Introduction

Method

Code and Explaination

The following code was written to automate data collection and compare accuracies of the three conditions.

clear

clc

%these keep track of how many participants are in each condition. It is also there to keep things

%’pretty’ in the data and avoid blank spots. Add or remove these for more or less conditions

c1s =1; %condition 1

c2s = 1; %condition 2

c3s = 1; %condition 3

for i=1:6 %have this go to the number of participants in the experiment

x = strcat('Participant', num2str(i), '.xls');

t = readtable(x);

a = table2array(t(2,2));

switch a %if you want more or less conditions, add or remove cases, coping the code under

%case 1, and changing the numbers to accommodate the new cases

case 1 %condition 1

condition1.subject(c1s).number = t(1, 2);

condition1.subject(c1s).condition = t(2, 2);

for j=1:10 %have this go to the number of trials in the experiment

condition1.subject(c1s).trialNum(j) = t{j, 3};

condition1.subject(c1s).piNum(j) = t{j, 4};

if char(t{j, 5}) == 'Y' || char(t{j, 5}) == 'y'

condition1.subject(c1s).response(j) = 1;

elseif char(t{j,5}) == 'N'|| char(t{j, 5}) == 'n'

condition1.subject(c1s).response(j) = 0;

end

end

c1s = 1 + c1s; %upticks number of people in the condition

case 2 %condition 2

condition2.subject(c2s).number = t(1, 2);

condition2.subject(c2s).condition = t(2, 2);

for j=1:10 %have this go to the number of trials in the experiment

condition2.subject(c2s).trialNum(j) = t{j, 3};

condition2.subject(c2s).piNum(j) = t{j, 4};

if char(t{j, 5}) == 'Y' || char(t{j, 5}) == 'y'

condition2.subject(c2s).response(j) = 1;

elseif char(t{j,5}) == 'N'|| char(t{j, 5}) == 'n'

condition2.subject(c2s).response(j) = 0;

end

end

c2s = 1 + c2s; %upticks number of people in the condition

case 3 %condition 3

condition3.subject(c3s).number = t(1, 2);

condition3.subject(c3s).condition = t(2, 2);

for j=1:10 %have this go to the number of trials in the experiment

condition3.subject(c3s).trialNum(j) = t{j, 3};

condition3.subject(c3s).piNum(j) = t{j, 4};

if char(t{j, 5}) == 'Y' || char(t{j, 5}) == 'y'

condition3.subject(c3s).response(j) = 1;

elseif char(t{j,5}) == 'N'|| char(t{j, 5}) == 'n'

condition3.subject(c3s).response(j) = 0;

end

end

c3s = 1 + c3s; %upticks number of people in the condition

end

end

%plots conditions

plot(condition1.subject.piNum, condition1.subject.response);

hold on

plot(condition2.subject.piNum, condition2.subject.response);

hold on

plot(condition3.subject.piNum, condition3.subject.response);

hold off

The basics of the code is a switch function that looks at the condition the subject was part of in the excel sheet. The cases are 1, 2, and 3. This is to sort each subject into one of three structs that correspond with the conditions (this is for plotting them later), these structs contain another struct called subject, which contains the variables the correspond to subject number, condition, trial number, pi number, and response. Each case has exactly the same code, except they output to the different condition structs. Inside the case, it takes data from an excel sheet and sets it to the appropriate variables. For trials, piNum, and In order to make the yes or no responses more manageable for MATLAB, there is an if else statement that changes Y to 1 and N to 0.