Maps in R!

ST465– Making Maps in R

Today we'll go over how to make maps in R and how to work with spatial data sets. Some of the key R packages that we'll work with are `tmap`, `mapview` for map making and `sf`, `spData`, `spDataLarge` for working with spatial data.

```
## R packages
setwd(getwd())
library(rmarkdown)
library(tmap) ## Very commonly used -- to make static and interactive plots
library(mapview)
library(leaflet)
library(shiny)
library(tidyverse)
library(spDataLarge) ## Data sets -- use commands in notes to download
library(spData) ## Data sets
library(sf) ## simple features
```

tmap

Let's work with the `world` and `coffee` data sets. For each, identify the type of object it is and its coordinate reference system (CRS).

```
#data("world")
world
```

```
## Simple feature collection with 177 features and 10 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -89.9 xmax: 180 ymax: 83.64513
## Geodetic CRS:  WGS 84
## # A tibble: 177 x 11
##    iso_a2 name_long        continent     region_un subregion          type                area_km2
##  * <chr>  <chr>            <chr>         <chr>     <chr>              <chr>                  <dbl>
##  1 FJ     Fiji             Oceania       Oceania   Melanesia          Sovereign country    19290.
##  2 TZ     Tanzania         Africa        Africa    Eastern Africa     Sovereign country   932746.  52
##  3 EH     Western Sahara   Africa        Africa    Northern Africa    Indeterminate         96271.
##  4 CA     Canada           North America Americas  Northern America   Sovereign country 10036043.  35
##  5 US     United States    North America Americas  Northern America   Country            9510744. 318
##  6 KZ     Kazakhstan       Asia          Asia      Central Asia       Sovereign country  2729811.  17
##  7 UZ     Uzbekistan       Asia          Asia      Central Asia       Sovereign country   461410.  30
##  8 PG     Papua New Guinea Oceania       Oceania   Melanesia          Sovereign country   464520.   7
##  9 ID     Indonesia        Asia          Asia      South-Eastern Asia Sovereign country  1819251. 255
## 10 AR     Argentina        South America Americas  South America      Sovereign country  2784469.  42
## # ... with 167 more rows
```

```
coffee_data
```

```
## # A tibble: 47 x 3
##    name_long                coffee_production_2016 coffee_production_2017
##    <chr>                                     <int>                  <int>
##  1 Angola                                       NA                     NA
##  2 Bolivia                                       3                      4
##  3 Brazil                                     3277                   2786
##  4 Burundi                                      37                     38
##  5 Cameroon                                      8                      6
##  6 Central African Republic                     NA                     NA
##  7 Congo, Dem. Rep. of                           4                     12
##  8 Colombia                                   1330                   1169
##  9 Costa Rica                                   28                     32
## 10 Côte d'Ivoire                               114                    130
## # ... with 37 more rows
```

```
world_coffee = dplyr::left_join(world, coffee_data, by = "name_long")
world_coffee
```

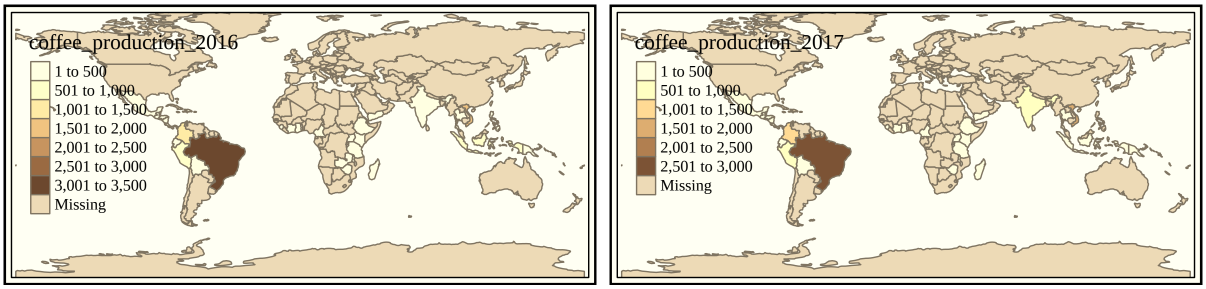
```
## Simple feature collection with 177 features and 12 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -89.9 xmax: 180 ymax: 83.64513
## Geodetic CRS:  WGS 84
## # A tibble: 177 x 13
##    iso_a2 name_long        continent     region_un subregion          type              area_~1     po
##    <chr>  <chr>            <chr>         <chr>     <chr>              <chr>               <dbl>   <dbl
##  1 FJ     Fiji             Oceania       Oceania   Melanesia          Sovereign count~  1.93e4  8.86
##  2 TZ     Tanzania         Africa        Africa    Eastern Africa     Sovereign count~  9.33e5  5.22
##  3 EH     Western Sahara   Africa        Africa    Northern Africa    Indeterminate     9.63e4 NA
##  4 CA     Canada           North America Americas  Northern America   Sovereign count~  1.00e7  3.55
##  5 US     United States    North America Americas  Northern America   Country           9.51e6  3.19
##  6 KZ     Kazakhstan       Asia          Asia      Central Asia       Sovereign count~  2.73e6  1.73
##  7 UZ     Uzbekistan       Asia          Asia      Central Asia       Sovereign count~  4.61e5  3.08
##  8 PG     Papua New Guinea Oceania       Oceania   Melanesia          Sovereign count~  4.65e5  7.76
##  9 ID     Indonesia        Asia          Asia      South-Eastern Asia Sovereign count~  1.82e6  2.55
## 10 AR     Argentina        South America Americas  South America      Sovereign count~  2.78e6  4.30
## # ... with 167 more rows, and abbreviated variable names 1: area_km2, 2: gdpPercap, 3: coffee_produc
```

