Rote Memorization Techniques

W241 Summer 2020 (Daniel Hedblom Wed 6:30 pm) Final Project

Tina Huang, Gunnar Mein, Jeff Li

8/12/2020

Contents

Introduction

Background

Rote memorization has a bad name in pedagogy, causing teachers to stay away from it and students to feel free to never practice it. However, in mathematics, physics, computer science, and other sciences, being able to recall certain snippets of knowledge quickly is imperative for both being able to perform calculations, for writing code in reasonable time, and to recognize patterns and form one's own "grammatical ability" to formulate approaches. It isn't really possible to imagine what a forest looks like, and how you would navigate it, until you can effortlessly recall what a tree is. Often, there are only a handful of vital concepts that a student needs to recall, but those they need to recall with absolutely no expenditure of time or doubt.

Teachers have, at times, built little structured exercises into their curricula to help students remember such snippets. One technique is for the student to read the fact/formula/code in question so that they have a memory of seeing the thing, writing it down in their own hand (or typing it, but I prefer hand-writing for this) so they have a memory of writing it, and lastly saying it out loud multiple times so they have memory of saying it.

This is all good and nice, but can we substantiate the validity of these techniques with an experiment? In other words, "is using multiple ways of remembering the same concept effective in aiding retention?"

Intuition

The justification for the memorization approach tested here comes mainly out of intuition. When we remember a line in a song, we don't remember just the words, we remember the sound of the vocalist singing them. When we remember a vacation at a beach, we remember the color of the sky and the sound of the waves and birds, not the abstract concepts "there were birds, the sky was blue". Instead, we reconstruct these abstract concepts from our memory of the experience of our senses.

When we need to memorize purely abstract concepts for their own sake, we can manufacture sense experiences by saying them out loud and writing them down (this might be less true for writing on a computer compared to paper and pencil, but stick with us here). It is our hope that these sense memories will be easier to recall, and aid the student in producing the right abstract facts to go along with them.

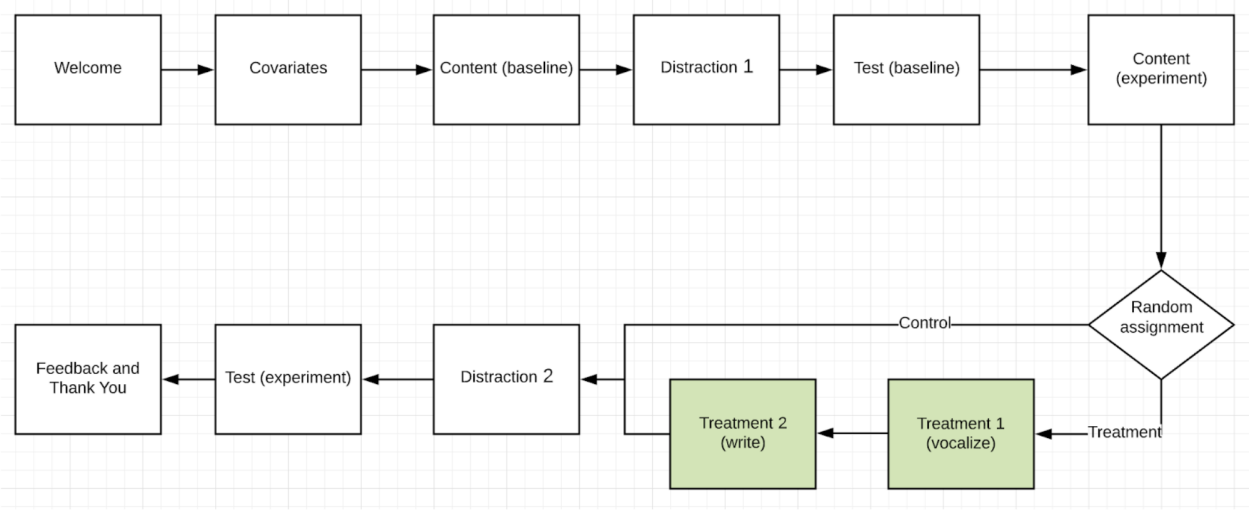
Hypothesis

Our specific hypothesis is, then, that students memorizing content by vocalizing them and writing them down should see a significantly higher score when evaluated over multiple such exercises, compared to students who only read the content quietly.

Experiment design

Basic structure

There were 2 portions of our test. The first portion was a "baseline" portion, where study participants would be presented 2 passages. The second portion would consist of the actual experiment, where our study participants would then be randomized into "treatment" or "control" groups.



Subjects were randomly assigned 2 topics in both Part 1 and Part 2 from a pool of 8 questions ranging from topics from "Units for the Calorie Content of Food" to the "History of the Submarine". After reading each passage, they were given a video distraction and then given 4 multiple choice questions on what they had read. Screen shots are provided in the appendix with sample passages and questions that people were presented with

Test subjects

Since one of our team members is a high school teacher in Washington state, we decided to solicit the students of this school to participate in this experiment. Due to a need for additional study participants, all of the team members also solicited people on social media, as well as family and friends in order to participate in this experiment, rounding out our "considered" subjects. "Assigned" were subjects who at least made it as far as entering and submitting covariants. Since there was major attrition beyond that point, the "actually evaluated" subjects are only those who completed the whole test. We have detailed data about which subjects attrited at what point, and can compute an average treatment effect on the treated (ATET).

The actual age distribution of participant who completed the test is displayed in the results section. Middle school to high school age range accounted for around a third of the total participants. The rest were distributed between the 20 to 80 years old, and the older portion skews toward the female gender.

Comparison of potential outcomes

The format of our experiment was a multiple choice test. Study participants would read a passage for a predetermined amount of time (generally around 2 minutes per passage, depending on length), and would subsequently be presented with a set of multiple choice questions. Possible scores on each test ranges from 0 (fully incorrect) to 8 (fully correct). What we are interested in is the difference in score between the pre-test (before treatment) and the post-test (after treatment). For the control group, the expected difference between tests is 0 in an ideal world. In the real world, subjects might get tired between tests and score lower on the second part, or also might get warmed up to the process of memorization and testing and score higher. However, with proper randomization, the expected value of score difference between treatment and control should be 0 if the treatment has no effect.

Randomization process

Participants were assigned to "control" and "treatment" groups randomly. The software did this right after covariates were gathered. In anticipation of collusion between students, we decided to cluster by time slot, in 5-minute increments. In hindsight, this was not necessary given our randomized draw from a question bank which made spillover effects unlikely, but it also did not hurt our process or errors too much. Students by-and-large took the tests in different time slots.

Pre-treatment-phase content and test

The first part of the experiment is the same for control and treatment - two content items are displayed along with questions we intend to ask, then a distraction takes the subject's mind off the content for about a minute, and then we test the subject on the two topics with 4 questions per topic, with 5 possible choices for each question. This gives us 8 "correct"/"incorrect" scores, which we record as a "baseline" score.

Treatment-phase content and test

The treatment of this experiment is in Part 2. The participants in the treatment group will read 2 passages with 4 questions for each passage that they will answer later on. At the top of the screen, there is instruction to read the answers to the questions shown out loud 5 times. This allows the participants to have repeated memory of speaking the concept. On the screen after, the participants will be shown the same passages and questions. At the top of this screen, there is instruction for participants to read the passage again and type out brief answers to the questions. This allows the participants to have written memory of the concept. Showing the passages and instructing the participants to read the passages on 2 different screens allows them to have repeated memory of seeing the concept. The treatment as a whole allows the participants to have repeated memory of seeing and speaking the concept and written memory of the concept and therefore tests our hypothesis. After these 2 screens, the participants will watch a 1 minute distraction video, then go to a screen where they will select correct answers for the questions in multiple choice format. We once again record their 8 correct/incorrect answers, this time marked as "experiment" scores.

Power calculation

Prior to conducting our experiment, we used a ballpark estimate in order to estimate how many observations we would need. We used an online power calculator (https://www.stat.ubc.ca/~rollin/stats/ssize/n2.html) in order to estimate our necessary effect size. Given that each study participant would answer 8 questions (with 5 multiple choice options), we used a binomial distribution to estimate our value for σ (arriving at an estimated σ of 1.131).

We estimated an effect size of getting on average one additional question right between treatment and control. With the experiment parameters of 0.05 and power calculation of .8, we estimated that we would need roughly 21 samples per group.

In retrospect, this back of the envelope calculation was not extremely helpful. We would have benefited greatly from a pretest, however, due to a software bug, we were unable to properly assess the results of our pretest. In retrospect, our estimated effect size was far too high. People generally were able to score quite well on the exam regardless of if they were in treatment or control.

Analysis of results

We ended the experiment at a time we set several days in advance, after discussing that new participation had ebbed off and that without further marketing, we could not expect more, and none of us had any grand ideas for producing more subjects. The day we picked lined up with our initial time line.

Loading and preprocessing the data

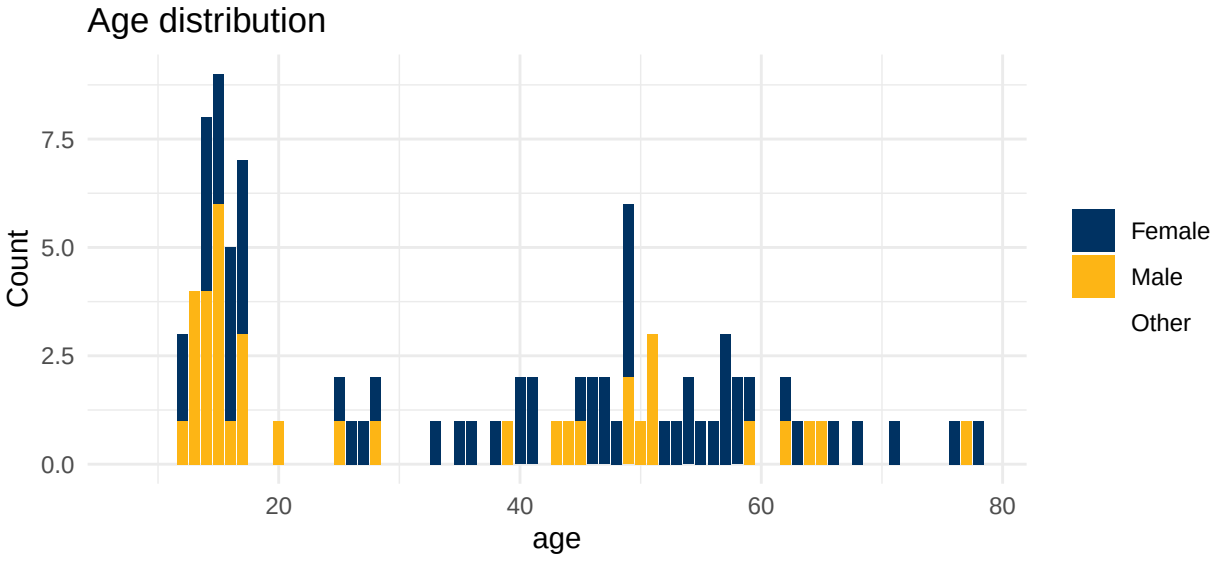
```
dataset = load_data()
dataset_long = load_data_long()
```

EDA

Table 1:

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
score	97	0.979	1.671	−2	0	2	5
score_pre	97	5.773	1.693	1	5	7	8
score_post	97	6.753	1.173	1	6	7	8
cluster	97	58.763	39.901	1	22	88	130
age	97	36.309	19.968	12	15	52	78
prior_knowledge	96	6.948	2.373	4.000	5.000	8.000	13.000
treat	97	0.485	0.502	0	0	1	1
reading	97	3.423	0.934	1	3	4	5
practice	97	2.278	0.933	1	1	3	5

At the end of the experiment, we had 50 subjects in control and 47 in treatment.



