Task 2.2

Harisankar Krishnadas

2023-12-13

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(sf)
```

```
## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
```

```
library(ggplot2)
library(rnaturalearth)
```

```
## Support for Spatial objects (`sp`) will be deprecated in {rnaturalearth} and will be removed in a fu
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
##
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
##
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
##
## The following object is masked from 'package:purrr':
##
##     transpose
```

```r
library(dplyr)
library(patchwork)
library(viridis)
```

```
## Loading required package: viridisLite
```

```r
library(grid)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```r
library(svglite)
```

2.2.1

```r
gadm2 <- read.csv('gadm2_aqli_1998_2021.csv')
gadm2_shp <- st_read('./gadm2_aqli_shapefile/aqli_gadm2_final_june302023.shp')
```

```
## Reading layer 'aqli_gadm2_final_june302023' from data source
##   '/Users/harisankar/Work/Harris/Job/EPIC India/Task 2.2/gadm2_aqli_shapefile/aqli_gadm2_final_june3
##   using driver 'ESRI Shapefile'
## Simple feature collection with 48155 features and 4 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -180 ymin: -59.48428 xmax: 180 ymax: 83.65833
## Geodetic CRS:  WGS 84
```

```r
dir_out <- "./Output/"
dir.create(dir_out, showWarnings = FALSE)
```

Calculating average of PM2.5 and life years lost (based on WHO and National guidelines)

from 1998 to 2021

```r
pm_columns <- grep("^pm", names(gadm2), value = TRUE)
gadm2$average_pm <- rowMeans(gadm2[pm_columns], na.rm = TRUE)

llpp_who_columns <- grep("^llpp_who", names(gadm2), value = TRUE)
```

```r
gadm2$average_llpp_who <- rowMeans(gadm2[llpp_who_columns], na.rm = TRUE)

llpp_nat_columns <- grep("^llpp_nat", names(gadm2), value = TRUE)
gadm2$average_llpp_nat <- rowMeans(gadm2[llpp_nat_columns], na.rm = TRUE)
```

Remove columns that are not required and creating column for relative life years lost

```r
col_to_remove <- c(grep("^pm", names(gadm2), value = TRUE),
                   grep("^llpp_who", names(gadm2), value = TRUE),
                   grep("^llpp_nat", names(gadm2), value = TRUE))
gadm2 <- gadm2[, !(names(gadm2) %in% col_to_remove)]
gadm2 <- na.omit(gadm2)
gadm2$relative_life_lost <- gadm2$average_llpp_who - gadm2$average_llpp_nat
```

List of top 10 polluted countries

