Psychology 205: Research Methods
Experiment 1: A study of false memory

William Revelle

Department of Psychology
Northwestern University
Evanston, Illinois USA

October, 2015

Outline

Overview and update of these slides

1. Several slides have been added to this presentation.
2. They are in the results of the recall (31-34) and of the recognition data (42-48)
3. For writing your paper, you should include some of these new figures as well as the new results.
4. Consider the conclusions on slide 49.
5. These new slides are meant to be as clear as possible.
6. Make sure that you look at all slides.

The fundamental challenge: accounting for variability

1. Data = Model + Residual
   - Total variability is made up of understood (modeled) and not understood (residual) variability
   - $\sigma^2_{total} = \sigma^2_{model} + \sigma^2_{residual}$
2. Good models explain more of the total variation
   - $Fit = \frac{\sigma^2_{model}}{\sigma^2_{total}}$
3. The challenge of research is to develop better models
4. The process of research is to reduce the residual
5. We do this by a progression of models, ranging from the very simple to the complex
6. We want to know how each model fits the data

The basic designs

1. Correlational/observational studies of the relationships between variables
   - Data can be any systematic set of observations
     - typically includes subject variables
     - survey research, clinical assessment, personality measurement
   - To what extent do one set of measures covary/correlate with another set of measures?
   - Can be used in predictive context – How much is a change in X associated with a change in Y?
   - Does not allow for causal inference
2. Experimental: The study of the effect of manipulated variables
   - Participants assigned to conditions to examine the causal effect of conditions
   - Between subject designs to control for order or learning effects.
   - Within subject designs control for between subject variability.
3. Quasi-experimental has appearance of experimental but does not include random assignment.

Types of Experimental Designs

1. Within Subjects
    - Removes the variability due to between person differences.
    - Has the problem of learning or fatigue related order effects.
    - Can control for order effects using counter balancing.
2. Between Subjects
    - Is less powerful because of between subject variability.
    - Does not have problem of learning or fatigue related order effects.
    - Can control for subject effects by randomization.
3. Mixed designs combine within subjects with between subjects.

Roediger and McDermott study

1. Meta-theoretical question
    - memory as photograph versus memory as reconstruction
    - ‘recovered’ childhood memories of trauma versus ‘false’ memories
    - legal testimony of accuracy of memory
2. More a demonstration of an effect than a test of competing theories
    - Alternative explanations for memory effects
        - connection strength models of memory
        - network models of association
    - Theoretical statement
        - not testing theory but rather testing phenomenon
        - need to get a robust measure of false memory in order to study it

Is memory like a photograph, or is memory like a story?

1. Bartlett and the idea of reconstructive memory
   - Recall of an experience is not just recalling facts, but is an attempt to reconstruct the events
2. Loftus and event reconstruction
   - Events are reconstructed
   - Questions can prime (perhaps incorrectly) a coherent story
3. Prior work by Deese showed that intrusion errors could be induced by using lists of high associates to a (non-presented) target word.
4. Prior work by Underwood showed that recognition errors have low probability
5. Roediger and McDermott paradigm rediscovered the Deese paradigm (for recall) and used recognition (ala Underwood).

Roediger and McDermott Study 1

1. Materials
   - 6 lists of 12 words with high associates of 6 target lures
   - recognition list
   - 12 studied words
   - 6 target lures
   - 12 weakly related
   - 12 unrelated
2. Procedure
   - verbal presentation of each list
   - free recall after each list
   - recognition 2 minutes after all lists had been presented
3. Results
   - recall shows serial position effects
   - intrusion errors almost as strong as low point of serial position
   - recognition errors are frequent

Roediger and McDermott Study 2

- Materials
  - 16 lists of words
  - each list has 15 words, all high associates of an unpresented target word
  - words are in order of associative strength
- Procedure
  - To examine the effect of prior recall, half the trials involve recall, half do not
- Results
  - Serial position effects show subjects follow instructions
  - Moderate level of false recognition

Our study

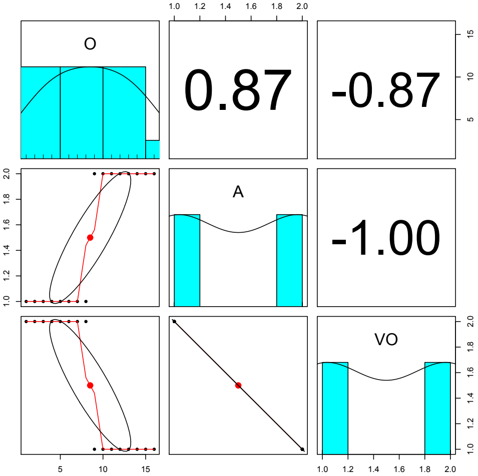
1. Replication and extension of Roediger and McDermott Based upon prior work in 205, observed lower rates of subsequent false recognition than R & M. Was this due to modality of presentation?
2. Within subject study (why?)
   - Modality of presentation (visual vs. oral)
   - Recall vs. no recall (math vs. recall)
3. Recall of presented words
   - Half the trials subjects recalled words
4. Recognition of presented and non-presented words
   - 1st, 8th and 10th words from list were on recognition list
   - Non-presented "lure" or "target" words
   - 32 non-presented, non cued words were also included to check for global willingness to respond

Issues in design

1. Within subject designs control for differences in motivation and ability by using each person as their own control
   - Each subject is a complete experiment
   - But conditions need to be independent of each other and of order effects
2. Two solutions
   - Complete randomization (used with many, many trials)
   - Counterbalancing of conditions against each other and against order
3. Consider a number of possible research orders

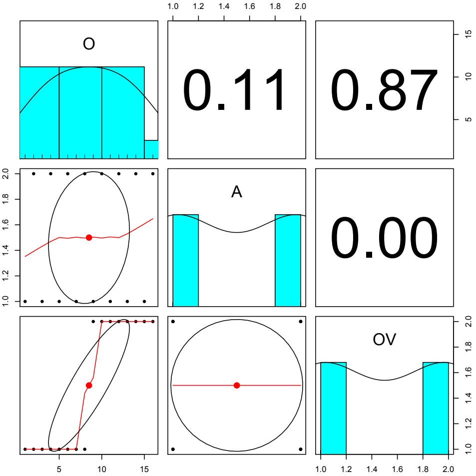
Complete confounding of variables and order

```
bad
> pairs.panels(bad)

 bad
     O  A  VO
1    1  A   V
2    2  A   V
3    3  A   V
4    4  A   V
5    5  A   V
6    6  A   V
7    7  A   V
8    8  A   V
9    9  B   O
10  10  B   O
11  11  B   O
12  12  B   O
13  13  B   O
14  14  B   O
15  15  B   O
16  16  B   O
```



