README.txt for climate indicators section

I found the data from PRISM climate group at Oregon State during my search for high quality climate data. I saw that it had data on max temperature, mean temperature, min temperature, and precipitation down to the county level in California for every month since 1981. However, I also observed that it was not easy to download the data in bulk.

The first step I took was to see if I could use wget from the command line to download the data and append it together. I built the getprism.sh file to do that, but realized that there was a middle part of the url that changed for each data download before the csv such that using special characters {} to download multiple counties no longer worked. For that reason, we divided the 58 counties to our group members and each downloaded some. Once the files were downloaded, I was able to append them using the command line (using getprism.sh), but found later that it was better to use STATA. The main reason for this choice was that the data for each county came with latitude and longitude coordinates but did not list the county name so I had to back into that link later (I also tried to spatially join the latitude and longitude to shapefiles, but had unresolvable issues related to the projection – more on that later).

Append and Data Cleaning with STATA:

* Once I had 58 csvs (1 per county) with the raw climate indicators (mean temp in month, max temp in month, precipitation in inches for month, and min temp in month) for all months over the time period of 1981-2018, I needed to read them into a single file
* I created a loop to read each of the 58 csv files and used a local variable to attach the number in which the file was read as an idnum column (the first file had an idnum equal to 1, second file idnum ==2, etc.)
  + This loop created a STATA data file for each of the input csvs
* Given how we divided the download of files using wget between the team members, I had a list of the file names that was (if order could be preserved) in alphabetical order
* Next, I found an alphabetical list of California counties online and created a matching key of ascending numbers
* In the STATA file I created a county variable and I filled that county with appropriate name based on the matching list. I used the alt trick you showed us in class in Jupyter Notebook cells to create STATA code (replacing county if idnum == 1, etc.). There isn’t a specific record of this because in the STATA code this results in 58 lines of “if idnum==1, replace = “county name”, but this wasn’t as repetitive as it looks.
* This was the main cleaning issue, but I also renamed variables and changed the date to the appropriate date filetype

Create the change variables:

* The dataset had mean and max temperatures, but it did not tell us how these things changed over time
* I consulted a professor and learned that I could get a measure of the monthly change by county by running a regression where I would regress time (year) on the various indicators (max\_temp, mean\_temp, precipitation) for each county and month
  + This would tell me how these indicators changed each year for this locality and month
* At this point, I ran into an issue with my matrix size (my version of STATA was not powerful enough to run this operation), so I divided this process into 6 stata files where I ran the analysis code on 10 counties each
* I created a file called prism\_runall.do and appended the resulting datasets together after running the analysis 10 counties at a time and outputting 6 intermediate files

Outputs from STATA files:

* From the first cleaning effort (inelegantly named prism\_cleaning\_0\_10v3.do), I saved a long data format, which had the raw indicators and all months over the 37 years called appended\_prism.dta
* Then after getting the indicator change measure and appending the files (in prism\_runall.do), I created an output called prism\_all.dta, which has one line per county with its change for each indicator (this change is a measure of yearly change so I multiplied by 37 in many cases to show total change)
* I made a post-analysis file to load data in ready-to-visualize formats in R in another STATA script file called prism\_viz.do
  + This created csvs with annual averages (across the months) for the indicators,

Visualizing data:

* I visualized the data in Jupyter Notebooks (with Python), Rstudio, and Datawrapper
* I used Datawrapper to create two visuals but I found the iFrame to be pretty inflexible so I ultimately remade one of the visuals in R. The one remaining visual is an interactive map
* The file Cali\_maps2.ipynb show my commented code in Jupyter Notebooks and the file yearmeanchange.R contains my commented R scripts for the other 4 visuals
* In sum, of the graphics that were used, I created 1 graphic in data wrapper, 2 using Python’s geopandas library, and 4 in R
  + R was an entirely new language for me and so making those graphics and translating the initial drafts from STATA was one of my primary challenges
  + Using the command line in the shell file to download the files (although ultimately only half successful) was another entirely new exercise for me
  + Finally, while I had used Python before, I did use this project to deepen and consolidate my knowledge and actually had to spend much of the project time trouble-shooting the issue with merging the county data correctly