In the age distribution of only people who completed the whole experiment, we can see a group of Eastside Prepartory School students under the age of 20, a female-leaning cluster of certain Facebook friends over the age of 50, and a mixed group of other social media recruits between those groups. Gender balance does not look bad.

Randomization checks

Check by regression

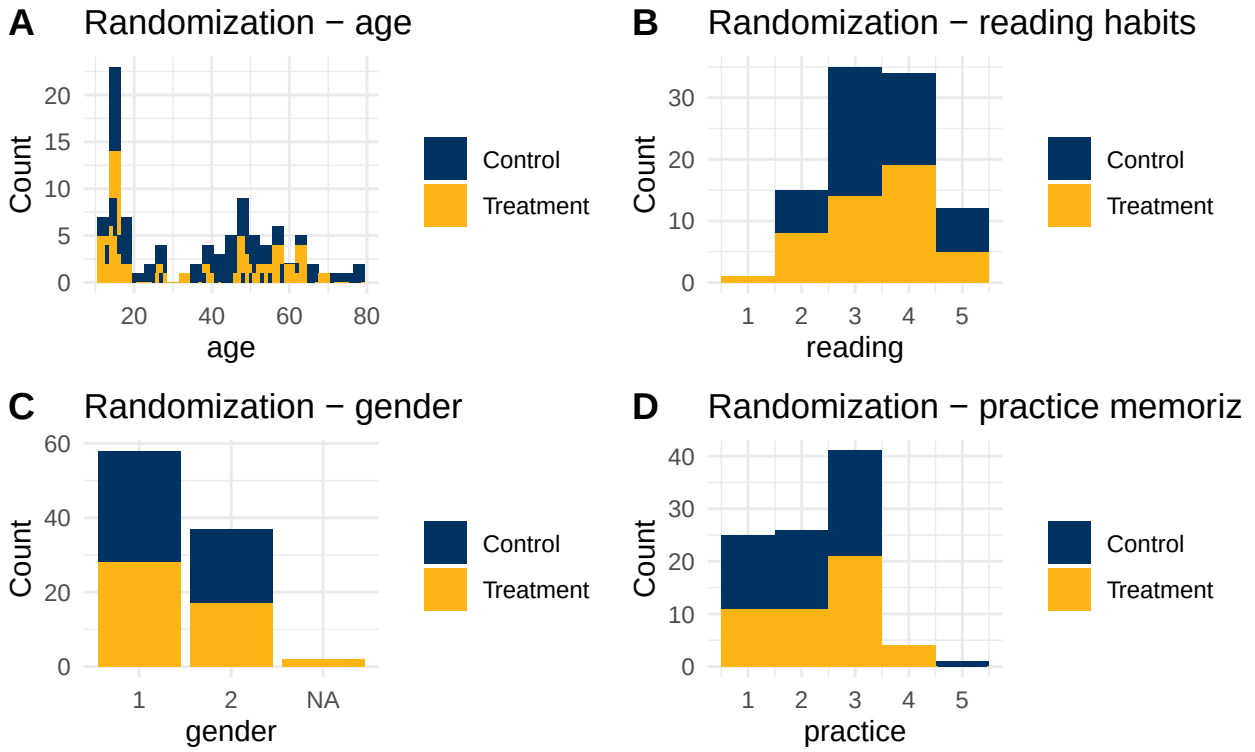
We used 2 methods to validate the randomization process worked. The first method is to create a regression model using covariate information collected to predict whether the participant was be assigned to control or treatment group. We created this regression model both for all participants who got assigned and for only participants who completed the survey. We concluded that the treatment assignment was random as none of the covariates had a significant predicting power. Details of the regression models are attached below.

Table 2:

	Dependent variable:
	treat
age	−0.007
	(0.011)
gender2	−0.318
	(0.504)
practice	0.126
	(0.247)
reading	−0.089
	(0.242)
prior_knowledge	0.090
	(0.104)
Constant	−0.332
	(1.065)
Observations	94
Log Likelihood	−64.045
Akaike Inf. Crit.	140.090
Note:	*p<0.1; **p<0.05; ***p<0.01

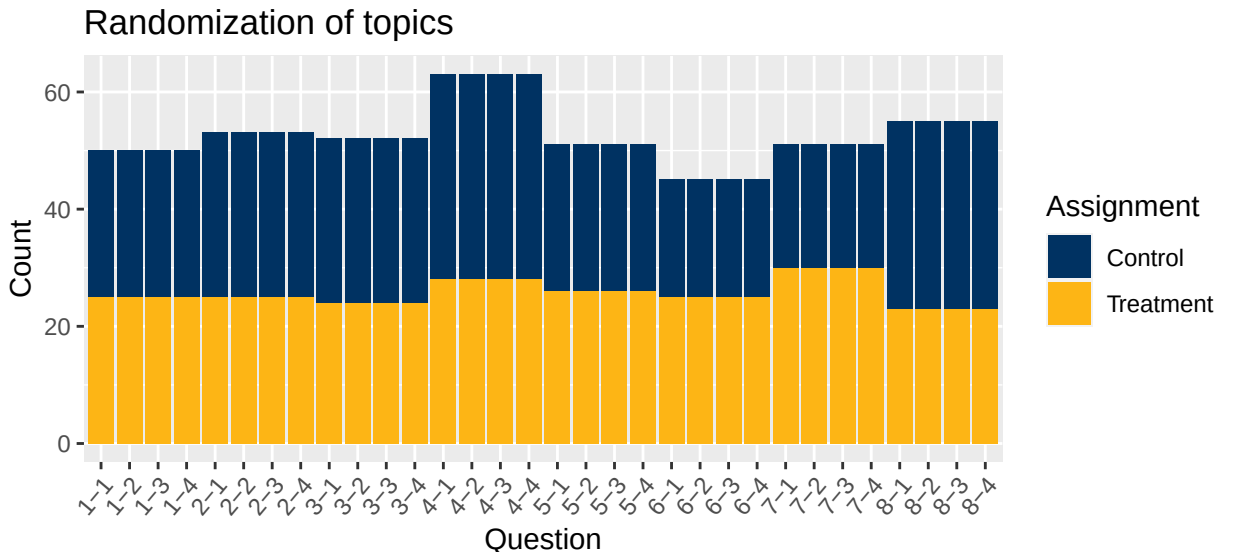
Visual check

The second method of validating randomization is to look at the distribution of control versus treatment assignment over participants' age, gender, self-reported memorization skill, and self-reported reading skill. As shown in the graphs below, the distribution of control versus treatment assignment are fairly even among all the participant groups. Therefore, we validated the randomization worked.



Did our question banks get randomized adequately? From a quick visual check, yes:

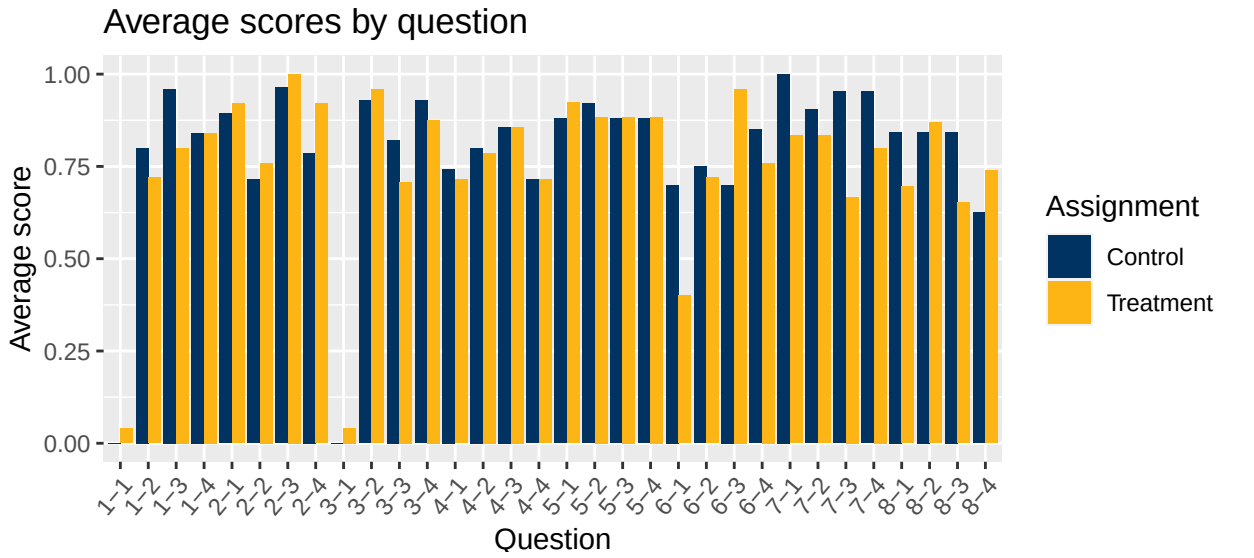
```
plot_rand_qs()
```



Scoring review

Were some questions harder than others?

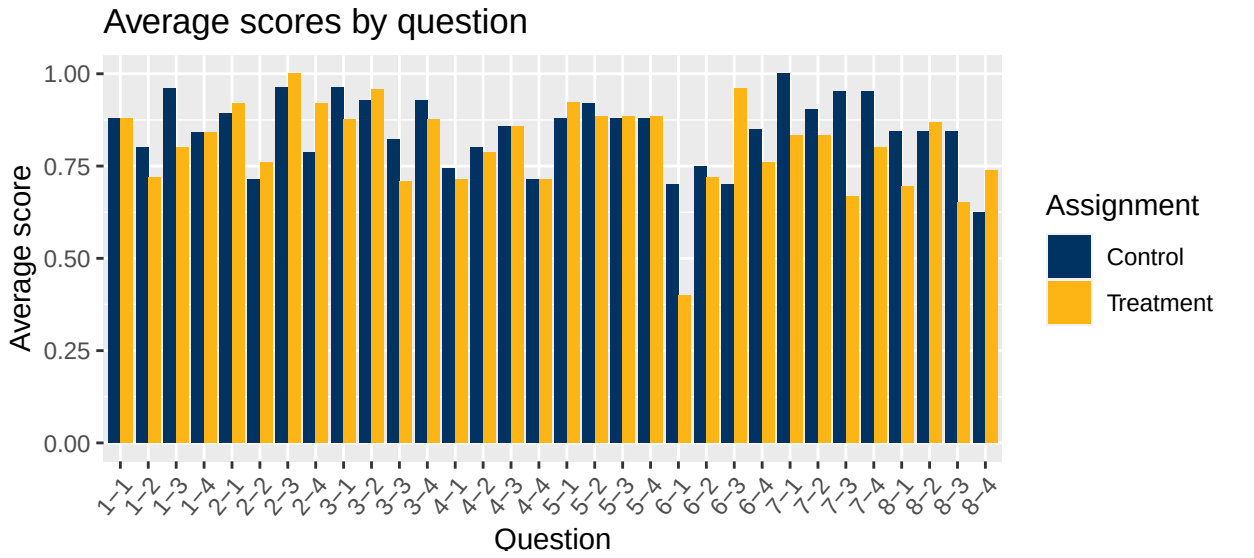
```
plot_qs()
```



This is odd. We accept that some of our questions were harder than others, but a score of almost 0 for item 1 question 1 and item 3 question 1 is unlikely. Also, subjects in the treatment group had a harder time by far with item 6 question 1.

