This week, I’ll show you how to make a slight modification. I’ll show you how to highlight specific countries according to a variable in your data frame.

In this code, we will re-use our data:

**CODE: MAPPING OIL PRODUCTION BY COUNTRY**

#==============

# LOAD PACKAGES

#==============

library(tidyverse)

library(sf)

library(rvest)

library(stringr)

library(scales)

#library(viridis)

#============

# SCRAPE DATA

#============

df.oil <- read\_html("https://en.wikipedia.org/wiki/List\_of\_countries\_by\_oil\_production") %>%

html\_nodes("table") %>%

.[[1]] %>%

html\_table()

#====================

# CHANGE COLUMN NAMES

#====================

colnames(df.oil) <- c('rank', 'country', 'oil\_bbl\_per\_day')

#=============================

# WRANGLE VARIABLES INTO SHAPE

#=============================

#----------------------------------

# COERCE 'rank' VARIABLE TO INTEGER

#----------------------------------

df.oil <- df.oil %>% mutate(rank = as.integer(rank))

df.oil %>% glimpse()

#---------------------------------------------------

# WRANGLE FROM CHARACTER TO NUMERIC: oil\_bbl\_per\_day

#---------------------------------------------------

df.oil <- df.oil %>% mutate(oil\_bbl\_per\_day = oil\_bbl\_per\_day %>% str\_replace\_all(',','') %>% as.integer())

# inspect

df.oil %>% glimpse()

#===========================

#CREATE VARIABLE: 'opec\_ind'

#===========================

df.oil <- df.oil %>% mutate(opec\_ind = if\_else(str\_detect(country, 'OPEC'), 1, 0))

#=========================================================

# CLEAN UP 'country'

# - some country names are tagged as being OPEC countries

# and this information is in the country name

# - we will strip this information out

#=========================================================

df.oil <- df.oil %>% mutate(country = country %>% str\_replace(' \\(OPEC\\)', '') %>% str\_replace('\\s{2,}',' '))

# inspect

df.oil %>% glimpse()

#------------------------------------------

# EXAMINE OPEC COUNTRIES

# - here, we'll just visually inspect

# to make sure that the names are correct

#------------------------------------------

df.oil %>%

filter(opec\_ind == 1) %>%

select(country)

#==================

# REORDER VARIABLES

#==================

df.oil <- df.oil %>% select(rank, country, opec\_ind, oil\_bbl\_per\_day)

df.oil %>% glimpse()

#========

# GET MAP

#========

map.world <- map\_data('world')

df.oil

#==========================

# CHECK FOR JOIN MISMATCHES

#==========================

anti\_join(df.oil, map.world, by = c('country' = 'region'))

# rank country opec\_ind oil\_bbl\_per\_day

# 1 67 Congo, Democratic Republic of the 0 20,000

# 2 47 Trinidad and Tobago 0 60,090

# 3 34 Sudan and South Sudan 0 255,000

# 4 30 Congo, Republic of the 0 308,363

# 5 20 United Kingdom 0 939,760

# 6 3 United States 0 8,875,817

#=====================

# RECODE COUNTRY NAMES

#=====================

map.world %>%

group\_by(region) %>%

summarise() %>%

print(n = Inf)

# UK

# USA

# Democratic Republic of the Congo

# Trinidad

# Sudan

# South Sudan

df.oil <- df.oil %>% mutate(country = recode(country, `United States` = 'USA'

, `United Kingdom` = 'UK'

, `Congo, Democratic Republic of the` = 'Democratic Republic of the Congo'

, `Trinidad and Tobago` = 'Trinidad'

, `Sudan and South Sudan` = 'Sudan'

#, `Sudan and South Sudan` = 'South Sudan'

, `Congo, Republic of the` = 'Republic of Congo'

)

)

#-----------------------

# JOIN DATASETS TOGETHER

#-----------------------

map.oil <- left\_join( map.world, df.oil, by = c('region' = 'country'))

#=====

# PLOT

#=====

# BASIC (this is a first draft)

ggplot(map.oil, aes( x = long, y = lat, group = group )) +

geom\_polygon(aes(fill = oil\_bbl\_per\_day))

#=======================

# FINAL, FORMATTED DRAFT

#=======================

df.oil %>% filter(oil\_bbl\_per\_day > 822675) %>% summarise(mean(oil\_bbl\_per\_day))