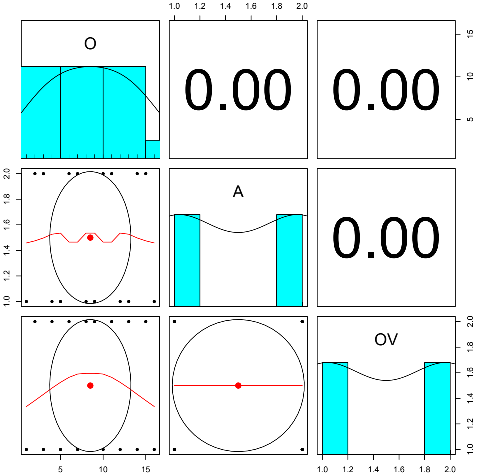
Variables are independent, but are confounded with order

```
> better
> pairs.panels(better)

> better
    O A OV
1   1 A  O
2   2 B  O
3   3 A  O
4   4 B  O
5   5 A  O
6   6 B  O
7   7 A  O
8   8 B  O
9   9 A  V
10 10 B  V
11 11 A  V
12 12 B  V
13 13 A  V
14 14 B  V
15 15 A  V
16 16 B  V
```



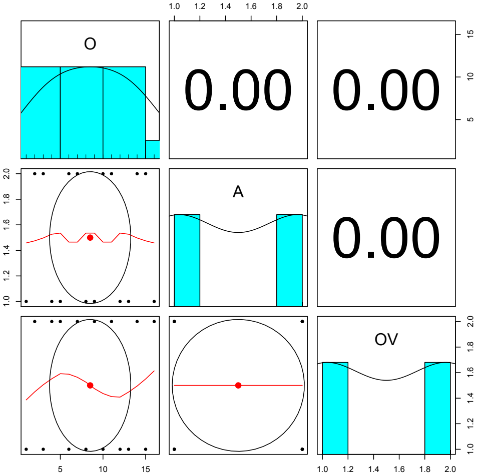
Variables are independent, and are independent of order, one way

```
> best
> pairs.panels(betteryet)

    O A OV
1   1 A  O
2   2 B  V
3   3 B  O
4   4 A  V
5   5 A  O
6   6 B  V
7   7 B  O
8   8 A  V
9   9 A  V
10 10 B  O
11 11 B  V
12 12 A  O
13 13 A  V
14 14 B  O
15 15 B  V
16 16 A  O
```



Variables are independent, and are independent of order

```
> best
> pairs.panels(best)

    O A OV
1   1 A  O
2   2 B  V
3   3 B  O
4   4 A  V
5   5 A  V
6   6 B  O
7   7 B  V
8   8 A  O
9   9 A  V
10 10 B  O
11 11 B  V
12 12 A  O
13 13 A  O
14 14 B  V
15 15 B  O
16 16 A  V
```



Multiple ways to present and analyze the data

1. Data analysis as a detective process (Descriptive statistics)
   - What happened?
   - What is a plausible description?
   - What are plausible alternative descriptions?
   - Be a strong critic.
2. Data analysis as a judicial process (Inferential statistics)
   - Are the results different from just random results?
   - How confident are you of the results?
   - Would the results be the same if you did it again?
   - How willing are you to be you will get the same result again?

Consider the Recall and Recognition data

1. How to describe it
    - Raw data
    - Summary statistics
    - Graphically
2. All tables and graphs are prepared by using the R computer package. For details on using R, consult the tutorials, particularly the short tutorial, listed in the syllabus
    - First, install R from http://r-project.org (just do this once)
    - Then, install the *psych* (just do this once)
        - install.packages(”psych”)
    - library(psych) #everytime you start R

The Very RAW data as entered into Excel – but just showing some of it

This is clearly not very useful. We need to think of ways to organize it.

Condition L1P1 L1P2 L1P3 L1P4 L1P5 L1P6 L1P7 L1P8 L1P9 L1P10 L1P11 L1P12 L1P13 L1P14 L1P15 L1Tot L2P1 L2P2 L2P3 L2P4 L2P5 L2P6 L2P7 L2P8 L2P9 L2P10 L2P11 L2P12 L2P13 L2P14 L2P15 L2Tot L3P1 L3P2 L3P3 L3P4 L3P5 L3P6 L3P7 L3P8 L3P9 L3P10 L3P11 L3P12 L3P13 L3P14 L3P15 L3Tot

2 1 0 1 1 1 0 0 0 0 0 0 0 0 1 1 6 1 1 1 1 1 0 0 0 1 0 0 0 0 0 1 7 1 1 0 1 1 1 0 0 0 1 0 1 0 0 1 0 7 1 1 1 0 1 0 1 1 0

1 0 0 0 1 1 1 9 2 1 1 1 1 1 1 0 0 1 1 0 1 1 1 0 11 1 1 1 1 1 1 1 0 1 1 0 1 1 0 1 12 1 1 1 1 1 1 1 1 1 0 0 1 0 1 1 1

12 2 1 1 0 0 1 1 1 0 1 0 1 1 0 1 1 10 1 0 1 1 0 1 1 1 1 0 0 0 0 1 1 9 2 1 0 1 0 1 1 1 1 1 0 1 1 1 1 1 12 1 1 0 1 0 1

1 0 1 0 1 1 1 1 1 11 2 1 1 1 1 1 1 1 0 1 1 1 1 1 0 0 12 1 1 1 1 1 1 0 0 1 1 1 0 1 1 1 12 1 0 0 0 1 0 1 1 1 1 0 1 1 1