```r
top_10_list <- gadm2 %>%
  filter(!is.na(average_pm)) %>%
  group_by(country) %>%
  summarize(max_pm = max(average_pm, na.rm = TRUE)) %>%
  arrange(desc(max_pm)) %>%
  slice_head(n = 10) %>%
  select(country)
```

Extracting the details top 10 countries and grouping data based on countries

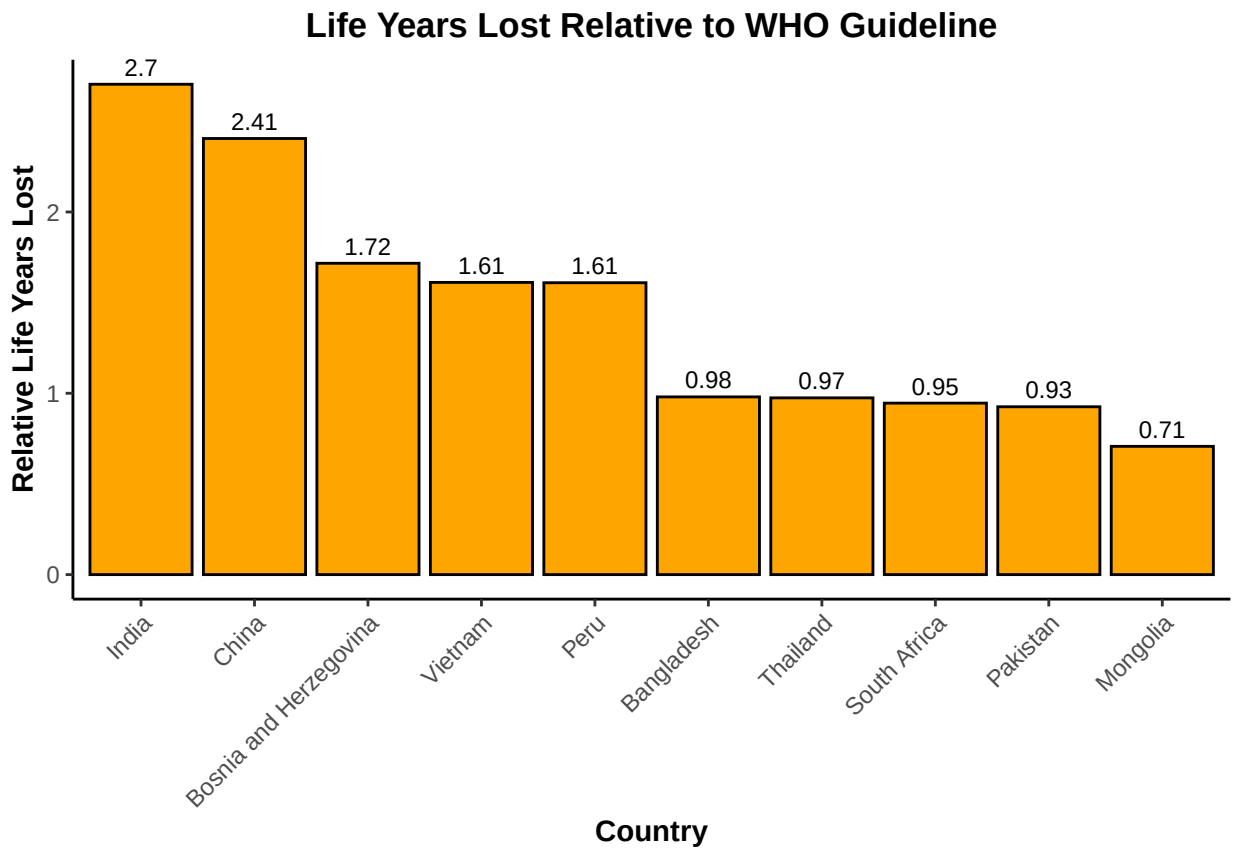
```r
top_10 <- gadm2 %>%
  filter(country %in% top_10_list$country) %>%
  select(-name_1, -name_2, -objectid_gadm2, -iso_alpha3) %>%
  group_by(country) %>%
  summarize(
    population_sum = sum(population, na.rm = TRUE),
    across(.cols = everything(), .fns = mean, na.rm = TRUE)
  )
```

```
## Warning: There was 1 warning in `summarize()`.
## i In argument: `across(.cols = everything(), .fns = mean, na.rm = TRUE)`.
## i In group 1: `country = "Bangladesh"`.
## Caused by warning:
## ! The `...` argument of `across()` is deprecated as of dplyr 1.1.0.
## Supply arguments directly to `.fns` through an anonymous function instead.
```

```
##
##   # Previously
##   across(a:b, mean, na.rm = TRUE)
##
##   # Now
##   across(a:b, \(x) mean(x, na.rm = TRUE))
```

Bar plot for relative life years lost for top 10 countries

```
years_lost_plot <- ggplot(top_10, aes(x = reorder(country, -relative_life_lost),
                                      y = relative_life_lost)) +
  geom_bar(stat = "identity", fill = "orange", color = "black") +
  geom_text(aes(label = round(relative_life_lost, 2)), vjust = -0.5, size = 3) +
  labs(title = "Life Years Lost Relative to WHO Guideline",
       x = "Country",
       y = "Relative Life Years Lost") +
  theme_minimal() + theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5, face = "bold"),
        axis.title.y = element_text(face = "bold"),
        axis.title.x = element_text(face = "bold"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.line.x = element_line(colour = "black"),
        axis.line.y = element_line(colour = "black"))


