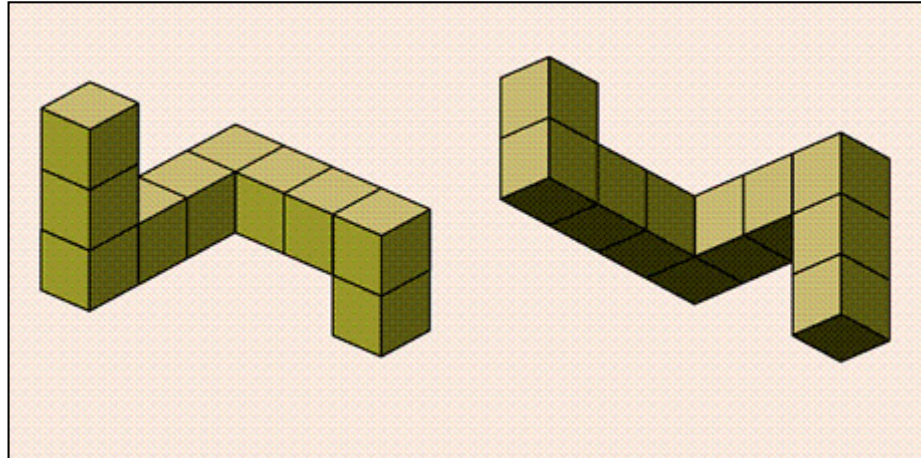

Mental Rotation Lab



How is abstract information represented by the mind? Across our previous labs, this question has been continuously lurking in the shadows. In our final lab, we will take a look directly at this issue through the lens of “mental rotation” tasks.

Imagine a dog walking down your favorite New York street. Now imagine the dog has only three legs instead of four. Now imagine putting back the fourth leg, but making the dog wear a sweater. These mental “simulations” are infinitely creative, but also can feel as vivid as our actual visual experience. When you mentally construct a picture or image in your mind it is known as *mental imagery*. Mental imagery is a classic research topic in cognitive psychology because - unlike perception or sensation research - it involves mental representations and processing of information which are not immediately perceptible (e.g., with your eyes). Psychologists know a lot about how the visual system helps us perceive our surroundings, but much less about how these perceptual representations support later abstract thinking.

The idea of mental images seem so familiar and common-place, yet it seems very difficult to get a handle on what form these representations actually take. Indeed, one of the most contentious debates in cognitive psychology in the late 1960s

was about the nature of mental imagery. Are mental images real? Are they even images? At first glance these questions may seem silly because the vividness of our imagination seems so obvious to us (we called this the *argument from introspection* in lecture). However, the central question in early work on mental imagery was if the mind actually represents information in the form of images (just like a JPEG or PDF figure you put in your paper) or if all abstract thought is mediated by propositional, symbolic, language-like processes.

The two sides of this debate (discussed in lecture) were the analog camp (represented by Kosslyn) and the propositional camp (represented by Pylyshyn). The analog camp argued that mental images are image-like representations that shared much in common with the lower level perceptual properties of our experience. Indeed, proponents of these theory likened mental images to the “surface displays generated on a cathode ray tube by a computer” (Kosslyn and Pomerantz, 1977). In contrast, the propositional camp argued that mental images were represented in a more abstract, language-like representation (see lecture notes for more on this).

So, what does mental rotation have to do with any of this?



Familiar objects rotated in various ways along the axis extending outward from the page.

Mental Rotation

The original mental rotation experiments were done by Roger Shepard and Jean Metzler in 1971. In these experiments, participants viewed pairs of objects like the ones at the top of the page and were asked to judge if they were the same object (just from a different view) or were different (in the sense of being a mirror image of the each other). The key manipulation was that the objects were rotated in various ways. For example, imagine you are holding a warm cup of coffee in your hand. Now rotate the cup 15 degrees to clockwise. Good. Now go a bit further, 30 degrees. Watch out! The liquid might fall out!

