

第七章 按键反应的记录、保存 和数据分析

Key Presses and Data analysis

单个实验试次Trial的流程

01 呈现刺激

Psychtoolbox (PTB) 中的Screen函数

???

02 记录 (按键) 反应

7.1 与PTB兼容的按键相关命令

- 新建一个m文件KbPractice
 - 功能：持续记录键盘上的按键，并在命令窗口显示对应的按键数字码以及按键时间；按SPACE键结束记录。

```
1 % KbPractice.m
2 % Displays the number of seconds that have elapsed
3 % when the user presses a key.
4 % a simplified version of KbDemo modified by YW, 2022
```

```
>> KbPractice
```

%在命令窗口运行KbPractice

```
Testing KbCheck and KbName: press a key to see its number
Press the 'SPACE' key to exit.
You pressed key 87 which is w typed at time 2.4043seconds
You pressed key 69 which is e typed at time 3.5383seconds
You pressed key 65 which is a typed at time 5.286seconds
You pressed key 82 which is r typed at time 5.7581seconds
You pressed key 70 which is f typed at time 6.828seconds
You pressed key 49 which is ! typed at time 12.7795seconds
You pressed key 50 which is @ typed at time 13.3814seconds
You pressed key 51 which is # typed at time 13.8795seconds
You pressed key 97 which is 1 typed at time 15.4673seconds
You pressed key 98 which is 2 typed at time 15.7394seconds
You pressed key 99 which is 3 typed at time 16.1774seconds
You pressed key 32 which is space typed at time 18.0832seconds
>> |
```

```
6 KbName('UnifyKeyNames');
7 WaitSecs(1.0);
8 disp('Testing KbCheck and KbName: press a key to see its number');
9 disp('Press the ''SPACE'' key to exit.');
```

```
10 escapeKey = KbName('SPACE');
```

• KbName

- 实现键盘按键数字码和所表示的按键名称之间的转换。

• KbName('UnifyKeyNames')

- 可以使程序使用在Windows系统和Mac系统下通用的按键

• WaitSecs

- 等待给定的时间，单位为秒。

• disp

- 在命令窗口显示字符串

%在命令窗口测试KbName

```
>> KbName('a')
ans =
    65
>> KbName(65)
ans =
a
```

```
11 ListenChar(2);  
12 while KbCheck; end % Wait until all keys are released.  
13 % This is a loop that flips around doing anything as long as KbCheck  
14 % reports back that a key is pressed on the keyboard. Means that the  
15 % program doesn't continue until all keys have been released.
```

- ListenChar(2): 监听除了Octave/MATLAB命令窗口以外的按键行为
 - m文件运行过程中记录到的按键不会显示在命令窗口
- ListenChar(0): 恢复MATLAB命令窗口的监听状态，从而可以执行命令窗口输入的命令

```
11 ListenChar(2);
12 while KbCheck; end % Wait until all keys are released.
13 % This is a loop that flips around doing anything as long as KbCheck
14 % reports back that a key is pressed on the keyboard. Means that the
15 % program doesn't continue until all keys have been released.
```

● KbCheck

- 实时检查是否有按键行为，确保在开始实验前键盘的按键处于空闲状态。

试一试： help KbCheck

%这是函数KbCheck, 三个输出变量, 没有输入变量

- [keysDown,secs,keyCode] = KbCheck;
- keysDown: 如果有按键被按到, keysDown=1, KbCheck=1; 如果没有按键, keysDown=0, KbCheck=0;
 - Secs: 按键的电脑系统时刻点
 - keyCode:
 - keyCode在Windows中是256*1的矩阵, 每一个元素代表的是一个按键
 - 如果某个按键(例如, 第65个)被按到, 那keyCode矩阵对应的元素(例如, 第65个元素)就为1, 其他元素都为0
 - 可通过find(keyCode)查看是哪个数字编码的按键
 - 可通过KbName(keyCode)查看按键对应的字符是什么

```
11 ListenChar(2);
12 while KbCheck; end % Wait until all keys are released.
13 % This is a loop that flips around doing anything as long as KbCheck
14 % reports back that a key is pressed on the keyboard. Means that the
15 % program doesn't continue until all keys have been released.
```

