Analytics Exam

Thank you for applying to work on the analytics team at the Analyst Institute and for agreeing to take this exam. We hope that this exam can give us a good sense of your skills and give you a good sense of whether you'd enjoy the kind of work we do.

We expect that certain portions of this exam may come easier to you than others, and we designed this exam to be challenging. If you are unable to finish certain portions, we strongly encourage you to submit an outline of your thinking.

You have 48 hours to complete this exam from the time you received it. To submit your test, please e-mail the results of Part 1 and Part 2 (which will be two links) to awang@analystinstitute.org with a subject line "NAME - Test Submission". More detailed submission instructions appear below.

There are two parts to this exam. Part 1 asks you to demonstrate your familiarity with experimental analysis. Part 2 asks you to complete a more general programming task.

Part I

In 2014, one of our partners was interested in determining the effectiveness of a mail persuasion program on likely voters. The campaign hoped their mailers would both persuade voters to support candidate Jane Smith over her opponents and mobilize voters to cast a ballot in the general election. Voters were randomly assigned to receive one of three interventions:

- No mailer (control)
- Mailer highlighting Smith's record on healthcare (ProHealth)
- Mailer highlighting Smith's record on education (ProTeach)

After the mail was sent, we conducted a phone survey asking voters which candidate they planned to support in the upcoming election. After the election, we consulted publicly-available state voter files to measure whether the targeted voters actually voted on Election Day. Note that while the state voter files (and thus, our measure of turnout) include everyone in the treatment universe, we only surveyed a subset of the universe.

The two main research questions were

1. Did either/both of the pro-Smith messages persuade voters to support Jane Smith?
2. Did either/both of the pro-Smith messages increase voters' likelihood to vote?

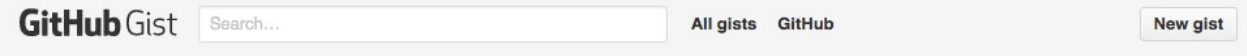
We have attached a zip file of the data. Your coworker started the analysis but didn't finish it. It is your job to make sure the code is complete, correct, and well-documented.

Please feel free to change the analysis code however you feel is necessary, and add comments to explain what your code does.

Before starting on Part 1, you may find it helpful to read the Analysis at AI document.

How to submit Part 1: Please create a secret Gist on GitHub.com. If you do not have an account on Github, you can sign up at https://github.com/join?source=login. (Feel free to just select the free plan.)

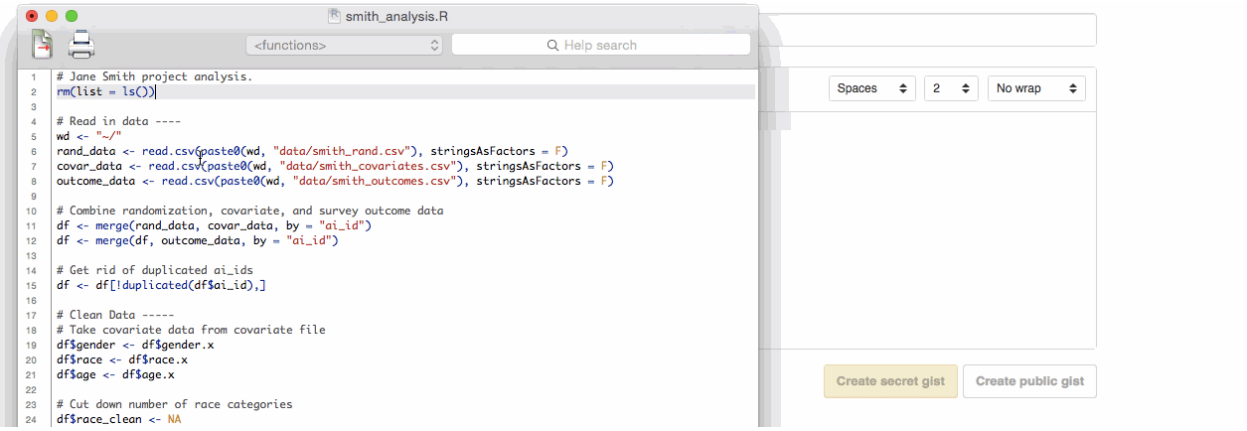
Once you have logged in, go to https://gist.github.com/ and click “New Gist” in the upper right hand corner:



Start by copy+pasting the original code sent to you from your co-worker and click **CREATE SECRET GIST**.

Name your gist `FIRSTNAME_LASTNAME_PART1.R`.

Then, when you’re ready, edit the Gist and copy+paste in the revised code and click Update Secret Gist. Include the link in your submission request.



Part II

Please write a function to merge two dataframes and display some diagnostic statistics.

- You may write this function in a language of your choice, though we prefer R or Python (in that order)
- You are encouraged to use existing merge functions in any library or package of your choice (the goal of this exercise is NOT to demonstrate your knowledge of matching algorithms but rather your ability to code)

- Please note all dependencies, e.g., if you import a library
- Use your judgment in deciding the appropriate input parameters
- Include documentation and usage instructions (including instructions on how to load data from a CSV file), with at least one example

For two arbitrary data frames, dfA and dfB, your function must be able to

- Allow the user to perform an INNER, LEFT (OUTER), RIGHT (OUTER), or FULL OUTER joins of dfA and dfB using exact matches on one or more columns
- The function should print the following diagnostics:
    - The number of rows from the returned dataset in dfA AND dfB
    - The number of rows from the returned dataset in dfA BUT NOT dfB
    - The number of rows from the returned dataset in dfB BUT NOT dfA
    - Any other diagnostics you feel would be useful
- The returned dataset should also include an indicator column called _merge that contains the following codes:
    - 1 if the rows are in dfA AND dfB
    - 2 if the rows are in dfA BUT NOT dfB
    - 3 if the rows are in dfB BUT NOT dfA

Your function will be evaluated on functionality, ease of use, speed, conciseness, and documentation.

How to Submit Part 2: Please create a secret Gist on Github. Copy and paste the code for your function and examples there. Name your gist `FIRSTNAME_LASTNAME_PART2.[extension]`. Include the link in your test submission.