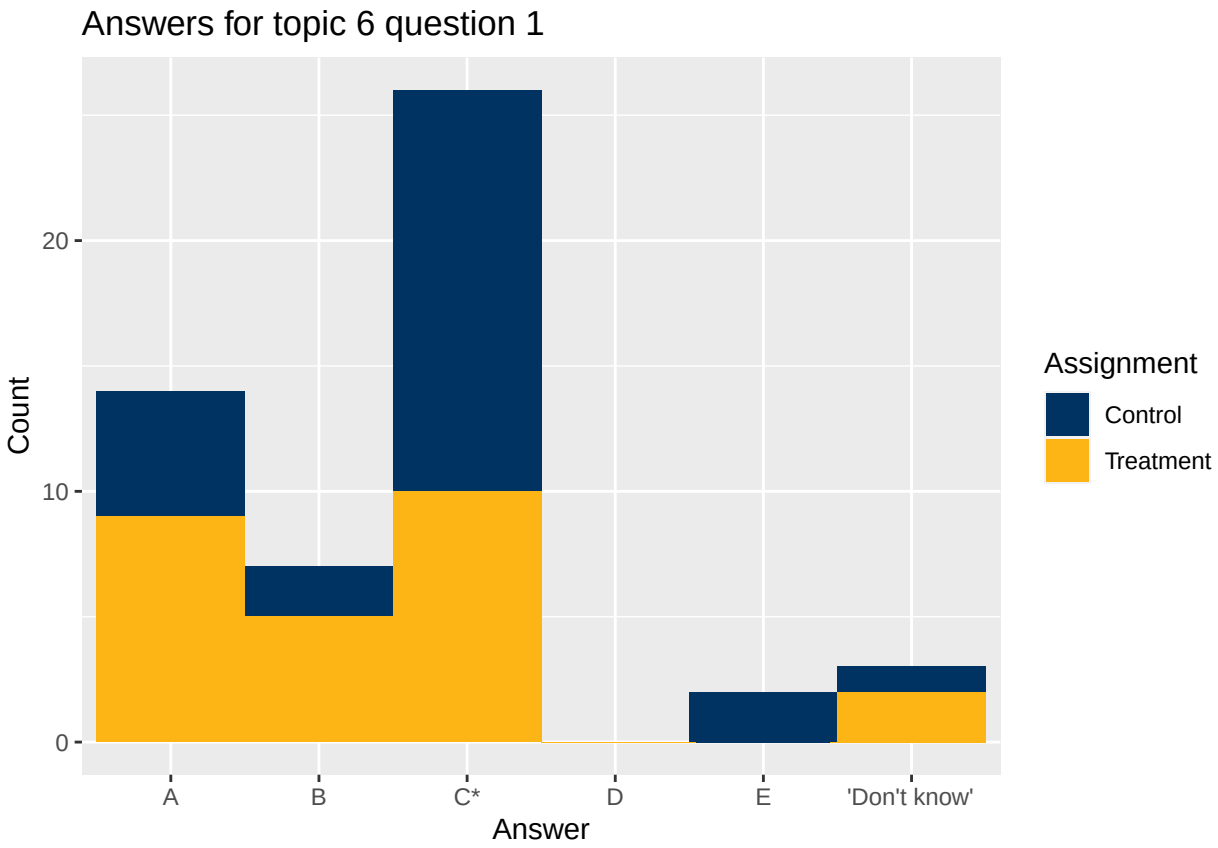
The first two items were analyzed and found incorrectly coded (details are in the appendix). Thankfully, our records include the actual answers, and so we could correct the scores:

```
correct_data() # makes a secondary CSV
dataset = load_data(TRUE)
dataset_long = load_data_long(TRUE)
plot_qs()
```



Topic number 6 question 1 is also discussed in the appendix. Compared to the other two, this one seems to simply have been a difficult question. Here is the distribution of answers:

```
plot_q_601()
```

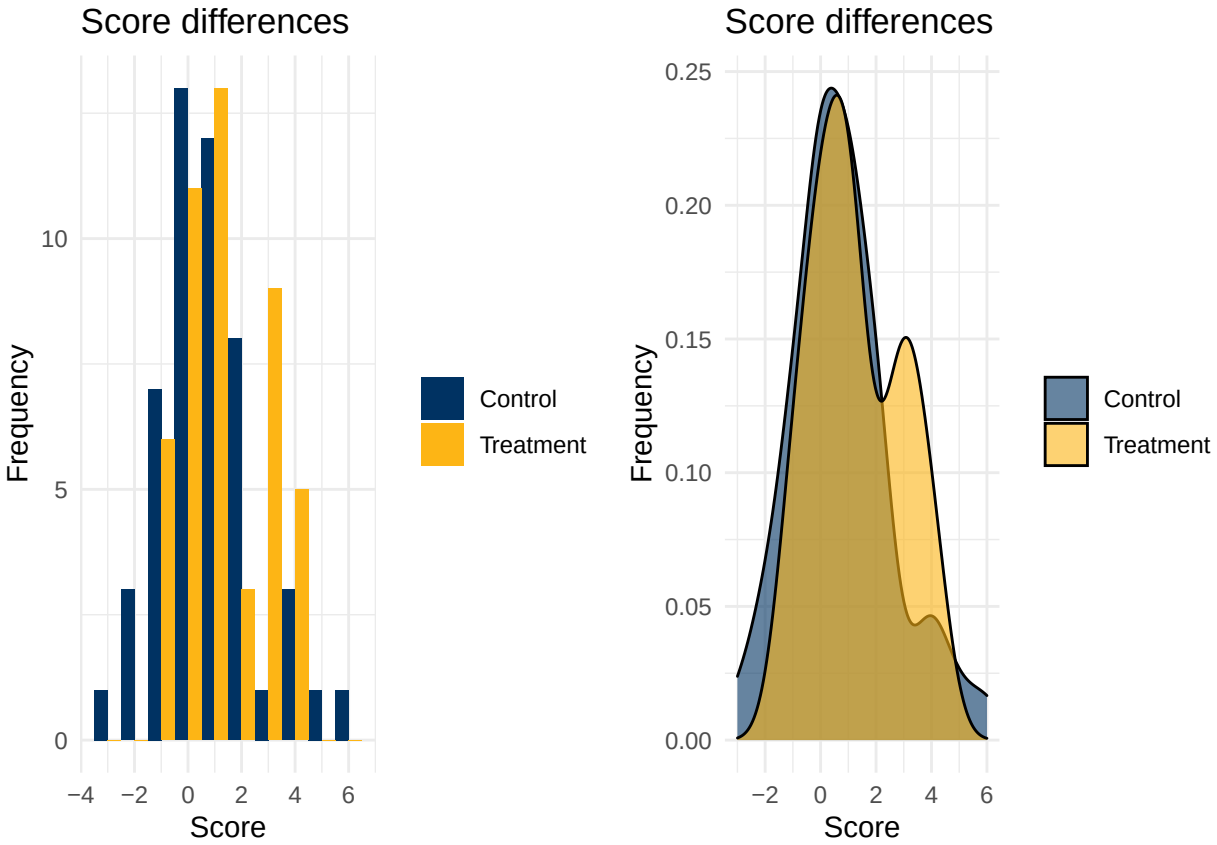


The correct answer was indeed the most frequent, so there is no real reason to doubt this data point.

The rest of this report uses the corrected data.

Glance at score distribution

As part of our EDA, we take a look at the distribution of differential (post-pre) scores between treatment and control. This is recorded in the variable `dataset`, in which we computed the differences in total scores between pre-tests and post-tests as the outcome:



We can see the distribution shifted to the positive in the treatment group, by about a little under a full score point. Let's see whether it holds up statistically.

Effect calculation

As discussed above, one primary method of effect calculation for us was looking at the differences between pre- and post-treatment-phase scores. In this spirit, we now compute the ATET throguh a regression of the difference on (1) only the treatment indicator `treat`:

$$y_{scoredelta} = \beta_0 + \beta_1(treat)$$

and (2) on the treatment indicator along with a number of covariates:

$$\begin{aligned}\Delta score =& \beta_0 + \beta_1(treat) + \beta_2(age) + \beta_3(priorknowledge) \\ &+ \beta_4(reading) + \beta_5(gender) + \beta_6(practice)\end{aligned}$$

The results are on the next page. The ATET of 0.5170 (0.331) for the simple model just misses statistically significance. In an earlier version of the results (slide deck), before the mis-coded results for topics 1 and 3 were discovered, the ATET was significant at the 5% level. Disappointingly, the inclusion of covariates did nothing to push the ATET of ATET 0.572 (0.346) over the significance level. Standard errors are cluster-robust.

Table 3: Regression Results with Clustered Standard Errors

	Dependent variable:	
	score	
	(1)	(2)
treat	0.517	0.572
	(0.331)	(0.346)
age		0.015
		(0.010)
prior_knowledge		−0.024
		(0.089)
reading		−0.091
		(0.174)
gender2		0.496
		(0.406)
practice		0.318
		(0.198)
Constant	0.760**	−0.235
	(0.265)	(0.828)
Observations	97	94
Adjusted R^2	0.013	0.007
Residual Std. Error	1.706 (df = 95)	1.735 (df = 87)
F Statistic	2.221 (df = 1; 95)	1.105 (df = 6; 87)

Alternative effect calculation

An alternative computation of effect was suggested by Prof. Reiley in the review of our results. Here, we use `dataset_long` in which we made an individual data point (row) for each answer for each test and subject. As a result, our outcome variable is now the probability of getting a correct score. This approach allows us to correlate prior knowledge and topics in a fine-grained way, and also correctly controls for questions of varying difficulty. Treatment is interacted on "test", which is the phase indicator (baseline/experiment):

$$\begin{aligned} logit(p_{correct}) = &\beta_0 + \beta_1(treat) + \beta_2(test) + \beta_3(treat : test) \\ &+ \beta_4(age) + \beta_5(priorknowledge) + \beta_6(reading) \\ &+ \beta_6(gender) + \beta_7(practice) \end{aligned}$$

Results are again on the next page. Using this approach, the ATET is now the interaction coefficient of 0.738 (0.333) for the simple model, and 0.766 (0.357) with all covariates. These results are significant at the 5% level. The interpretation of the ATET for this model is that of an odds-ratio: Odds of scoring a correct answer rose by 115.11 % (covariate model). In the context of our experiment, think of this as (in a back-of-the-envelope calculation) raising an already high 80% probability of getting the correct answer, to about 90%.

