

R Readiness Assignment

Introduction to R, RStudio, and ggplot

This faux assignment is intended to give you experience working with the R program. If this were a real assignment, you would submit your responses to each of the questions below in a printed document and label the sections as indicated below within your printed document. All graphics would be resized so that they do not take up more room than necessary and would have an appropriate caption. Any mathematics/equations would also need to be appropriately typeset within the document.

Preparation

Open RStudio and install the following packages, if you have not already installed them:

- arm
- dplyr
- ggplot2
- ggthemes
- plotrix
- psych
- sm

Once these have been installed successfully, you should not need to install them again.

Open a new script file. Save the script file as **Assignment-00.R**. Save all of the R syntax you use to answer the questions on this assignment in this script file.

Denote each question in the script file using comments. For example,

```
#####  
### Question 1  
#####
```

Add comments throughout your syntax as liberally as you feel is necessary to help you recall what the syntax does in the future.

Part I

In 2013, Andy read 40 books. The number of pages Andy read each month is reported in Table 1.

Table 1 *Number of pages read per month.*

| Month | Pages |
|-----------|-------|
| January | 1453 |
| February | 422 |
| March | 848 |
| April | 1679 |
| May | 1655 |
| June | 1630 |
| July | 710 |
| August | 557 |
| September | 978 |
| October | 920 |
| November | 647 |
| December | 2698 |

1. Use the `data.frame()` function to enter these data into a data frame called `reading`. There should be two columns in the data frame. The first column you should name `month` and the second should be named `pages`.
2. Use the `sum()` function to find the total number of pages Andy read in 2013.
3. Use the `sm.density()` function from the `sm` package to create a density plot of the marginal distribution of `pages`.
4. Use the `describe()` function from the `psych` package to summarize the marginal distribution of `pages`.
5. Enter the following syntax:

```
plot(reading$pages, type = "b")
```

6. Explain in 1–2 sentences what the plot shows.

Part II

Use RStudio to open the *goodreads-2016.csv* dataset and assign it into an object called `read`. Use the data to answer the following questions.

Enter the following syntax.

```
library(dplyr)
p = read %>%
  filter(bookshelf == "read") %>%
  group_by(month_read) %>%
  summarize(pages = sum(pages))
p
```

7. Explain (in no more than 5 sentences) what the `filter()`, `group_by()` and `summarize()` syntax does in the syntax above.

Part III

Again, use the data in the `read` object you created in Part II. The next several questions will have you use `ggplot()` to create a lineplot displaying the total number of pages read by month of the year. While each question contains more instruction to add detail to your plot, there is no need to put a separate plot in your document for each question. Include one plot in your document that has all of the elements asked for.

8. Using `ggplot()`, create a lineplot (similar to the plot seen [here](#)). This plot should show the total number of pages read per month. The plot should include points and lines as in the example. It should also be displayed chronologically (e.g., January should be the first month).

After you have successfully created the plot, “pretty” it up by doing the following:

- **Axes:** Read the *Axes* page of the [Cookbook for R website](#). Add to the syntax above to change the label on the *x*- and *y*-axis of the plot.
- **Color:** Change the line color to some non-default color of your choice. (Note that if you are printing the document using a non-color printer that the color choice you make should print well in greyscale.)
- **Labels:** Read the *scale_x_continuous* page of the [ggplot2 documentation website](#). Add to the syntax you wrote for the previous question to add a break and label for each of the 12 months.

Include this plot in a word-processed document. Resize the plot so it does not take up any more space than necessary and add an appropriate caption.

Part IV

Again, use the data in the `read` object you created in Part II. Now, you will use `ggplot()` to create a mean-and-error-bar plot of the *average* number of pages read per month. Mean-and-error-bar plots are also referred to as Tie-Fighter plots because they resemble a Tie-Fighter from the Star Wars movies.

9. Change the `dplyr` syntax from above to compute (1) the average number of pages read per month—rather than the total number read per month; and (2) the standard deviation of the pages read per month. (Hint: The output should be a data frame with three columns rather than two columns.)
10. Using `ggplot()`, create a mean-and-error-bar plot (i.e., Tie-Fighter plot) similar to the [plot seen here](#). Be sure the plot has appropriate axes and labels, and is numbered and has a caption. Include this plot in a word-processed document. Resize the plot so it does not take up any more space than necessary and add an appropriate caption.

Part V

One more time, use the data in the `read` object you created in Part II. Now, you will use `ggplot()` to create a lineplot plot of the *average* number of pages read per month for each year of Andy's GoodReads data.

11. Again change the **dplyr** syntax from above to compute the average number of pages read per month—rather than the total number read per month (you do not need the standard deviation of the pages read per month this time). This time however, also group the output by year. (Hint: The output should again be a data frame with three columns.)
12. Using `ggplot()`, create a lineplot by month where each year's data is displayed in a separate line (six lines total). Read the *Mapping variable values to colors* section of the *Colors* page of the [Cookbook for R website](#). Be sure the plot has appropriate axes and labels, and is numbered and has a caption. Include this plot in a word-processed document. Resize the plot so it does not take up any more space than necessary and add an appropriate caption.