# 3190373

df.oil %>% filter(oil\_bbl\_per\_day < 822675) %>% summarise(mean(oil\_bbl\_per\_day))

# 96581.08

ggplot(map.oil, aes( x = long, y = lat, group = group )) +

geom\_polygon(aes(fill = oil\_bbl\_per\_day)) +

scale\_fill\_gradientn(colours = c('#461863','#404E88','#2A8A8C','#7FD157','#F9E53F')

,values = scales::rescale(c(100,96581,822675,3190373,10000000))

,labels = comma

,breaks = c(100,96581,822675,3190373,10000000)

) +

guides(fill = guide\_legend(reverse = T)) +

labs(fill = 'bbl/day'

,title = 'Oil Production by Country'

,subtitle = 'Barrels per day, 2016'

,x = NULL

,y = NULL) +

theme(text = element\_text(family = 'Gill Sans', color = '#EEEEEE')

,plot.title = element\_text(size = 28)

,plot.subtitle = element\_text(size = 14)

,axis.ticks = element\_blank()

,axis.text = element\_blank()

,panel.grid = element\_blank()

,panel.background = element\_rect(fill = '#333333')

,plot.background = element\_rect(fill = '#333333')

,legend.position = c(.18,.36)

,legend.background = element\_blank()

,legend.key = element\_blank()

) +

annotate(geom = 'text'

,label = 'Source: U.S. Energy Information Administration\nhttps://en.wikipedia.org/wiki/List\_of\_countries\_by\_oil\_production'

,x = 18, y = -55

,size = 3

,family = 'Gill Sans'

,color = '#CCCCCC'

,hjust = 'left'

)

Before we do any plotting, let’s just inspect the data:

map.oil %>% glimpse()

# Observations: 99,338

# Variables: 9

# $ long -69.89912, -69.89571, -69.94219, -70.00415, -70.06612,...

# $ lat 12.45200, 12.42300, 12.43853, 12.50049, 12.54697, 12.5...

# $ group 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, ...

# $ order 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 16, 17,...

# $ region "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", "Aruba", ...

# $ subregion NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...

# $ rank NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...

# $ opec\_ind 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...

# $ oil\_bbl\_per\_day NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA...

So what do we have here?

The data contains variables that enable us to plot countries as polygons: long, lat, region, and group.

The data also contains a variable called oil\_bbl\_per\_day, which is essentially the amount of oil produced by the country per day.

Let’s make a basic map that plots this data, with each country “filled in” according to the amount of oil it produces.

#=====

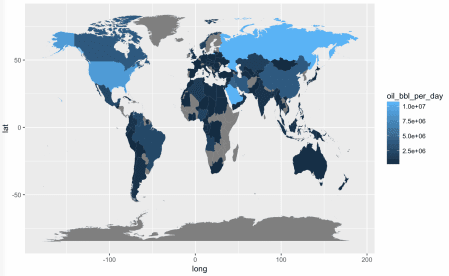
# PLOT

#=====

# BASIC (this is a first draft)

ggplot(map.oil, aes( x = long, y = lat, group = group )) +

geom\_polygon(aes(fill = oil\_bbl\_per\_day))

[[](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)

Next, let’s make a slightly different map. Here, we’re going to remove the mapping to the fill aesthetic, and we’ll going to map a different variable – opec\_ind – to the color aesthetic.

#----------------------------------------------

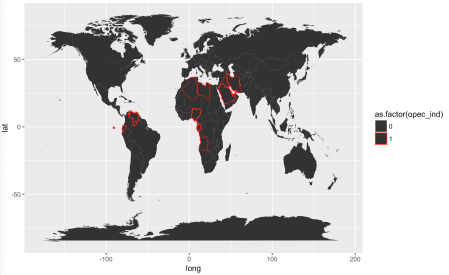
# PLOT with red highlight around OPEC countries

#----------------------------------------------

ggplot(map.oil, aes( x = long, y = lat, group = group )) +

geom\_polygon(aes(color = as.factor(opec\_ind))) +

scale\_color\_manual(values = c('1' = 'red', '0' = NA))

[[](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)

Essentially, what we’ve done here, is used the color aesthetic in combination with scale\_color\_manual() to manipulate the border color of the countries. Specifically, we have just highlighted OPEC countries with the color red.

