Psychology 205: Research Methods in Psychology Using R to examine our data

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

April, 2021

Outline

Using R to get and describe the data

Some basic data manipulation

Inferential Statistics

References

Using R to analyze our data – Some preliminaries

1. Make sure that you have installed R on your computer.
2. Make sure that you then installed the *psych*.
3. See the short and longer tutorials on how to do this.
4. Once they are installed (you only need to do this once) then you need to use the `library(psych)` command at the beginning of every session.

Using R to examine the experimental data

The data from our experiment were taken from the Qualtrics file and saved as a .csv file from Excel. The web address is http://personality-project.org/courses/205/205.recognition.21.csv. You can view the in your browser or read it directly into R. If you read it in your browser, you can select it all, copy the data to your clipboard, and read the clipboard (option 1)

R code

```
library(psych)  #start each session with this
#go to the website to read the data for today
#Option 1  Copy the file to your clipboard and then read the clipboard
my.data <- read.clipboard.csv()
#option 2  Read the remote file directly
filename <-
   "http://personality-project.org/courses/205/205.recognition.21.csv"
my.data <- read.file(filename)  #read the data
dim(my.data)  #find the dimensions of the data (subjects x variables)
```

```
filename <-
+    "http://personality-project.org/courses/205/205.recognition.21.csv"
> my.data <- read.file(filename)  #read the data
Data from the .csv file http://personality-project.org/courses/205/205.recognition.21.csv has been loaded
> dim(my.data)  #find the dimensions of the data read using either option 1 or 2
[1] 15 81
```

The data file on the server using a webbrowser

```
group,plain,fruit,beautiful,cherry,strong,banana,horn,high,white,water,flow,throne,doctor,sofa,dough
1,1,3,1,4,1,2,1,1,4,1,1,1,3,4,4,4,4,4,1,1,4,1,3,4,2,4,4,3,1,1,4,1,3,4,2,1,1,4,1,1,4,1,4,1,2,1,1,2,4,
2,4,4,2,2,2,4,2,2,4,2,2,2,3,4,2,4,2,2,2,2,2,2,4,2,3,4,4,4,2,2,3,4,4,4,2,2,2,3,2,2,4,2,4,2,2,2,2,3,3,
1,1,1,1,3,1,4,1,1,4,1,1,1,1,4,4,4,4,2,2,1,2,1,3,4,2,4,2,2,1,1,4,1,4,1,1,1,1,4,1,1,4,1,4,2,1,1,1,4,4,
2,1,1,1,4,1,4,1,1,4,2,1,1,1,3,4,4,2,4,1,1,4,1,4,4,2,1,1,1,1,1,4,1,2,4,1,1,1,4,1,1,4,1,4,1,1,1,1,2,4,
2,1,3,1,3,1,3,1,1,4,2,1,1,2,2,3,3,4,4,1,1,3,1,4,4,4,4,4,4,2,1,3,1,3,4,2,1,1,4,2,1,4,1,4,1,1,1,1,3,4,
1,1,4,1,4,1,4,1,2,4,1,2,2,1,4,4,4,2,2,1,1,3,1,3,4,1,4,1,1,1,1,3,1,4,3,1,1,1,4,1,1,4,1,1,1,1,1,1,1,4,
2,2,3,1,4,1,2,1,2,4,2,2,1,2,4,4,2,4,4,2,1,4,1,4,2,3,2,2,3,1,1,4,1,2,4,2,1,2,2,1,2,4,1,2,1,1,1,1,2,4,
2,3,2,2,2,3,3,2,2,4,3,3,3,2,4,2,4,4,3,2,3,2,2,4,4,4,4,4,3,2,2,3,2,2,2,2,3,2,4,2,1,4,4,4,3,3,3,3,2,2,
2,1,1,1,4,2,2,2,2,4,2,2,2,1,3,4,4,4,2,1,1,2,2,2,3,4,1,2,2,1,2,4,1,2,2,1,1,2,2,2,1,2,1,1,1,1,1,1,2,3,
1,1,1,1,3,2,4,2,2,4,2,2,2,2,4,2,4,4,4,2,2,2,2,3,4,2,2,2,3,2,1,3,2,3,4,2,1,1,4,1,1,3,1,4,1,1,1,1,2,4,
1,2,4,3,4,3,2,2,1,4,3,3,2,4,2,4,3,4,3,2,1,3,2,2,4,2,4,3,3,2,1,3,2,4,2,1,1,3,3,3,2,3,3,3,2,1,3,1,3,4,
1,2,3,3,4,2,4,1,1,3,2,2,2,3,4,4,4,4,4,2,1,2,2,4,4,2,4,3,4,2,1,4,2,4,4,2,1,1,4,2,2,4,3,4,2,2,2,2,4,3,
2,2,3,1,3,1,4,3,1,4,1,1,1,4,4,4,2,4,3,1,1,1,1,4,4,4,4,4,4,2,1,2,1,4,4,2,2,2,3,2,2,4,2,4,2,2,2,2,4,4,
2,1,3,1,4,1,4,1,1,3,2,2,1,3,1,1,1,3,1,1,1,1,1,1,4,1,4,2,3,2,2,3,2,4,1,1,2,2,4,2,2,3,2,4,2,2,1,1,2,2,
2,2,4,1,4,2,4,1,2,4,2,2,1,3,4,2,4,4,3,1,2,4,2,3,4,4,4,4,2,2,1,4,2,4,4,2,2,2,4,2,1,4,3,4,2,3,1,2,4,4,
```