- while KbCheck; end

- 只要KbCheck=1（检查到有按键被按住），就一直保持按键检查状态，直到没有任何一个按键被按住，即所有按键都处于空闲状态，为随后“正式”开始检查按键行为做好准备。

```

17 startSecs = GetSecs;           %GetSecs获取当前电脑系统时间
18 while 1
19     % while 1 is always true, so this loop will continue indefinitely.
20     % Line 28 forceably breaks us out of the loop if the escape key is
21     % pressed.
22
23     % Check the state of the keyboard.
24     % See if a key is currently pressed on the keyboard. If not, we skip
25     % the next for loop from lines 20-38, and basically check again almost
26     % immediately.
27
28     [ keyIsDown, seconds, keyCode ] = KbCheck;   %KbCheck检查是否有按键行为
29
30     % If the user is pressing a key,
31     % then display its code number and name. %如果检测到按键行为, 显示按键内容及时间
32     if keyIsDown
33
34         % Note that we use find(keyCode) because keyCode is an array.
35         str=['You pressed key ', num2str(find(keyCode)), ' which is ', KbName(keyCode), ' typed at time ', ...
36             num2str(seconds - startSecs), 'seconds'];
37
38         disp(str);
39         % Display which key has been pressed.
40
41         % If the key that has been pressed is the escape key break out of all loops
42         % including the indefinite while loop.
43         if keyCode(escapeKey)
44             break; %如果检测到的按键行为是结束键, 则break,
45         end       %跳出while 1 end'死循环'
46
47         while KbCheck; end
48         % If the user holds down a key for more than a microsecond,
49         % KbCheck will report multiple events, since computers are faster
50         % than people's fingers
51         % To condense multiple 'keyDown' events into a single event,
52         % once a key has been pressed
53         % we wait until all keys have been released
54         % before going through the loop again
55
56     end
57 end
58 ListenChar(0);

```

7.2 与PTB兼容的按键相关函数1

- 新建一个函数m文件waitTilltime.m
 - 在给定时间内记录并存储按键内容及时间。

%这是函数waitTilltime, 两个输出变量, 两个输入变量 (第2个输入变量不是必须)

```
1 function [keys,RT] = waitTilltime(waitTime,startTime)
2
3 % [keys,RT] = waitTill(waitTime,[startTime])
4 %
5 % Returns a vector of keys pressed and the timing of the presses during an
6 % interval of 'waitTime' seconds. By default, the clock starts within the
7 % function, but if startTime is provided then the function will return
8 % waitTime seconds after startTime was defined with 'GetSecs'.
9 %
10 % An empty variable is returned if no key was pressed during the interval.
11 %
12 % 3/24/09 Written by G.M. Boynton at the University of Washington
```

```
>> [keys,RT] = waitTilltime(5)
keys =
{
    [1,1] = e
    [1,2] = r
    [1,3] = t
}
RT =
    2.2399    3.5617    4.2635
```

```
>> [keys,RT] = waitTilltime(8)
keys =
{
    [1,1] = 1
    [1,2] = 2
    [1,3] = 3
    [1,4] = 4
    [1,5] = 5
    [1,6] = 6
}
RT =
    1.4135    1.5936    1.7635    1.9695    2.2353    2.4215
```

%在命令窗口运行函数waitTilltime

```

14 %Initialize the output variables
15 keys = {};
16 RT = [];
17
18 %Read the clock if no clock time was provided
19 if ~exist('startTime')
20     startTime = GetSecs;
21 end
22
23 %Give a warning if the waiting interval is zero or less
24 if GetSecs-startTime > waitTime
25     disp('Warning! waitTill: waiting interval is less than zero')
26 end
27
28 %Turn off the output to the command window
29 ListenChar(2);
30
31 nKeys = 0;
32 %loop until waitTime seconds has passed since startTime was defined
33 while GetSecs-startTime < waitTime
34     %see if a key is pressed
35     [ keyIsDown, timeSecs, keyCode ] = KbCheck;
36     if keyIsDown %a key is down: record the key and time pressed
37         nKeys = nKeys+1;
38         RT(nKeys) = timeSecs-startTime;
39         keys{nKeys} = KbName(keyCode);
40         %clear the keyboard buffer
41         while KbCheck; end
42     end
43 end
44 %Turn on the output to the command window
45 ListenChar(0);
46

```

Q: 如何设置必须要按键才行?

