ESM 263 Assignment 1: Cartography

Linus Blomqvist

1/19/2021

First we'll load the data. The `sf` function `st_read` understands what format the file is so it needs no further specification.

```
hw1 <- st_read("data/HW1/HW1.gpkg", layer = "CountryWatch")
```

```
## Reading layer `CountryWatch' from data source `/Users/linusblomqvist/Dropbox/0 UCSB/1 ESM 263/esm-263
## Simple feature collection with 250 features and 61 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -180 ymin: -90 xmax: 180 ymax: 83.6236
## geographic CRS: WGS 84
```

We can verify that we now have an `sf` object:

```
class(hw1)
```

```
## [1] "sf"         "data.frame"
```

It's a good idea to check if the data is projected, which it is not. `TRUE` here means that it is in long/lat form, which is not projected.

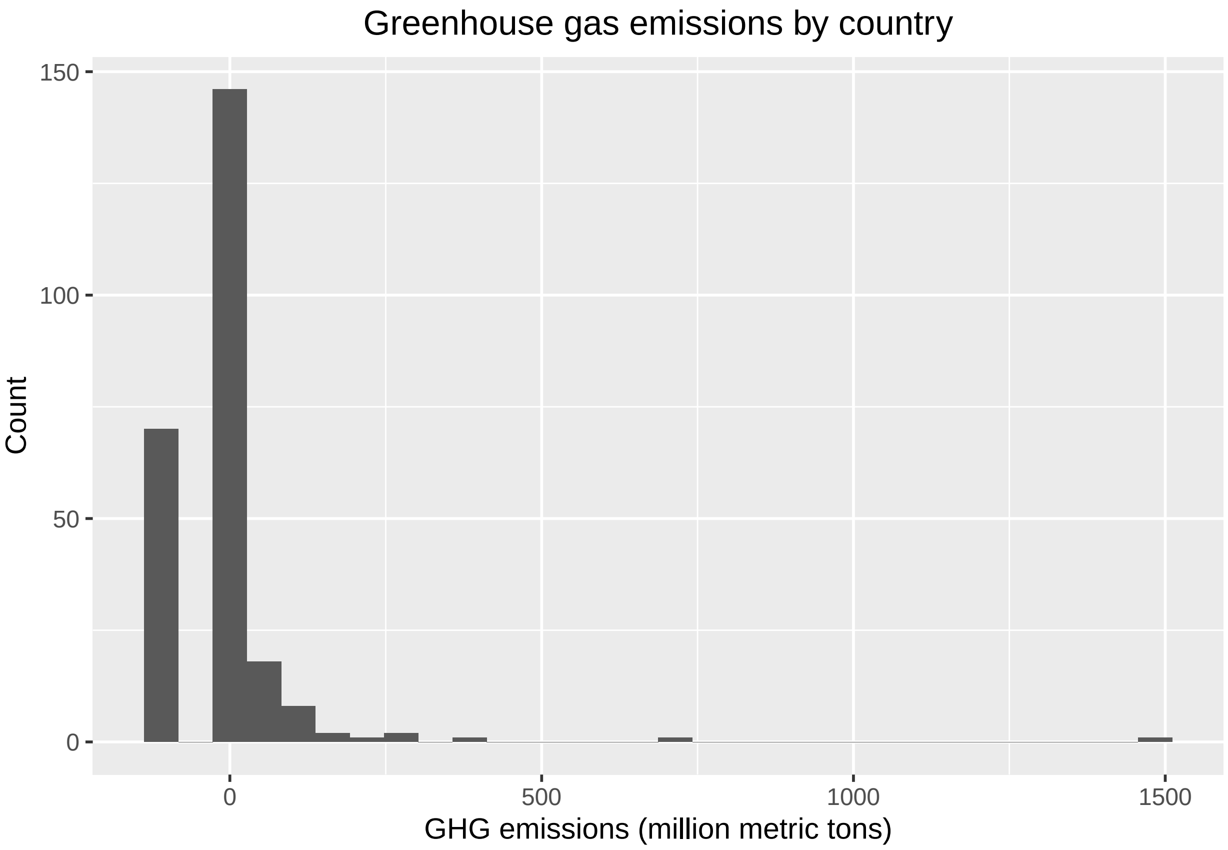
```
st_is_longlat(hw1)
```

```
## [1] TRUE
```

For this assignment, I don't think we need to project the sf object, but we would need to do that if we wanted to calculate things like area.