A very bad way to show the data – show everthing

R code

```
my.data
```

```
   group plain fruit beautiful cherry strong banana horn high white water flow throne doctor sofa dough cu
1      1     1     3         1      4      1      2    1    1     4     1    1      1      3    4     4
2      2     4     4         2      2      2      4    2    2     4     2    2      2      3    4     2
3      1     1     1         1      3      1      4    1    1     4     1    1      1      1    4     4
4      2     1     1         1      4      1      4    1    1     4     2    1      1      1    3     4
5      2     1     3         1      3      1      3    1    1     4     2    1      1      2    2     3
6      1     1     4         1      4      1      4    1    2     4     1    2      2      1    4     4
7      2     2     3         1      4      1      2    1    2     4     2    2      1      2    4     4
8      2     3     2         2      2      3      3    2    2     4     3    3      3      2    4     2
9      2     1     1         1      4      2      2    2    2     4     2    2      2      1    3     4
10     1     1     1         1      3      2      4    2    2     4     2    2      2      2    4     2
11     1     2     4         3      4      3      2    2    1     4     3    3      2      4    2     4
12     1     2     3         3      4      2      4    1    1     3     2    2      2      3    4     4
13     2     2     3         1      3      1      4    3    1     4     1    1      1      4    4     4
14     2     1     3         1      4      1      4    1    1     3     2    2      1      3    1     1
15     2     2     4         1      4      2      4    1    2     4     2    2      1      3    4     2
   death foot chair cold low thimble chilly man anger physician needle beard thread nurse note barge hot
1      2    4     4    3   1       1      4   1     3         4      2     1      1     4    1     1   4
2      3    4     4    4   2       2      3   4     4         4      2     2      2     3    2     2   4
3      2    4     2    2   1       1      4   1     4         1      1     1      1     4    1     1   4
4      2    1     1    1   1       1      4   1     2         4      1     1      1     4    1     1   4
5      4    4     4    4   2       1      3   1     3         4      2     1      1     4    2     1   4
6      1    4     1    1   1       1      3   1     4         3      1     1      1     4    1     1   4
7      3    2     2    3   1       1      4   1     2         4      2     1      2     2    1     2   4
8      4    4     4    3   2       2      3   2     2         2      2     3      2     4    2     1   4
9      4    1     2    2   1       2      4   1     2         2      1     1      2     2    2     1   2
10     2    2     2    3   2       1      3   2     3         4      2     1      1     4    1     1   3
11     2    4     3    3   2       1      3   2     4         2      1     1      3     3    3     2   3
12     2    4     3    4   2       1      4   2     4         4      2     1      1     4    2     2   4
13     4    4     4    4   2       1      2   1     4         4      2     2      2     3    2     2   4
14     1    4     2    3   2       2      3   2     4         1      1     2      2     4    2     2   3
```