%如果还处在等待按键反应的时间内

%检查按键行为

%计算总的按键次数

%计算当次按键反应时间 (当前时间-起始时间)

%查看按键内容

7.2 与PTB兼容的按键相关函数2

- 新建一个函数m文件waitTillKey.m
 - 一直等待按键，直到检测到按键为止；没按键就一直等待。

%这是函数waitTillkey, 两个输出变量, 没有输入变量

```
1 function [keys,RT] = waitTillkey
2 % [keys,RT] = waitTill2
3 % Returns a vector of keys pressed and the timing of the presses .
4 % By default, the clock starts within the
5 % function, but if startTime is provided then the function will return
6 % waitTime seconds after startTime was defined with 'GetSecs'.
7 % 3/24/09 Written by G.M. Boynton at the University of Washington, modified
```

```
9 %Initialize the output variables
10 keys = {};
11 RT = [];
12
13 %Turn off the output to the command window
14 ListenChar(2);
15
16 nKeys = 0;
17 startTime = GetSecs;
18 %loop until a key is pressed
19 while 1
20     %see if a key is pressed
21     [ keyIsDown, timeSecs, keyCode ] = KbCheck;
22     if keyIsDown %a key is down: record the key and time pressed
23         nKeys = nKeys+1;
24         RT(nKeys) = timeSecs-startTime;
25         keys{nKeys} = KbName(keyCode);
26         %clear the keyboard buffer
27         while KbCheck; end
28         break
29     end
30 end
31 %Turn on the output to the command window
32 ListenChar(0);
```

A: 设置while 1的'死循环'

小结

- 常见的按键记录方式 **注意：这两个函数的名称可以任意命名**

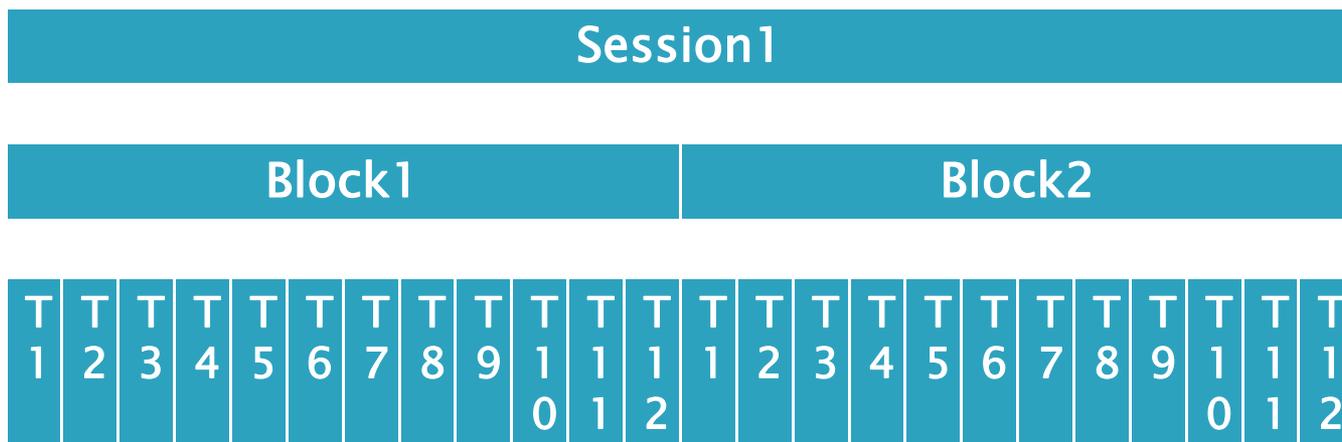
waitTilltime

- 在给定时间内按键
- 不管有没有按键，时间到了执行下一个命令

waitTillkey

- 必须要按键
- 如果按键，就执行下一个命令
- 如果没有按键，就一直停留

当1个试次的刺激呈现和按键反应能分别通过Screen函数和waitTilltime/waitTillkey实现，接下来1个block试次呢，1个session的试次呢？



