 values indicate that their parameters are similar in nature. Model 3 is the worst model which does not aptly estimate the unknown parameters of the actual distribution of the data.

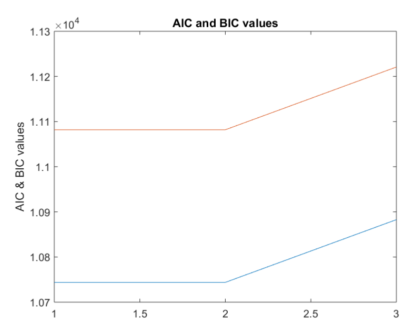

Figure 31. Visualizing AIC and BIC values for the three models represented by x-axis indices.

2.9 Model Recovery and Confusion Matrix

The simulations are performed 100 times for each of the models and the AIC, BIC values are calculated for each of the simulations across the three models. The AIC and BIC values are similar but model 1 is on the lower side on an average. The best fit is the model with the minimum AIC or BIC values. Labels for models are given 1 for being the best fit and 0 otherwise. All the relevant functions are in the code zip. Figure 32 and 33 reveal the reliability of the models.

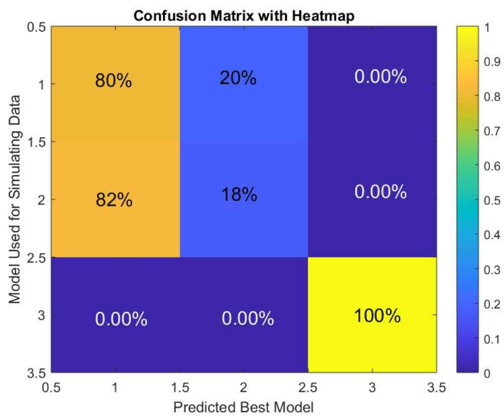

Figure 32. Confusion matrix computed on AIC values of the three models.

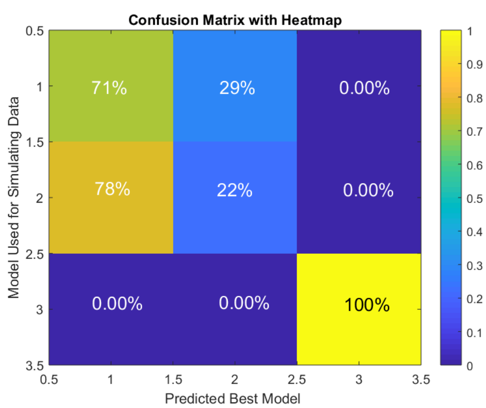

Figure 33. Confusion matrix computed on BIC values of the three models.

The confusion matrix is computed separately using the AIC values and BIC values to understand the effect of these performance parameters. The third model is the most robust model which is always the best fit whereas model 1 and 2 do not exactly represent the simulated data in relative terms. The parameters value used to simulate the data is [0.35 5.5] as provided initially for task (a) experiment. The confusion matrix values are sensitive to the initial parameters being used to generate the normal distribution. It could be argued that the model 1 and 2 are not reliably separable.

2.10 Interpretation

Firstly, we simulated the choices and rewards (data) using some random parameters to understand the governing distributions dependence on these parameters, for the provided data.

In task (b), we explored these parameters to simulate data and calculated average reward to validate the sensibility of using those parameter values to model the data.

Further in task(c), the negative log likelihood is computed. This computes the resemblance of our observed data (simulated or provided) to the prior in terms of probability.

In (d), maximizing the log likelihood or minimizing the negative log likelihood, estimates the parameter values closest to the parameters used to sample the unknown population of data. Correlation is computed to adjudge if any of the parameters could be used to distinguish between groups and thus to model MDD.

In section or task (f), the two-sample t-test is performed to ascertain if the parameter values viz. the learning rate and the inverse temperature are sampled from the same distribution or not.

In task (g), the estimated parameters are used to generate the distribution that they are possibly sampled from. The distribution is again used to sample the parameters and then those are fitted to validate if the model could regenerate the same distribution from the fitted values or in other words, recover the parameters.

In task (h), other models are utilized to perform the same task as model 1 and see if they better fit the provided data.

In task (i), the fit of the model is quantified using AIC and BIC values to adjudge the best model to represent the data and generate the closest estimated parameters that generated the provided data.

In task (j), this is a crucial step to verify that the model regenerates the closest estimated parameters for data generated by itself. It is like a reverse engineering process to validate the robustness of the model.

Task (f) results very modestly indicated that the inverse temperature parameter came from a different distribution for differently perceived participants. The subsequent tasks reveal that both the inverse temperature and the learning rate are equally modeled and regenerated using the same model assuming single distribution for each of them. Since, the correlation for inverse temperature is always on the relatively lower side compared to the learning rate for the implemented models, it could be argued that this substantiates the hints given by the t-test.

In general, the sample is not good enough to represent the underlying distribution and thus the models overfit and fit noise for exact replication. Thus, the reliability performance of these models is not ideal (except model 3) for resampled values created from fitted parameters. The provided data seems to be insufficient for the model to be able to generalize and model the underlying behavior.

Note: The Gradient question is attempted in MATLAB file nll_withgradient.m but is not reported as the results are absurd and probably wrong with fitted values of 0 for each sample set.