We've created a new data set called world_coffee, what type of object is it and what is its CRS?

Let's plot the spatial object using the R package `tmap`.

```
## tmap_mode "view" set to interactive viewing
tmap_mode("plot")
facets = c("coffee_production_2016", "coffee_production_2017")

tm_shape(world_coffee) +
  tm_polygons(facets) +
  tm_facets(nrow = 1, sync = TRUE)
```



Now let's use the `World` data set (not that `world` and `World` are different data sets.)

```
data("World")
World
```

```
## Simple feature collection with 177 features and 15 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -89.9 xmax: 180 ymax: 83.64513
## Geodetic CRS:  WGS 84
## First 10 features:
##    iso_a3                   name           sovereignt                continent                area  po
## 1     AFG            Afghanistan          Afghanistan                     Asia   652860.000 [km^2] 284
## 2     AGO                 Angola               Angola                   Africa  1246700.000 [km^2] 127
## 3     ALB                Albania              Albania                   Europe    27400.000 [km^2]  36
## 4     ARE   United Arab Emirates United Arab Emirates                     Asia    71252.172 [km^2]  47
## 5     ARG              Argentina            Argentina            South America  2736690.000 [km^2] 409
## 6     ARM                Armenia              Armenia                     Asia    28470.000 [km^2]  29
## 7     ATA             Antarctica           Antarctica               Antarctica 12259213.973 [km^2]
## 8     ATF Fr. S. Antarctic Lands               France Seven seas (open ocean)     7257.455 [km^2]
## 9     AUS              Australia            Australia                  Oceania  7682300.000 [km^2] 212
## 10    AUT                Austria              Austria                   Europe    82523.000 [km^2]  82
##                 income_grp gdp_cap_est life_exp well_being footprint inequality      HPI
## 1             5. Low income    784.1549   59.668        3.8      0.79 0.42655744 20.22535 MULTIPOLYGON
## 2   3. Upper middle income   8617.6635       NA         NA        NA         NA       NA MULTIPOLYGON
## 3   4. Lower middle income   5992.6588   77.347        5.5      2.21 0.16513372 36.76687 MULTIPOLYGON
## 4  2. High income: nonOECD  38407.9078       NA         NA        NA         NA       NA MULTIPOLYGON
```

```
## 5   3. Upper middle income  14027.1261   75.927        6.5      3.14 0.16423830 35.19024 MULTIPOLYGON
## 6   4. Lower middle income   6326.2469   74.446        4.3      2.23 0.21664810 25.66642 MULTIPOLYGON
## 7  2. High income: nonOECD 200000.0000       NA         NA        NA         NA       NA MULTIPOLYGON
## 8  2. High income: nonOECD 114285.7143       NA         NA        NA         NA       NA MULTIPOLYGON
## 9     1. High income: OECD  37634.0832   82.052        7.2      9.31 0.08067825 21.22897 MULTIPOLYGON
## 10    1. High income: OECD  40132.6093   81.004        7.4      6.06 0.07129351 30.47822 MULTIPOLYGON
```

HPI stands for “Happy Planet Index” and provides a quantification of a country’s “happiness”. More information here: https://happyplanetindex.org Let’s check it out:

```
tmap_mode("view")

tm_shape(World) +
    tm_polygons("HPI")
```

Let’s look at the HPI along with economy type:

```
tmap_mode("view")
tm_shape(World) +
    tm_polygons(c("HPI", "economy")) +
    tm_facets(sync = TRUE, ncol = 2)
```

