PUBH 7462 Homework 3

Due 2/17/2022

General Expectations

Throughout the assignment, please:

- Use meaningful file names ("_" or "-" seperated)
- Use meaningful variable names
- 'Good' `R` style (white space, etc.)
- Consistent style (choose a style and stick to it)
- Appropriate titles, axes labels, and legend titles/group names
- Get into the habit of commenting your code chunks

- Use relative paths, ex. "./data"

With respect to the knitted .RMD:

- Omit extra output (anything from `R` with ## for example)
- Make sure your code chunks are visible with `echo = TRUE`
- Make sure your inline R works properly and `round()` digits

With respect to Github:

- Make sure your repository is public so the TA and myself can view it

- Keep the repository 'tidy' and well organized
- Use meaningful filenames such that a stranger who happened upon the repository could surmise what's going on

With respect to data visualizations in general:

- **Never output a tibble() or data.frame() as a table**
  - Please use `df %>% gt() %>% tab_header("")`
- **Github Documents – always save with `gtsave` in `./figures/` folder, and call inline with `![](filepath_to_fig)`**
- Remember, a *good* data visualization should be self-explanatory
- This means that I shouldn't need to read your code to know what's going on in the plot
- I find it useful to imagine your audience knows little to nothing about what you're doing prior to seeing the plot (as is often the case)

Problem 1. Github repository (10pts)

- Please set up a repository named `pubh7462_hw3_`*your-email -handle*
- Connect to it Rstudio with an .Rproj
- Create a `/data` folder, add to the `.gitignore`
- Include all necessary .md figure files
- Keep the repository 'tidy', no extra files or folders

Problem 2. Best Practices and Consistent Style (10pts)

Problem 3. Instacart (80pts)

Instacart is an online grocery service that allows you to shop online from local stores. In MN, partner stores include Cub Foods, ALDI, Costco, HyVee and more. Instacart offers same-day delivery, and items that users purchase are (alledgedly) delivered within 2 hours.

Data description

"The Instacart Online Grocery Shopping Dataset 2017" was acquired from their website here on June 24, 2017. The version of the Instacart data that we will use in this class can be found on Canvas here

Context The original data is quite extensive, and the data linked to at the top of this page for use in the class represents a cleaned and limited version of the data. The dataset contains 1,384,617 observations of 131,209 unique users, where each row in the dataset is a product from an order. There is a single order per user in this dataset.

Variables There are 15 variables in this dataset –

- `order_id`: order identifier
- `product_id`: product identifier
- `add_to_cart_order`: order in which each product was added to cart
- `reordered`: 1 if this prodcut has been ordered by this user in the past, 0 otherwise
- `user_id`: customer identifier
- `eval_set`: which evaluation set this order belongs in (Note that the data for use in this class is exclusively from the "train" eval_set)
- `order_number`: the order sequence number for this user (1=first, n=nth)
- `order_dow`: the day of the week on which the order was placed
- `order_hour_of_day`: the hour of the day on which the order was placed
- `days_since_prior_order`: days since the last order, capped at 30, NA if `order_number=1`
- `product_name`: name of the product
- `aisle_id`: aisle identifier
- `department_id`: department identifier
- `aisle`: the name of the aisle
- `department`: the name of the department

Instructions Please put the `instacart.csv` in your `/data` folder and answer the following questions with these data. **Note that with larger data, it may be helpful to start programming with a smaller subset of the data first.** Once that is running properly, you can rerun it on the entire data. In addition `cache = TRUE` can speed up your knitting process, *but be careful* when doing so, it may be useful to rerun your final version with `cache = FALSE` to make sure everything is working properly.

For all `gt` tables, please create a `./figures` folder, save your `gt` as a .png with `gtsave`, and embed in with .RMD with `![](./figures/your_gt.png)`. *Note* - you may need to use `results = "hide"` or `eval = FALSE` to supress the table from outputting in the code chunk during the save.

3.1 Please create a `gt` summary table which describes the overall `mean`, `median`, and `sd` of the *number of days since last order*, *number of items purchased*, *number of aisles*, *number of departments*, *order number*, and *percent of items reordered*; for all customers' orders in these data. (20pts)

- Please include a title
- No 'tidy' variable names (i.e. "No. Items" or "# Items" not "n_items")
  - `stringr` can help with this
- No need to colour the table
- May need the `unique()` function

3.2 Create a visualization in `ggplot 2` which displays the number of orders (of all products) per aisle, with aisles ordered in a meaningful way. (20pts)

- Please use `stringr` to clean up aisle names and turn them into titles (i.e. Fresh Vegetables vs. fresh vegetables)
- Please order the aisles meaningfully
- Please colour the plot according to this order

Hint - will need to utilize `fig.height =` in the R chunk to display all the aisles properly.

3.3 What are the top 6 aisles in the the top 6 departments by items purchased? Create a `ggplot 2` visualization which displays this information, with departments and aisles ordered in a meaningful way. (20pts)

- Please use `stringr` to clean up aisle/department names and turn them into titles (i.e. Fresh Vegetables vs. fresh vegetables)
- Please order the aisles/departments meaningfully
- Please colour and arrange the plot according to this order

Hint - Will need to ungroup prior to cleaning aisle/dept names and `fct_reorder2()` to order properly.

3.4 What are the top 5 aisles by items purchased and what are the top 5 items purchased in each of those aisles? Please display your answer in a single `gt` table. (20pts)

- Ensure that all names are titles and not variable_names
- Order the Aisles and Product names by number of purchases (descending)
- Colour the number of purchases (descending, c("white", "a colour of your choosing"))