In the Shepard & Metzler (1971) experiments, the time it took people to decide if an object was the same (just rotated) or different (a mirror reflection) depended on the **angle of rotation** (see Figure on the right is taken from the original paper - one of our target readings). In other words, it would take you longer to decide if the 30 degree rotated version of your cup was the same as the upright version relative to the 15 degree rotated version. The idea is that in order to make the same/mirror judgement, people **actually mentally rotate the object**. Bigger angles require more time to “rotate” in your mind just as they would in the real world (see Figure below). As mentioned in class, Cooper (1976) performed an even more interesting followup which provided even stronger evidence that people rotate objects when doing the matching task. Thus, most people agree that people do seem to perform some kind of mental rotation.

Why is this surprising?

Take a moment to think about this result: the speed of making a same/mirror judgment depends

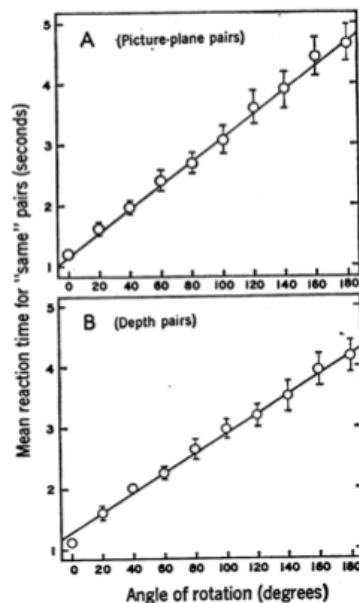


Fig. 2. Mean reaction times to two perspective line drawings portraying objects of the same three-dimensional shape. Times are plotted as a function of angular difference in portrayed orientation: (A) for pairs differing by a rotation in the picture plane only; and (B) for pairs differing by a rotation in depth. (The centers of the circles indicate the means and, when they extend far enough to show outside these circles, the vertical bars around each circle indicate a conservative estimate of the standard error of that mean based on the distribution of the eight component means contributed by the individual subjects.)

on the angle at which the object pair is rotated. It is a trivial result, but also kind of profound. Think about all the possible ways the experiment *might* have worked out. Perhaps the representations we have to thinking about objects is more sophisticated. Instead of mentally rotating objects until they match we might be able to do a quick match based on some other property. Why rotate? Why not do some other type of transformation? Rotation implies stepping through a large number of intermediate states (1 degree, 2 degree, 3 degree, etc...), not all of which are critical to the actual same/mirror judgment.

The mental rotation experiments of the early 70s suggested that the abstract, mental processing of objects in our mind followed similar principals to those in the real world. This result was exciting to the analog theorists (like Kosslyn) who felt the mental images were like perceptual representations. On the other hand, it is not obvious how the propositional theory might explain this finding. There is nothing inherent to the propositional theory that would predict that objects would be matched based on rotations. Indeed, when you think about the propositional theory, it gives you some ideas about alternative algorithms or procedures for doing that task that are independent of angle (for example matching features and relations between different parts of the object). Of course, the propositional theory could be modified to account for this result (assuming that matching is done with a sequence of small, incremental “transformations” of some abstract representation). However, this seems like an ad-hoc add-on to the theory rather than a natural implication of the view the representations are propositional.

In the end, the field decided that, despite these intriguing results, the mental imagery debate was unlikely to resolved on the basis of behavioral data alone (Anderson, 1978). In effect, both the analog and propositional theory made a cardinal sin in theorizing about cognitive function: **they each considered only the nature of the representation and not the process which operates over those representations.** As we have discussed this semester extensively, specification of both the representation and the processes that operates over those representations is critical for any cognitive theory. As time as gone on, the debate about mental imagery are a little less contentious, although the question of the

substrate of mental representation is still very active. The analog/imagery perspective is now most often associated with the *embodied cognition* movement (which argues that much of thinking is tied to the spatial properties of the world and our interactions with the environment). In addition, there have been a number of attempts to “blend” both the analog and proposition ideas (similar to the dual-code theory of Anderson, 1978). One example of this is the “perceptual symbols systems” approach from Barsalou (1999) which argues for a propositional representation that operates *over* images.

Mental rotation remains one of the core phenomena that cognitive psychologists study. In our lab we are going to replicate the mental rotation experiment as an excuse to learn a bit about linear regression.