We continue to make maps, this time with the `metro` data set. What information is provided? What is the geometry type and what is the CRS?

```
## reading in data
data(metro)
metro
```

```
## Simple feature collection with 436 features and 12 fields
## Geometry type: POINT
## Dimension:     XY
## Bounding box:  xmin: -123.1193 ymin: -37.814 xmax: 174.7767 ymax: 60.16925
## Geodetic CRS:  WGS 84
## First 10 features:
##            name            name_long iso_a3 pop1950 pop1960 pop1970 pop1980  pop1990  pop2000  pop20
## 2         Kabul                Kabul    AFG  170784  285352  471891  977824  1549320  2401109  37223
## 8       Algiers El Djazair  (Algiers)    DZA  516450  871636 1281127 1621442  1797068  2140577  24320
## 13       Luanda               Luanda    AGO  138413  219427  459225  771349  1390240  2591388  45084
## 16 Buenos Aires         Buenos Aires    ARG 5097612 6597634 8104621 9422362 10513284 12406780 142458
## 17      Cordoba              Cordoba    ARG  429249  605309  809794 1009521  1200168  1347561  14592
## 25      Rosario              Rosario    ARG  554483  671349  816230  953491  1083819  1152387  12980
## 32      Yerevan              Yerevan    ARM  341432  537759  778158 1041587  1174524  1111301  10655
## 33     Adelaide             Adelaide    AUS  429277  571822  850168  971856  1081618  1141623  12179
## 34     Brisbane             Brisbane    AUS  441718  602999  904777 1134833  1381306  1666203  20330
## 37    Melbourne            Melbourne    AUS 1331966 1851220 2499109 2839019  3154314  3460541  39511
```

```
tmap_mode("view")
tm_basemap("Stamen.Watercolor") +
tm_shape(metro) + tm_bubbles(size = "pop2020", col = "red") +
tm_tiles("Stamen.TonerLabels")
```

You can also play around with the style in `tmap`.

```
tmap_style("classic")
## tmap style set to "classic"
## other available styles are: "white", "gray", "natural", "cobalt", "col_blind", "albatross", "beaver"

tm_shape(World) +
    tm_polygons("HPI", legend.title = "Happy Planet Index")
```

We can also look through some data from New Zealand. As the others, what type of object is it and what is the CRS? How it different from others and why might that be the case?

```
## New Zealand
nz
```

```
## Simple feature collection with 16 features and 6 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: 1090144 ymin: 4748537 xmax: 2089533 ymax: 6191874
## Projected CRS: NZGD2000 / New Zealand Transverse Mercator 2000
## First 10 features:
##                 Name Island Land_area Population Median_income Sex_ratio                      ge
## 1          Northland  North 12500.561     175500         23400 0.9424532 MULTIPOLYGON (((1745493 600
## 2           Auckland  North  4941.573    1657200         29600 0.9442858 MULTIPOLYGON (((1803822 590
## 3            Waikato  North 23900.036     460100         27900 0.9520500 MULTIPOLYGON (((1860345 585
## 4      Bay of Plenty  North 12071.145     299900         26200 0.9280391 MULTIPOLYGON (((2049387 583
## 5           Gisborne  North  8385.827      48500         24400 0.9349734 MULTIPOLYGON (((2024489 567
## 6         Hawke's Bay  North 14137.524     164000         26100 0.9238375 MULTIPOLYGON (((2024489 567
## 7           Taranaki  North  7254.480     118000         29100 0.9569363 MULTIPOLYGON (((1740438 571
## 8  Manawatu-Wanganui  North 22220.608     234500         25000 0.9387734 MULTIPOLYGON (((1866732 566
## 9         Wellington  North  8048.553     513900         32700 0.9335524 MULTIPOLYGON (((1881590 548
## 10        West Coast  South 23245.456      32400         26900 1.0139072 MULTIPOLYGON (((1557042 531
```

```
legend_title = expression("Area (km"^2*")")
map_nza = tm_shape(nz) +
  tm_fill(col = "Land_area", title = legend_title) + tm_borders()
map_nza + tm_style("bw")
```

```
map_nza + tm_style("classic")
```

```
map_nza + tm_style("cobalt")
```

```
map_nza + tm_style("col_blind")
```