0 0 8 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 13 1 1 1 1 1 1 1 1 0 0 0 1 0 0 1 1 10 1 1 1 1 1 1 0 1 0 1 0 1 1 1 1 1 12 1 0 1

0 1 0 0 1 0 1 0 1 1 1 1 0 8 2 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 14 1 0 1 0 1 1 1 0 1 0 0 1 1 1 1 10 2 1 1 1 1 1 1 0 1 0 1

1 1 1 1 1 13 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 14 1 1 0 1 1 1 1 1 1 1 0 0 0 1 1 1 11 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 13 2 1 1 1 1 1 1 1 0 1 1 1 0 1 1 0 12 1 0 1 1 1 1 1 1 1

1 1 1 1 1 1 14 1 1 1 1 1 1 0 1 0 1 0 1 0 1 0 0 9

The raw data as read into R replacing blanks with NA

```
> recall <- read.clipboard.tab()
> recall <- recall[-1]
> recall
   Condition L1P1 L1P2 L1P3 L1P4 L1P5 L1P6 L1P7 L1P8 L1P9 L1P10 L1P11 L1P12 L1P13 L1P14
1          1    1    0    1    1    1    0    0    0    1     0     1     0     0     1
2          1    1    1    0    1    0    1    1    0    1     0     0     0     1     1
3          1    1    1    1    1    1    1    1    1    0     0     1     0     1     1
4          1    0    0    0    1    0    1    1    1    1     0     1     1     1     0
5          1    1    0    1    1    0    1    1    1    1     1     1     1     1     1
6          1    1    1    1    1    1    1    1    0    0     0     1     0     0     1
7          1    1    1    1    1    1    0    1    0    1     0     1     1     1     1
8          1    0    1    0    1    0    0    1    0    1     0     1     1     1     1
9          1    1    0    1    1    1    1    1    1    1     0     0     0     1     1
10         1    1    1    1    1    1    1    1    1    1     1     0     0     1     1
11         1    1    1    1    1    1    0    1    0    1     0     1     0     1     0
12         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
13         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
14         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
15         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
16         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
17         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
18         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
19         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
20         2   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
21        NA   NA   NA   NA   NA   NA   NA   NA   NA   NA    NA    NA    NA    NA    NA
```

A bit of strange code (can be appreciated or ignored)

```
recall <- read.clipboard.tab()
dim(recall)
[1]  21 564

W <- seq(2, 257, 16)
W
 [1]   2  18  34  50  66  82  98 114 130 146 162
        178 194 210 226 242

w <- outer(W,0:15,"+")


w
```

1. First copy the data to the clipboard and then read the clipboard into the recall data.frame
2. How big is this data frame? (What are the dimensions?)
3. Create a vector to show where each list is
4. Then create a vector to show how to add up the items

```
 [1]   2  18  34  50  66  82  98 114 130 146 162 178 194 210 226 242

     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16]
 [1,]    2    3    4    5    6    7    8    9   10    11    12    13    14    15    16    17
 [2,]   18   19   20   21   22   23   24   25   26    27    28    29    30    31    32    33
 [3,]   34   35   36   37   38   39   40   41   42    43    44    45    46    47    48    49
  ...
[16,]  242  243  244  245  246  247  248  249  250   251   252   253   254   255   256   257
```

Find means for each person for each position

```
rec <- matrix(NA,nrow=21,ncol=15)
for (i in 1:15) {rec[,i] <- rowMeans(recall[w[,i]],na.rm=TRUE)}
colnames (rec) <- paste0("P",1:15,"")
rownames(rec) <- paste0("S",1:21,"")
rec
```

```
       P1    P2    P3    P4    P5    P6    P7    P8    P9   P10   P11   P12   P13   P14   P15
S1  1.000 0.750 1.000 0.750 0.625 0.625 0.625 0.750 0.750 0.625 0.875 0.750 0.750 1.000 0.750
S2  0.875 0.625 0.875 0.625 0.500 0.500 0.875 0.625 0.625 0.625 0.375 0.625 0.875 0.500 0.750
S3  0.875 1.000 0.750 1.000 0.750 0.625 0.750 0.875 0.500 0.375 0.625 0.500 1.000 0.750 0.875
S4  0.750 0.375 0.625 0.750 0.625 0.500 0.375 0.500 0.500 0.250 0.375 0.375 0.875 0.750 0.875
S5  1.000 0.750 1.000 0.750 0.750 0.750 0.875 0.750 1.000 0.875 0.875 0.625 1.000 0.750 0.750
S6  0.875 0.875 1.000 1.000 1.000 1.000 1.000 0.625 0.500 0.625 0.750 0.875 0.875 0.625 1.000
S7  1.000 0.875 0.625 0.625 0.875 0.375 0.750 0.875 0.500 0.375 0.750 0.875 0.750 0.750 0.875
S8  0.375 0.875 0.375 0.625 0.500 0.375 0.625 0.375 0.625 0.500 0.750 0.500 0.625 0.875 0.875
S9  1.000 0.750 0.875 0.750 0.750 0.500 0.875 0.625 0.500 0.125 0.500 0.500 0.375 0.625 0.875
S10 0.875 0.625 0.750 0.875 0.500 0.500 0.625 0.750 0.875 0.625 0.375 0.625 0.500 0.375 0.750
S11 0.875 0.750 0.875 0.750 0.750 0.625 0.500 0.750 0.500 0.375 0.750 0.125 1.000 0.750 0.875
S12 0.875 0.750 1.000 0.875 0.750 0.750 0.500 0.500 0.625 0.625 0.375 0.750 0.750 0.625 1.000
S13 1.000 1.000 0.875 0.750 0.625 0.750 0.750 0.500 0.750 1.000 0.375 0.750 0.875 0.750 0.750
S14 0.875 0.750 0.750 0.625 0.500 0.750 0.750 0.625 0.500 0.500 0.375 0.500 0.375 0.750 0.750
S15 1.000 0.750 0.750 0.875 0.500 1.000 0.750 0.750 1.000 0.625 0.750 1.000 0.875 0.750 0.625
S16 1.000 1.000 1.000 1.000 0.750 0.625 0.750 0.750 0.750 0.500 0.750 0.625 0.625 0.625 0.750
S17 1.000 0.875 1.000 0.500 0.875 0.750 0.875 0.750 1.000 0.750 0.750 1.000 1.000 1.000 1.000
S18 1.000 0.875 1.000 0.875 1.000 0.750 0.875 1.000 0.750 1.000 1.000 1.000 1.000 0.875 1.000
S19 0.500 0.500 0.250 0.000 0.000 0.000 0.125 0.000 0.000 0.125 0.125 0.125 0.125 0.000 0.000
S20 1.000 0.750 0.875 0.750 0.875 0.750 0.875 0.875 0.625 0.875 0.750 0.500 0.875 0.875 0.875
S21 0.875 1.000 0.750 0.625 0.625 0.625 0.750 0.750 0.750 0.750 1.000 0.875 0.375 0.625 0.375
```