Table 4: Alternate Regression Results with Clustered Standard Errors

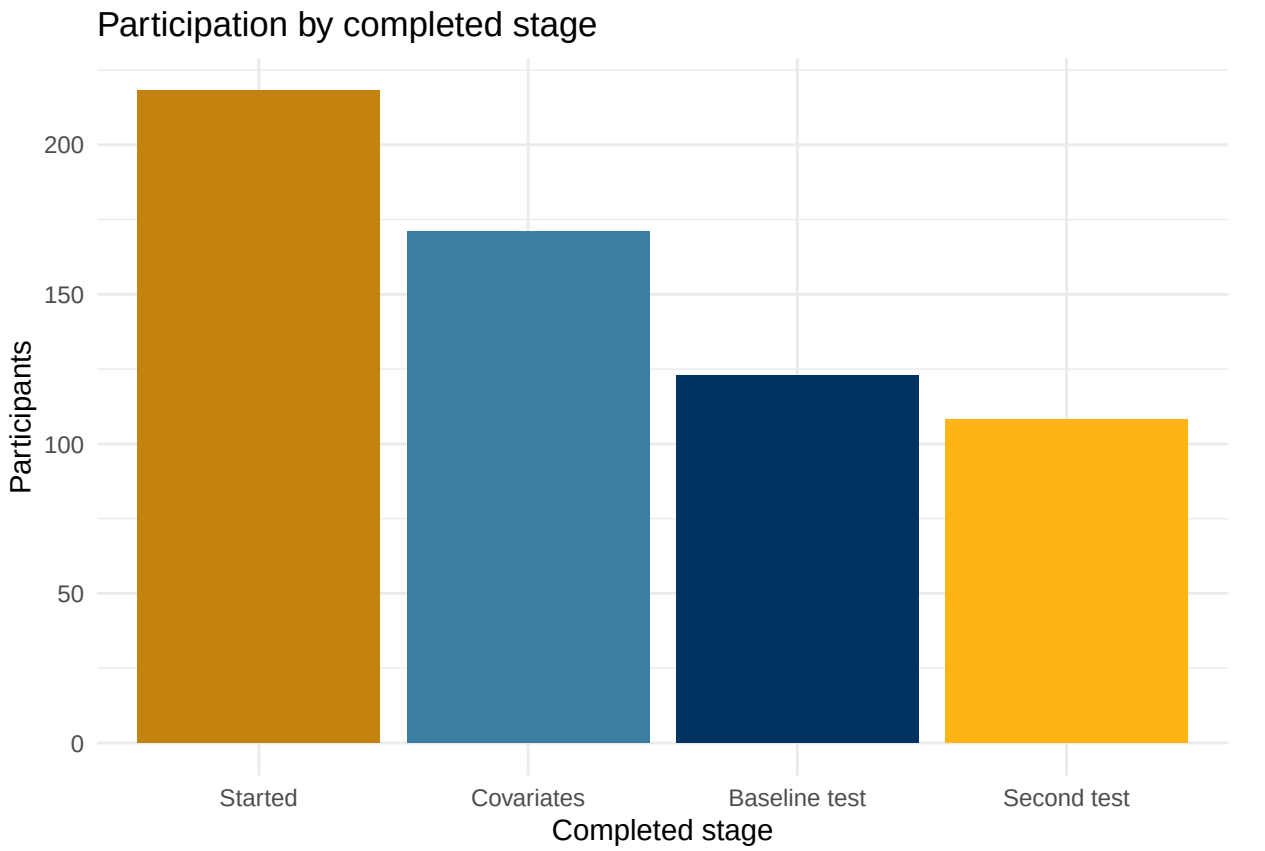
	Dependent variable:	
	score	
	(1)	(2)
treattrue	−0.483 (0.257)	−0.618* (0.270)
testexperiment	0.584* (0.253)	0.661* (0.265)
question1-2		−0.888 (0.514)
question1-3		−0.000 (0.561)
question1-4		−0.350 (0.497)
question2-1		−0.035 (0.652)
question2-2		−1.328* (0.563)
question2-3		1.682 (1.075)
question2-4		−0.589 (0.544)
question3-1		0.311 (0.613)
question3-2		0.626 (0.754)
question3-3		−1.027 (0.553)
question3-4		0.060 (0.661)
question4-1		−1.276* (0.570)
question4-2		−0.901 (0.472)
question4-3		−0.433 (0.551)
question4-4		−1.360** (0.481)
question5-1		0.086 (0.525)
question5-2		0.086 (0.527)
question5-3		−0.126 (0.598)
question5-4		−0.126 (0.539)
question6-1		−1.947*** (0.568)
question6-2		−0.997* (0.491)
question6-3		−0.280 (0.590)
question6-4		−0.600 (0.512)
question7-1		0.222 (0.668)
question7-2		−0.172 (0.579)
question7-3		−0.746 (0.584)
question7-4		−0.172 (0.499)
question8-1		−0.963 (0.556)
question8-2		−0.446 (0.570)
question8-3		−1.072* (0.479)
question8-4		−1.553** (0.530)
knowledge		0.080 (0.146)
age		−0.008 (0.005)
gender1		−0.202 (0.273)
gender2		0.616* (0.288)
practice		0.090 (0.152)
reading		−0.046 (0.140)
treattrue:testexperiment	0.738* (0.333)	0.766* (0.357)
Constant	1.429*** (0.202)	2.226** (0.791)
Observations	1,680	1,676
Akaike Inf. Crit.	1,501.871	1,470.937

Discussion of experiment

Attrition

We faced an attrition issue in this experiment: 22% of participants dropped between pressing "start" and submitting covariates, a further 28% of the remaining participants dropped between submitting covariates and completing the first part of the test. 37% of participants dropped between submitting covariates and the end of the test:

```
plot_participation()
```



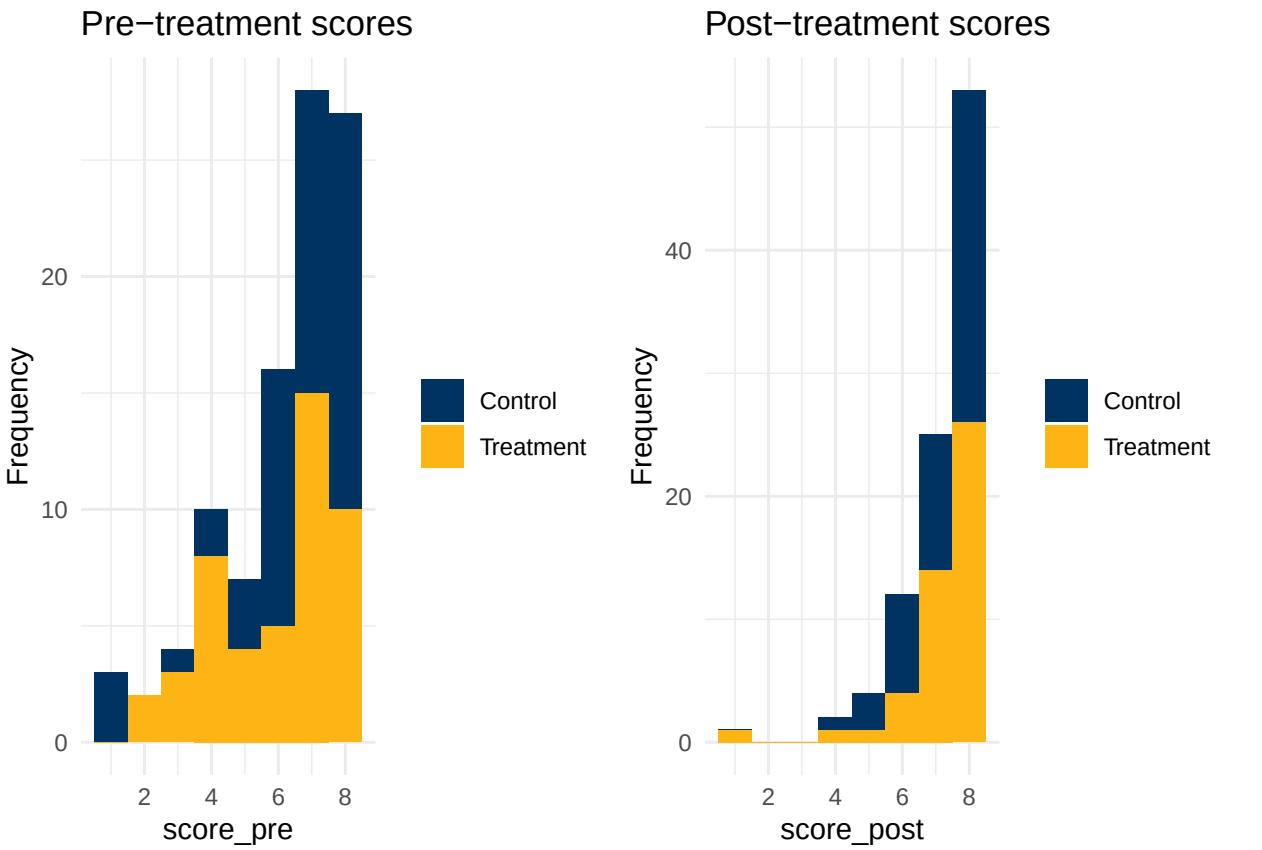
Potential reasons for attrition include: The test took a little over 10 minutes which is too long for voluntary participants, the content of the passages were not interesting to the participants, and technical difficulties such as the website is not compatible with mobile users, and pressing the "go back" button on browser will confuse the server logic and disqualify the test.

From our covariate balance check earlier, which was done using only subject which had completed the whole experiment, we have no indication to be worried about an attrition imbalance between treatment and control. Asymmetries in covariates across attrition stages are possible, but since this is not a long-term study and we are not comparing results between potentially different groups, the level of attrition only hurts our generalizability. Subjects of different abilities - or other varying attributes - might have dropped off before the actual test, meaning our sample set might not reflect any relaistic slice of any general population.

Calibration of questions / general scoring

A look at the pre- and post-treatment scores for only the treatment group is instructive:

```
plot_scores()
```



Subjects scored well on our questions to begin with, there wasn't all that much room for improvement. This suggests that our content was too easy to memorize, or that our multiple-choice questions were too easy to answer.