print(years_lost_plot)
```



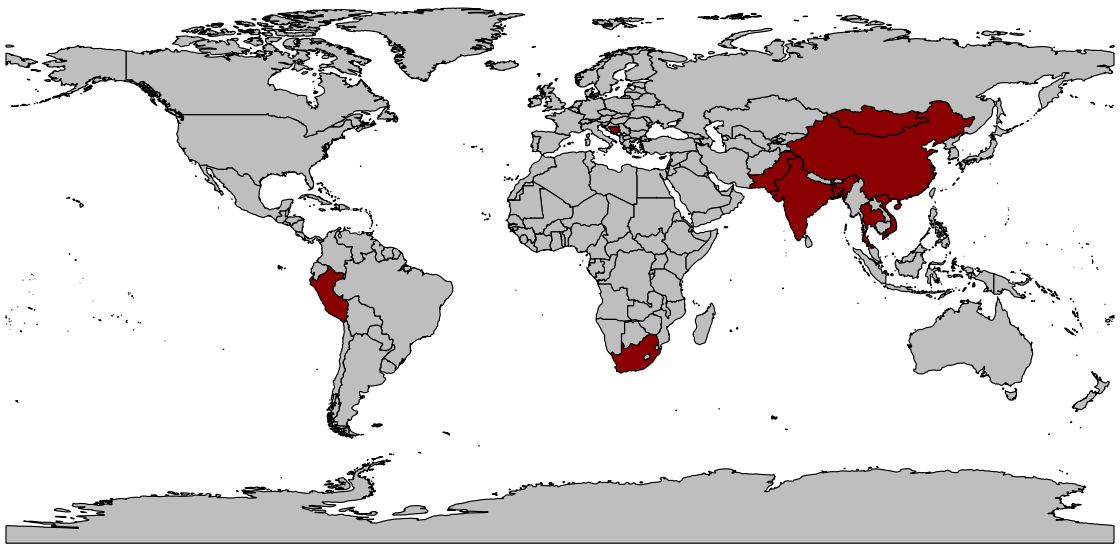

```
ggsave(file.path(dir_out, "Years lost to WHO.png"), plot = years_lost_plot,
       device = "png")
```

```
## Saving 6.5 x 4.5 in image
```

Plotting the top 10 countries in the world map

```
world <- ne_countries(scale = "medium", returnclass = "sf")

top_10$country_lower <- tolower(top_10$country)
world$name_lower <- tolower(world$name_long)

world$fill_color <- 'gray'

world$fill_color[world$name_long %in% top_10$country] <- 'darkred'

base_map <- ggplot() +
  geom_sf(data = world, aes(fill = fill_color), color = "black") +
  scale_fill_manual(values = c('darkred', 'gray'), guide = "none") +
  theme_minimal() +
  theme(legend.position = "none", plot.background = element_rect(
    fill = "white", color = NA), panel.background = element_rect(
```

```
    fill = "white", color = NA), panel.grid = element_blank())

top_10_map <- base_map +
  labs(title = 'Global Map with Top 10 Countries by Average Life Years Lost',
       fill = '') +
  theme(plot.title = element_text(size = 16, face = "bold", hjust = 0.5),
        axis.text = element_blank(),
        axis.title = element_blank(),
        axis.ticks = element_blank(),
        panel.grid = element_blank())


print(top_10_map)
```




```
ggsave(file.path(dir_out, "Top 10 polluted - Global Map.png"),
       plot = top_10_map, device = "png")
```

```
## Saving 6.5 x 4.5 in image
```

2.2.2

Merging dataframe (gadm2) with shapefile (gadm2_shp)

```
gadm2 <- merge(gadm2_shp, gadm2, by.x = "obidgadm2", by.y = "objectid_gadm2")
```

Listing Eastern and Western European countries extracting these country data

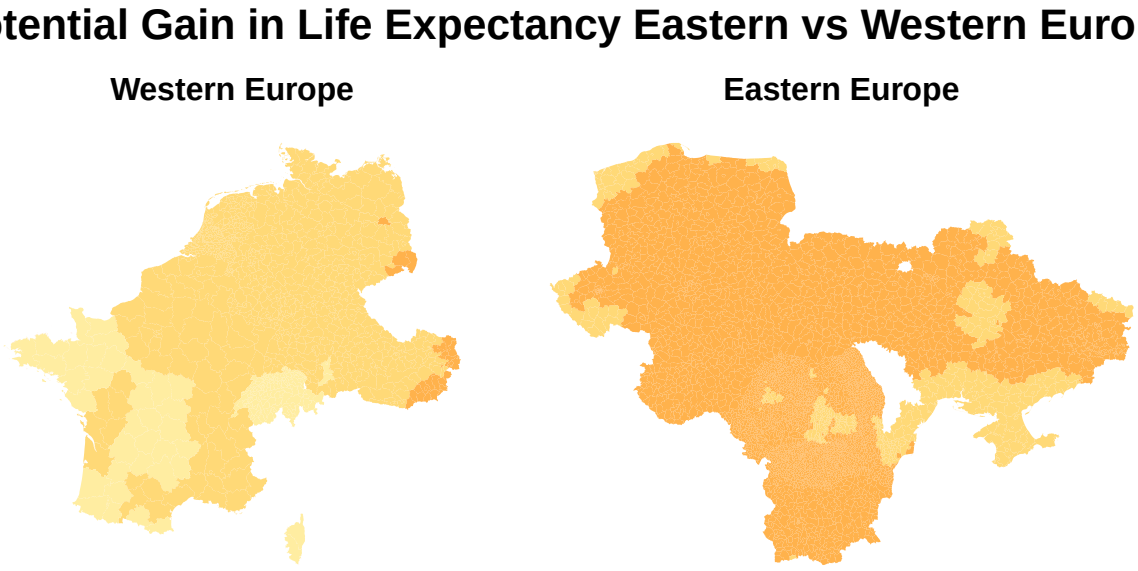
to a new dataframe

