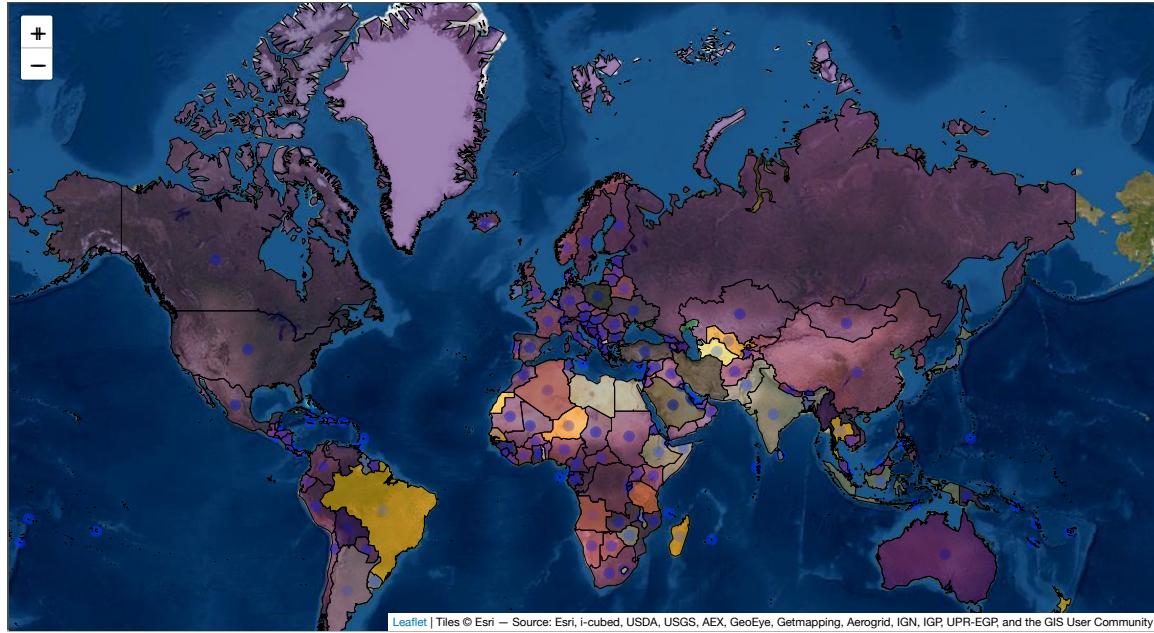




Spatial Data Analysis: Polygons and data frame in Leaflet: Workflow



• Data Collection

This is the dataset I found from databank.worldbank.org on women labour force globally for each country. The dataset is downloadable in CSV format. Reading from local data source the header of the file can be called using the following syntax in R. The output of the syntax is displayed in the output box below the code lines.

```
>data <- read.csv("//file-location")
>head(data)
```

```
>head(data)
Data.Source: World.Development.Indicators
      X
1 Country Name          Country Code Indicator Name
2 Aruba                  ABW Labor force, female (% of total labor force)
3 Afghanistan            AFG Labor force, female (% of total labor force)
4 Angola                 AGO Labor force, female (% of total labor force)
5 Albania                ALB Labor force, female (% of total labor force)
6 Andorra               AND Labor force, female (% of total labor force)
7 X.1 X.2 X.3 X.4 X.5 X.6 X.7 X.8 X.9 X.10 X.11 X.12 X.13 X.14 X.15 X.16 X.17
1 Indicator Code 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975
2 SL.TLF.TOTL.FE.ZS NA NA
3 SL.TLF.TOTL.FE.ZS NA NA
4 SL.TLF.TOTL.FE.ZS NA NA
5 SL.TLF.TOTL.FE.ZS NA NA
6 SL.TLF.TOTL.FE.ZS NA NA
7 X.18 X.19 X.20 X.21 X.22 X.23 X.24 X.25 X.26 X.27 X.28 X.29 X.30 X.31 X.32 X.33
1 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992
2 NA NA
3 NA NA
4 NA NA
5 NA NA
6 NA NA
7 X.34 X.35 X.36 X.37 X.38 X.39 X.40 X.41 X.42
1 1992.00000 1993.00000 1994.00000 1995.00000 1996.00000 1997.00000 1998.00000 1999.00000 2000.00000
2 NA NA
3 15.61551 15.51161 15.4761 15.43395 15.30132 15.1999 15.18657 15.18567 15.07723
4 48.34041 48.34830 48.3869 48.3555 48.3245 48.30392 48.281104 48.26988 49.08077
5 44.04471 44.38473 44.70900 44.70200 44.12405 43.98255 43.12955 42.25334 41.70331
6 NA NA
7 NA NA
```