Suggestions for future experiments

- Pre-study: Extend pre-study to around 2 to 3 weeks so we can collect enough results for better estimation of expected effect size for power calculation. We were unable to do this in the experiment because of the time constraints and discovering testing website code error in the later stage of pre-study.
- Increase test question difficulty level: We found that regardless of treatment or control, people were generally able to score pretty well in the questions we posed them. We believe that a longer pre-study period would allow for future experimenters to better assess question difficulty and adjust accordingly.
- Shorter content: Some of the memorization questions were longer than others in some cases, and we received feedback from study participants that the study was too long. This was also reflected within the attrition in our survey. We suggest that future experiments take into consideration how difficult the memorization task actually is.
- Hard-code page flow: Given the time and resource constraint of this experiment, the testing website cannot record and display correctly if the participants press the "go back" button on the browser. Ironically, this is partly because we made a very serious effort to track where the user is in the process, and to react correctly to "back arrow" events. However, a couple of years ago, the authors of Google Chrome decided to make the back arrow to do precisely what the users expect it to do, and implemented a quite extraordinary level of machinery to ensure that the server *cannot* detect whether someone did this. It might be possible to implement a more robust server logic, but it seemed improbable and was certainly not possible given the time line we were on. A simpler website with more hard-coding of logic into the HTML/JS client side of the logic might prove to be more robust.
- Conduct this experiment in a supervised room: Having participants take this study in a monitored room setting for 10 minutes can minimize the potential collaboration which results in inflated or inaccurate test scores. Having participants in a room can also prevent attrition.
- Random order of baseline and treatment passages: Current design of the treatment group displays baseline passage in part one of the test and treatment passage in part two of the test. Because reading and memorization require mental concentration, a potential concern of the current design is that participants might lose attention when they are in part two of the test. Therefore randomizing the order of baseline and treatment passage will remove the effect of concentration level on test results.

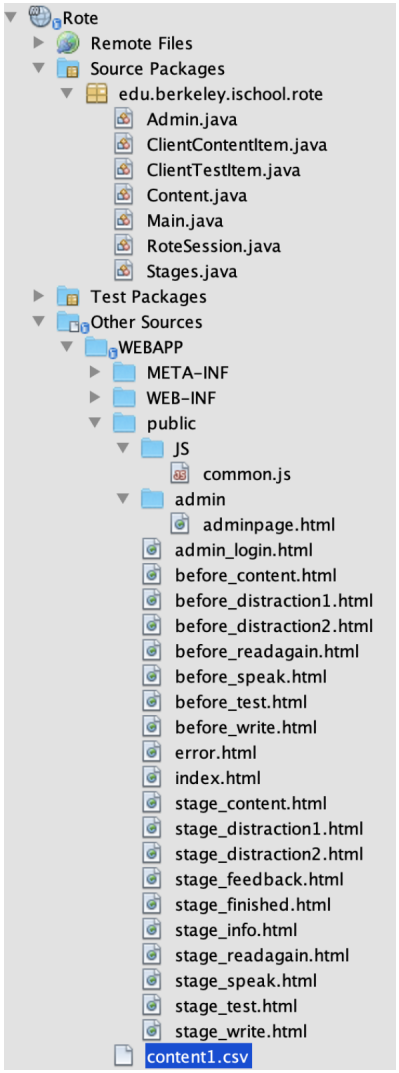
Appendix

Code structure

Rote is a Java project utilizing SparkJava as a bare-bones web-server framework. The client consists of plain HTML and JavaScript with a minimal dose of bootstrap CSS.

The project ran locally on a Tomcat server on MacOS for development and debugging, and on a similar Tomcat setup inside a Microsoft Azure Linux box for the real experiment. All code was developed in Apache NetBeans 11.3.

The repository is available at https://github.com/gunnarmein-ucbischool/w241-rote.



Logs

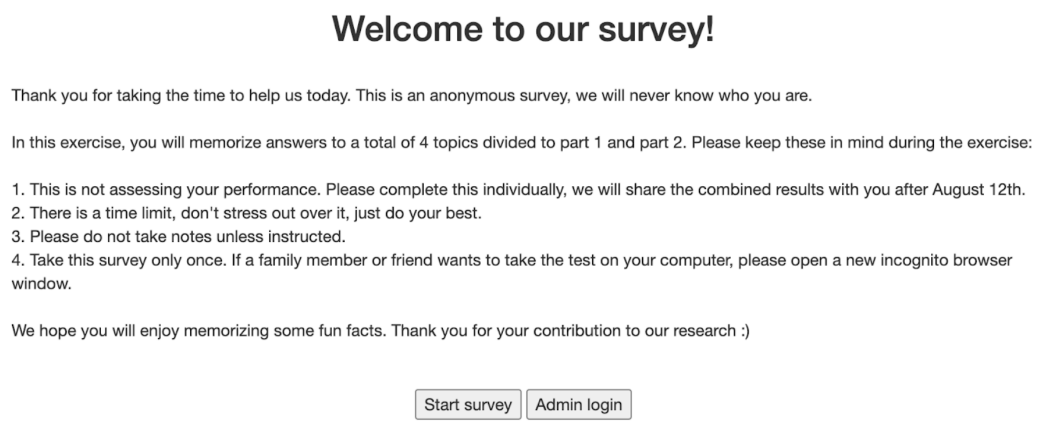
The general operations log, covariates log, written submissions log, and test results log can be viewed at:

https://drive.google.com/drive/folders/1t1wtW139ow35JUcuspJsWoYNjJzNNaK_?usp=sharing

It can also be found in the "analysis" folder of the GitHub repo.

Screen shots

Welcome



Welcome to our survey!

Thank you for taking the time to help us today. This is an anonymous survey, we will never know who you are.

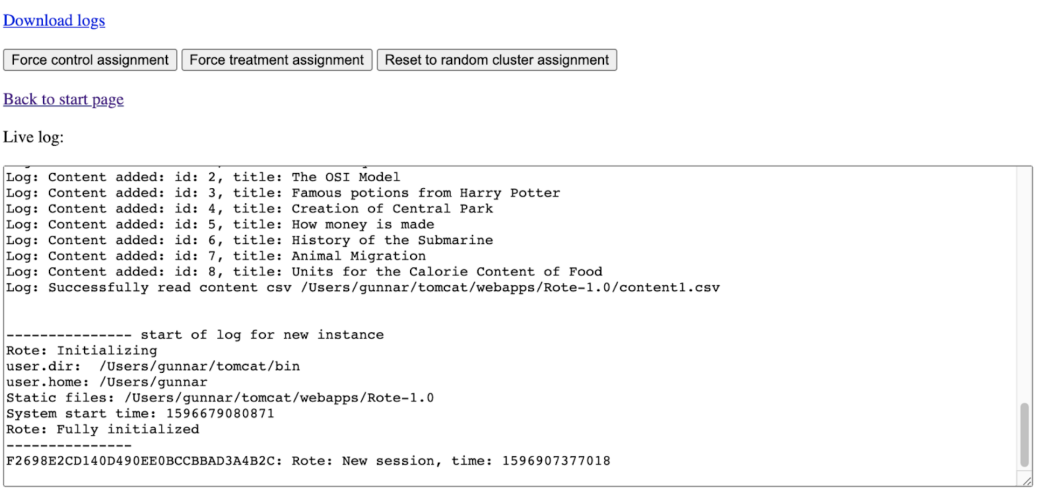
In this exercise, you will memorize answers to a total of 4 topics divided to part 1 and part 2. Please keep these in mind during the exercise:

1. This is not assessing your performance. Please complete this individually, we will share the combined results with you after August 12th.
2. There is a time limit, don't stress out over it, just do your best.
3. Please do not take notes unless instructed.
4. Take this survey only once. If a family member or friend wants to take the test on your computer, please open a new incognito browser window.

We hope you will enjoy memorizing some fun facts. Thank you for your contribution to our research :)

Start survey | Admin login

Admin UI



Download logs

Force control assignment | Force treatment assignment | Reset to random cluster assignment

Back to start page

Live log:

```
Log: Content added: id: 2, title: The OSI Model
Log: Content added: id: 3, title: Famous potions from Harry Potter
Log: Content added: id: 4, title: Creation of Central Park
Log: Content added: id: 5, title: How money is made
Log: Content added: id: 6, title: History of the Submarine
Log: Content added: id: 7, title: Animal Migration
Log: Content added: id: 8, title: Units for the Calorie Content of Food
Log: Successfully read content csv /Users/gunnar/tomcat/webapps/Rote-1.0/content1.csv


--------------- start of log for new instance
Rote: Initializing
user.dir:  /Users/gunnar/tomcat/bin
user.home: /Users/gunnar
Static files: /Users/gunnar/tomcat/webapps/Rote-1.0
System start time: 1596679080871
Rote: Fully initialized
---------------
F2698E2CD140D490EE0BCCBBAD3A4B2C: Rote: New session, time: 1596907377018
```

Gathering covariates

DON'T GO BACK!!! The test will not work and your results will be lost

About You

Please tell us a tiny bit about yourself. This is all the information we will store, nothing can be associated with you. The more you can tell us, the more meaning our research will have.

Age:

Gender:

- ○ Male
- ○ Female
- ○ Other
- ○ Prefer not to say

How much do you read?

- ○ Not at all
- ○ Little
- ○ Some
- ○ A substantial amount
- ○ All the time

How much practice do you have with memorization?

- ○ None
- ○ Little
- ○ Some
- ○ A substantial amount
- ○ Expert

How much knowledge do you have about the topic "The OSI Model"?

- ○ None

Intermission (one example only)

DON'T GO BACK!!! The test will not work and your results will be lost

Coming up: Memorization

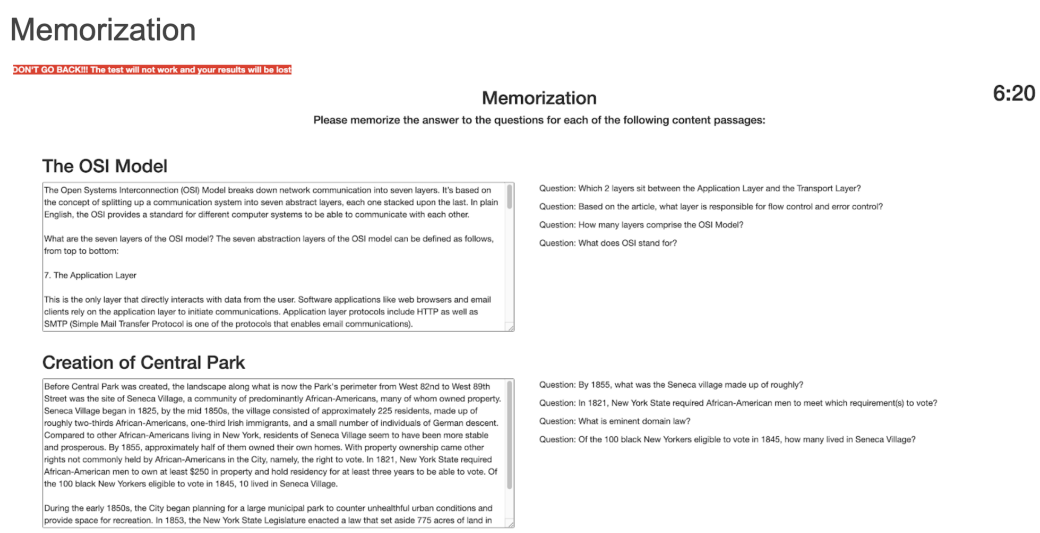
Thank you!