A bad way to treat the data – show everybody - but just the last 16 items

R code

```
my.data[66:81]
```

```
my.data[66:81]
   anger.1 black.1 bread.1 chair.1 cold.1 doctor.1 foot.1 fruit.1 girl.1 high.1 king.1 man.1 mountain.1 m
1        0       0       0       1      0        0      1       0      0      0      0     0          0
2        1       1       1       1      1        1      1       1      0      0      0     1          0
3        1       1       1       1      1        1      1       0      0      0      0     0          0
4        0       0       0       0      0        0      0       0      0      0      0     0          0
5        1       1       1       1      1        0      1       1      0      0      0     0          0
6        0       0       0       0      0        0      1       1      0      0      0     0          0
7        1       0       0       0      0        0      1       0      0      0      0     0          0
8        1       0       1       1      0        0      1       1      0      0      0     0          0
9        0       0       0       0      0        0      0       0      0      0      0     0          0
10       0       0       0       0      0        0      0       0      0      0      0     0          0
11       1       1       1       0      1        1      0       1      0      1      1     1          0
12       1       0       1       1      1        0      0       1      0      0      0     0          0
13       1       1       1       1      1        1      1       1      0      0      0     0          0
14       0       0       0       0      0        0      0       0      0      0      0     0          0
15       1       0       1       1      0        0      1       1      0      0      0     0          1
>
```

Examine the data using `headTail` and `describe`

R code

```
headTail(my.data,from=1, to =10) #just the first few columns
headTail(my.data, from=66)  #show the last 16 columns
```

```
 headTail(my.data,from=1, to =10) #just the first few columns
    group plain fruit beautiful cherry strong banana horn high white
1       1     1     3         1      4      1      2    1    1     4
2       2     4     4         2      2      2      4    2    2     4
3       1     1     1         1      3      1      4    1    1     4
4       2     1     1         1      4      1      4    1    1     4
...   ...   ...   ...       ...    ...    ...    ...  ...  ...   ...
12      1     2     3         3      4      2      4    1    1     3
13      2     2     3         1      3      1      4    3    1     4
14      2     1     3         1      4      1      4    1    1     3
15      2     2     4         1      4      2      4    1    2     4
> headTail(my.data, from=66)  #show the last 16 columns
    anger.1 black.1 bread.1 chair.1 cold.1 doctor.1 foot.1 fruit.1 girl.1 high.1 king.1 man.1 mountain.1
1         0       0       0       1      0        0      1       0      0      0      0     0          0
2         1       1       1       1      1        1      1       1      0      0      0     1          0
3         1       1       1       1      1        1      1       0      0      0      0     0          0
4         0       0       0       0      0        0      0       0      0      0      0     0          0
...     ...     ...     ...     ...    ...      ...    ...     ...    ...    ...    ...   ...        ...
12        1       0       1       1      1        0      0       1      0      0      0     0          0
13        1       1       1       1      1        1      1       1      0      0      0     0          0
14        0       0       0       0      0        0      0       0      0      0      0     0          0
15        1       0       1       1      0        0      1       1      0      0      0     0          1
>
```

Describe the data – Just the last 16 columns

These are the 0/1 of did you see this word. None of these words appeared, but the first 8 were primed by related words.

R code

```
describe(my.data[66:81] )  #we specify the column numbers that we want
```

```
describe(my.data[66:81] )  #we specify the column numbers that we want
           vars  n mean   sd median trimmed mad min max range  skew kurtosis   se
anger.1       1 15 0.60 0.51      1    0.62   0   0   1     1 -0.37    -1.98 0.13
black.1       2 15 0.33 0.49      0    0.31   0   0   1     1  0.64    -1.69 0.13
bread.1       3 15 0.53 0.52      1    0.54   0   0   1     1 -0.12    -2.11 0.13
chair.1       4 15 0.53 0.52      1    0.54   0   0   1     1 -0.12    -2.11 0.13
cold.1        5 15 0.40 0.51      0    0.38   0   0   1     1  0.37    -1.98 0.13
doctor.1      6 15 0.27 0.46      0    0.23   0   0   1     1  0.95    -1.16 0.12
foot.1        7 15 0.60 0.51      1    0.62   0   0   1     1 -0.37    -1.98 0.13
fruit.1       8 15 0.53 0.52      1    0.54   0   0   1     1 -0.12    -2.11 0.13
girl.1        9 15 0.00 0.00      0    0.00   0   0   0     0   NaN      NaN 0.00
high.1       10 15 0.07 0.26      0    0.00   0   0   1     1  3.13     8.39 0.07
king.1       11 15 0.07 0.26      0    0.00   0   0   1     1  3.13     8.39 0.07
man.1        12 15 0.13 0.35      0    0.08   0   0   1     1  1.95     1.93 0.09
mountain.1   13 15 0.07 0.26      0    0.00   0   0   1     1  3.13     8.39 0.07
music.1      14 15 0.07 0.26      0    0.00   0   0   1     1  3.13     8.39 0.07
needle.1     15 15 0.07 0.26      0    0.00   0   0   1     1  3.13     8.39 0.07
river.1      16 15 0.13 0.35      0    0.08   0   0   1     1  1.95     1.93 0.09
```

Note that R renamed these variables because they had already appeared as column names.