• Data Analysis

The dataset is spread over a range of years. For simple spatial data analysis I have created a new data frame with country names and the data values in column 2019.

```
>Country.name <- data$Country.Name
>Women.labour.2019 <- data$X2019
>data_2019 <- data.frame(Country.name, women.labour.2019)
>data_2019
```

```
>data_2019
      Country.name women.labour.2019
1          Arab Emirates          NA
2          Afghanistan        21.60739
3             Angola        50.270279
4             Albania       43.044282
5           Andorra          NA
6          Arab World       20.363081
7 United Arab Emirates     17.479732
8            Argentina      43.269628
9            Armenia       44.225470
10           American Samoa        NA
11          Antigua and Barbuda        NA
12             Australia      46.641605
13            Austria       46.417745
14            Azerbaijan      48.631628
15            Burundi        51.857644
16            Belgium        46.225621
17            Benin         49.226740
```



```

18 Burkina Faso 44.730881
19 Bangladesh 30.514651
20 Bulgaria 46.114512
21 Bahrain 20.141999
22 Bahamas, The 48.216161
23 Bosnia and Herzegovina 40.577755
24 Belarus 48.950164
25 Belize 38.793913
26 Benin 0.0
27 Bolivia 44.343475
    
```

• Spatial Data Collection

For spatial analysis of the data collected, a shape file was downloaded from [pvBjörn Sandvik](#).
The shapefile can be imported into the session using 'readOGR' function in the package 'rgdal'.

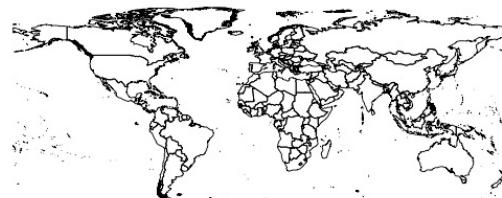
```
>library(rgdal)
>data_spatial <- readOGR("//file-location")
>summary(data_spatial)
```

```

>summary(data_spatial)
Object of class SpatialPolygonsDataFrame
Coordinates:
   min       max
x -180 180.0000
y -90 83.57027
Is projected: FALSE
proj4string : [+proj=longlat +datum=WGS84 +no_defs]
Data attributes:
  FIPS ISO2 ISO3 UN NAME
Length:246 Length:246 Length:246 Min. 4.0 Length:246
Class :character Class :character Class :character 1st Qu.:215.0 Class :character
Mode :character Mode :character Mode :character Median :449.0 Mode :character
Mean :431.8 Mean :431.8 Mean :431.8
3rd Qu.:650.5 Max. :894.0 Max. :894.0
Max. :11638094.0
  LAT POP2005 REGION SUBREGION LON
Min. :-80.4460
1st Qu.: -0.3025
Median : 10.0
Mean : 16.4075
3rd Qu.: 39.1067
Max. :11638094.0
  
```

• Spatial Data Frame

```
>plot(data_spatial)
```



• Pointset Data + Spatial data integration

Integrated data table for shape file and tabular data in order to assign the coordinates from the shapefile to the tabular data using the country name as the primary key.

To create an integrated spatial data first check the class of each data table:

```
>class(data_2019)
```

```
>class(data_spatial)
[1] "SpatialPolygonsDataFrame"
attr(,"package")
[1] "sp"
```

```
>class(data_spatial)
```

```
>class(data_spatial)
[1] "SpatialPolygonsDataFrame"
attr(,"package")
[1] "sp"
```



