

R Bootcamp

Homework 3: Forensic Budget Analysis using operations on data frames

Marguerite A. Butler

February 8, 2018

DUE: Thursday February 8 by 5pm, turned in by GitHub or by email to mbutler808@gmail.com. Please put “710 Homework 3” in the subject line.

Instructions

Start by debugging the code in the stub, and then continuing on with the tasks requested in the following assignment. **Turn in your homework by creating a script of R commands (a text file with the ending .R).** Make sure to include comments in the code anyone reading it can understand what the code does, and please answer questions in your code with comments (please indicate task number in comments too).

Problems

Your friend comes to you for help with budgeting. He can't figure out where all his money is going (besides rent and mandatory bills like electricity, etc.). His expenditure over one week is in the file `budget.csv`. You asked him to provide a spreadsheet with a column for date, description (which store/vendor), a budget category (groceries, coffee, etc.), and the cost.

Start off by downloading `budget.csv` and putting it in your working directory for this homework. Create a script to do all of the following. Read in the `budget.csv` file. Change the column names to `date`, `description`, `category`, `amount` to avoid a lot of typing and upper case. **Test each line of code to make sure each step works as expected. If there is a problem, debug it. Make sure you understand each object and each line of code.** Complete all the tasks below. Check at the bottom for the output that you should turn in with your script. Don't forget to add comments!

1. How much was the total expenditure? Assuming that October 1 is a Monday, calculate the total expenditure split by weekday (Mon - Thurs) and weekend (Fri-Sun).
2. Let's figure out what is happening day by day. Use indexing or the `aggregate()` function to find the total expenditure by date and budget category.
3. Make a report for your friend. Create a data frame of summary statistics by date. Produce a data frame reporting the expenditures (amount): One row per date, and on the columns, total, mean, max. You may want to look back at the tutorial chapter 8. Name this `report.bydate`
4. Do the same by category instead of date. Name this data frame `report.bycategory`
5. Your friend tells you that he budgets \$200/month for food, and \$100/month for entertainment. Does this match with his expenditures? Produce a report that splits out the data by his two categories. You notice that you have to lump several categories together to match his food. Create a new data frame `food` by indexing the `dat` data frame. Include in this data frame the categories `groceries`, `lunch`, `dinner`, and `coffee`.
6. Similarly, make the data frame `entertainment` and include in it `entertainment`, `beer`.
7. Do a subcategory analysis – For `food` produce a data frame by category (this will contain the subcategories of food) and include as before one row per category, and on the columns, total, mean, max.
8. Do a subcategory analysis – For `entertainment` produce a data frame by category and include as before one row per category, and on the columns, total, mean, max.
9. Do you see any problems here? Should your friend think about adding some budget categories? Make a minimum of 2 bar charts to illustrate the problems relative to his planned budget.
10. Turn in - your script with comments (make sure it runs without error!). Your data frames `report.bydate.csv`, `report.bycategory.csv`, `food.bycategory.csv`, and `entertainment.bycategory.csv`, as well as pdf's of your figures.
11. Extra Credit - We are repeatedly aggregating by date or category, so turn this into a function where we get total, mean, and max, by category and return back the merged summary statistics