Required Readings

Shepard, R.N. and Metzler, J. (1971) “Mental rotation of three-dimensional object.” *Science*, 117, 701-703.

Takano, Y. and Okubo, M. (2002) “Mental rotation.” in Lynn Nadel (Ed.), *Encyclopedia of Cognitive Science*, London: Macmillan.

[these papers will be available from the class website]

Optional Readings

Anderson, J.R. (1978) “Arguments concerning representations for mental imagery” *Psychological Review*, 85(4), 249-277.

Kosslyn, S.M., Ganis, G. and Thompson, W.L. (2001) “Neural foundations of imagery” *Nature Reviews Neuroscience*, 2, 635-642.

Further Reading

Hugdahl, K., Thomsen, T. and Erslund, L. (2006) “Sex differences in visuo-spatial processing: An fMRI study of mental rotation” *Neuropsychologia*, 44(9), 1575-1583.

Flusberg, S.J. and Boroditsky, L. (2010) “Are things that are hard to physically move also hard to imagine moving?” *Psychonomic Bulletin and Review*, 18, 158-164.

[the other papers are available via Google scholar accessed from NYU’s network]

Lab Steps

#1: Collect Data.....

To collect data for our experiment we are going to use a web based version of the mental rotation task. Unfortunately, since the computers in our lab are without internet, you will do this at home. The URL for the experiment is here: http://psych.hanover.edu/JavaTest/Cognition/Cognition/mentalrotation_instructions.html

When you click the link at the bottom of the page that says "Click here to start the experiment" a window will pop up that should take over your entire screen. The first page lets you configure various aspects of the experiment. We want to test the following parameters:

Stimulus Type	Original 3D
Number of Rotation Angles:	7
Should the left stimulus be allowed to rotate or be fixed?	Rotate
Stimulus Size:	300 pixels
Number of Trials per Level:	15
Dist. from Fixation:	0.150
Delay before stimuli:	1000
Stimulus on till response:	YES

When you have this entered in correctly, press "Done" to begin the experiment. A fixation cross will appear in the center of the screen. Press the space bar when you are ready to begin. On each trial view the pairs of items. If they are the same item (ignoring rotation) press 's'. If these are mirror reflections (i.e., you can't rotate one to make it the same as the other), press 'm' (for mirror reflection).

#2: Get results and process them.....

When you get done, the experiment will display a set of results. Write down or take a screen shot of

the display. Finally, at the bottom of the screen is a button that says "Show Trial Data". When you press this it will pop up your data in a format looking like this:

```
Trial : Angle : Stimulus : Response : RT(msec)
1 : 180 : Mirror : Same : 10421.0
2 : 90 : Same : Same : 469.0
3 : 90 : Mirror : Same : 272.0
4 : 0 : Same : Mirror : 376.0
```

Select all the data in the window and copy-paste it into a text file (using text edit on the mac or wordpad on the PC). Save the data to a text file on your computer (like Todd.txt).

Next, open the data in Excel. When you open the data the "Text Import Wizard" will appear. In the first option choose "Delimited". Also choose to start the import at row 2 (skipping the header). Next, excel will ask you for what delimiter you want to use. In this file, the colon is the divider, so choose "other" and enter the ":" character. You should see that excel detects the columns in the preview window. Click Finish (you can skip the next step of the import wizard).

Now that you have your data, we need to format it a bit for R. First lets add a "subject number" column. Using the subject number you received in class make a new column that just has your subject number over and over. To do this, select the first column of data (the trial number) and choose "insert column". Then in the first column type your subject number. Then select the bottom corner of that top cell and drag downwards. Here is what my data looks like now (I pretended to be subject 13):

	A	B	C	D	E	F
1	13	1	180	Mirror	Same	10421
2	13	2	90	Same	Same	469
3	13	3	90	Mirror	Same	272
4	13	4	0	Same	Mirror	376
5	13	5	90	Same	Same	112
6	13	6	0	Mirror	Mirror	79
7	13	7	0	Mirror	Same	8
8	13	8	90	Mirror	Mirror	87
9	13	9	0	Mirror	Same	87
10	13	10	180	Mirror	Mirror	80
11	13	11	90	Same	Same	47
12	13	12	90	Same	Mirror	95
13	13	13	0	Same	Same	80

Describe columns

Now, choose "save as" and save your data as <subject number>.txt. Make sure the format is "tab delimited text".