You will see 2 passages of text, with 5 questions each. The objective is to be able to answer the questions.
Glance at the questions, read the passages, read the questions again.
Memorize the answers.

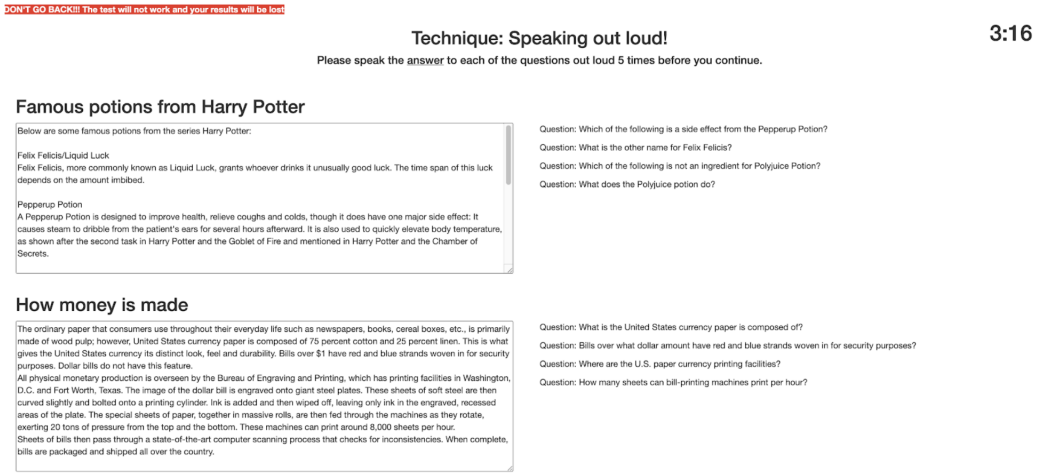
Press "Continue" when ready ...

Continue

Memorization



Treatment (part 1)



Treatment (part 2)

DON'T GO BACK!!! The test will not work and your results will be lost

Technique - Write it down!

3:17

Please look at the content again, and write down your brief answers in the boxes:

Famous potions from Harry Potter

Below are some famous potions from the series Harry Potter:

Felix Felicis/Liquid Luck
Felix Felicis, more commonly known as Liquid Luck, grants whoever drinks it unusually good luck. The time span of this luck depends on the amount imbibed.

Pepperup Potion
A Pepperup Potion is designed to improve health, relieve coughs and colds, though it does have one major side effect: It causes steam to dribble from the patient's ears for several hours afterward. It is also used to quickly elevate body temperature, as shown after the second task in Harry Potter and the Goblet of Fire and mentioned in Harry Potter and the Chamber of Secrets.

Question: Which of the following is a side effect from the Pepperup Potion? Answer:

Question: What is the other name for Felix Felicis? Answer:

Question: Which of the following is not an ingredient for Polyjuice Potion? Answer:

Question: What does the Polyjuice potion do? Answer:

How money is made

The ordinary paper that consumers use throughout their everyday life such as newspapers, books, cereal boxes, etc., is primarily made of wood pulp; however, United States currency paper is composed of 75 percent cotton and 25 percent linen. This is what gives the United States currency its distinct look, feel and durability. Bills over $1 have red and blue strands woven in for security purposes. Dollar bills do not have this feature.
All physical monetary production is overseen by the Bureau of Engraving and Printing, which has printing facilities in Washington, D.C. and Fort Worth, Texas. The image of the dollar bill is engraved onto giant steel plates. These sheets of soft steel are then curved slightly and bolted onto a printing cylinder. Ink is added and then wiped off, leaving only ink in the engraved, recessed areas of the plate. The special sheets of paper, together in massive rolls, are then fed through the machines as they rotate, exerting 20 tons of pressure from the top and the bottom. These machines can print around 8,000 sheets per hour.
Sheets of bills then pass through a state-of-the-art computer scanning process that checks for inconsistencies. When complete, bills are packaged and shipped all over the country.

Question: What is the United States currency paper is composed of? Answer:

Question: Bills over what dollar amount have red and blue strands woven in for security purposes? Answer:

Question: Where are the U.S. paper currency printing facilities? Answer:

Question: How many sheets can bill-printing machines print per hour? Answer:

Test

DON'T GO BACK!!! The test will not work and your results will be lost

Time to put your memory to the test!

6:24

The OSI Model:

Which 2 layers sit between the Application Layer and the Transport Layer?
- ○ Presentation Layer, Session Layer
- ○ Data Link Layer, Presentation Layer
- ○ Transmission Layer, Network Layer
- ○ DataLink Layer, Physical Layer
- ○ None of the above
- ○ I don't remember

Based on the article, what layer is responsible for flow control and error control?
- ○ Presentation Layer
- ○ Data Link Layer
- ○ Transport Layer
- ○ The Network Layer
- ○ The Session Layer
- ○ I don't remember

How many layers comprise the OSI Model?
- ○ 5
- ○ 6
- ○ 7
- ○ 8
- ○ 9
- ○ I don't remember

What does OSI stand for?
- ○ Operating System Interface
- ○ Open System Internet
- ○ Open Systems Interconnection
- ○ Operating Systems Interconnect
- ○ None of the above
- ○ I don't remember

Creation of Central Park:

By 1855, what was the Seneca village made up of roughly?
- ○ two-thirds Irish, one-third African-Americans immigrants
- ○ two-thirds African-Americans, one-third Irish immigrants
- ○ two-thirds African-Americans, one-third German immigrants
- ○ two-thirds German, one-third African-Americans immigrants
- ○ None of the above
- ○ I don't remember

In 1821, New York State required African-American men to meet which requirement(s) to vote?
- ○ Own property and hold residency for at least five years
- ○ Own at least $250 in property and hold residency for at least three years
- ○ Own at least $250 in property and have no criminal record
- ○ Have paid at least $250 in property tax and hold residency for at least five years
- ○ None of the above
- ○ I don't remember

What is eminent domain law?
- ○ The law allows the government to take private land for public use with compensation paid to the landowner
- ○ The law allows the government to take private land for public use by providing alternative property to the landowner
- ○ The law allows the government to take private land for public use by providing tax credit to the landowner
- ○ The law allows the government to take private land for public use by sharing ownership of the new establishment with the landowner
- ○ None of the above
- ○ I don't remember

Of the 100 black New Yorkers eligible to vote in 1845, how many lived in Seneca Village?
- ○ 25
- ○ 15
- ○ 10
- ○ 45
- ○ None of the above
- ○ I don't remember

Feedback

DON'T GO BACK!!! The test will not work and your results will be lost

Feedback

Do you have any feedback on your experience? Anything we should improve?

(500 characters or fewer, please)

Submit and finish

Investigation of low-scoring questions

Topic 1: "The Baroque Period", time limit 150 s

The Baroque period was anticipated before 1600, although that date remains as a convenient marker for the start of the period. It is a period of dramatic expression, of a vigorous, highly ornamented art. An era of absolute monarchies, each court had its own group of musicians, both vocal and instrumental. **The Doctrine of Affections of the Baroque relates to the portrayal of emotions through music.** A recognized musical vocabulary expressed certain emotions. Within this overall context, composers used musical techniques to vividly describe the meaning of the words. Rising passages are found at words such as ""resurrection,"" ""heaven,"" etc. Descending passages were used for such phrases as ""to the depths"" and ""descended into hell."" It was also a period of scientific discovery and reasoning. New findings in the sciences were vitally important to knowledge on the continent. Although we are usually concerned with Baroque music of the continent, it was during this period that the settlement of the New World began. Rhythmic energy, coupled with a strong melodic thrust, makes performances of Baroque music appealing and satisfying to musicians, both amateur and professional.

Characteristics of Baroque choral music include:

1. Vertical structure rather than linear
2. Major-minor tonality established
3. Figured bass
4. Outer voice polarity—soprano melody over a figured bass
5. New counterpoint—subordinate to the harmony
6. Concertato style important
7. Terraced dynamics
8. Instruments influenced texture
9. Form determined by musical considerations
10. Doctrine of Affections
11. Virtuosity and improvisation are important elements
12. Steady pulsating rhythm—barlines introduced

Although 1600 is generally acknowledged as the beginning of the Baroque period, Renaissance characteristics are found long after that. The two styles, stile antico and stile moderno existed side by side, particularly in the early Baroque. Composers often wrote in both styles; consequently, a conductor must look beyond the name and dates of a composer to determine the style of the music.

Question 1 "Based on the article, what does the""Doctrine of Affections"" relate to?"

- A: Love and empathy
- B: portrayal of emotions through music
- C: scientific reasoning
- D: Strong Melodic thrusts
- E: None of the above

The answer marked correct in content file is "E: None of the above". However, reading the text quickly reveals that "B" should have been marked as correct instead.

Topic 3 - "Famous potions from Harry Potter", time limit 130s

Below are some famous potions from the series Harry Potter:

Felix Felicis/Liquid Luck Felix Felicis, more commonly known as Liquid Luck, grants whoever drinks it unusually good luck. The time span of this luck depends on the amount imbibed.

Pepperup Potion A Pepperup Potion is designed to improve health, relieve coughs and colds, though it does have **one major side effect: It causes steam to dribble from the patient's ears** for several hours afterward. It is also used to quickly elevate body temperature, as shown after the second task in Harry Potter and the Goblet of Fire and mentioned in Harry Potter and the Chamber of Secrets.

Polyjuice Potion The Polyjuice Potion allows the drinker to assume the appearance of someone else for an hour or more depending on the quantity. The potion only causes a physical and voice transformation of the drinker, but clothing is not affected. Its ingredients include fluxweed, knotgrass, lacewing flies, leeches, powdered Bicorn horn, and shredded Boomslang skin. The final component is a bit of the individual to be impersonated; strands of hair are most often used for this purpose.

Skele-Gro Skele-Gro is a medicinal potion that can regrow missing or removed bones, though it tastes terrible and the process is very slow and extremely painful. In Chamber of Secrets, a bewitched Bludger breaks Harry's arm while he plays Quidditch, and Gilderoy Lockhart, the incompetent Defence Against the Dark Arts teacher, accidentally removes his bones instead of mending them. As a result, Harry takes a dose of the potion and spends the night in the hospital wing.

Veritaserum Veritaserum is a very powerful truth potion. The name ""Veritaserum"" derives from the Latin word Veritas, meaning truth. Three drops of this potion are all that is needed to force anyone to respond to any question with the truth.

Question 1 "Which of the following is a side effect from the Pepperup Potion?"

- A: Causes snot to dribble from the nose
- B: Causes you to be really energetic
- C: Causes steam to dribble out the ears
- D: Gives you indigestion
- E: None of the above

The answer marked as correct in our content file is once again "E: None of the above". Instead, "C" should have been marked as correct.