每个试次的刺激和按键反应信息都需要“各归其位”：预留位置，存储数据，便于后续分析数据！

作答

7.3 把刺激呈现和（按键）反应记录相结合

%这是脚本m文件StroopStimResponse, 也是Stroop实验程序m文件

```
1 % StroopStimResponse.m
2 %
3 % This program presents stroop like stimuli and record keypress response.
4 % Written by YW, 2022
5
6 try
7     clear all
8     close all
9
10    subid='1';
11    subname='zhangsan';
12
13    % predefine parameters
14    BlockNum=1;% how may block in this experiment
15    StimDuration=2000;% 2000ms
16    ISImin=1000;% the minnum interstimulus interval (ISI)
17    ISImax=1500;% the maxmum interstimulus interval (ISI)
18    TxtSize=100;% the size of the texts
19    TxtFont='Arial'; % the font of the texts
20
21    Words={'Red', 'Green', 'Blue'};
22    StimColor={[255 0 0];[0 255 0];[0 0 255]};
23    count=0;
24    for i=1:length(Words)
25        for j=1:length(StimColor)
26            count=count+1;
27            Stim(count).word=Words{i}; % stimulus word
28            Stim(count).color=StimColor{j};% stimulus color
29        end
30    end
31
32    % to preset variables 'results' for recording data
33    results.word = NaN*ones(BlockNum,length(Stim));
34    results.color = NaN*ones(BlockNum,length(Stim));
35    results.response = NaN*ones(BlockNum,length(Stim));
36    results.reactiontime = NaN*ones(BlockNum,length(Stim));
37
```

%设置被试的编号和姓名信息, 每个被试做实验时, 运行一次实验程序

提醒: Octave路径不能包含中文

文件浏览器

K:/StroopOctave2022

名称

- ▶ Data
- ▶ MyStroopPic
- MyStroopStim.m
- StroopStim.m
- StroopStimResponse.m

%预设结构变量results存储数据, 大小为block行, trial列, 从而把所有block的所有trial数据都能存储下来; NaN是not a number, 因为数据中可能既有数字又有字符

```
51 % blank the Screen
52 starttime=Screen('Flip',wPtr); % collect the time for the first flip with starttime
53 for ii=1:BlockNum %从第1个block到最后一个block
54
55 Screen('TextSize', wPtr , TxtSize);
56 Screen( 'TextFont', wPtr ,TxtFont);
57 StimRand=Stim(randperm(length(Stim))); % randomize all stimuli
58
59 for jj=1:length(Stim) %从第1个trial到最后一个trial
60 TxtRect= Screen('TextBounds',wPtr,StimRand(jj).word); % get the size of each stimulus text
61 TxtLoc=[round(rect(3)/2-TxtRect(3)/2),round(rect(4)/2-TxtRect(4)/2)]; % compute the upperleft and
62 Screen('DrawText', wPtr, StimRand(jj).word, TxtLoc(1), TxtLoc(2),StimRand(jj).color);
63 starttime=Screen(wPtr, 'Flip', starttime+(ISI(ii,jj)*monitorFlipInterval)); % time in secs
64 [keys,RT] = waitTilltime(StimDuration*monitorFlipInterval, starttime);%time in secs
65 starttime=Screen('Flip', wPtr);% time in secs
66
67 %%% save keypress response
68 if strcmp(StimRand(jj).word,Words{1})==1
69 results.word(ii,jj)=1; % word1 'Red'
70 elseif strcmp(StimRand(jj).word,Words{2})==1
71 results.word(ii,jj)=2;% word2 'Green'
72 elseif strcmp(StimRand(jj).word,Words{3})==1
73 results.word(ii,jj)=3;% word3 'Blue'
74 end
75 results.color(ii,jj)=find(StimRand(jj).color~=0) % color 1=red,2=green, 3=blue
76
77 if ~isempty(RT)
78 results.reactiontime(ii,jj)=RT;
79 results.key{ii,jj}=keys{1};
80
81 if (results.key{ii,jj}=='f' && results.color(ii,jj)==1)...
82 ||(results.key{ii,jj}=='j' && results.color(ii,jj)==2)...
83 ||(results.key{ii,jj}=='k' && results.color(ii,jj)==3)
84 results.response(ii,jj) = 1;% f for color red, j for color green, k for color blue are correct
85 else
86 results.response(ii,jj) = 0;% incorrect
87 end
88 end
89 end
90 end
```