Email me the file and I will collect up everyone's data into a single structure suitable for simple analysis in R.

#3: Analysis!

Step 1

Read in your data into R using `read.table()`. Remember there is not header line (`header=F`)

Step 2

Name the columns of the file to something that makes sense to you using the `names(data)` command.

Step 3

Now we want to compute the overall accuracy of each subject. To do this we want to count the number of times each person was "correct" and find the average hit rate.

Unfortunately, the mental rotation experiment online does not create a separate column for "hit" but we can make one ourselves. One of the columns of the data shows the stimulus `SAME=0`, `MIRROR=1`, and another shows the response (also `SAME=0`, `MIRROR=1`). If the stimulus type and the response are the *same*, then it is a hit, and if it is different it is a miss. We can add a new column to our data that codes if the trial was a hit or a miss like this:

```
data$hit=(data$stim==data$resp)
```

Now, if you look at the new column `data$hit`, it will be `TRUE` or `FALSE` depending on a hit or a miss.

Finally, we just want to compute the average or mean hit rate as a function of subject. To do this we use `tapply`. I won't give you the exact command but it should look something like this (refer to previous labs if you need a reference)

```
accs = tapply( ????, ?????, mean)
```

And remember, you want to find the mean hit rate (in your new column `data$hit`) as a function of subject number.

Record the mean and standard deviation of the accuracies (remember you can do `mean()` or `sd()` of the `accs` result). Look closely at the results. How did people do? Are there particular people who did very poor accuracy? Also look at accuracy for same/mirror trials separately. Are one type of trial harder than the other?

Required Analysis

Compute the mean accuracy for the group. Do a t-test to ensure that performance is greater than chance ($\mu=0.5$, $p<0.05$). You should report the standard ($M=X$, $SD=X$, $t(X)=X$, $p<X$) along with what the null hypothesis is.

Because some people didn't perform that well (some may have been distracted), if you want to drop an individual from the analysis you can do so by excluding their subject number. For example,

```
newdata=data[data$subj!=3, ]
```

Will drop subject 3. To drop more than one you just use lots of "&" (ANDs) i.e.:

```
data$subj!=3&data$subj!=4
```

You will want to come up with a "fair" standard for exclusion and explain this clearly in your paper.

Step 4

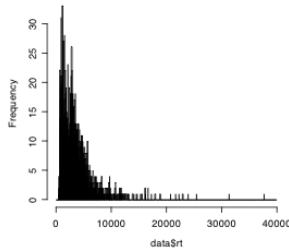
As another clean up, we probably want to drop trials where people took too long to respond (40s is pretty long). Using the code and directions from Lab 2, plot a histogram of the reaction times for everyone in the experiment (do not drop any data yet). Then, by looking at the overall distribution decide on a reasonable cutoff for excluding RT data (this could also be the median +2*std. dev. we used in Lab 2). A basic version of

Pearson's product-moment correlation

```
data: data$rt and data$angle
t = 5.819, df = 1573, p-value = 7.158e-09 <<< this is the value of the stats
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.09646338 0.19317328
sample estimates:
 cor
0.1451650 <<<<<<<<<< This is the correlation coefficient of your test
```

An example output from R's `cor.test()` function. The words in bold help you interpret the result.

the RT histogram for ALL the data is shown below:



Drop trials with reaction times $<$ (i.e., less than) the cutoff you choose and rename this data subset “cleandata” (see lab 2 for reference, but this should be familiar).

Step 5

Select data from SAME and MIRROR trials separately (remember this is coded in the “stimulus” column). Later, we will want to analyze the SAME/MIRROR information separately. Create a new variable called `samedata` and `mirrordata` and select from your `cleandata` structure all trials where the `trialtype` actually was SAME or MIRROR, (remember these uppercase words are codes that you have to define in R in order to use them by name).