Find the totals for each list

1. The total number recalled for each list was entered as the 16th element for each list
2. We have these data in the spread sheet
3. We can recover them by addressing every 16th position starting at position 17

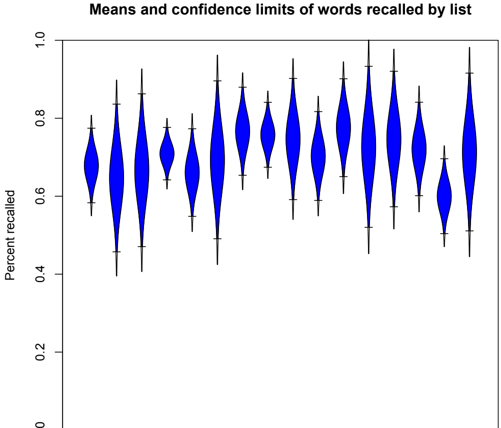
```
tot<- seq (17,257,16)
 recall[,tot]
```

	L1Tot	L2Tot	L3Tot	L4Tot	L5Tot	L6Tot	L7Tot	L8Tot	L9Tot	L10Tot	L11Tot	L12Tot	L13Tot	L14Tot	L15
1	7	NA	NA	11	9	NA	NA	14	NA	14	14	NA	NA	13	
2	9	NA	NA	11	6	NA	NA	12	NA	10	11	NA	NA	12	
3	12	NA	NA	12	11	NA	NA	11	NA	10	14	NA	NA	10	
4	8	NA	NA	10	9	NA	NA	12	NA	9	9	NA	NA	6	
5	13	NA	NA	11	13	NA	NA	12	NA	13	15	NA	NA	14	
6	10	NA	NA	12	13	NA	NA	13	NA	14	13	NA	NA	13	
7	12	NA	NA	13	12	NA	NA	12	NA	11	8	NA	NA	12	
8	8	NA	NA	11	7	NA	NA	12	NA	6	7	NA	NA	12	
9	11	NA	NA	8	10	NA	NA	10	NA	9	11	NA	NA	8	
10	13	NA	NA	9	12	NA	NA	7	NA	8	11	NA	NA	7	
11	9	NA	NA	9	7	NA	NA	10	NA	12	15	NA	NA	12	
12	NA	6	7	NA	NA	12	10	NA	12	NA	NA	14	13	NA	
13	NA	11	12	NA	NA	9	13	NA	12	NA	NA	12	12	NA	
14	NA	10	9	NA	NA	9	10	NA	9	NA	NA	11	9	NA	
15	NA	12	11	NA	NA	14	12	NA	14	NA	NA	11	10	NA	
16	NA	12	12	NA	NA	10	10	NA	13	NA	NA	11	12	NA	
17	NA	14	10	NA	NA	14	13	NA	12	NA	NA	13	14	NA	
18	NA	13	14	NA	NA	14	15	NA	14	NA	NA	14	14	NA	

Show the data by person and by list: Is there an pattern?

R code

```
error.bars(recall[,tot]/15,main="Means and confidence limits
 of words recalled  by list",xlab="List number",
                ylab="Percent recalled",ylim=c(0,1))
```



Describe the List data

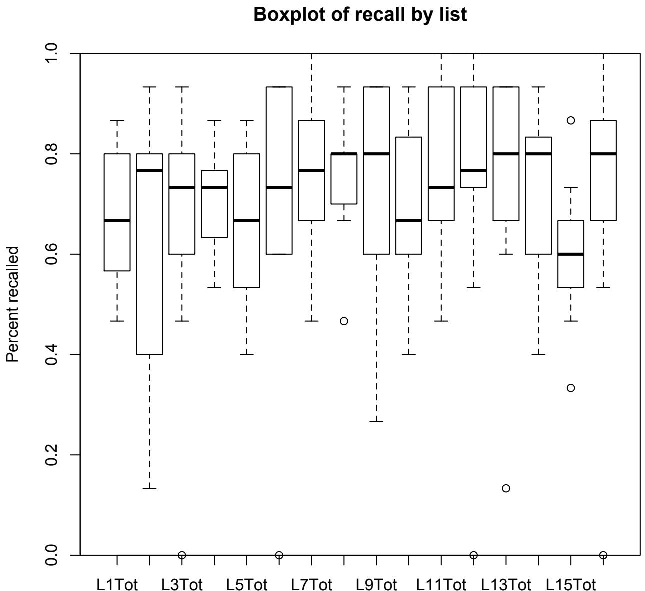
describe(recall[,tot]/15)

```
        vars  n mean   sd median trimmed  mad  min  max range  skew ku
L1Tot      1 11 0.68 0.14   0.67    0.68 0.20 0.47 0.87  0.40  0.01
L2Tot      2 10 0.65 0.26   0.77    0.68 0.15 0.13 0.93  0.80 -0.73
L3Tot      3 10 0.67 0.27   0.73    0.72 0.15 0.00 0.93  0.93 -1.27
L4Tot      4 11 0.71 0.10   0.73    0.71 0.10 0.53 0.87  0.33 -0.24
L5Tot      5 11 0.66 0.17   0.67    0.67 0.20 0.40 0.87  0.47 -0.18
L6Tot      6 10 0.69 0.28   0.73    0.75 0.20 0.00 0.93  0.93 -1.27
L7Tot      7 10 0.77 0.16   0.77    0.78 0.15 0.47 1.00  0.53 -0.27
L8Tot      8 11 0.76 0.12   0.80    0.77 0.10 0.47 0.93  0.47 -0.91
L9Tot      9 10 0.75 0.22   0.80    0.78 0.20 0.27 0.93  0.67 -0.99
L10Tot    10 11 0.70 0.17   0.67    0.71 0.20 0.40 0.93  0.53 -0.10
L11Tot    11 11 0.78 0.19   0.73    0.79 0.30 0.47 1.00  0.53 -0.26
L12Tot    12 10 0.73 0.29   0.77    0.78 0.20 0.00 1.00  1.00 -1.45
L13Tot    13 10 0.75 0.24   0.80    0.80 0.20 0.13 0.93  0.80 -1.48
L14Tot    14 11 0.72 0.18   0.80    0.73 0.10 0.40 0.93  0.53 -0.62
L15Tot    15 11 0.60 0.14   0.60    0.60 0.10 0.33 0.87  0.53  0.00
L16Tot    16 10 0.71 0.28   0.80    0.77 0.15 0.00 1.00  1.00 -1.45
```