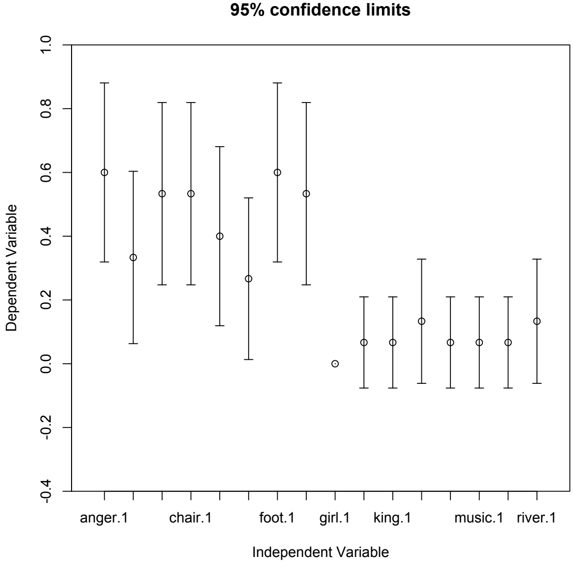
Can also show this graphically

Here we explore 4 different ways of drawing the results.

R code

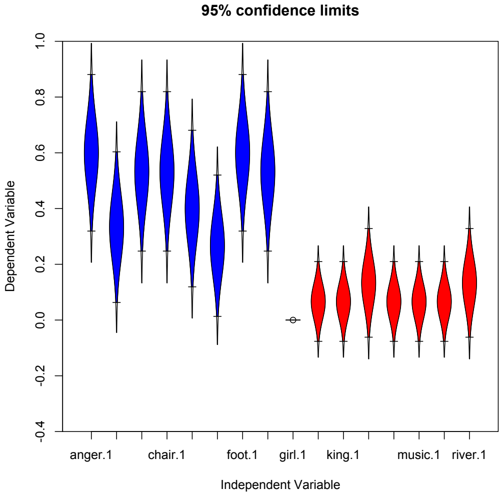
```
error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)),
                               eyes=FALSE) #the basic plot
error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)))
  #with 'cats eyes"  and somewhat improved by specifying x and y axes

error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)),
   ylab=" Recognition",xlab="Prompt", las=2,
     main="False Recognition means and 95% confidence") #means with confidence
 error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)),
   ylab=" Recognition",xlab="Prompt", las=2,
   main="False Recognition means and 95% confidence", eyes=FALSE)

error.dots(my.data[66:81]) #another way to show the results
```

The False Recognition results and their confidence



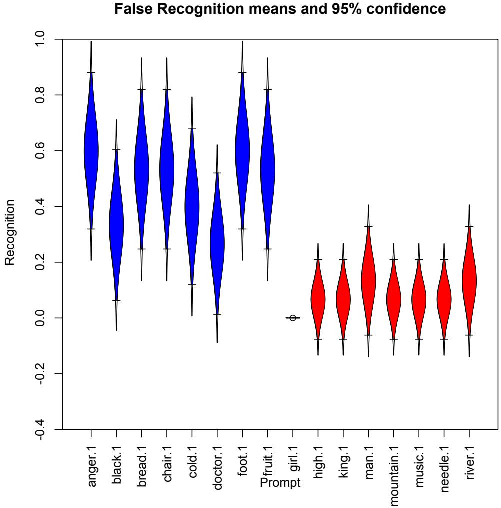
```
error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)),  eyes=FALSE) #the basic plot
```

The False Recognition results and their confidence with "cats eyes" to show confidence