I will map the greenhouse gas emissions variable. Before I do so, I'll do some sanity checks, starting with a histogram.

```
hw1 %>%
  ggplot(aes(GGAS_EMS98)) +
    geom_histogram() +
    labs(title = "Greenhouse gas emissions by country",
         x = "GHG emissions (million metric tons)", y = "Count") +
    theme(plot.title = element_text(hjust = 0.5))
```



There are 70 values below zero and they are all -99.

```
hw1 %>%
  filter(GGAS_EMS98 < 0) %$%
  unique(GGAS_EMS98) # unique values less than 0
```

```
## [1] -99
```

According to the documentation, this means there's no data. This is better represented with `NA`, so I'll change that. I'll make a new variable just to preserve the original.

```
hw1$ggas <- hw1$GGAS_EMS98
hw1$ggas[hw1$ggas < 0] <- NA
```

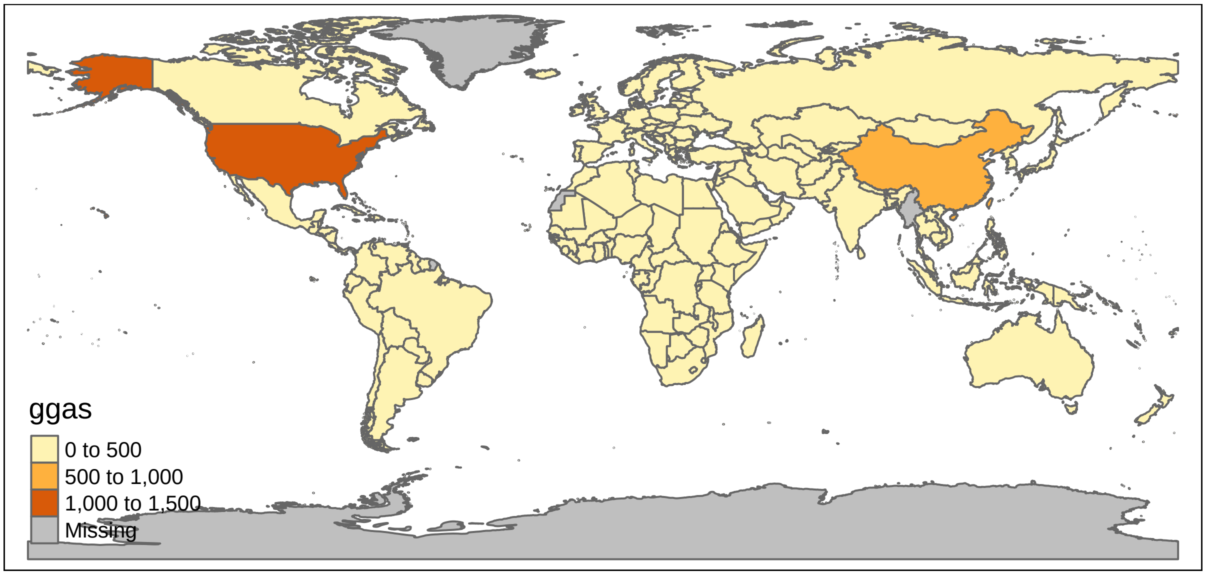
I'm also curious which countries have the extremely large emissions.

```
hw1 %>%
  filter(ggas > 300) %$%
  CNTRY_NAME
```

```
## [1] "Russia"        "United States" "China"
```

