merging

[version: Thursday 20th September, 2018 10:31]

due in 2 weeks

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

but if you have person-level data, that is, your u/a is a person, then it may be more challenging, because you will need 1 other dataset about the same person; what may help is if you have some 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 m:1 merge

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

1. (next week's class; so for now just look for data as per the above): 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! dataset you merge must be other real dataset; that is, you cannot generate "fake" new datasets as I may in the class by artificially splitting (collapsing, etc) my dataset; in short, data used for merge must not have been originally in the same dataset!

hints

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

---

general directions (always the same):

- i will show your code in class and possibly post some of your code or link to it–again, as per our core values–opensource, transparency, sharing; but if you'd like to keep your code private, that's fine–just let me know, and i will keep your code secret (no penalty, except that you may get little less feedback–usually if we discuss your code in the class, you will benefit from it!)
- you must submit all the code that was executed from the very beginning starting with the very raw data as per replication principle
- all ps are mostly cumulative–you can, and should, include much of previous code you've written for this class; 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, email us, stop by our offices, etc
- because you are only submitting code, it must load data from Internet–just put your data onto your own website, wordpress, google drive, etc; (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 using **sample**); and it is also easier to experiment on small datasets
- keep it simple! drop unnecessary vars; and even retain only certain, say most important, observations; keep it manageable; it is 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 idea to submit ps as early as possible–we will probably give you some comments; if not, email us and ask for comments!
- it is 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, it 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 the Sakai's dropbox, or GIT repo; ps are due by the beginning of the next class unless indicated otherwise, eg "due in 2 weeks"; late ps are 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 data management techniques to build new a dataset that can be used to answer interesting questions– say in few sentences (as a comment) 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; typically: to develop a new dataset (that has not existed before) that can be used to answer some exciting questions: 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: even at the beginning, already in ps1, say why you use data you are using, is it best, does it serve the purpose; also, feel free to ask me questions in comments