Step 6

We are most interested in the relationship between the ANGLE of rotation and RT. Within BOTH the

`samedata` and `mirrordata`, select only trials where people responded correctly (if they made a mistake we are less interested in their data, but could analyze it later). This should be something like

```
sdc=samedata[samedata$hit==TRUE,]
```

where `sdc` might refer to “same data correct” (remember variable names are arbitrary and up to our preferences).

Step 7

Next, we can compute the correlation between RT and angle in each of the conditions. The format of this command is like the t-test. We tell R the two variables we want to assess the relationship between and then compute the correlation. In addition, R will do a statistical test (basically a t-test) to tell you if the correlation is significantly different than zero (which means there is no relation). Positive values of the correlation coefficient mean that RT is increased as angle increases. Negative values mean that RT is decreasing as angle increases.

The command for computing the correlation is:

```
cor.test(XXX$rt, XXX$angle)
```

And some example output is provided in the figure at the top of the page. To report this specific example you would write that there was a positive correlation between reaction time and angle, $r=.145$, $t(1573) = 5.819$, $p<.001$.


```

Call:
lm(formula = angle ~ rt, data = data)

Residuals:
    Min       1Q   Median       3Q      Max
-117.120 -46.994  -1.067  44.356 108.181

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 54.641373  10.356878   5.276 7.35e-07 ***
rt          0.008845   0.002196   4.028 0.000108 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 56.31 on 103 degrees of freedom

Multiple R-squared: 0.1361,    Adjusted R-squared: 0.1277 <<< - the Multiple R-squared is the same
as the correlation coefficient
you get from cor() but
squared. R-squared tell you
the proportion of variance
accounted for in the data.
This is low here (13%)
meaning there is lots of
fluctuations in RT not simply
explained by changes in
angle. This is true even
though the line fits better than
expected by chance.

F-statistic: 16.22 on 1 and 103 DF, p-value: 0.0001079 <<< - the Fstat = 16.22, df = 103, p = .00011
this is the same output you get from
cor.test() on the same data, only the
t-value is the square root of the F
value!

```

An example output from R's `lm()` function. The words in bold help you interpret the result.

Required Analysis (see Step 6)

An overall regression analysis that reports the overall effect of ANGLE on RT for both the same and mirror trials. You will report the estimated INTERCEPT and SLOPE along with the corresponding t-tests. In addition, record the overall "Adjusted R-squared and the associated F-test" to assess the overall correlation between Angle and RT. All of this is provided from `cor.test()` or `lm()`. The report should be something like "The best-fit slope to the overall data was 0.008, $t(X)=X$, $p<X$. Similarly, the best fit slope was 54.64, $t(X)=X$, $p<X$. The correlation between angle and reaction time was $R^2=.127$, $F(X,X)=X$, $p<X$."

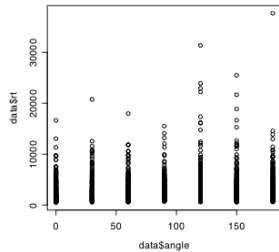
You should perform this analysis both for the "same" and "mirror" trials.

Step 8

Plot a scatter plot of data vs. angle using the plot command:

```
plot(XXX$angle, XXX$rt)
```

(where XXX is replaced with the appropriate data variable). You should get something like this:



Now, we would like to perform a linear regression on the data to see what the relationship between angle and RT is. To do this we use the `lm()` function which finds the best fit line through the data.

```
reg = lm(rt~angle,data)
```

or

```
reg = lm(data$rt~ data$angle)
```

The first version means “predict RT from ANGLE, using the data in the ‘data’ variable”. The second command more explicitly says predict the RT column from the ANGLE column. Both are identical just one requires less typing! The result is now stored in the `reg` variable. We can get information about the quality of the fit and the significance of various fit parameters by typing

```
summary(reg)
```

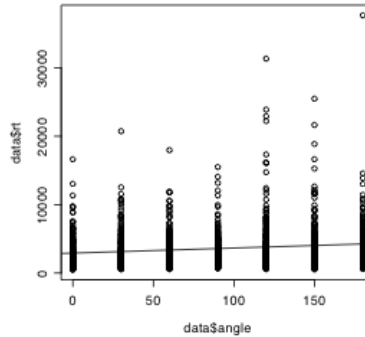
The breakdown from output this is shown at the top of this page.