Topic 6: "History of the Submarine", time limit 70 s

The submarine's origins trace back as far as 332 B.C., when it was said that Alexander the Great was lowered into the ocean inside a glass barrel so he could study the life of fish. This concept was then pushed aside until it reappeared again in 1578 in a publication called Inventions or Devises, written by William Bourne. Bourne described the idea of submerging a boat by altering its volume. Around 1620, Cornelis Jacobszoon Drebbel, a Dutch engineer, covered a rowboat in greased leather. A dozen oarsmen row beneath the surface of the Thames, breathing through snorkel tubes. He named his boat Drebbel I, and it is considered by many to have been the first functioning submarine. In 1863, the French Plongeur ('Diver') which was powered by engines run on compressed air, became the first submarine that did not rely on human propulsion for momentum. The real breakthrough, and the birth of the modern submarine, came courtesy of John Phillip Holland, towards the end of the 19th century. He became the first designer to successfully unite three new pieces of technology - the electric motor, the electric battery, and the internal combustion engine - to create the first recognizably modern submarine, USS Holland. Submarines were first widely used during World War I, and are now used in many navies large and small.

Question 1 "Which submarine was the first recognizably modern submarine?"

- A: Drebbel I
- B: Plongeur ('Diver')
- C: USS Holland
- D: Nautilus
- E: None of the above

The correct answer was marked in our content file as "C", and as one can see from the text, this is indeed the correct answer. However, it is a more subtle question than others in our content set.

Full code for wrangling, analysis and graphs

```
library(dplyr)
library(anytime)
library(lubridate)
library(tidyr)
library(ggplot2)
library(ggpubr)
library(stargazer)
library(sandwich)
library(lmtest)
library(png)
library(grid)

load_data = function(corrected=FALSE){
  # import files
  rote_cov_original <- read.csv(file="rote_cov.csv", sep=',') #covariates
  if (corrected) {
    rote_test_original <- read.csv(file="rote_test_corrected.csv", sep=',') #test file
  } else {
    rote_test_original <- read.csv(file="rote_test.csv", sep=',') #test file
  }

  # create copies
  rote_cov <- rote_cov_original
  rote_test <- rote_test_original

  # convert unix epoch time to datetime
  # cov
  rote_cov$session_start_time <- as.POSIXct(rote_cov$session_start_time/1000, origin="1970-01-01")
  rote_cov$cov_submit_time <- as.POSIXct(rote_cov$cov_submit_time/1000, origin="1970-01-01")

  # test
  rote_test$session_start_time <- as.POSIXct(rote_test$session_start_time/1000, origin="1970-01-01")
  rote_test$test_submit_time <- as.POSIXct(rote_test$test_submit_time/1000, origin="1970-01-01")
  rote_test$test_time <- rote_test$test_submit_time - rote_test$session_start_time

  # remove rows with any NAs in the answers(indicates they did not complete the test)
  rote_test <- rote_test %>% drop_na("a11", "a12", "a13", "a14", "a21", "a22", "a23","a24")

  rote_test_baseline <- rote_test %>% filter(test == 'baseline')
  rote_test_experiment <- rote_test %>% filter(test == 'experiment') %>%
    select('session_id','item_id1','a11','a12', 'a13', 'a14', 'a21', 'a22', 'a23','a24',
           'item_id2', 'c11','c12', 'c13', 'c14', 'c21', 'c22', 'c23','c24')

  rote_test <- inner_join(rote_test_baseline, rote_test_experiment, by="session_id" )


  # remove rows with missing covariates
  # convert age, remove outliers for age
  suppressWarnings(
    rote_cov$age <- ifelse(as.numeric(as.character(rote_cov$age)) > 100,
                           NA, as.numeric(as.character(rote_cov$age))))
  rote_cov <- rote_cov %>% drop_na("age")
```

```r
#fix gender
rote_cov$gender <- ifelse(rote_cov$gender == 'other', NA, rote_cov$gender)


# on test file, establish if testee answer matches actual answer.
# if it matches, set var to 1, else 0.
# notation will be o11, o12, etc (o is for outcome)
# x = baseline
# y = treatment
rote_test$o11x <- ifelse(rote_test$a11.x == rote_test$c11.x, 1, 0)
rote_test$o12x <- ifelse(rote_test$a12.x == rote_test$c12.x, 1, 0)
rote_test$o13x <- ifelse(rote_test$a13.x == rote_test$c13.x, 1, 0)
rote_test$o14x <- ifelse(rote_test$a14.x == rote_test$c14.x, 1, 0)
rote_test$o21x <- ifelse(rote_test$a21.x == rote_test$c21.x, 1, 0)
rote_test$o22x <- ifelse(rote_test$a22.x == rote_test$c22.x, 1, 0)
rote_test$o23x <- ifelse(rote_test$a23.x == rote_test$c23.x, 1, 0)
rote_test$o24x <- ifelse(rote_test$a24.x == rote_test$c24.x, 1, 0)

rote_test$o11y <- ifelse(rote_test$a11.y == rote_test$c11.y, 1, 0)
rote_test$o12y <- ifelse(rote_test$a12.y == rote_test$c12.y, 1, 0)
rote_test$o13y <- ifelse(rote_test$a13.y == rote_test$c13.y, 1, 0)
rote_test$o14y <- ifelse(rote_test$a14.y == rote_test$c14.y, 1, 0)
rote_test$o21y <- ifelse(rote_test$a21.y == rote_test$c21.y, 1, 0)
rote_test$o22y <- ifelse(rote_test$a22.y == rote_test$c22.y, 1, 0)
rote_test$o23y <- ifelse(rote_test$a23.y == rote_test$c23.y, 1, 0)
rote_test$o24y <- ifelse(rote_test$a24.y == rote_test$c24.y, 1, 0)

rote_test$score_pre <- rote_test$o11x + rote_test$o12x + rote_test$o13x +
  rote_test$o14x + rote_test$o21x + rote_test$o22x + rote_test$o23x + rote_test$o24x
rote_test$score_post <- rote_test$o11y + rote_test$o12y + rote_test$o13y +
  rote_test$o14y + rote_test$o21y + rote_test$o22y + rote_test$o23y + rote_test$o24y

rote_test$score =  rote_test$score_post - rote_test$score_pre

#on covariates file, sum knowledge scores together
rote_cov$knowledge_cov_pre <- rote_cov$knowledge1 + rote_cov$knowledge2
rote_cov$knowledge_cov_post <- rote_cov$knowledge3 + rote_cov$knowledge4
rote_cov$prior_knowledge <- rote_cov$knowledge_cov_post + rote_cov$knowledge_cov_pre

#convert necessary columns to boolean
rote_test$treat <- ifelse(rote_test$treat == "false", 0 , 1)
rote_cov$treat <- ifelse(rote_cov$treat == "false", 0 , 1)

#get distinct ids
rote_test <- rote_test[!duplicated(rote_test$session_id),]
rote_cov <- rote_cov[!duplicated(rote_cov$session_id),]

#inner join 2 datasets
dataset <- inner_join(rote_test, rote_cov, by="session_id" )

dataset$treat <- dataset$treat.x
dataset$cluster <- dataset$cluster.x
dataset$gender <- as.factor(dataset$gender)
```

```
  dataset <- dataset %>%
    filter %>%
    select("session_id", "score", "score_pre", "score_post", "cluster", "gender", "age",
           "prior_knowledge", "treat","reading", "practice")

}

load_data_long = function(corrected=FALSE){
 # import files
  rote_cov_original <- read.csv(file="rote_cov.csv", sep=',') #covariates
  if (corrected) {
    rote_test_original <- read.csv(file="rote_test_corrected.csv", sep=',') #test file
  } else {
    rote_test_original <- read.csv(file="rote_test.csv", sep=',') #test file
  }

  # create copies
  rote_cov <- rote_cov_original
  rote_test <- rote_test_original

  # remove rows with any NAs in the answers(indicates they did not complete the test)
  rote_test <- rote_test %>% drop_na("a11", "a12", "a13", "a14", "a21", "a22", "a23","a24")

  # remove rows with missing covariates
  # convert age, remove outliers for age
  suppressWarnings(
    rote_cov$age <- ifelse(as.numeric(as.character(rote_cov$age)) > 100,
                           NA, as.numeric(as.character(rote_cov$age))))
  rote_cov <- rote_cov %>% drop_na("age")

  #fix gender
  rote_cov$gender <- ifelse(rote_cov$gender == 'other', NA, rote_cov$gender)


  # on test file, establish if testee answer matches actual answer.
  # if it matches, set var to 1, else 0.
  # notation will be o11, o12, etc (o is for outcome)
  # x = baseline
  # y = treatment
  rote_test$o11 <- ifelse(rote_test$a11 == rote_test$c11, 1, 0)
  rote_test$o12 <- ifelse(rote_test$a12 == rote_test$c12, 1, 0)
  rote_test$o13 <- ifelse(rote_test$a13 == rote_test$c13, 1, 0)
  rote_test$o14 <- ifelse(rote_test$a14 == rote_test$c14, 1, 0)
  rote_test$o21 <- ifelse(rote_test$a21 == rote_test$c21, 1, 0)
  rote_test$o22 <- ifelse(rote_test$a22 == rote_test$c22, 1, 0)
  rote_test$o23 <- ifelse(rote_test$a23 == rote_test$c23, 1, 0)
  rote_test$o24 <- ifelse(rote_test$a24 == rote_test$c24, 1, 0)

  # now flip to long form
  rote_test = rote_test %>%
    gather("question", "score", o11:o24) %>%
    mutate(question = as.integer(substr(question,2,3))) %>%
    mutate(item_id = ifelse(question >= 20, item_id2, item_id1)) %>%
```

```
    mutate(treat = as.factor(ifelse(treat == "false", 0 , 1))) %>%
    mutate(subject = session_id) %>%
    mutate(test = as.factor(test)) %>%
    select(subject, test, item_id, question, score) %>%
    distinct(subject, test, question, .keep_all = TRUE)

  # massage covariates file
  rote_cov = rote_cov %>%
    mutate(subject = session_id) %>%
    mutate(gender = as.factor(ifelse(is.na(gender),2,gender-1))) %>%
    distinct(subject, .keep_all = TRUE)
  #inner-join 2 datasets
  dataset <- inner_join(rote_test, rote_cov, by="subject" )


  dataset = dataset %>%
    mutate(knowledge = ifelse(item_id == item_id1, knowledge1, 0)) %>%
    mutate(knowledge = ifelse(item_id == item_id2, knowledge2, knowledge)) %>%
    mutate(knowledge = ifelse(item_id == item_id3, knowledge3, knowledge)) %>%
    mutate(knowledge = ifelse(item_id == item_id4, knowledge4, knowledge)) %>%
    mutate(question = as.factor(paste(item_id, "-", as.character(question %% 10), sep="", collapse=NULL)
    select(subject, cluster, treat, age, gender, reading, practice, knowledge, test, question, score)
  dataset
}

correct_data = function(){
  df = read.csv(file="rote_test.csv", sep=',')

    df[df$item_id1 == 1, "c11"] = 2
  df[df$item_id2 == 1, "c21"] = 2
  df[df$item_id1 == 3, "c11"] = 3
  df[df$item_id2 == 3, "c21"] = 3

  write.csv(df, file="rote_test_corrected.csv")
}

print_stats = function() {
  print(paste("After cleaning, the number of rows in our dataset is:",
              toString(nrow(dataset))))
  print(paste("After cleaning, the number of observations in treatment is:",
              toString(sum(dataset$treat))))
  print(paste("After cleaning, the number of observations in control is:",
              toString(nrow(dataset) - sum(dataset$treat))))
}


print_summary = function() {
  stargazer(dataset,
            header= F,
            title = "Summary Table of Data",
            type="latex") #flip type between text and latex
}
```