Oops, there was something wrong with the data

1. Note on the previous slide that the minimum for some positions was zero.
2. This does not look right.
3. Lets explore the data graphically to see what is happening.
4. It turns out that one person systematically had very poor recall.
5. How should we treat such an outlier?

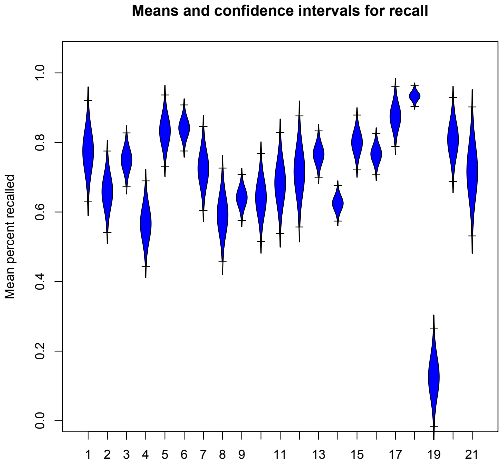
Boxplot the data to try to figure out what is happening



Means by subject show the problem

R code

```
error.bars(t(recall[,tot]),ylab="Mean percent recalled",xlab="Subject",
    main="Means and confidence intervals for recall")  #plot the sujbects
```



What to do with outliers?

1. Clearly subject 19 was behaving differently from the others.
2. We do not know why but we should drop him/her.
3. In any write up, we need to say that we dropped one subject for poor performance.
4. Drop the subject `rec <- recall[-19,]  #drops the subject`
5. `Examine the serial position effect without subject 19`

Now look at the serial position curves

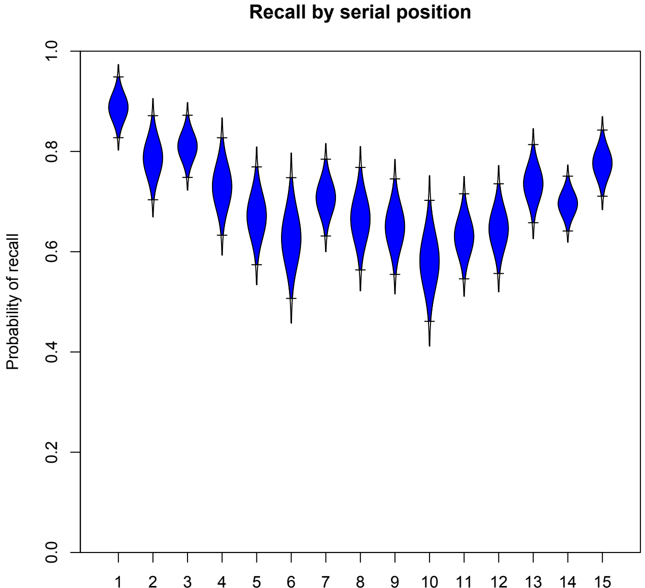
We do the same trick of organizing the data as we did before, but this time, we organize it by list position instead of by subject.

R code

```
position <- matrix(NA,nrow=15,ncol=16)
for (i in 1:15) {position[i,] <- colMeans(recall[w[,i]],na.rm=TRUE)}
describe(t(position))
error.bars(t(position),ylim=c(0,1),ylab="Probability of recall",
                xlab="Serial position",main="Recall by serial position")
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
1	1	16	0.89	0.11	0.90	0.90	0.13	0.64	1.00	0.36	-0.85	-0.48	0.03
2	2	16	0.79	0.16	0.80	0.79	0.16	0.50	1.00	0.50	-0.34	-1.29	0.04
3	3	16	0.81	0.12	0.80	0.81	0.11	0.64	1.00	0.36	0.33	-0.95	0.03
4	4	16	0.73	0.18	0.80	0.73	0.24	0.40	1.00	0.60	-0.12	-1.29	0.05
5	5	16	0.67	0.18	0.70	0.68	0.12	0.18	1.00	0.82	-0.96	1.22	0.05
6	6	16	0.63	0.23	0.64	0.65	0.24	0.09	0.91	0.82	-0.67	-0.22	0.06
7	7	16	0.71	0.14	0.71	0.71	0.16	0.45	0.91	0.45	-0.13	-1.26	0.04
8	8	16	0.67	0.19	0.70	0.67	0.20	0.30	0.91	0.61	-0.34	-1.23	0.05
9	9	16	0.65	0.18	0.67	0.67	0.19	0.18	0.90	0.72	-0.89	0.49	0.04
10	10	16	0.58	0.23	0.59	0.59	0.20	0.09	0.90	0.81	-0.65	-0.55	0.06
11	11	16	0.63	0.16	0.60	0.63	0.17	0.36	0.91	0.55	0.03	-0.81	0.04
12	12	16	0.65	0.17	0.67	0.65	0.20	0.36	0.90	0.54	-0.33	-1.23	0.04
13	13	16	0.74	0.15	0.80	0.75	0.15	0.40	0.91	0.51	-0.66	-0.60	0.04
14	14	16	0.70	0.10	0.70	0.69	0.12	0.55	0.91	0.36	0.32	-0.83	0.03
15	15	16	0.78	0.12	0.80	0.78	0.15	0.50	1.00	0.50	-0.25	-0.39	0.03

Recall varies by serial position



Does recall vary by mode of presentation?

This will require some recoding

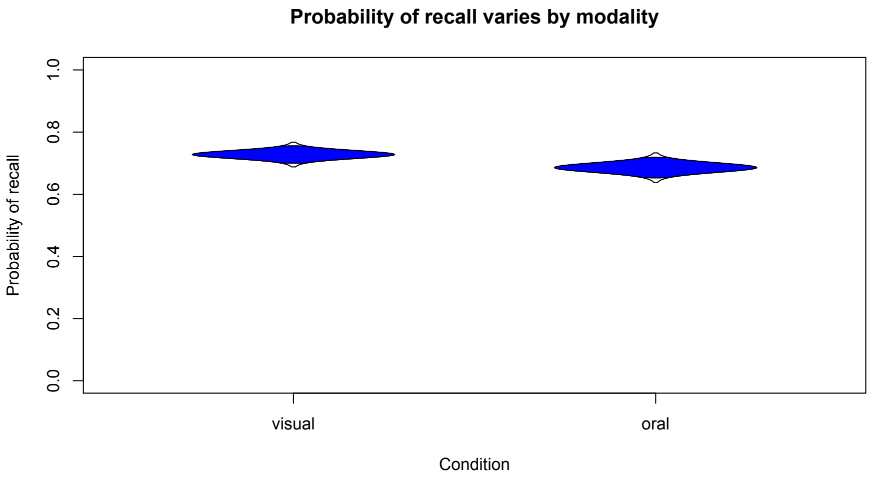
R code

