Exam 2 - Data Science for the Social World

Michael G. Findley[*] Michael Denly[†]

Instructions

This is a two-hour, open-book, open-internet exam. Each question will be worth 4 points. As soon as you click on the exam on Canvas is when your timer starts. If you submit the exam late, you will be penalized by 1 point for each minute late that you submit. For example, if you submit 5 minutes late, you will lose 5 points on your overall exam grade.

In terms of your submission, Canvas will only allow you to submit one file. That one file should be either a PDF file or Word document corresponding to the output of your `R` Markdown file. Kindly note that we will not accept Google Docs, and that you must write your exam using `R` Markdown. We will not accept exams that use a regular `R` script and they copy/paste outputs onto a Word Document or PDF file. The PDF or Word file must be generated using `R` Markdown, and we will be able to easily discern when that is not the case. Using a comment on your Canvas submission of your PDF file or Word document output from `R` Markdown, please also provide a link to a GitHub repo. The latter should contain your `R` Markdown `.Rmd` file and the exam dataset–basically, everything you would need such that your repo is communicating with your files on your computer. You may name the repo anything that you would like, but maybe something like "exam2" would be appropriate. Since GitHub provides time stamps for everything, we will be able to discern if you modify the files outside your two-hour exam window; in short, please respect the two-hour window. We will not accept exams that fail to provide a GitHub repo link with an `R` Markdown file and dataset to accompany the Canvas submission.

Please work independently. You may *not* consult anyone in the class or outside the class for help, and you may not post the exam questions on the Stack Exchange, Google Groups, or any similar website–though you may visit these websites or others. Please also do not discuss the questions or answers over WhatsApp, GroupMe, text message, or any other platform, especially because everyone will be taking the exams at different times. We will be monitoring accordingly, and anyone who violates any one of these policies will receive a zero on the exam.

Please annotate your `R` code chunks in your `R` Markdown file with comments, or make sure that the text surrounding it sufficiently explains what you are doing. Essentially, your `R`

[*] Professor, Department of Government, UT Austin, mikefindley@utexas.edu

[†] PhD Candidate, Department of Government, UT Austin, mdenly@utexas.edu

Markdown file should mimic that notes files that we submit on Canvas to accompany the video lectures. We will remove points when you do not provide clear comments or explanation to tell us exactly what you are doing with your code.

We have endeavored to make the exam self-explanatory, but feel free to email the instructors and the TA if you have questions. At least one of us will be available over email for the entire 4-hour exam. However, please email all three of us if you have a question (i.e., do not email only one or two of us), because we will be taking shifts.

One final note: if you have an SSD accomodation, please submit your exam under "Exam 2 (accomodation)". Otherwise, please submit on Canvas under "Exam 2".

And one final hint: use your time wisely. If you can't answer one question, move on to the next one, and come back to it once you are done with the ones that you can answer more quickly. Good luck!

Questions

1. Please clear the environment in `R`.
2. Load the "inequality" dataset into `R`, and save the data frame as 'inequality_data'.
3. Is this dataset a cross-sectional or panel dataset? Explain why in words and provide some `R` code to prove that your answer is correct.
4. The data frame contains a variable called `inequality_gini`. It corresponds to the inequality Gini index, which "measures the extent to which the distribution of income (or, in some cases, consumption expenditure) among individuals or households within an economy deviates from a perfectly equal distribution." In simple terms, there is a lot of inequality when there are a lot of rich people and a lot of poor people but not a lot of middle-class people. There is low inequality when most people are earning about the same amount of income. Scandinavian countries like Sweden and Denmark tend to have the most optimal Gini index scores. Using the `subset` command, provide the `inequality_gini` scores for Denmark and Sweden.
5. Since Brazil started the *Bolsa Familia* conditional cash transfer program in 1990s, inequality in Brazil has decreased significantly. Just the same, inequality in Brazil is very high comparatively. Using the `subset` command, please show the `inequality_gini` score for Brazil.
6. Given your answers to the previous questions, is it better to have a high or low `inequality_gini` scores?
7. Use the `head` command to get a quick peak at the data frame.
8. Write a function called "accent.remove" to remove the accent on Belarus, apply that function, and run the `head` command again to show that you removed the accent.

9. Sort the data by the countries with the lowest `inequality_gini` scores and then run the `head` command again to show what the top 5 countries are.

10. What is the mean `inequality_gini` score? Provide the relevant `R` code.

11. Using the `ifelse` command, create two new dummy variables, `high_inequality` and `low_inequality`, which takes values of either zero or one. The `low_inequality` variable should correspond to countries with `inequality_gini` scores below the mean. The `high_inequality` variable should correspond to countries with `inequality_gini` scores above the mean. (Note: we will not accept answers that do not use the `ifelse` command to create the variables.)

12. Run a cross-tab using the `high_inequality` and `low_inequality` variables that you created in the previous question. The cross-tab should provide the mean `inequality_gini` score and number of observations for each category of inequality. (Note: if you had trouble using the `ifelse` command, we couldn't provide points for the previous question. However, you can create the variables using the indexing method)

13. The World Bank, the African Development Bank, and the Bill and Melinda Gates Foundation are all working on reducing inequality in Africa. Write a `for` loop that prints the names of these three actors. (Note: we will not accept answers that do not provide a `for` loop.)

14. Use this website to find a variable from the World Development Indicators that you think is correlated with inequality. Tell us what variable you picked and why you picked it. (Don't worry if your prediction is not correct. We just want you to provide some rationale.)

15. Import that variable directly into `R`. (Note: if you are having trouble, read Mike Denly's Canvas announcement from the other day.)

16. Rename the variable that you imported into something that we can actually understand.

17. Merge the new variable into the other dataset, using `inequality_data` as the `x` and and your new data frame as the `y`. When merging use the command that only keeps the rows in your `x` data frame. Call your new data frame `merged_df`. Ensure that you have no variables with `.x` or `.y` at the end of them in your new `merged_df`, while at the same time ensuring there are still variables like `country` and `year`.

18. In `merged_df`, remove the missing data on the basis of `inequality_gini` and your new variable that you took from the World Development Indicators.

19. Using the `filter` command and piping method, only keep the data with `inequality_gini` scores greater than 30. Save the new data frame as `data_greater_30`. (Note: we will not accept answers using `subset`.)

20. Using `data_greater_30`, use to `R` to count how many countries have the sequence "ai" in their name.

21. Use any command from the `apply` family to take the sum of `inequality_gini` in `data_greater_30`.

22. Label your variables in `merged_df`. Any labels will suffice.
23. Save the labeled data frame as a Stata dataset called `final_data`.
24. Save all of the files (i.e. `.Rmd`, `.dta`, `.xlsx`, `.pdf`/Word Doc), push them to your GitHub repo, and provide us with the link to that repo.