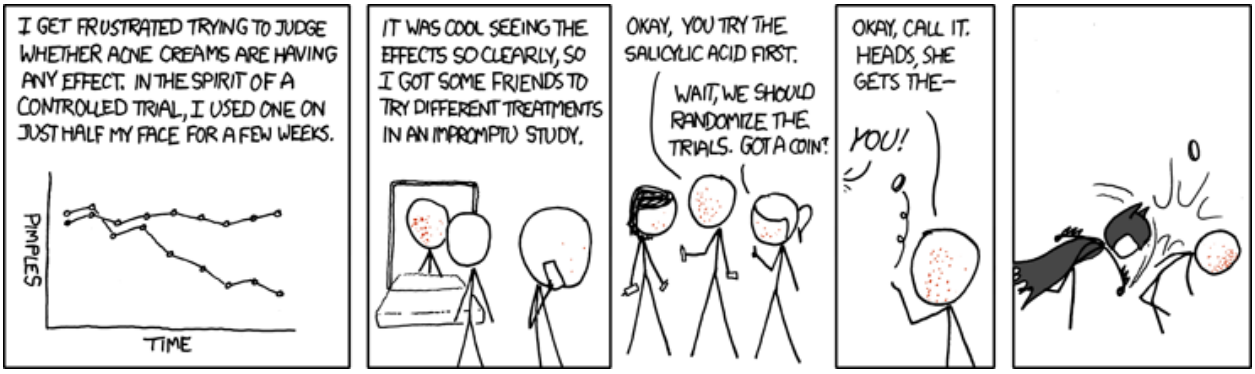
```
result_regressions = function() {
  regression1 <- lm(score ~ treat ,data=dataset)
  regression2 <- lm(score ~ treat + age + prior_knowledge + reading + gender + practice,data=dataset)
  clustered_errors1 <- vcovCL_1c <- vcovCL(regression1, cluster = dataset[ , 'cluster'])
  clustered_errors2 <- vcovCL_2c <- vcovCL(regression2, cluster = dataset[ , 'cluster'])
  stargazer(regression1, regression2,
            header = F,
            type = "latex",
            omit.table.layout= "n",
            keep.stat = c("adj.rsq", "n", "f", "ser", "aic", "wald"),
            se = list(sqrt(diag(clustered_errors1)),sqrt(diag(clustered_errors2))),
            star.cutoffs = c(0.05, 0.01, 0.001),
            title="Regression Results with Clustered Standard Errors")
}

result_regressions_alternate = function() {
  regression1 = glm(score ~ treat*test,
              data=dataset_long, family=binomial(link="logit"))
  regression2 = glm(score ~ treat*test+question+knowledge+age+gender+practice+reading,
              data=dataset_long, family=binomial(link="logit"))
  clustered_errors1 <- vcovCL_1c <- vcovCL(regression1, cluster = dataset_long[ , 'cluster'])
  clustered_errors2 <- vcovCL_2c <- vcovCL(regression2, cluster = dataset_long[ , 'cluster'])
  stargazer(regression1, regression2,
            header = F,
            single.row=TRUE,
            type = "latex",
            omit.table.layout= "n",
            keep.stat = c("adj.rsq", "n", "f", "ser", "aic", "wald"),
            se = list(sqrt(diag(clustered_errors1)),sqrt(diag(clustered_errors2))),
            star.cutoffs = c(0.05, 0.01, 0.001),
            title="Alternate Regression Results with Clustered Standard Errors")
}

cov_check_regression = function() {
  glm(treat~age+gender+practice+reading+prior_knowledge,
      data=dataset, family=binomial(link="logit"))
}


plot_rand_gender = function (){
  dataset %>% ggplot()+
    geom_bar(aes(x=gender, fill=as.factor(treat)),
             position="stack") +
    theme_minimal() +
    #scale_fill_brewer(palette="Dark2") +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Count") +
    ggtitle("Randomization - gender") +

    theme(legend.title=element_blank())
}
```

```
plot_rand_age = function (){
  dataset %>%
    ggplot(aes(x=age, fill=as.factor(treat)))+
    geom_histogram(position="stack", binwidth=3) +
    stat_count()+
    theme_minimal() +
    #scale_fill_brewer(palette="Dark2") +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Count") +
    ggtitle("Randomization - age") +

    theme(legend.title=element_blank())
}

plot_rand_reading = function (){
  dataset %>% ggplot(aes(x=reading, fill=as.factor(treat)))+
    geom_histogram(position="stack", binwidth=1) +
    stat_count()+
    theme_minimal() +
    #scale_fill_brewer(palette="Dark2") +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Count") +
    ggtitle("Randomization - reading habits") +

    theme(legend.title=element_blank())
}

plot_rand_practice = function (){
  dataset %>% ggplot(aes(x=practice, fill=as.factor(treat)))+
    geom_histogram(position="stack", binwidth=1) +
    stat_count()+
    theme_minimal() +
    #scale_fill_brewer(palette="Dark2") +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Count") +
    ggtitle("Randomization - practice memorizing") +

    theme(legend.title=element_blank())
}

plot_scores = function() {
  dataset %>% ggplot(aes(x=score, fill=as.factor(treat)))+
    geom_histogram(position="dodge") +
    stat_count()+
    theme_minimal() +
    #scale_fill_brewer(palette="Dark2") +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Frequency") +
    ggtitle("Score difference between baseline and experiment") +
```

```
    theme(legend.title=element_blank())
}

plot_gender = function() {
  ggplot(dataset, aes(x=as.factor(gender))) +
  geom_bar() +
  ggtitle("Distribution of Gender")
}

plot_age = function() {
dataset %>% ggplot(aes(x=age, fill=gender))+
    geom_bar(position="stack") +
    stat_count()+
    theme_minimal() +
    #scale_fill_brewer(palette="Dark2") +
    scale_fill_manual(values=c("#003262", "#FDB515", "#C4820E"),
                      labels=c("Female", "Male", "Other"))+
    ylab("Count") +
    ggtitle("Age distribution") +
    theme(legend.title=element_blank())
}

plot_score_diffs = function () {
  dataset %>% ggplot(aes(x=score, fill=as.factor(treat)))+
    geom_histogram(position="dodge", binwidth=1) +
    theme_minimal() +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Frequency") +
    xlab("Score")+
    ggtitle("Score differences") +
    theme(legend.title = element_blank())
}

plot_score_diffs_smooth = function () {
  dataset %>% ggplot(aes(x=score, fill=as.factor(treat)))+
    geom_density(alpha=0.6)+
    theme_minimal() +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Frequency") +
    xlab("Score")+
    ggtitle("Score differences") +
    theme(legend.title = element_blank())
}

plot_score_pre = function () {
  dataset %>% ggplot(aes(x=score_pre, fill=as.factor(treat)))+
    geom_histogram(position="stack", binwidth=1) +
    theme_minimal() +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Frequency") +
```

```
    ggtitle("Pre-treatment scores") +
    theme(legend.title = element_blank())
}

plot_score_post = function () {
  dataset %>% ggplot(aes(x=score_post, fill=as.factor(treat)))+
    geom_histogram(position="stack", binwidth=1) +
    theme_minimal() +
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    ylab("Frequency") +
    ggtitle("Post-treatment scores") +
    theme(legend.title = element_blank())
}

plot_scores = function() {
  p1 = plot_score_pre()
  p2 = plot_score_post()
  ggarrange(p1, p2, nrow=1, ncol=2)
}

plot_diffs_all = function() {
  p1 = plot_score_diffs()
  p2 = plot_score_diffs_smooth()
  ggarrange(p1, p2, nrow=1, ncol=2)
}

plot_participation = function() {
  df_a = data.frame(id=1:4,stage=c("Started", "Covariates", "Baseline test", "Second test"),
                    n=c(218, 171, 123, 108))
  df_a %>% ggplot(aes(x=stage, y=n, fill=stage))+
    geom_bar(stat="identity") +
    theme_minimal() +
    scale_x_discrete(limits=c("Started", "Covariates", "Baseline test", "Second test")) +
    scale_fill_manual(values=c("#003262", "#3B7EA1", "#FDB515", "#C4820E"))+
    ylab("Participants") +
    xlab("Completed stage")+
    theme(legend.position="none")+
    ggtitle("Participation by completed stage")
}

plot_rand = function() {
  p1 = plot_rand_age()
  p2 = plot_rand_reading()
  p3 = plot_rand_gender()
  p4 = plot_rand_practice()
  ggarrange(p1,p2,p3,p4,
            labels = c("A","B","C","D"),
            ncol = 2, nrow = 2)
}


plot_rand_qs = function() {
```

```
  dataset_long %>%
    ggplot(aes(x=question, fill=treat))+
    geom_bar(stat="count")+
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    theme(axis.text.x = element_text(angle = 50, hjust = 1))+
    labs(fill = "Assignment")+
    ylab("Count")+
    xlab("Question")+
    ggtitle("Randomization of topics")
}

plot_qs = function() {
  dataset_long %>%
    group_by(treat, question) %>%
    summarise(m = mean(score), .groups="keep") %>%
    ggplot(aes(x=question, y=m, fill=treat))+
    geom_bar(stat="identity", position="dodge")+
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    theme(axis.text.x = element_text(angle = 50, hjust = 1))+
    labs(fill = "Assignment")+
    ylab("Average score")+
    xlab("Question")+
    ggtitle("Average scores by question")
}

plot_q_601 = function() {
  df = read.csv(file="rote_test_corrected.csv", sep=',')
  df %>%
    drop_na(a11, a21) %>%
    filter(item_id1 == 6 | item_id2 == 6) %>%
    mutate(answer = ifelse(item_id1 == 6, a11, a21)) %>%
    ggplot(aes(x=as.numeric(answer), fill=treat))+
    geom_histogram(position="stack", binwidth=1)+
    stat_count()+
    scale_fill_manual(values=c("#003262", "#FDB515"),
                      labels=c("Control", "Treatment"))+
    labs(fill = "Assignment")+
    ylab("Count")+
    xlab("Answer")+
    ggtitle("Answers for topic 6 question 1") +
    scale_x_discrete(limits=c("A", "B", "C*", "D", "E", "'Don't know'"))
}
```



https://xkcd.com/700/