```
vis.recall <-  rowSums(memory.data[,c("L1Tot","L2Tot","L7Tot",
    "L8Tot","L11Tot","L12Tot","L13Tot","L14Tot")],na.rm=TRUE)
oral.recall <-  rowSums(memory.data[,c("L3Tot","L4Tot","L5Tot",
    "L6Tot","L9Tot","L10Tot","L15Tot","L16Tot")],na.rm=TRUE)
recall.df <- data.frame(visual=vis.recall,oral=oral.recall)/(4*15)
 describe(recall.df)
 t.test(recall.df[,"visual"],recall.df[,"oral"],paired=TRUE)
```

```
       vars  n mean   sd median trimmed  mad  min  max range  skew kurtosis   se
visual    1 21 0.73 0.15   0.75    0.75 0.07 0.18 0.93  0.75 -1.94     5.03 0.03
oral      2 21 0.69 0.18   0.72    0.71 0.17 0.07 0.93  0.87 -1.67     3.80 0.04
```

```
Paired t-test

data:  recall.df[, "visual"] and recall.df[, "oral"]
t = 2.6216, df = 20, p-value = 0.01634
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.008594821 0.075532163
sample estimates:
mean of the differences
             0.04206349
```

Recall varies by modality of presentation



How to describe the recall results

You may use or paraphrase the following

1. Recall of words from lists that were presented visually ($\bar{X} = .73, sd = .15$) were recalled more than were words from lists presented orally ($\bar{X} = .69, sd = .18$), ($t_{20} = 2.62, p < .02$) (Figure XX).
2. Remember to have a figure caption for this figure that explains what those strange shapes (cats' eyes) are.

The recognition data may be examined many different ways

1. First start with the easiest: What is true versus false recognition
2. Does this vary by condition?
3. Convert raw numbers to appropriate percentages

R code

```
recog <- read.clipboard.tab()
recog[,"realrecog"] <- recog[,"realrecog"]/48
recog[,"falsemem"] <- recog[,"falsemem"]/16
describe(recog)
describeBy(recog,"Condition")  #do it by condition
```

Overall recognition results show that real words are recognized more than false ones

R code

```
describe(recog)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
Condition	1	21	1.48	0.51	1.00	1.47	0.00	1.00	2.00	1.00	0.09	-2.08	0.11
PrsRRTot	2	21	19.10	6.13	18.00	18.47	4.45	9.00	37.00	28.00	1.04	1.35	1.34
PrsRnRTot	3	21	0.52	0.60	0.00	0.47	0.00	0.00	2.00	2.00	0.57	-0.80	0.13
PrsnRRTot	4	21	19.10	5.80	19.00	19.88	1.48	0.00	28.00	28.00	-1.62	3.48	1.26
PrsnRnRTot	5	21	9.29	3.84	9.00	9.18	4.45	3.00	17.00	14.00	0.12	-0.70	0.84
PrmRRTot	6	21	0.95	1.12	1.00	0.76	1.48	0.00	4.00	4.00	1.11	0.51	0.24
PrmRnRTot	7	21	0.19	0.51	0.00	0.06	0.00	0.00	2.00	2.00	2.44	5.06	0.11
PrmnRRTot	8	21	4.52	2.94	4.00	4.53	2.97	0.00	10.00	10.00	-0.02	-1.19	0.64
PrmnRnRTot	9	21	10.24	3.03	10.00	10.24	2.97	5.00	16.00	11.00	0.01	-1.07	0.66
realrecog	10	21	0.80	0.09	0.79	0.80	0.09	0.62	0.94	0.31	-0.17	-0.81	0.02
falsemem	11	21	0.34	0.17	0.31	0.35	0.19	0.00	0.62	0.62	-0.11	-1.13	0.04

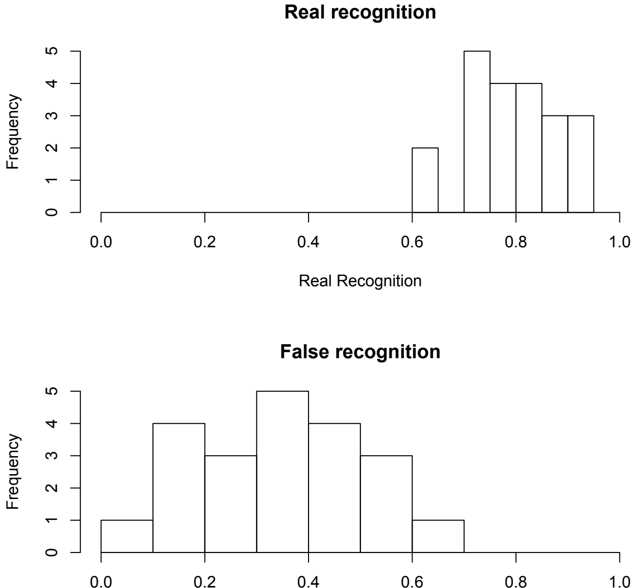
Graphically show the difference between real and false recognition

