

STAT40620

Data Programming with R

Assignment 2

Instructions

- This assignment is due on Friday 8th November 2019 at 5pm.
- You should submit it to the ‘Assignment 2’ assignment object in Brightspace.
- You should submit two files only:
 1. `Rmd` file detailing the commented code you used to obtain your answers.
 2. final document in either `pdf` or `Word` which should contain answers to the questions below.
 - If you created an HTML file, please convert it to pdf. You can use Google Chrome: **File > Print > Destination [Change...] > select Save as PDF.**
- You may submit it multiple times before the deadline, but only the last version will be marked.
- There is a maximum of 10 marks for this assignment. This assignment is worth 10% of your final grade.
- The marks available for each question are shown in brackets.
- Late submissions will score 0, unless a “Late Submission of Coursework” form is submitted.
- You may have to discover and learn some new functions. Use `help()` and `help.search()` to find what you need.
- Some tips on using R Markdown are given at the end of this document.

The file `mly532.csv` contains the Historical Monthly Data recorded at the Dublin Airport Met Éireann Weather Observing Station from November 1941 to May 2019¹. The data set contains the following variables:

`year` Year
`month` Month
`rain` Precipitation Amount (mm)
`maxtp` Maximum Air Temperature (C)
`mintp` Minimum Air Temperature (C)
`mnmax` Mean Maximum Temperature (C)
`mnmin` Mean Minimum Temperature (C)
`gmin` Grass Minimum Temperature (C)
`wdsp` Mean Wind Speed (knot)
`maxgt` Highest Gust (knot)
`sun` Sunshine duration (hours)

1. Complete your assignment using R Markdown, check that all the output and code are correctly shown in your final document. Knit your document frequently to fix errors. Once completed, submit the `Rmd` file and the resulting `pdf` or `word` document which shows all your code. [0.5]
2. Load in the data as an object called `DublinAirport`. Notice that you have to skip the first 19 lines contained in the file before beginning to read data. Line 20 contains the column names. Display the structure of the dataset. [0.5]
3. Transform the column `months` to a factor with labels given by the month names. [Two hints: (1) look at the arguments `levels` and `labels` of the function `factor`. (2) you may want to make use of the built-in constant `month.name`] [1]
4. Use the `aggregate` function to compute which month has on average the highest and the lowest Precipitation Amount. [1]
5. Create a new column which contains a factor indicating the season:
 - Winter: December, January, February,
 - Spring: March, April, May,
 - Summer: June, July, August,
 - Autumn: September, October, November

[Hint: again, look at the arguments `levels` and `labels` of the function `factor`.] [1]

6. Assign to the `DublinAiport` object the classes `WeatherData` and `data.frame`. [0.5]

¹Source: <https://www.met.ie/climate/available-data/historical-data>

7. Write an S3 summary method for an object of class `WeatherData` which produces the following statistical summaries the `rain`, `maxtp`, `mintp`, `maxgt` variables split by season: mean, standard deviation, minimum, maximum. Ignore the missing values in the calculations.

Test your function on the `DublinAirport` data set and comment your findings. [2.5]

8. Create an S3 plot method for the class `WeatherData` that produces the following plots.

(1) Plot of the monthly Air Temperature (C) (`maxtp`, `mintp`).

(2) Plot of the Precipitation Amount (mm) (`rain`).

(3) Plot of the Highest Gust (knot) (`maxgt`).

- The user must be able to decide which years to plot. By default it will use the data from 2015 until 2018.
- The user must be able to decide which plot to draw (i.e, only one of the three, two of the three, or all three plots). By default the function will create all three plots.
- The plots must be on a single panel.
- The plots must have meaningful labels and/or titles, and a legend if needed.
- Test your function on the `DublinAirport` data set.

[3]

Tips for R Markdown

- Be aware that a common error is to give the same label to two different code chunks!

```
```{r cars}
summary(cars)
```
```

```
```{r cars}
plot(cars)
```
```

You can fix this by changing the label to one of them:

```
```{r cars2}
plot(cars)
```
```

- If you want to improve the appearance of your plot in your knitted document you can set up the dimension of your figure:

```
```{r, fig.height = 10, fig.width = 7, fig.align = "center"}
plot(Nile)
```
```

- In case of an error in your code, add the option `error = TRUE` into the R chunk to run the code, show the error message on the knitted file. For example:

```
```{r, error = TRUE}
x <- "a"
sum(a)
```
```

- For all the available options for the R chunk, you can see here: <https://yihui.name/knitr/options/>
- R Markdown website: <https://rmarkdown.rstudio.com/>
- R Markdown cheatsheet is available here: <https://www.rstudio.com/resources/cheatsheets/#rmarkdown>