```
eastern_europe <- c('Belarus', 'Bulgaria', 'Czechia', 'Hungary',
                    'Moldova', 'Poland', 'Romania', 'Slovakia', 'Ukraine')
western_europe <- c('Austria', 'Belgium', 'France', 'Germany',
                    'Luxembourg', 'Netherlands', 'Switzerland')
gadm2$region <- case_when(
  gadm2$name0 %in% eastern_europe ~ "Eastern Europe",
  gadm2$name0 %in% western_europe ~ "Western Europe",
  TRUE ~ NA_character_
)

east_west_eu <- gadm2 %>%
  filter(!is.na(region))

east_west_eu$potential_gain <- east_west_eu$average_llpp_who -
  east_west_eu$average_llpp_nat
```

Plotting West vs East potential gain in life expectancy map

```
color_scale <- c('#ffffff', '#ffeda1', '#ffd977', '#ffb24c', '#fd8d3d',
                 '#fc4e2a', '#e31b1b', '#bd0026', '#800126')

assign_color <- function(value) {
  cuts <- c(0, 0.1, 0.5, 1, 2, 3, 4, 5, 6, Inf)
  colors <- color_scale
  return(cut(value, breaks = cuts, labels = colors, include.lowest = TRUE))
}

east_west_eu$color <- assign_color(east_west_eu$potential_gain)

maps <- list()
for (region_name in unique(east_west_eu$region)) {
  regional_data <- filter(east_west_eu, region == region_name)
```

```
  maps[[region_name]] <- ggplot() +
    geom_sf(data = regional_data, aes(fill = color), color = NA, size = 0.2) +
    scale_fill_identity() +
    theme_minimal() +
    theme(panel.grid = element_blank(),
          axis.text = element_blank(),
      axis.title = element_blank(),
      plot.title = element_text(hjust = 0.5, size = 12, face = 'bold')) +
    labs(title = paste( region_name, sep = ""),
         fill = '')
}

e_w_map <- wrap_plots(maps, ncol = 2) +
  plot_annotation(
    title = "Potential Gain in Life Expectancy Eastern vs Western Europe",
    theme = theme(plot.title = element_text(
      size = 16, hjust = 0.5, face = "bold"))
  )
print(e_w_map)
```

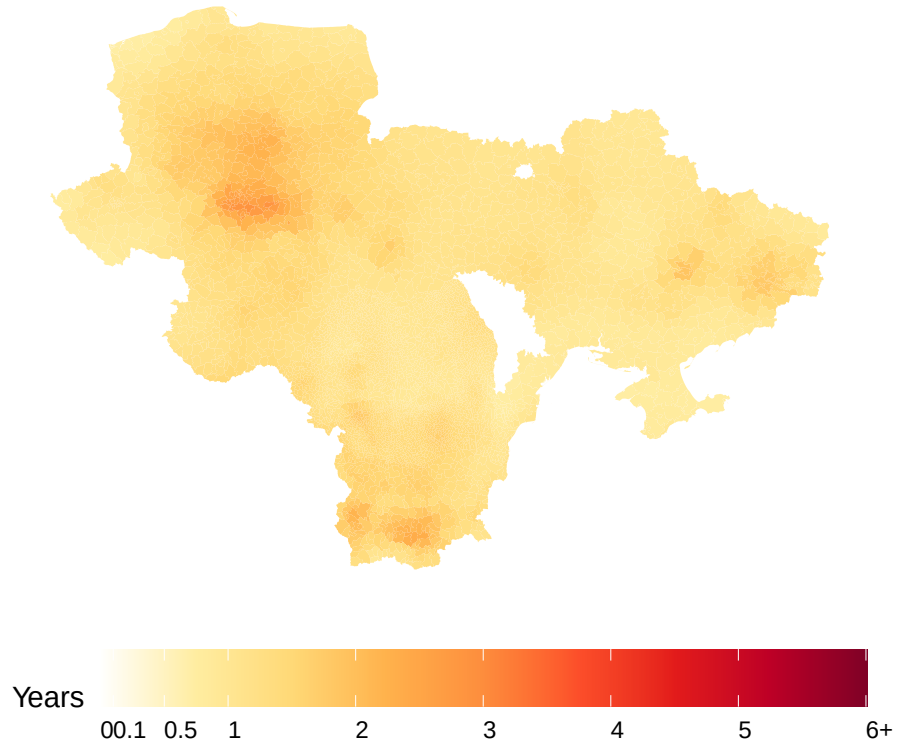