1. Perhaps the best way to compare group differences is graphically.
2. We can do this with a histogram to show the distribution
3. Or with an error bars plot (with the within option = TRUE)

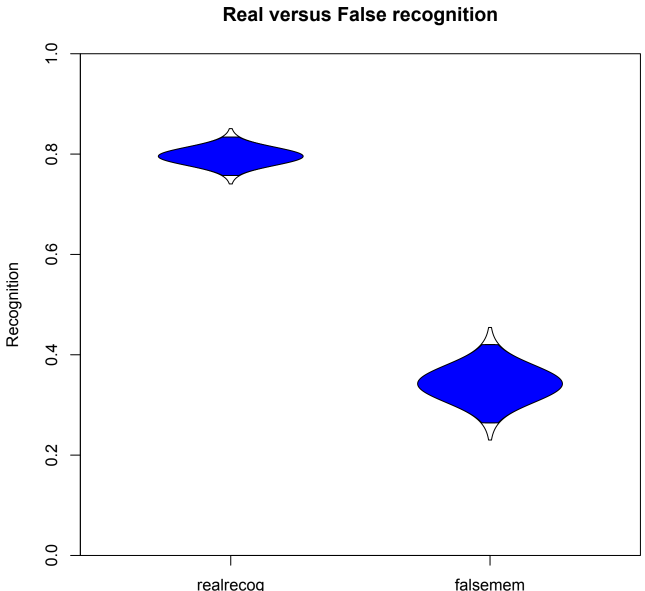
R code

```
op <- par(mfrow=c(2,1))   #do a two row graphic
hist(recog[,"realrecog"],xlab="Real Recognition",
            main="Real recognition",xlim=c(0,1))
hist(recog[,"falsemem"],xlab="False Recognition",
                  main="False recognition",xlim=c(0,1))
 op <- par(mfrow=c(1,1))
 error.bars(recog[10:11],ylim=c(0,1),within=TRUE,
  ylab="Recognition",xlab="Type of recognition",
      main="Real versus False recognition")
```

Real versus false recognition



Real versus false recognition



Final analysis

1. There were some mistakes in the original data as reported, we have cleaned that up and can report the recognition results.
2. We can examine the recognition data as a function of mode of presentation.
3. This requires some manipulation of the raw scores to break it out by mode
4. The next slide shows what we did, the subsequent slides are more useful in showing what we found

Various R commands to do the recoding ot the recognition data

The dataframe is just our data sheets transcribed into a long vector for each person.

R code

```
v <- c(1,2,7,8,11:14)
o <- c(3:6,9,10,15,16)
vrr <- rowSums( memory.data[,v+307],na.rm=TRUE)
orr <- rowSums( memory.data[,o+307],na.rm=TRUE)
vrnR <-  rowSums( memory.data[,v+324],na.rm=TRUE)
ornR <- rowSums( memory.data[,o+324],na.rm=TRUE)
vnrR <-  rowSums( memory.data[,v+341],na.rm=TRUE)
onrR <- rowSums( memory.data[,o+341],na.rm=TRUE)
 vnrnR <- rowSums( memory.data[,v+358],na.rm=TRUE)
onrnR <- rowSums( memory.data[,o+358],na.rm=TRUE)
vFrR <- rowSums( memory.data[,v+375],na.rm=TRUE)
oFrR <- rowSums( memory.data[,o+375],na.rm=TRUE)
vFrnR <- rowSums( memory.data[,v+392],na.rm=TRUE)
oFrnR <- rowSums( memory.data[,o+392],na.rm=TRUE)
vFnrR <- rowSums( memory.data[,v+409],na.rm=TRUE)
oFnrR <- rowSums( memory.data[,o+409],na.rm=TRUE)
 vFnrnR <- rowSums( memory.data[,v+426],na.rm=TRUE)
oFnrnR <- rowSums( memory.data[,o+426],na.rm=TRUE)
 visrecog <-( vrr+vnrR)/24
oralrecog <- (orr + onrR)/24
vFoil <- (vFrR + vFnrR)/8
oFoil <- (oFrR + oFnrR)/8
recog <- data.frame(vrr,orr,vrnR,ornR,vnrR,onrR,vnrnR,onrnR,vFrR,oFrR,
vFrnR,oFrnR,vFnrR,oFnrR,vFnrnR,oFnrnR,visrecog,oralrecog,vFoil,oFoil,condition)
```