%waitTilltime在刺激呈现时间2s内等待按键

%将刺激的特征以数字形式存储在结果变量results.word和results.color两个字段中

提醒：存储数据的方法不只这一种方式

```

51 % blank the Screen
52 starttime=Screen('Flip',wPtr); % collect the time for the first flip with starttime
53 for ii=1:BlockNum
54
55     Screen('TextSize', wPtr , TxtSize);
56     Screen('TextFont', wPtr ,TxtFont);
57     StimRand=Stim(randperm(length(Stim))); % randomize all stimuli
58
59     for jj=1:length(Stim)
60         TxtRect= Screen('TextBounds',wPtr,StimRand(jj).word); % get the size of each stimulus text
61         TxtLoc=[round(rect(3)/2-TxtRect(3)/2),round(rect(4)/2-TxtRect(4)/2)]; % compute the upperleft and
62         Screen('DrawText', wPtr, StimRand(jj).word, TxtLoc(1), TxtLoc(2),StimRand(jj).color);
63         starttime=Screen(wPtr, 'Flip', starttime+(ISI(ii,jj)*monitorFlipInterval)); % time in secs
64         [keys,RT] = waitTilltime(StimDuration*monitorFlipInterval, starttime);%time in secs
65         starttime=Screen('Flip', wPtr);% time in secs
66
67         %%% save keypress response
68         if strcmp(StimRand(jj).word,Words{1})==1
69             results.word(ii,jj)=1; % word1 'Red'
70         elseif strcmp(StimRand(jj).word,Words{2})==1
71             results.word(ii,jj)=2;% word2 'Green'
72         elseif strcmp(StimRand(jj).word,Words{3})==1
73             results.word(ii,jj)=3;% word3 'Blue'
74         end
75         results.color(ii,jj)=find(StimRand(jj).color~=0); % color 1=red,2=green, 3=blue
76
77         if ~isempty(RT)
78             results.reactiontime(ii,jj)=RT;
79             results.key{ii,jj}=keys{1};
80
81             if (results.key{ii,jj}=='f' && results.color(ii,jj)==1)...
82                 ||(results.key{ii,jj}=='j' && results.color(ii,jj)==2)...
83                 ||(results.key{ii,jj}=='k' && results.color(ii,jj)==3)
84                 results.response(ii,jj) = 1;% f for color red, j for color green, k for color blue are correct
85             else
86                 results.response(ii,jj) = 0;% incorrect
87             end
88         end
89     end
90 end

```

%将反应时间、按键内容以及按键正确与否存储在 results.reactiontime, key, response三个字段中

7.4 导出并永久保存实验数据文件

- save将指定变量导出到指定文件夹中的mat文件
 - mat文件是MATLAB可识别的文件格式
- 导出为其他数据格式文件
 - 导出txt文件的命令: fprintf (help fprintf查看用法)

