Homework 1 in Introduction to R

Hi, Liis! This is your home assignment.

The grading might be somewhat unusual for you – it is partly individual, partly group-based. Group-work based grading is pretty much a standard in the business schools. The point is to make people work together, to give you some exposure to all the problems and benefits of group work, to give you some experience in how to manage collaboration. It might be stressful – your group-mate might turn out to be a total asshole etc (and I am really very sorry for this) – but this is exactly what life brings you. The grading will be a mild version of what business schools usually use – your final grade consists of 80% of what your own assignment is worth, and 20% of what your group-mates assignments are worth (in many of the business schools no one cares about what you personally did – only that the work is done). And I have deliberately assigned you to groups randomly – you'll have a chance to get to know each other :). Deal with this however you want – do all of your assignments together, or do your own assignment first and then come together to discuss where you got stuck[1]. I will only care if someone will be skipping the course altogether, otherwise it is up to you to help your group mates.

Your group is in Table 1.

Table 1: Your groupmates

name	email
Liis Roosaar	liis.roosaar@ut.ee
Lala Rustamli	rustamlilala@gmail.com
Yolandah Chido Chinyani	yolachinyani@gmail.com

It will take some time to do this, but doing it will be very much like what the real work with R would look like: thinking a bit, searching, trying, failing and trying again. Most, but not all of it would be covered in handouts.

Note that you all have different data and may have marginally different exercises – copying each others code verbatim will not work, **read your assignment carefully**. But the exercises are similar, so asking each other in case you are stuck will help.

Your excercise – messy data, dates and stuff

The data you see in the real world is usually messy. It has some unexpected values in it, the date formats are rather random etc.

I have generated some random data for you (each one of you has a different file with randomly different problems in it). I need you to read the data in, clean it, convert date variables to dates and graph it.

Your **personal** data file is at:

`http://www.ut.ee/~iseppo/A20783.csv`

1) What I want you to do first is to generate the following graph.

[1] I would very much advise you to think it through yourself though and not copy-paste. This is the only way to learn and you can do it.

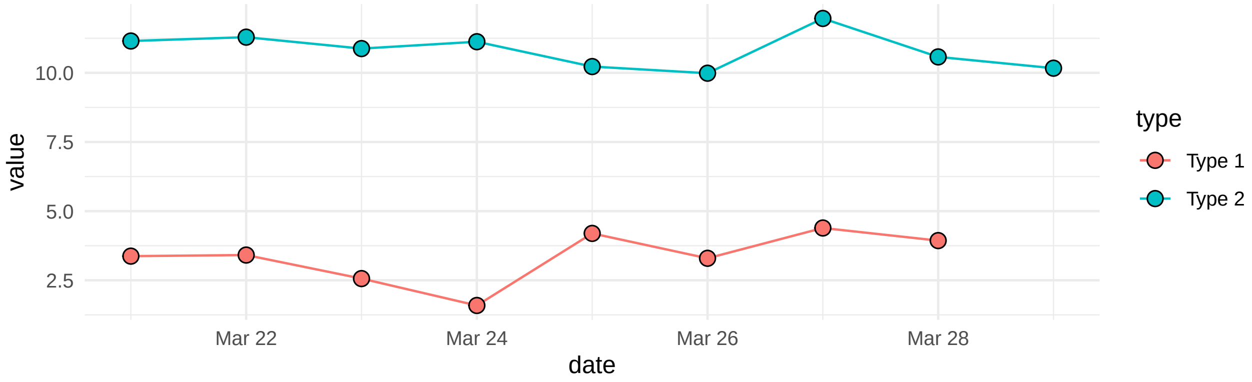

Figure 1: The first graph, 40% done

Save it as a png file with width of 8 inches and height 3 inches (scale=1), but don't upload this yet (this would only give you 50% of the points) – let's see if you can do more.

Tips:

- First read in the data[2] and take a look at it. Are there any funny values there? There will be, I've made sure of it. There are now couple of ways to deal with this – you can tell R to treat stuff that obviously is NA as NA-s while reading the data in, and/or you can read it in and convert the data to numeric[3].
- Take a look at the date variable. What kind of convention does it seem to follow? There are a number of different conventions in the world – most of the world uses DMY (day-month-year), but you'll find that in big parts of Asia they tend to use YMD, in the US the MDY convention and in actual data you'll see every possible and impossible way imaginable. Use the appropriate functions from the **lubridate** package[4] to convert your date variable so that R can understand it.
- There seem to be two geoms at the graphs – points and a line. The points are filled with a color, the lines are also of different color. Use a shape=21 for points, and use fill aesthetics for points (not color, as we used in the class – color usually colors the borders, fill fills the inside). Set the size of the points to 3.
- Also – add `theme_minimal()`.

