Workflow guide

Steps: each should be contained in separate code (e.g., separate ".R" files, code chunks, or folders)

1. Data Organization - create project folder linked to GitHub
    a. If Github repo already exists, pull this onto your local and work from within it
    b. Each project folder, whether created by you or pulled from GitHub, should include the following folders:
        i. Data - Keep all raw data in this folder and do not modify the raw data
        ii. Output - Write out all cleaned versions of data (make sure the file name includes "clean" to distinguish from raw) and analysis output here
        iii. Figures - Write out all figures here
        iv. R - Save all R scripts here
    c. Create other folders as needed (e.g., "Literature", "ms", etc.) but add to .gitignore file so they are not pushed up to GitHub
    d. Add dates to files that will have multiple versions (e.g., different runs of an analysis, versions of figures, etc.) using the format YYYY-MM-DD to the beginning of file names so files are sorted accordingly
2. Data Cleaning - do NOT edit the raw data file, instead do ALL cleaning within your code
    a. Spelling mistakes
    b. Capitalization inconsistencies
    c. Refer to the coding style guide for naming conventions
    d. End with tidy ("long") format data. See Wickham for more on tidy format
3. Documentation
    a. Data - A text README file should be created for every project and a full description of the raw data should be included here
    b. Code - All code should be extensively commented such that each step of the analysis is clear to any reader without that individual knowing R. Follow the style guide for additional notes on commenting code
4. Data Plotting and Validation - We recommend producing simple scatter plots, boxplots, check that outliers are not errors
5. After data cleaning and plotting/validation, output data into its final, clean format (write this file to the "Output" folder. Everyone in the research team should start from the same cleaned dataset. They should also have access to or at least be aware of what was done to get to this point (i.e., Steps 1, 2, and 3).
6. Data Analysis - have separate scripts to process, filter, aggregate, summarize data as needed for different types of analyses
7. GitHub
    a. Pull all changes before beginning work on your code
    b. All code changes should be pushed to GitHub on a regular basis (it is better to commit more rather than less)
    c. Aim to include as descriptive commit messages as possible
    d. In general, do not push the "Data," "Output," or "Figures" folders up to GitHub (i.e., include these folders in your .gitignore file) because they are often too large

e. Do not push up non-analysis folders (e.g., “Literature”, “ms”, etc.). Instead add to .gitignore

f. When setting up a GitHub project, default to private settings. If a project is made public, carefully check for any data that may have been pushed up to ensure it is okay for this to be public