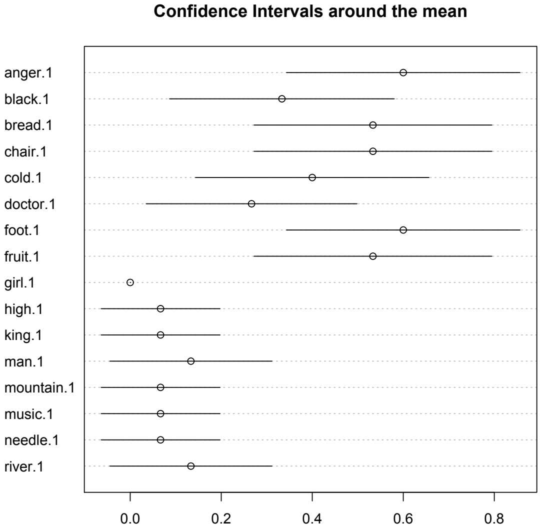
```
error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)))  #the basic plot with colored cats eyes
```

The False Recognition results and their confidence with “cats eyes” to show confidence – improved



```
error.bars(my.data[66:81],col=c(rep("blue",8),rep("red",8)),ylab=" Recognition",xlab="Prompt",
 las=2, main="False Recognition means and 95% confidence")
```

The False Recognition results and their confidence using error.dots



```
error.dots(my.data[66:81],sort=FALSE) #another way to show the results
```

Create some summary data, adjust for number of trials

We can do basic arithmetic on our data to combine the columns in a useful manner.

R code

```
Cued <- rowSums(my.data[66:73])/8 #how many false recognitions that were cued
Control <- rowSums(my.data[74:81]) /8 #How many false recognitions for non-cue
CuedA <- rowSums(my.data[c(66,69,71,72)])/4 #the eyes open condition for group
CuedB <- rowSums(my.data[c(67,68,70,72)])/4 #the eyes open condition for group
Diff = Control - Cued
#Save these as a  data.frame
data.df <- data.frame(group = my.data[,"group"],Cued=Cued,Control=Control,
       CuedA=CuedA,CuedB=CuedB, Diff=Diff)
data.df  #show the data
error.bars.by(CuedA + CuedB + Control ~ group, data=data.df) #draw iit
```

Show the data.frame

R code

```
data.df
```

```
 data.df
   group  Cued Control CuedA CuedB   Diff
1      1 0.250   0.000  0.50  0.25 -0.250
2      2 1.000   0.125  1.00  1.00 -0.875
3      1 0.875   0.125  1.00  1.00 -0.750
4      2 0.000   0.000  0.00  0.00  0.000
5      2 0.875   0.000  0.75  1.00 -0.875
6      1 0.250   0.000  0.25  0.25 -0.250
7      2 0.250   0.000  0.50  0.25 -0.250
8      2 0.625   0.000  0.75  0.50 -0.625
9      2 0.000   0.000  0.00  0.00  0.000
10     1 0.000   0.000  0.00  0.00  0.000
11     1 0.750   0.625  0.50  0.75 -0.125
12     1 0.625   0.000  0.50  0.50 -0.625
13     2 1.000   0.000  1.00  1.00 -1.000
14     2 0.000   0.000  0.00  0.00  0.000
15     2 0.625   0.250  0.75  0.50 -0.375
```

Descriptive statistics of the results

R code

```
describe(data.df)
```

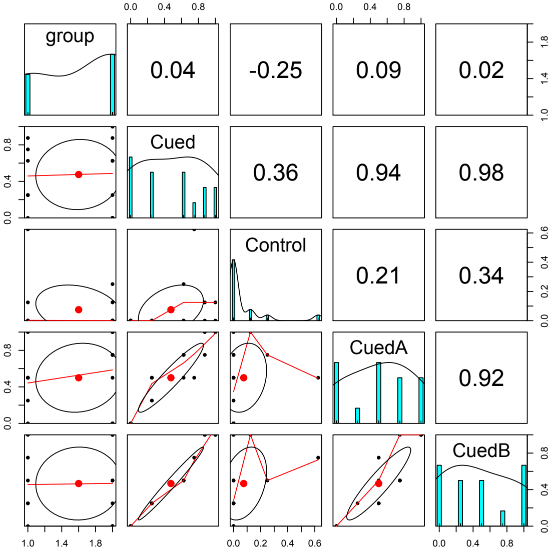
```
        vars  n mean   sd median trimmed  mad min  max range  skew kurtosis   se
group      1 15 1.60 0.51   2.00    1.62 0.00   1 2.00  1.00 -0.37    -1.98 0.13
Cued       2 15 0.48 0.38   0.62    0.47 0.56   0 1.00  1.00 -0.02    -1.71 0.10
Control    3 15 0.07 0.17   0.00    0.04 0.00   0 0.62  0.62  2.31     4.58 0.04
CuedA      4 15 0.50 0.38   0.50    0.50 0.37   0 1.00  1.00 -0.12    -1.52 0.10
CuedB      5 15 0.47 0.40   0.50    0.46 0.74   0 1.00  1.00  0.20    -1.62 0.10
```

But it is also useful to show it graphically. We use the `pairs.panels` function to do this.