Not surprising. Now we can start building the map. I'll make one with all the `tmap` defaults just to get a picture of what we have and what the map needs to look like.

```
tm_shape(hw1) +
  tm_polygons("ggas")
```

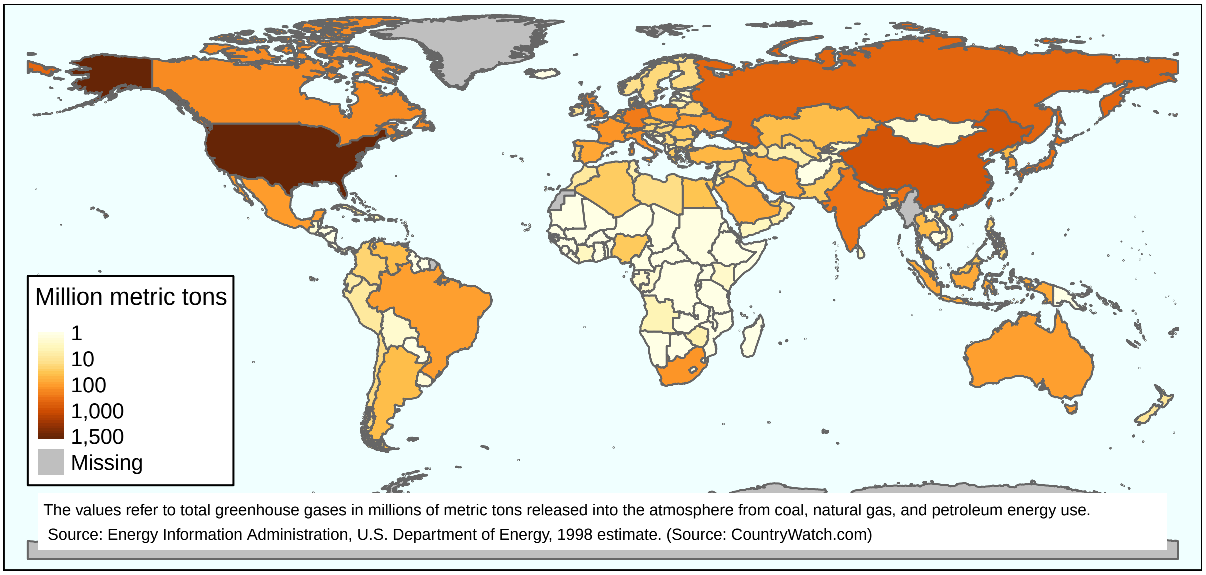


Two things to note. First, we want what is referred to as sequential palette by the package `RColorBrewer`. I chose one where the color goes from yellow to brown. This seems appropriate for pollution. Second, the data are really skewed, so the map will be easier to read if plotted on a log scale.

I'm also including a definition and credits at the bottom.

```
tm_shape(hw1) +
  tm_polygons(col = "GGAS_EMS98",
              style = "log10",
              title = "Million metric tons",
              palette = "YlOrBr",
              breaks = c(0,1,2,3, log10(1500)))+
  tm_layout(main.title = "Greenhouse gas emissions by country",
            main.title.position = c("center", "TOP"),
            main.title.size = 1.2,
            legend.position = c(0.02, 0.15),
            legend.title.size = 0.9,
            legend.text.size = 0.7,
            legend.bg.color = "white",
            legend.frame = TRUE,
            bg.color = "azure1") +
  tm_credits("The values refer to total greenhouse gases in millions of metric tons released into the a
             position = c("center", "bottom"),
             bg.color = "white")
```

Greenhouse gas emissions by country



The values refer to total greenhouse gases in millions of metric tons released into the atmosphere from coal, natural gas, and petroleum energy use.
Source: Energy Information Administration, U.S. Department of Energy, 1998 estimate. (Source: CountryWatch.com)