Now, let’s combine the two techniques: we will fill in the color of the countries using the fill aesthetic, and we will highlight the OPEC countries by mapping a variable to the color aesthetic.

#-------------------------------------------

# PLOT

# - red highlight for OPEC

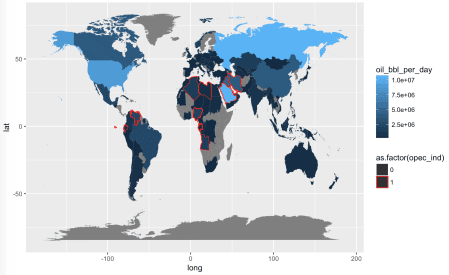
# - fill value corresponds to oil production

#-------------------------------------------

ggplot(map.oil, aes( x = long, y = lat, group = group )) +

geom\_polygon(aes(color = as.factor(opec\_ind), fill = oil\_bbl\_per\_day)) +

scale\_color\_manual(values = c('1' = 'red', '0' = NA))

[[](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)

There’s more that we’ll need to do to create the finalized version, but all things considered, this is pretty good. It essentially shows the information we want to display … it just needs some formatting.

So, now lets create the final, formatted map.

#=====================

# FINAL, FORMATTED MAP

#=====================

ggplot(map.oil, aes( x = long, y = lat, group = group )) +

geom\_polygon(aes(fill = oil\_bbl\_per\_day, color = as.factor(opec\_ind))) +

scale\_fill\_gradientn(colours = c('#461863','#404E88','#2A8A8C','#7FD157','#F9E53F')

,values = scales::rescale(c(100,96581,822675,3190373,10000000))

,labels = comma

,breaks = c(100,96581,822675,3190373,10000000)

) +

guides(fill = guide\_legend(reverse = T)) +

labs(fill = 'Barrels per day\n2016'

,color = 'OPEC Countries'

,title = 'OPEC countries produce roughly 44% of world oil'

,x = NULL

,y = NULL) +

theme(text = element\_text(family = 'Gill Sans', color = '#EEEEEE')

,plot.title = element\_text(size = 28)

,plot.subtitle = element\_text(size = 14)

,axis.ticks = element\_blank()

,axis.text = element\_blank()

,panel.grid = element\_blank()

,panel.background = element\_rect(fill = '#333333')

,plot.background = element\_rect(fill = '#333333')

,legend.position = c(.18,.36)

,legend.background = element\_blank()

,legend.key = element\_blank()

) +

annotate(geom = 'text'

,label = 'Source: U.S. Energy Information Administration\nhttps://en.wikipedia.org/wiki/List\_of\_countries\_by\_oil\_production\nhttps://en.wikipedia.org/wiki/OPEC'

,x = 18, y = -55

,size = 3

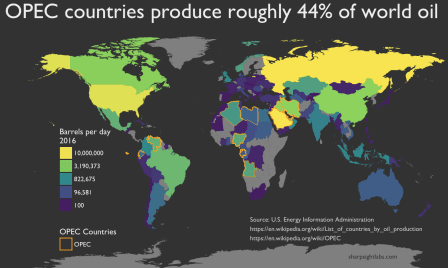
,family = 'Gill Sans'

,color = '#CCCCCC'

,hjust = 'left'

) +

scale\_color\_manual(values = c('1' = 'orange', '0' = NA), labels = c('1' = 'OPEC'), breaks = c('1'))

[[](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)](http://sharpsightlabs.com/join-newsletter-rapid-mastery/)

Let’s point out a few things.

First, the fill color scale has been carefully crafted to optimally show differences between countries.

Second, we are simultaneously using the highlighting technique to highlight the OPEC countries.

Finally, notice that we’re using the title to “tell a story” about the highlighted data.

All told, there is a lot going on in this example.

**Sign up now, and discover how to rapidly master data science**

It’s possible to learn and master data science tools faster than you thought possible.

Even though the example in this blog post is complicated, it is very easy to learn to create visualizations like this, if you know what tools to learn, how to practice those tools, and how to put those tools together.

Sharp Sight is dedicated to teaching you how to master the tools of data science as quickly as possible. We teach data science, but we also *teach you how to learn*.