2) 40% of the maximum is a nice to have, but lets see if you can get more. You can easily add 10% by **adding a title, subtitle and caption to the graph** (let the title be your name, subtitle "Graph for homework" and caption "Source: some randomly generated data" – if you can't make some special characters in your name to work then don't worry):

Tips:

[2] Remember – if `read.csv()` does not work, `read.csv2()` usually does

[3] Pay close attention that you do not convert a factor variable straight to numeric – you first need to convert it into character, and only then convert the character type to numeric – you will get wrong numbers otherwise! If you are using a newer version of R it will not read it in as factor variables, so you will not see this problem.

[4] Remember – there were functions like `ymd()`, `dmy()` etc available in the lubridate package, you should find some additional information from http://r4ds.had.co.nz/dates-and-times.html.

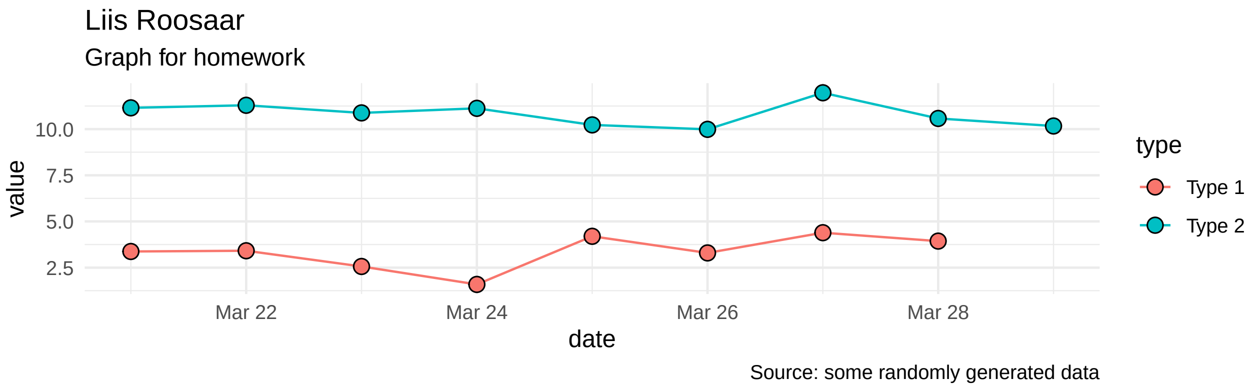

Figure 2: the second graph – 50% done

- Google

3) Now you are at 50%. Add another 10% by changeing the color of the lines and the inside color of the points so that Type 1 would be blue and Type2 would be black (as it is on Figure 3).

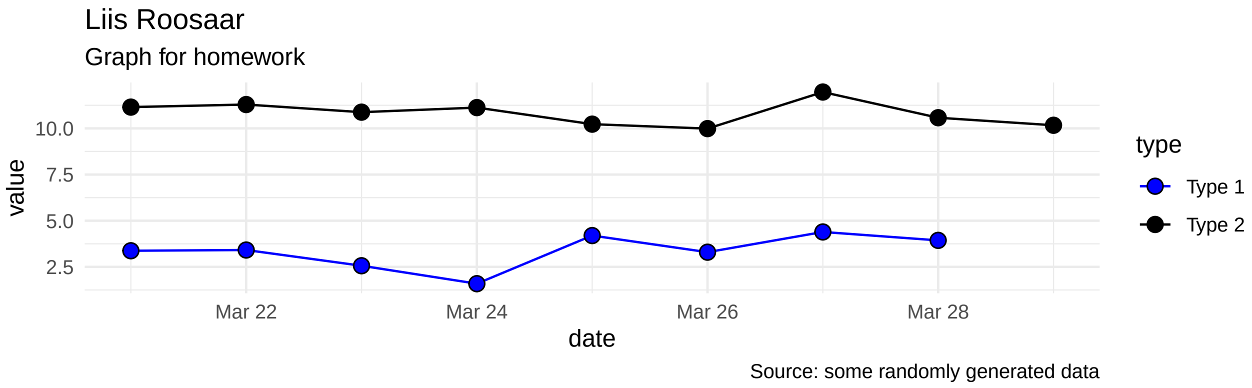

Figure 3: the third graph – 60% done

Tips:

- Think of how you set the colors here – which aesthetics did you use? I think you have used both fill and color. You will now need to use the corresponding `scale_xxx_manual()` to set the values yourself. We have done it in the classroom.

4) Another 10% will be gained by changing the orientation of x-axis labels and removing the x-axis title. There are many ways to do the latter – you can remove it entirely in a `theme()` call or just set it to be an empty string `("")`. Both work for us now. Cookbook for R will help you through either way. Another tip: if you are removing it in a theme() environment, add it **after the** `theme_minimal()`**-call** (if you have added the `theme_minimal()`, as I have).