As you see the `lm()` gives you all the information you need to do a linear regression. You should repeat the above analysis for both the SAME and MIRROR data, recording the information about the correlation coefficient (r), the best fit slope, intercept and associated stats, etc..

The next thing to try is to plot the best fit regression line on the plot too:

```
plot(XXX$angle, XXX$rt)
reg=lm(rt~angle,data=XXX)
abline(reg)
```

For example,



Repeat this for the same and mirror data. However, you will NOT put these graphs in your report. These are exploratory for revealing the trends in the overall data.

Step 9

So far, we’ve been focused on the aggregate data. Now we want to look closer at individual subject. This is really our critical analysis (we care if the effect obtains within individuals rather just in the group). For example, we could plot the relationship between angle and reaction time for just subject 1:

```
plot(data[data$subj==1,]$angle, data[
data$subj==1,]$rt)
```

Similarly we could do the linear regression analysis just on this subject:

```
plot(data[data$subj==1,]$angle, data[
data$subj==1,]$rt)
```

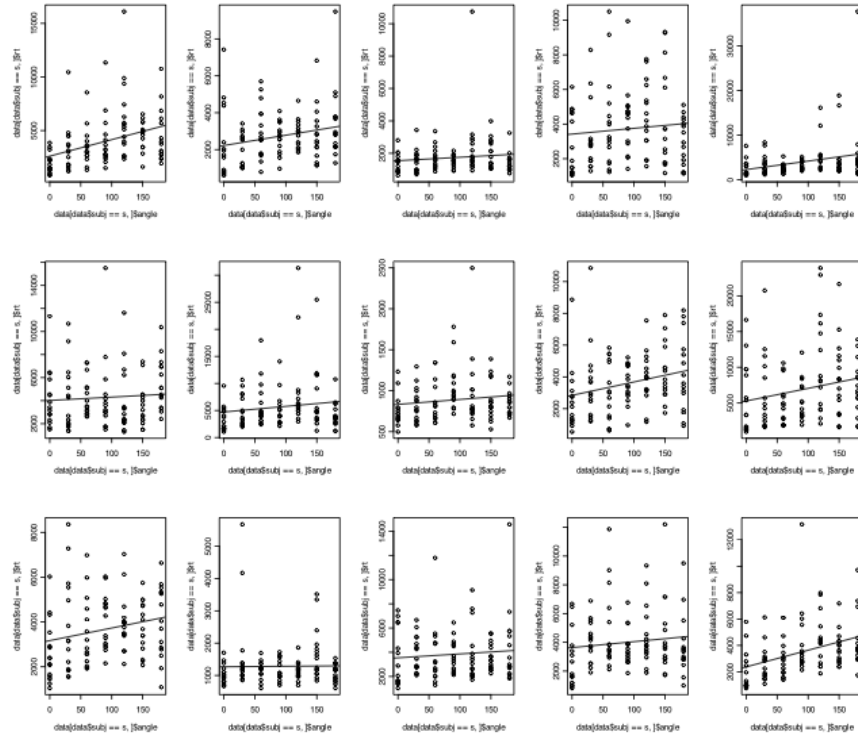
```
reg=lm(rt~angle,data=data[data
$subj==1,])
```

```
summary(reg)
abline(reg)
```

Now lets say we want to look at each person in the experiment and see how well they appear to be mentally rotating. Imagine you have 15 people in class and you want to plot all their data together, but separately.

```
quartz() # windows() on a pc
```

```
par(mfrow=c(3,5)) # 3 columns, 4 rows
```

An example output for a set of individual subject (plotting both their scatter plots and the best-fit regression line).

```

#plot each one along with regression
#line

# repeat for subject 1

plot(data[data$subj==1,]$angle, data
[data$subj==1,]$rt)

reg=lm(rt~angle,data=data[data
$subj==1,])

summary(reg)
abline(reg)

# repeat for subject 2

plot(data[data$subj==2,]$angle, data
[data$subj==2,]$rt)

reg=lm(rt~angle,data=data[data
$subj==2,])

summary(reg)

# repeat for subject 3

plot(data[data$subj==3,]$angle, data
[data$subj==3,]$rt)

reg=lm(rt~angle,data=data[data
$subj==3,])

summary(reg)

```

```
abline(reg)
```

... and so on.