Create combined data frame from point set data and spatial data using merge function on the name columns for each table:

```
>spatial_dataframe <- merge(data_spatial, data_2019, by.x="NAME", by.y="Country.name")
>summary(spatial_dataframe)
```

```
>summary(spatial_dataframe)
Object of class SpatialPolygonsDataFrame
Coordinates:
    min   max
x  -180 180.0000
y  -90  83.57027
Is projected: FALSE
proj4string : [+proj=longlat +datum=WGS84 +no_defs]
Data attributes:
      NAME          VIPS           ISO2           ISO3          UN
  Min. :  i     0.0 Length:246  Length:246  Length:246  Min. : 4.0
  1st Qu.: 44.5 Class :character 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.:215.0
  Median : 5515.5 Mode :character Median :19.00 Median :30.00 Median :429.0
  Mean   : 52696.1          Mean :65.43 Mean   :54.84 Mean   :13.29
  3rd Qu.: 34708.8          3rd Qu.:142.00 3rd Qu.:61.00 3rd Qu.: 50.01
  Max.   :1638094.0          Max. :150.00 Max.   :155.00 Max.   :179.22
      AREA        POP2005       REGION      SUBREGION      LON
  Min. :  i     0.0 Length:246  Min. : 0.00  Min. : 0.00  Min. :-178.13
  1st Qu.: 44.5 Class :character 1st Qu.: 2.00  1st Qu.: 14.00  1st Qu.:-50.16
  Median : 5515.5 Mode :character Median :19.00  Median :30.00  Median :17.66
  Mean   : 52696.1          Mean :65.43  Mean   :54.84  Mean   :13.29
  3rd Qu.: 34708.8          3rd Qu.:142.00 3rd Qu.:61.00  3rd Qu.: 50.01
  Max.   :1638094.0          Max. :150.00 Max.   :155.00 Max.   :179.22
      LAT
  Min. :-80.4460  Min. :13.38
  1st Qu.:-10.2525 1st Qu.:44.44
  Median : 16.5110  Median :44.16
  Mean   : 16.4075  Mean :41.41
  3rd Qu.: 39.1067  3rd Qu.:47.13
  Max.   : 78.8300  Max. :55.77
  NA's   : 89
```

Here, the spatial data frame is merged with the data table containing the percentage of women in the workforce in year 2019.

• Leaflet map

Add the leaflet libraries

```
>library("leaflet")
```

Add spatial_dataframe data in leaflet map

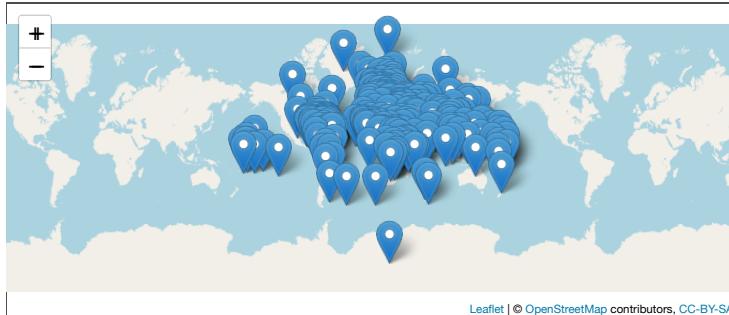
```
> map1 <- leaflet()%>
  addMarkers(data=spatial_dataframe,
  lon = longitude,
  lat= latitude)
```



• Leaflet map + tiles

Add tiles

```
>map1 %>% addTiles()
```



- Leaflet map + tiles + polygons

Add tiles

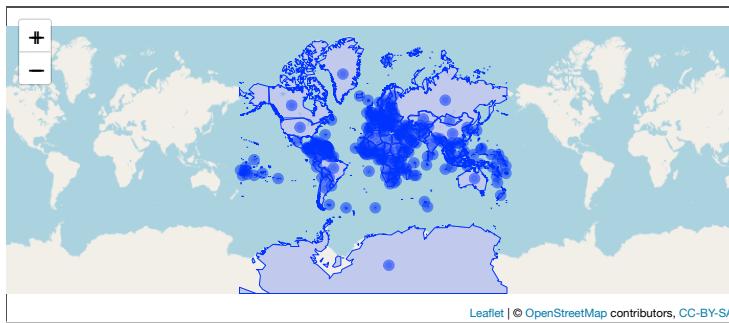
```
>map1 %>% addPolygons(data=integrated_df)
CORRECTION: replace integrated_df with spatial_dataframe
```



- Leaflet map + tiles (options)