```
pdf_file <- file.path(dir_out, "Gain in life exp - East vs West EU.pdf")
ggsave(pdf_file, e_w_map, width = 14, height = 7)
```

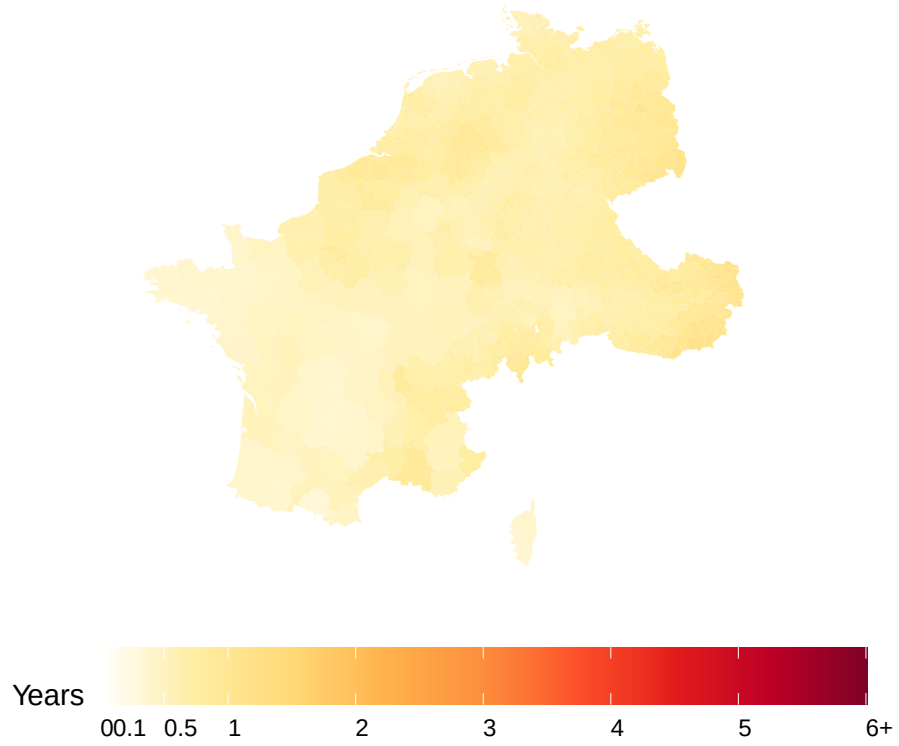
There was an issue is generating a common legend. Below code create separate

maps for Eastern and Western Europe with legends.

```
color_scale <- c('#ffffff', '#ffeda1', '#ffd977', '#ffb24c', '#fd8d3d'
                 , '#fc4e2a', '#e31b1b', '#bd0026', '#800126')

maps <- list()
for (region_name in unique(east_west_eu$region)) {
  regional_data <- filter(east_west_eu, region == region_name)