Otherwise the `theme_minimal()` will override it again and you will see no changes!

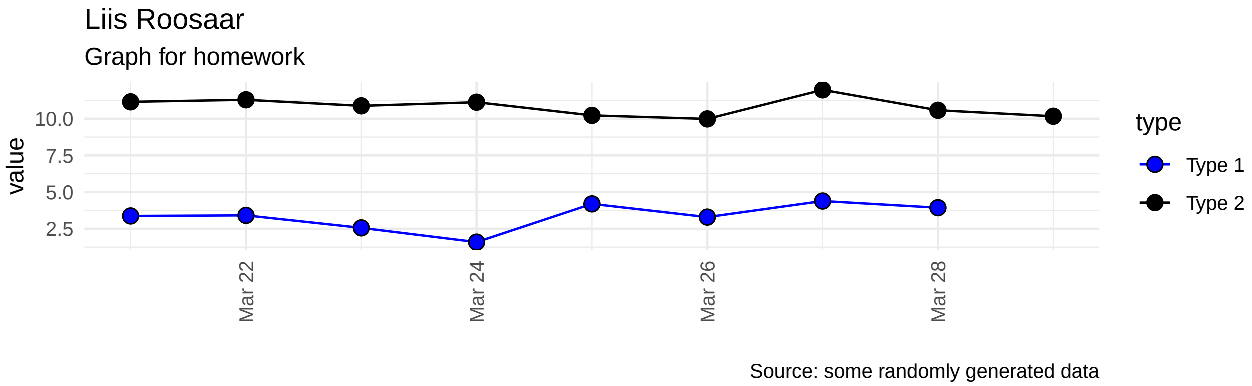

Figure 4: and we have a 70% done graph

5) Ok, you have 70% of the maximum already. The next 30% is actually just one line of code, but it is the hardest as I am not going to give you many tips of how you can manage this, other than you would need to use `scale_x_date()`, tell it to use `date_breaks()` so that between each break is **1 days** and specify the labels with `date_labels` – so that they would show either weekdays or months as on the graph you are seeing (you will find help either in R help system – `?strptime` should help to find the correct way to set this up – or on the internet)[5]. Check out Figure 5.

And it may show weekdays/months in your native language – it is absolutely ok. If you want them in different language (and in practice you will at some point – just to generate reports and graphs in different languages) you might start you inquires from here: `http://stackoverflow.com/questions/20577764/set-locale-to-system-default-utf-8`.

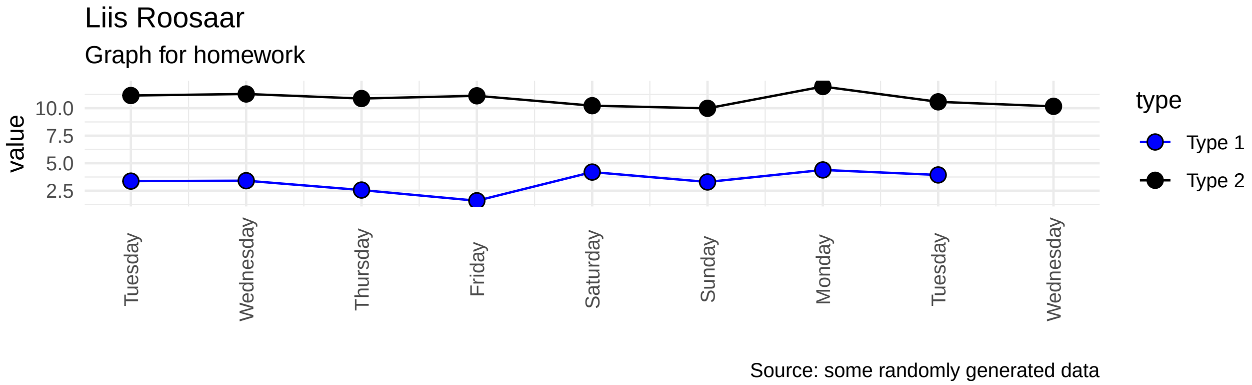

Figure 5: and we have a 100% done graph

Upload the **latest graph you did (no codes needed this time)** to Moodle (remember to save it so that it's width is set to 8, height to 3 and using scale=1). If

[5] If you are using some older versions of ggplot you might need to load in a **package called** `scales` for this to work, I don't really know – seems to work for me without this now.

you need any help, first ask your teammates. If this does not help (It is possible I have some major screw-up in the exercises), contact the teaching assistants or me. Please **start your subject line of the email as “R20”**, it is much easier to keep track this way!