Will give you something like on the next page.

How does it look? Does RT increase as a function of angle in your data?

You should make two plots like the above, one for the same data and one for mirror from each person.

Remember this means doing it separately for both:

```
samedatacorrect[data$subj==s,]
mirrordatacorrect[data$subj==s,]
```

Make two plots like the above, one for the same data and one for mirror data from each person. Remember this means doing it separately for both.

Required Analysis

A plot of each individual's data along with the best-fit regression line for SAME and MIRROR trials (hits only). This results in two figures (each containing lots of little figures)

Bonus! (for R fans)

The above analysis can be tedious (running the same set of analyses for each subject). A better way is to let the computer handle the tedious stuff by use of a FOR loop. A for loop will repeat the same chunk of code multiple times. Here is a snippet of code that does a simple for loop. See if you can extend this to handle the above code better.

```
for (s in unique(data$subj)) {
  print(s)
}
```

Step 10

Analyze the *parameters* of the best fit line for each person. Each time you run the `lm()` command on an individual, the linear regression model gives you a slope and an intercept. One thing we are interested in is if the slope for each person is (on average) greater than zero. We might also be interested in the value of the intercept (which seems quite obviously > 0.0 since it reflects the time required to decide if two identical objects are the same or different). We can start with an empty list/data structure:

```
fitdata=c()
```

Then for each person, we can compute the best-fit regression line and save the coefficients:

```
# for subject 1
s=1

# fit their data
reg=lm(rt~angle,data=data[data$subj==s,])
intercept=coef(reg)[1] # get intercept
slope=coef(reg)[2] # get slope
```

Required Analysis

Count the number of subjects who have a significantly POSITIVE slope (i.e., look at the output of `summary(reg)` in for each person). Since we are performing multiple comparisons here, use a more conservative p-value (.05/# of comparisons). If you had 14 subjects then $.05/14 \approx .003$. To report this, basically just describe it as "12/14 subjects had a significantly positive slope." Repeat this for the mirror trials. The expectation is that more people have a positive slope in the SAME trials compared to MIRROR. If you want, you can conduct a binomial test (look it up in R) to what the odds of getting N/14 subjects to have a positive slope if the true probability was 0.5 (the null hypothesis).

```

# add parameters to our list
fitdata=rbind(fitdata,c
(intercept,slope))
# for subject 2
s=2
reg=lm(rt~angle,data=data[data
$subj==s,]) intercept=coef(reg)[1] #
get intercept
slope=coef(reg)[2] # get slope
fitdata=rbind(fitdata,c
(intercept,slope))

```

and so on...

At the end you want to do:

```
fitdata=as.data.frame(fitdata)
```

```
names(fitdata) = c
('intercept','slope')
```

to convert our parameter into an array/matrix. At the end `fitdata` should look like below and contain the best fit intercept and slope (i.e., angle) for each person.

Finally, we can do a series of t-tests to see if these parameter are (on average) different than zero. For example:

```
t.test(fitdata[,1],
mu=0,alternative="two.sided")
```

```
t.test(fitdata[,2],
mu=0,alternative="two.sided")
```

Required Analysis

The final part of step 11, you created a table that has the intercept and slope for each person (it was called `fitdata` in the handout). You should do a one-sample t-test to see if the intercept and slope are different than zero overall. Report the mean and sd for the intercept and slope as well. Repeat this for the mirror data. **This results in FOUR t-tests (intercept, slope for both same & mirror).**

Required Analysis

Compute two paired t-tests which compare the best fit slope/intercept for same and mirror trials. This will require you to make two `fitdata` matrices, one for same trials and some for mirror (maybe `fitmirrordata` and `fitsamedata`). This answers the question "is the slope significantly higher in the SAME condition compared to the MIRROR condition?" This results in TWO t-test (intercept comparison and slope comparison). Report in the standard way.