Example Notebook

The code chunk below is the only special code for R Notebooks: We want to make sure that every code chunk uses the working directory of the R Project (and not nessarily the path where the Notebook is saved). A special feature of R Notebook / Markdown is that it uses diverges from the working directory of the project.

# Intro

Creating maps with ggplot is easy. You just have to supply longitudial data as the x variable and lattitudial data as the y variable and additionally give a group variable (so that R knows, which rows are to be filled together).

This kind of data (long, lat, group) is something which you can get from external sources (we present two options below). Then you have to merge this data with your values (e.g. what you want to fill into the map). So in addition, you need a fourth variable (like the code or the name of the geography).

# Setup

What do we always do first? Besides the tidyverse, we need the rwolrdmap package (which you have to install).

## Warning: package 'rworldmap' was built under R version 3.5.3

# Country level map with rworldmap

eurost <- read\_csv2("data/eurostat\_data.csv")  
  
eurost <- eurost %>%  
 filter(time==2014)  
  
mpdta <- map\_data("world")

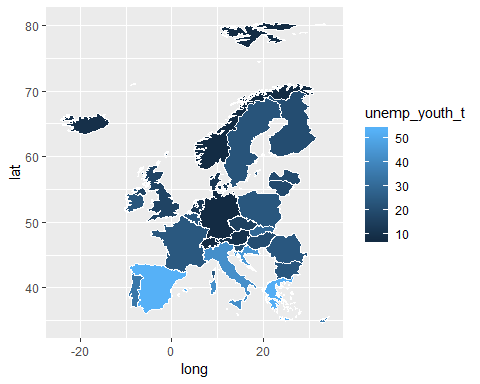
## Warning: package 'maps' was built under R version 3.5.3

eurost2 <- inner\_join(x=eurost, y=mpdta, by=c("geo\_name" = "region"))

The basic ggplot can be done with geom\_polygon. As we said, you supply longitude as x and lattitudial as y data, group them according to the group indicator and fill in your variable of interest (youth unemployment.)

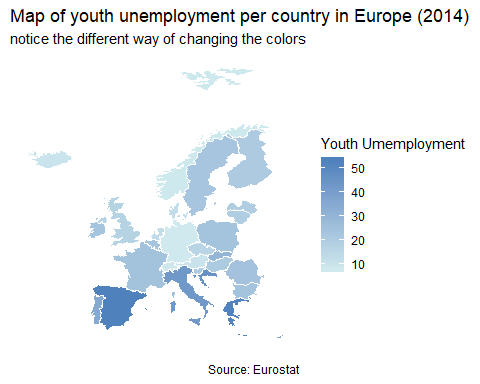
That’s it.

P <- ggplot()+   
 geom\_polygon(data = eurost2, aes(x=long, y = lat, group = group, fill=unemp\_youth\_t),  
 colour = "white", size = 0.1)  
P



Mapping is a normal ggplot operation like anything else. You can change the plot in the normal way.

P2 <- P+  
 theme(panel.grid.minor = element\_line(colour = NA),   
 panel.background = element\_rect(fill = NA, colour = NA),  
 axis.text.x = element\_blank(),  
 axis.text.y = element\_blank(), axis.ticks.x = element\_blank(),  
 axis.ticks.y = element\_blank(), axis.title = element\_blank())+  
 scale\_fill\_gradient(name = "Youth Umemployment",   
 low = rgb(207,233, 238, maxColorValue = 255),  
 high = rgb(79, 129, 189, maxColorValue = 255))+  
 labs(title = "Map of youth unemployment per country in Europe (2014)",  
 subtitle = "notice the different way of changing the colors",  
 caption = "Source: Eurostat")  
  
  
P2



# NUTS2 level plots with eurostat

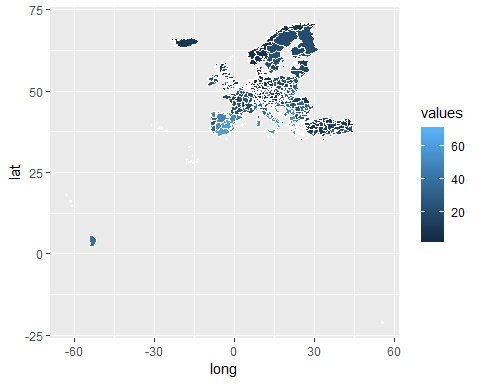
The eurstat package, which you already know by now, has the function get\_eurostat\_geospatial which allows you to retrieve a data frame with longitudial, lattitudial, and group data for maps on different levels of agglomeration.

The overall procedure is the same as above. Once we have data on long, lat, and group, we merge it with our data of interest.

library("eurostat")  
  
df <- get\_eurostat\_geospatial(output\_class = "df",   
 # data in a data.frame in order to be able to work with ggplot in a tidy way  
   
 resolution = "20",   
 # the higher the resolution, the more processing power needed  
   
 nuts\_level = "2"  
 # which NUTS level do you want  
 )   
  
data\_b2 <-  
 get\_eurostat(id = "yth\_empl\_110",  
 time\_format = "num")  
  
data\_b2 <- data\_b2 %>%  
 filter(age=="Y15-24",  
 sex=="T",  
 time==2014) %>%  
 subset(nchar(as.character(geo))>3)  
  
df2 <- inner\_join(data\_b2, df, by="geo")

And the plot in the exact same way.

new <- ggplot()+   
 geom\_polygon(data = df2, aes(x=long, y = lat, group = group, fill=values),  
 colour = "white", size = 0.1)  
  
new



Again, we simply modify this chart. To zoom into the plot, we use the function coord\_map which builds upon the mapproj package (which you have to install first).

library(mapproj)

## Warning: package 'mapproj' was built under R version 3.5.3

new2 <- new+  
 theme(panel.grid.minor = element\_line(colour = NA),   
 panel.background = element\_rect(fill = NA, colour = NA),  
 axis.text.x = element\_blank(),  
 axis.text.y = element\_blank(), axis.ticks.x = element\_blank(),  
 axis.ticks.y = element\_blank(), axis.title = element\_blank())+  
 scale\_fill\_gradient(name = "Youth Umemployment",   
 low = rgb(207,233, 238, maxColorValue = 255),  
 high = rgb(79, 129, 189, maxColorValue = 255))+  
 coord\_map(xlim = c(-30, 45), ylim = c(30, 75))+  
 labs(title = "Map of youth unemployment per NUTS2 region in Europe (2014)",  
 subtitle = "We zoom into main continental Europe with `coord\_map()` for better readability",  
 caption = "Source: Eurostat")  
new2