%设置好文件名称字符串, 包括被试编号和姓名信息, 赋值给变量datafile

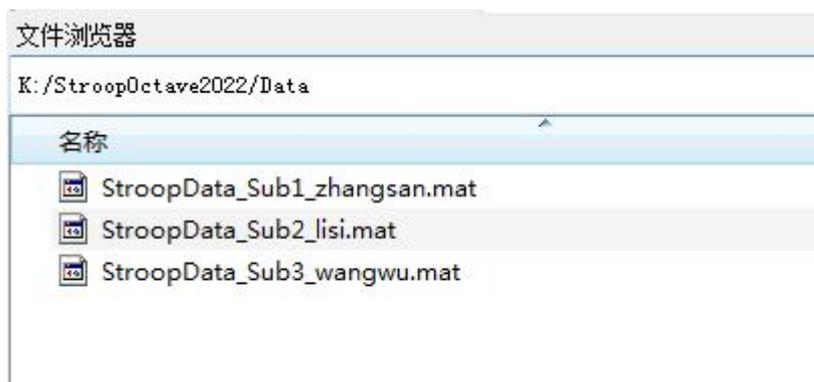
```
96 % save the datafile to disc for each participant
97 datafile=['StroopData_Sub' subid '_' subname];
98 path=pwd;
99 save ([path '\Data\' datafile '.mat'], 'results'); % save as mat file
```

*%保存的文件格式
为mat文件*

*%要保存的变量是
results*

怎样运行StroopStimResponse这个m文件呢?

命令窗口>>StroopStimResponse



提醒：每个被试保存一个数据文件；所有数据统一放置在\Data子文件夹下

作答

7.5 分析实验数据

```
1  % StroopDataAnalysis.m
2  %
3  % To compute the grand average result of all participants or subjects
4  % written in 2022.
5
6  clear all
7  close all
8
9  % load each subject's mat data file
10 path=pwd;
11 filemat=dir([path '\Data\*.mat']); %dir获取当前路径下有多少个被试的mat数据文件
12 for i=1:length(filemat)
13     allresults(i) = load([path '\Data\' filemat(i).name]); % load all subjects'
14 end %load逐个读取被试的数据mat文件，并存储在
15 %allresults的不同位置
16 %predefine the final data variable
17 congacc=zeros(1,length(allresults));
18 incongacc=zeros(1,length(allresults)); %预设每个被试不同类型试次（字色字义一致 vs.
19 congprt=zeros(1,length(allresults)); %字色字义不一致）的平均数据（准确率&反应时）
20 incongprt=zeros(1,length(allresults)); %存储在相应变量中
```

%通过循环命令依次计算每个被试不同类型试次的平均数据

%筛选出‘字色字义一致’和‘字色字义不一致’两种类型的试次分别计算

```
22 for i=1:length(allresults)
23     % each subject's acc of congruent and incongruent conditions
24     congresp=allresults(i).results.response(allresults(i).results.word==allresults(i).results.color); % congruent trials
25     congacc(i)=mean(congresp);
26
27     incongresp=allresults(i).results.response(allresults(i).results.word~=allresults(i).results.color); % incongruent trials
28     incongacc(i)=mean(incongresp);
29
30     % each subject's rt of congruent and incongruent conditions
31     congrreact=allresults(i).results.reactiontime(allresults(i).results.word==allresults(i).results.color);
32     congrt(i)=mean(congrreact(congresp==1)); %只计算准确试次的反应时（常规操作）
33
34     incongreact=allresults(i).results.reactiontime(allresults(i).results.word~=allresults(i).results.color);
35     incongrt(i)=mean(incongreact(incongresp==1));
36
37 end
38
39 %最后的最后，计算所有被试不同类型试次的总平均数据
40 congaccmean=mean(congacc(~isnan(congacc))); congaccsd=std(congacc(~isnan(congacc)));
41 incongaccmean=mean(incongacc(~isnan(congacc))); incongaccsd=std(incongacc(~isnan(congacc)));
42 % mean rt of all subjects
43 congrtmean=mean(congrt(~isnan(congacc))); congrtsd=std(congrt(~isnan(congacc)));
44 incongrtmean=mean(incongtrt(~isnan(congacc))); incongrtsd=std(incongtrt(~isnan(congacc)));
```

Q: 如何使用之前章节学习的数据绘图命令，呈现所有被试在两种实验条件下（字色字义一致 vs. 字色字义不一致）的总平均数据（准确率&反应时）？

怎样运行StroopDataAnalysis这个m文件呢?

命令窗口>>StroopDataAnalysis

作答

开始PRACTICE啦！