  # Create the map with horizontal legend at the bottom
  maps[[region_name]] <- ggplot() +
    geom_sf(data = regional_data, aes(fill = average_llpp_who), color = NA,
            size = 0.2) +
    scale_fill_gradientn(
      colors = color_scale,
      breaks = c(0, 0.1, 0.5, 1, 2, 3, 4, 5, 6),
      labels = c("0", "0.1", "0.5", "1", "2", "3", "4", "5", "6+"),
      limits = c(0, 6),
      guide = guide_colorbar(title = "Years", barwidth = 20, barheight = 1.5,
                             label.position = "bottom", label.hjust = 0)
    ) + theme_minimal() +
    theme(
      panel.grid = element_blank(),
      axis.text = element_blank(),
      axis.title = element_blank(),
      plot.title = element_text(hjust = 0.5, size = 16, face = 'bold'),
      legend.position = "bottom"
    ) +
    labs(
      title = paste(region_name,
                    "- Potential Gain in Life Expectancy", sep = " "),
      fill = 'Potential Gain'
    )
}

print(maps$`Eastern Europe`)
```




```
print(maps$`Western Europe`)
```




```
pdf_file <- paste0(dir_out, "Gain in life exp - East vs West EU - Seperate.pdf")
pdf(pdf_file, width = 14, height = 8)
for (region_name in names(maps)) {
  print(maps[[region_name]])
}
dev.off()
```

```
## pdf
##   2
```

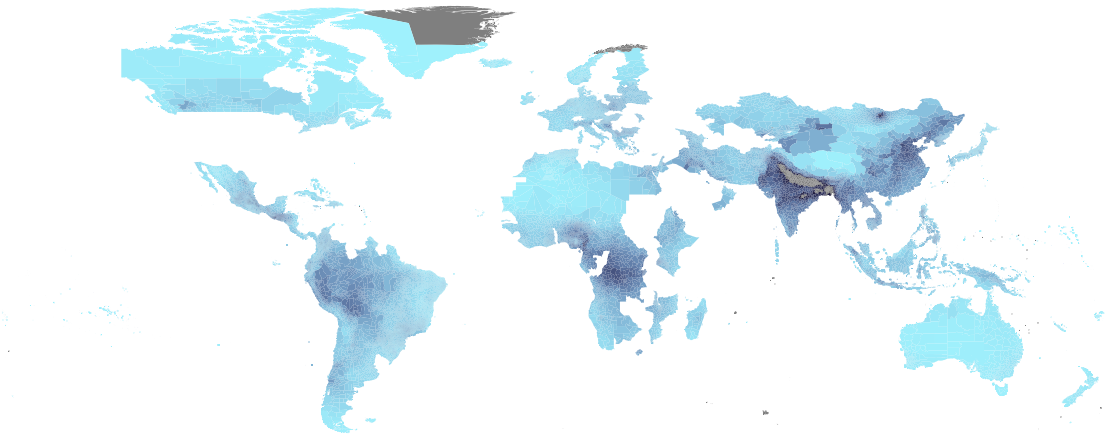
2.2.3

Merging the dataframes and subset it to two columns on pm2021 and geometry

```
pm_data <- read.csv('gadm2_aqli_1998_2021.csv')
pm_data <- subset(pm_data, select = c(objectid_gadm2, pm2021))
pm_data <- merge(pm_data, gadm2_shp, by.x = "objectid_gadm2",
                 by.y = "obidgadm2")
pm_data <- subset(pm_data, select = c(pm2021, geometry))
```

Plotting global pollution map 2021 (based on PM2.5)

```
color_scale <- c("#a1f5ff", "#92d4eb", "#82b5d5", "#7197bf", "#5f7aa6",
                 "#4e5e8b", "#3c456f", "#2b2d55", "#1a1638")

global_aqi <- ggplot() +
  geom_sf(data = pm_data, aes(fill = pm2021, geometry = geometry), color = NA) +
  scale_fill_gradientn(colors = color_scale, limits = c(0, 70),
                       breaks = c(0, 5, 10, 20, 30, 40, 50, 60, Inf),
                       labels = c("0", "5", "10", "20", "30", "40", "50",
                                  "60", "70+")) +
  labs(fill = "PM2.5 (µg/m³)", title = "Global Pollution Map 2021") +
  theme_minimal() +
  theme(
    panel.grid = element_blank(),
    axis.text = element_blank(),
    axis.ticks = element_blank(),
    plot.title = element_text(hjust = 0.5, face = "bold"),
    legend.position = "bottom",
    legend.direction = "horizontal"
  ) +
  guides(fill = guide_colorbar(barwidth = 15, barheight = 1,
                               title.position = "top", title.hjust = 0.5,
                               keywidth = 1))

print(global_aqi)
```



```
ggsave(file.path(dir_out, "Global PM25 2021.svg"), plot = global_aqi,
       device = "svg", dpi = 320)
```

```
## Saving 6.5 x 4.5 in image
```