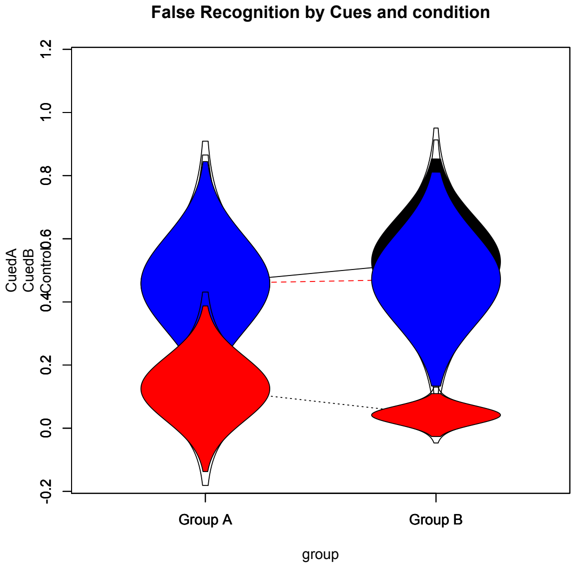
R code

```
pairs.panels(data.df)
```

A graphical display of our data using `pairs.panels`



Recognition by condition for cued and uncued words



How likely are these differences between Cued and Control due to chance? The use of Student's t test

1. We can test how likely these differences between Cued words and uncued (control words) are by comparing their means to the standard deviations.
2. Mean Cued = .48 with standard deviation of .38
3. Mean Control = .07 with standard deviation of .17
4. Student's "t" = $\frac{\bar{x_1} - \bar{x_2}}{\sqrt{\sigma_1^2/N_1 + \sigma_2^2/N_2}} = 3.69$
5. Pooled standard deviation $= \sqrt{\frac{N_1\sigma_1^2 + N_2 * \sigma_2^2}{N_1 + N_2}}$
6. Effect size = d = $\frac{\bar{x_1} - \bar{x_2}}{sd_{pooled}}$ is not sensitive to sample size
7. But t is sensitive to sample size: $t = d\sqrt{N_1 + N_2}$
8. The recommendation is to always report Effect size (e.g., Cohen's d) as well as the t test value Cohen (1988)

Using R to find the t value: two ways

R code

```
t.test(Cued, Control, data= data.df,var.equal=TRUE)
```

```
Two Sample t-test

data:  Cued and Control
t = 3.6892, df = 28, p-value = 0.0009605
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.1779017 0.6220983
sample estimates:
mean of x mean of y
    0.475     0.075
```

R code

```
t2d(3.689,28)   #convert t to d
```

```
1.39
```

R code

```
t.test(Cued, Control, data= data.df,paired=TRUE)
```

```
        Paired t-test

data:  Cued and Control
t = 4.2982, df = 14, p-value = 0.0007362
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.2003998 0.5996002
sample estimates:
mean of the differences
                    0.4
t2d(4.297,14) = 2.29 (Cohen's d)
```

Interpreting the t-test in English

1. Formally: The probability of their being no difference between the Cued and the Control condition was very small (p = .0009 or p = .0007)
2. Or, more normally: Recognition for cued words (mean = .475, sd = .38) was much greater than the recognition for uncued words (mean =.07, sd = .17) with paired $t_{14} = 4.30$, $p < 0.0007362$ with an effect size (Cohen, 1988) of 2.29.
3. Words that were not presented but were high frequency associates of word lists that were presented were much likely to be falsely reocognized (mean = .475, sd = .38) than were control words that were high frequency associates of word lists that were not presented (mean =.07, sd = .17), paired $t_{14} = 4.30$, $p < 0.0007362$ with an effect size (Cohen, 1988) of 2.29.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed ed.). Hillsdale, N.J.: L. Erlbaum Associates.