The basic descriptive statistics of the recognition data

R code

```
describe(recog)
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
vrr	1	21	9.24	2.23	9.00	9.24	1.48	5.00	14.00	9.00	0.02	-0.63	0.49
orr	2	21	9.00	2.61	9.00	8.94	2.97	4.00	15.00	11.00	0.27	-0.50	0.57
vrnR	3	21	0.38	0.59	0.00	0.29	0.00	0.00	2.00	2.00	1.14	0.17	0.13
ornR	4	21	0.14	0.36	0.00	0.06	0.00	0.00	1.00	1.00	1.90	1.69	0.08
vnrR	5	21	9.38	2.33	9.00	9.47	1.48	3.00	14.00	11.00	-0.51	0.77	0.51
onrR	6	21	10.52	2.09	10.00	10.59	1.48	6.00	14.00	8.00	-0.14	-0.56	0.46
vnrnR	7	21	5.00	2.47	5.00	4.76	2.97	2.00	10.00	8.00	0.46	-0.82	0.54
onrnR	8	21	4.33	2.11	5.00	4.47	2.97	0.00	7.00	7.00	-0.36	-1.07	0.46
vFrR	9	21	0.29	0.46	0.00	0.24	0.00	0.00	1.00	1.00	0.88	-1.28	0.10
oFrR	10	21	0.57	0.87	0.00	0.41	0.00	0.00	3.00	3.00	1.31	0.71	0.19
vFrnR	11	21	0.10	0.30	0.00	0.00	0.00	0.00	1.00	1.00	2.56	4.81	0.07
oFrnR	12	21	0.10	0.30	0.00	0.00	0.00	0.00	1.00	1.00	2.56	4.81	0.07
vFnrR	13	21	1.71	1.31	2.00	1.65	1.48	0.00	4.00	4.00	0.12	-1.22	0.29
oFnrR	14	21	2.90	2.19	3.00	2.82	2.97	0.00	7.00	7.00	0.20	-1.26	0.48
vFnrnR	15	21	5.86	1.46	6.00	5.88	1.48	3.00	8.00	5.00	-0.04	-0.98	0.32
oFnrnR	16	21	4.48	2.18	5.00	4.41	2.97	1.00	8.00	7.00	0.04	-1.41	0.48
visrecog	17	21	0.78	0.11	0.79	0.78	0.12	0.54	0.92	0.38	-0.41	-0.71	0.02
oralrecog	18	21	0.81	0.09	0.79	0.81	0.12	0.67	1.00	0.33	0.34	-1.05	0.02
vFoil	19	21	0.25	0.17	0.25	0.24	0.19	0.00	0.62	0.62	0.12	-0.53	0.04
oFoil	20	21	0.43	0.28	0.38	0.44	0.37	0.00	0.88	0.88	0.00	-1.47	0.06
condition	21	21	1.48	0.51	1.00	1.47	0.00	1.00	2.00	1.00	0.09	-2.08	0.11

Test these differences using a simple t-test

1. First we test the real recognition as a function of visual vs. oral presentation.
2. Then we do the same with the False Recognition.
3. For both of these we use the paired t-test which recognizes the subjects are the same for both conditions.
4. This reports the t-test of the difference, The means were reported in the previous slide.

R code

```
t.test(recog[,"visrecog"],recog[,"oralrecog"],paired=TRUE)
t.test(recog[,"vFoil"],recog[,"oFoil"],paired=TRUE)
```

```
Paired t-test

data:  recog[, "visrecog"] and recog[, "oralrecog"]
t = -1.7083, df = 20, p-value = 0.1031
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.083730536  0.008333711
sample estimates:
mean of the differences
            -0.03769841


Paired t-test

data:  recog[, "vFoil"] and recog[, "oFoil"]
t = -2.8684, df = 20, p-value = 0.0095
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.31871340 -0.05033422
sample estimates:
mean of the differences
             -0.1845238
```

Find some useful means

R code

```
real <- (recog[, "visrecog"] + recog[, "oralrecog"])/2
falsemem <- (recog[,"vFoil"] + recog[,"oFoil"])/2
describe(real)
describe(falsemem)
t.test(real,falsemem,paired=TRUE)
```

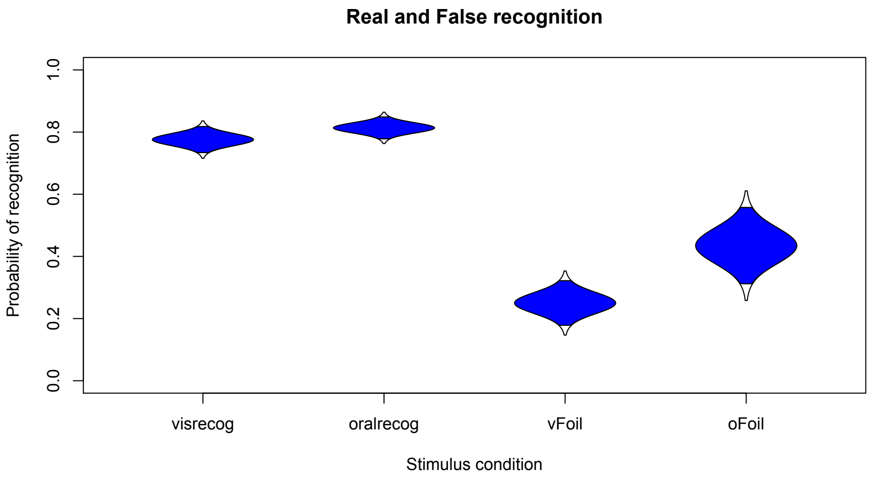
```
describe(real)
  vars  n mean   sd median trimmed  mad  min  max range  skew kurtosis   se
1    1 21 0.79 0.09   0.79     0.8 0.09 0.62 0.94  0.31 -0.14    -0.79 0.02
describe(falsemem)
  vars  n mean   sd median trimmed  mad min  max range  skew kurtosis   se
1    1 21 0.34 0.17   0.31    0.35 0.19   0 0.62  0.62 -0.11    -1.13 0.04

Paired t-test

data:  real and falsemem
t = 9.97, df = 20, p-value = 3.328e-09
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.3577318 0.5470301
sample estimates:
mean of the differences
               0.452381
```

Report this in the results section

You are welcome to take these verbatim, or put into your own words. Remember: say it in words, say it in numbers, say it in statistics.

1. Although the mean recognized following visual presentation ($\bar{X} = .78, sd = .11$) was less than the mean following oral presentations ($\bar{X} = .81, sd = .09$), this difference was not significant ($t = -1.71, df = 20, p = .10$).
2. However, unpresented words (Foils) that were high associates of the presented words were falsely recognized more following Oral presentation ($\bar{X} = .43, sd = .28$) than following visual presentation ($\bar{X} = .25, sd = .17$), ($t_{20} = -2.87, p < .01$) (Figure **??**)
3. As expected, words presented were recognized more ($\bar{X} = .25, sd = .09$) than words that were not presented ($\bar{X} = .25, sd = .17$) ($t_{20} = 10.69, p < .001$).

Real and False Recognition



Inserting figures

1. The previous figure can go into your manuscript. Cut and paste into a pdf.
2. Remember to come up with a suitable figure caption describing what is being shown.

What do we conclude?

1. The modality of presentation makes a difference.
2. Recall was better for words that were seen rather than those that were heard.
3. False Recognition was greater (worse) for words that were heard rather than seen.
4. This suggests that visual presentation improves accuracy of memory.