mapview

Let's use `mapview` now, another common R package for constructing spatial maps. We'll use the `trails` and `franconia` data set. What kind of objects are they and what is its CRS?

```
trails
```

```
## Simple feature collection with 543 features and 3 fields
## Geometry type: MULTILINESTRING
## Dimension:     XY
## Bounding box:  xmin: 528933.9 ymin: 5415754 xmax: 715816.7 ymax: 5597457
## Projected CRS: WGS 84 / UTM zone 32N
## First 10 features:
##                                                                   FGN    FKN    district
## 1                                                 003756/Kunigundenweg 003756 Oberfranken MULTILINESTI
## 2  004037/Jakobsweg (Almerswind-Coburg-Lichtenfels-Bamberg-Nuernberg) 004037 Oberfranken MULTILINESTI
## 3                005160/Steigerwaelder Jakobsweg (Bamberg-Uffenheim) 005160 Oberfranken MULTILINESTI
## 4                                      022650/Sieben-Fluesse-Wanderweg 022650 Oberfranken MULTILINESTI
## 5             005316/Rennsteigverein 1896 e.V. / Bamberger Rennsteig 005316 Oberfranken MULTILINESTI
## 6                                       012029/Steigerwald-Panoramaweg 012029 Oberfranken MULTILINESTI
## 7                             013633/Frankenwaldverein / Markgrafenweg 013633 Oberfranken MULTILINESTI
## 8                                          023964/Rot-Main-Auen-Weg 023964 Oberfranken MULTILINESTI
## 9                                                007443/Jean-Paul-Weg 007443 Oberfranken MULTILINESTI
## 10                                               007443/Jean-Paul-Weg 007443 Oberfranken MULTILINESTI
```

```
franconia
```

```
## Simple feature collection with 37 features and 6 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: 8.975926 ymin: 48.8625 xmax: 12.27535 ymax: 50.56422
## Geodetic CRS:  WGS 84
## First 10 features:
##    NUTS_ID  SHAPE_AREA SHAPE_LEN CNTR_CODE                 NAME_ASCI                       geometry
## 1    DE241 0.006736012 0.3926225        DE  Bamberg, Kreisfreie Stadt MULTIPOLYGON (((10.92581 49...
## 2    DE242 0.008424469 0.6247263        DE Bayreuth, Kreisfreie Stadt MULTIPOLYGON (((11.58157 49...
## 3    DE243 0.005982341 0.5185471        DE   Coburg, Kreisfreie Stadt MULTIPOLYGON (((10.95355 50...
## 4    DE244 0.007329480 0.4569815        DE      Hof, Kreisfreie Stadt MULTIPOLYGON (((11.93067 50...
## 5    DE245 0.146698316 3.4819699        DE         Bamberg, Landkreis MULTIPOLYGON (((10.87615 50...
## 6    DE246 0.159489736 3.6242023        DE        Bayreuth, Landkreis MULTIPOLYGON (((11.70657 50...
## 7    DE247 0.074698748 2.6954234        DE          Coburg, Landkreis MULTIPOLYGON (((10.88654 50...
## 8    DE248 0.079746707 1.7712298        DE                  Forchheim MULTIPOLYGON (((11.26376 49...
## 9    DE249 0.112934151 2.7544708        DE             Hof, Landkreis MULTIPOLYGON (((11.91988 50...
## 10   DE24A 0.081960299 1.9393830        DE                    Kronach MULTIPOLYGON (((11.36979 50...
```

What does it look like if we only plot the trails data?

```
trails%>%
  mapview()
```

What about the franconia data set?

```
franconia%>%
  mapview()
```

We'd like to plot the trails onto the map, but they're not the same CRS. In lecture, we saw that we can use a suite of functions starting with `st_` to modify a spatial object. We'll use a few here to change the crs and plot the data sets together:

```r
## first, using st_transform() function we changed the CRS
trails %>%
  st_transform(st_crs(franconia)) %>%
  st_intersection(franconia[franconia$district == "Oberfranken", ]) %>%
  st_collection_extract("LINE") %>%
  mapview(color = "red", lwd = 3, layer.name = "trails") +
  mapview(franconia, zcol = "district", burst = TRUE)
```

Lastly, suppose you also wanted to know about any breweries along the trails. Let's check out the `breweries` data set. What kind of object is it and what is its CRS?

```r
breweries
```

What does a map of the breweries look like?

```r
breweries%>%
  mapview()
```

Let's combine them all:

```r
trails %>%
  st_transform(st_crs(franconia)) %>%
  st_intersection(franconia[franconia$district == "Oberfranken", ]) %>%
  st_collection_extract("LINE") %>%
  mapview(color = "red", lwd = 3, layer.name = "trails") +
  mapview(franconia, zcol = "district", burst = TRUE) +
  breweries
```

tmap

What if we tried to do this in `tmap`?

```r
tm_shape(franconia) + tm_fill("district") +
tm_shape(trails) + tm_lines() +
tm_shape(breweries) + tm_dots()
```