manipulating data and merging

[version: Wednesday 13th September, 2023 16:13]

due in 2 weeks

warning! it may be difficult and time consuming: may be difficult to find data to merge with your initial data, so plan ahead–you need at least 1 new dataset to merge with your initial data: need a variable that is common across 2 datasets; it's pretty easy if you have agg data, say your u/a is city, county, state, country or time: year, month, etc

but if u/a a person, then more challenging because need 1 other dataset about the same person

look for geography or time in your dataset that varies–say multiple cities, counties, states, or multiple months, years, etc–then you should be able to find other datasets that contain that same information and then be able to merge pretty easily: it will be many to one merge

plan ahead and look for even more data: next ps will ask you to do 5 more merges

1. as before, in few sentences explain why these data: like a mini few sentence research abstract; extend and improve it
2. do at least 2x each : map/recode, replace on condition, subset/slice , groupby/agg –they all must be interesting and make substative (not just technical) sense!that is we are building and exploring a useful and meaningful dataset–the code must be helpful and loggical for that purpose
3. (next week's class) merge your dataset with at least 1 other dataset (as always, start early and email me if you get stuck); you may merge on u/a or geography (eg state) or time, (eg year) or some characteristics eg occupation; again, you may often contribute by just merging your data with other data!

hints

1. don't forget to check everything, e.g. after merge... correctness is important and difficult

---

general directions (always the same):

- i will show your code in class and possibly repost, as per our core values–opensource and transparency, but if you'd like to keep it private, let me know, you just may get less feedback
- you must submit all the code that was executed from the very beginning starting with the very raw data as per replication principle; if data too big to fit online, then just start with eg "to fit data online took 10% random sample"
- ps are cumulative–can and should include much of previous code; can also use code you've written outside of this class (other classes, projects, etc)–but you have to clearly mark the code that has not been written for this class–otherwise, scholastic dishonesty!
- use your own dataset; again if you do not have a dataset, ask for help finding it
- you are only submitting code, so it must load data from Internet: `https://theaok.github.io/generic/howToPutDataOnline.html` (when you put data into any public space, try not to violate data copyrights... I haven't heard of anyone having problems with that, but be careful–for instance you may subset dataset to few vars and smaller sample); and it is also easier to learn on small datasets
- keep it simple! at the beginning of your notebook drop unnecessary vars; and even retain only fewer obs; keep it manageable; much easier to learn using simple data; can always complicate later!; much better to do it right using simple data than do it wrong using complex data!
- have nice structure in your file: sections, subsections, etc; may also have multiple files
- great to copy code from others; again, one of the rules for this class is 'be lazy': don't reinvent the wheel, whatever you are coding, has already been done, google things often; but of course you cannot submit 100% code by someone's else
- if you do something extra/fancy that is relevant and closely related to the assignment questions, it will be extra credit
- use coding rules that we've learned so far
- submit (only) the code into git repo; ps are due by the beginning of the next class unless indicated otherwise, eg "due in 2 weeks"; late ps not accepted
- we are on the way to developing the final project with these ps: as we progress, your ps should start resembling a coherent and logical project where you use learned techniques to answer interesting questions– say in few sentences (probably at the beginning) why are you doing what you are doing–that is, answer the "so what question": "ok, you're gonna run all that code, and so what?" what's the goal of all that, why are you doing this? you need a compelling justification for what you are doing: say what are those questions you want to answer; be brief, say couple sentences, and definitely not more than say 100 lines, typically 10-50 lines is enough; related: say why you use data you are using, is it best, does it serve the purpose?; and can ask us questions in comments

- be prepared do present your code in class (if time), just briefly, key points, couple minutes

- if you work in a group of 2 or 3 people make it 2x or 3x better, eg If ps asks for joining 2 datasets, and there are 3 of you, then just join 6, etc, just do 3x more or better
- always have a brief description/interpretation of a map, say few sentences or a pargraph or max few; also may list problems you've encountered and ask questions
- always have exact links to all of the source data (so that i could create the map myself); note: exact links, eg do not say census.gov, but give full url to the data–i must be able o find it; sometimes there is no generic URL–then give steps: what I need to click to get the data!