Microsoft: DAT209x Programming in R for Data Science

12. Graphics in R > Lab > Lab

Bookmark

Consider the following data:

```
my.data<-data.frame(federal.states=c("Baden-Württemberg","Bayern","Berlin",
                                     "Brandenburg","Bremen","Hamburg","Hessen",
                                     "Mecklenburg-Vorpommern","Niedersachsen",
                                     "Nordrhein-Westfalen","Rheinland-Pfalz",
                                     "Saarland","Sachsen","Sachsen-Anhalt",
                                     "Schleswig-Holstein","Thüringen"),
                    Population=c(10716644,12691568,3469849,2457872,661888,1762791,
                    6093888,1599138,7826739,17638098,4011582,989035,4055274,
                    2235548,2830864,2156759))
```

The data consists of the names of the German 'Länder' and their populations as of December 31, 2014. Source:Statistiches Bundesamt, Germany.

Let's overlay a map of Germany with this information.

Question 1

(1/1 point)

Install and/or load the ggplot2 and ggmap package before you proceed.

Next, examine my.data using the `str()` function. Notice that the länder are stored as factor. Convert these to character.

Which command could you use to perform the task?

- my.data$federal.states<-is.character(my.data$federal.states)
- my.data$federal.states<-as.character(my.data$federal.states) ✔
- my.data$federal.states<-is.string(my.data$federal.states)
- my.data$federal.states<-as.string(my.data$federal.states)

EXPLANATION

Question 2

(1/1 point)

Next, get the geocodes for the German länder.

Which command could you use to perform the task?

- latlon <- get_map(my.data)
- latlon <- get_map(my.data$federal.states)
- latlon <- geocode(my.data)
- latlon <- geocode(my.data$federal.states) ✔

EXPLANATION

Question 3

(1/1 point)

You will received 2 warning messages. The request for geocodes failed for "Baden-ürttemberg" and "Thüringen". This is because `geocode()` cannot handle the German 'umlaut' (¨). In particular when dealing with letters which aren't standard English, you should be a little careful with `geocode()`. We therefore handle two locations separately. For Baden-Württemberg, we simply remove the umlaut. For Thüringen, we remove the umlaut, but also specify that the location is in Germany. Otherwise `geocode()` will find a location in western Austria; you should also be careful with possibly multiple location names.

Which two commands you need to run to fix this issue?

- [ ] my.data$federal.states[0]<-"Baden-Wurttemberg"
- [x] my.data$federal.states[1]<-"Baden-Wurttemberg" ✔
- [ ] my.data$federal.states[15]<-"Thuringen Germany"
- [x] my.data$federal.states[16]<-"Thuringen Germany" ✔

✔

EXPLANATION

Question 4

(1/1 point)

Get the geocodes for the German länder again. This time you shouldn't receive any warning messages. Assign the two variables from `latlon` to `my.data` respectively.

Which two commands could you use to perform the task?

- [ ] my.data <- latlon

- [x] my.data$lon <- latlon$lon; my.data$lat <- latlon$lat ✔
- [x] my.data <- cbind(my.data,latlon) ✔
- [ ] my.data <- rbind(my.data,latlon)

✔

EXPLANATION

Question 5

(1/1 point)

With this in place, proceed to get the geocodes of Germany and a raster map with a proper zoom factor.

Which command could you use to perform the task?

- ( ) Germany <- ggmap(get_map(location=my.data,zoom=6), extent="panel")
- (•) Germany <- ggmap(get_map(location="Germany",zoom=6), extent="panel") ✔

- Germany <- ggmap(get_map(location=Germany,zoom=6), extent="panel")
- Germany <- ggmap(get_map(location=germany_center,zoom=6), extent="panel")

EXPLANATION

Question 6

(1/1 point)

The last step is to overlay the map with the population sizes. Consider to set the following options: col for color, alpha for transparency, and size for the size of data according to the data frame's population.

```
circle_scale <- 0.000002
Germany + geom_point(aes(x=lon, y=lat),
                     data=  my.data  ,
                     col=   "red"    ,
                     alpha=  0.4
                     size=  my.data$pop
                            ulation      *circle_scale
```