Whitney Domico

The dopamine hypothesis is outdated: A meta-analysis of the complexities of the biochemistry of schizophrenia

**Debug 1: Exploratory analysis of HIV and patient outcomes**

***File Sharing: Mine to Them***

I emailed an instruction sheet to Rowan and Nagendra that detailed how to get access to my repository to find my README and project files. I had created a zip file of all the necessary files to run my code, and so they received that directly from the repository (as the instructions indicated). This included my linear R script and all the necessary comma-separated and Excel files.

***My Work with Their Code***

Nagendra and Rowan’s code is well-commented, so it is not difficult to follow. The first problem I encountered when I ran the code was that all of the file paths had to be changed before it would run. Additionally, one file – TxData.csv – had somehow not managed to get added to their zip file, so they emailed it to me separately. Overall, though, the code ran smoothly, and my outputs matched theirs.

I managed to break their code in two ways. Firstly, I added a fake patient to the end of their dataset (“Patient 242”). Because they had used hard numbers in their code (i.e., “1:241” instead of “1:length(x)”), the code fully ignored the 242nd patient. Similarly, I removed Patient 1 from their data file, and this caused a congruence error because the code attempted to add a column with 241 rows to the data frame that now only had 240 patients.

So, overall, my suggestions were:

* Create an instruction sheet that clearly states which data files are needed and/or implement the zip file into the repository for easier download.
* For all hard numbers, replace with length(x) functions so that if someone wants to use the code for lists of patients that are not exactly 241 people long, they still can. This would increase robustness.

***Their Work with My Code***

The download of the files went smoothly, and the code ran through for both Nagendra and Rowan (with aid of the instruction sheet and code commenting). Rowan noted that my decision to have the user input their own data into Excel sheets and reupload them into the script is “awkward” and “leaves opportunity for error.” He noted that, even though the user was only supposed to input either “0” or “1,” that every number input that was not equal to 1 was counted as a 0. He also disliked my decision to have the user rename the reuploaded file and suggested that I instead use the same name. He also mentioned that the graphs were not well-labeled (no axis titles, etc.), which made it difficult to understand what the data meant.

Nagendra had a strange error that occurred once when he reran the script where data frame rows that were supposed to be deleted were instead converted to *NAs*. We could not replicate this error.

Overall, I made the following changes to my script:

* Implemented an “else” function that stopped the code if the user provided an input that was not 0 or 1.
* Got rid of the need to rename the reuploaded files and instead had the script stop if the reuploaded file had the same number of columns as the original file (because the user is expected to have added a column).
* Updated graph labels and deleted some that were unnecessary.

**Debug 2: Honeybee and Wild Bee Flower Preferences in Agroecosystems of the Pacific NW**

***File Sharing: Mine to Them***

I shared the files in the same manner as Debug 1.

***My Work with Their Code***

Ty and Jonathan’s code is well-commented and easy to understand. Their code ran through smoothly the first time I ran it. The major issue that I found was that there was no consistency in their sub-setting of their data frames. For example, for one year, they only kept flowers with >130 visits, for another they only kept flowers with >10 visits, etc. I suggested that they do something more consistent – like top 10% or top 15 flowers.

Some more picky notes I had for them was that their graphs did not have the genus names italicized, and that the axis labels overlapped and were difficult to read. I suggested they italicize the genus names and switch the angle of the labels to 90 degrees with a “vjust” of 0, so that they were more separated.

So, overall, my suggestions were:

* Make the sub-setting consistent instead of choosing arbitrary values for each year.
* Make graphs more user friendly by making the axis labels easier to read and make them more correct by italicizing the genus names.

***Their Work with My Code***

The download of the files went smoothly, and the code ran through for both Ty and Jonathan (with aid of the instruction sheet and code commenting). Ty agreed with Rowan (from the previous debug) that the Excel sheet – user input choice was awkward; however, I do think that this is necessary for this project (if I had a few more months then I could feasibly consider automating this process). I had added an error message (after Rowan suggested it) that told the user if they put in a number that was not 0 or 1, but Ty said it was not descriptive enough to properly advise the user.

Jonathan’s main concern was that my code was not commented thoroughly enough, and that it was confusing for users who were not familiar with the biology and genetic information it contained.

Both Ty and Jonathan noted that I may want to make it easier for the user if they wanted to add an extra PMCID to analysis, which I agree with; however, again, it is not feasible with the time constraint of a semester-long class.

Overall, I made the following changes to my script:

* Made comments more descriptive and added extra information to my instruction file.
* Made error messages more descriptive.