Display options for marker and polygon layers: radius, weight

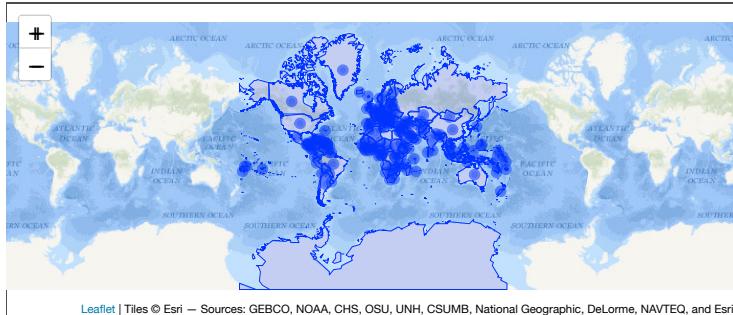
```
>leaflet() %>%
  addCircleMarkers(data=integrated_df,
    lon = longitude,
    lat = latitude,
    radius=3) %>%
  addTiles()%>%
  addPolygons(data=integrated_df,
    opacity=1,
    weight=0.5)
CORRECTION: replace integrated_df with spatial_dataframe
```



- Leaflet map options (basemap)

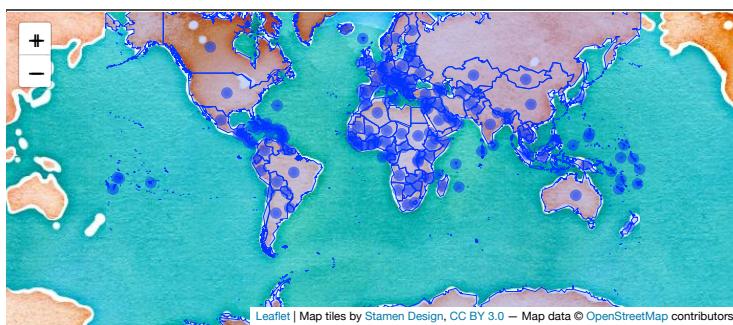
The layer tiles for base map by default is the [OpenStreetMaps](#). To select custom maps tiles from providers, update the code to:

```
>leaflet()%>%
  addCircleMarkers(data=integrated_df,
    lon = longitude,
    lat = latitude,
    radius=3)%>%
  addProviderTiles(providers$Esri.OceanBasemap)%>%
  addPolygons(data=integrated_df,
    opacity=1,
    weight=0.5)
CORRECTION: replace integrated_df with spatial_dataframe
```



```
>leaflet() %>%
  addCircleMarkers(data=integrated_df,
    lon = longitude,
    lat = latitude,
    radius=1)
  addProviderTiles(providers$Stamen.Watercolor) %>%
  addPolygons(data=integrated_df,
    opacity=1,
    weight=0.5)

CORRECTION: replace integrated_df with spatial_dataframe
```



• Color Map

The range of the percentage value of women in the workforce for each country has a variance ranging from a max of 55.77 and a min of 13.38. I got these ranges from the [summary of the data frame](#) created earlier. To effectively present the 'percentage of the women in the workforce for each country', a color map highlights each country according to the corresponding data values from the [data collected](#) is required.

To apply the color scale to the polygons in the shape file data there are various methods. The simplest one I could find is explained here:

```
1. Set Color Palette
>color_palette <- ("inferno")

2. No of Bins
>bins <- 20

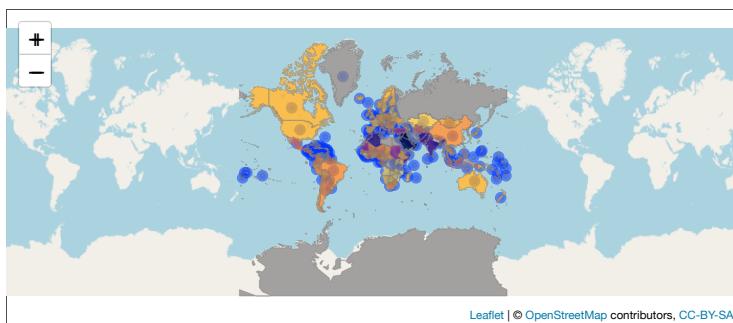
3. Data unavailable
>if.na <- "#000000"

4. Associated data range
>range <- spatial_dataframe$women.labour.2019

5. Colorbin variance
> variance <- colorBin(color_palette, range, bins, na.color = if.na)
```

Use the variance as the range fillcolor range in the leaflet function

```
>leaflet() %>%
  addCircleMarkers(data=spatial_dataframe,
    lon = longitude,
    lat = latitude,
    radius=3) %>%
  addTiles() %>%
  addPolygons(data=spatial_dataframe,
    weight=1,
    fillOpacity = 0.7,
    strokeDash = c(5,5),
    fillColor = variance(range),
    color = "grey")
```

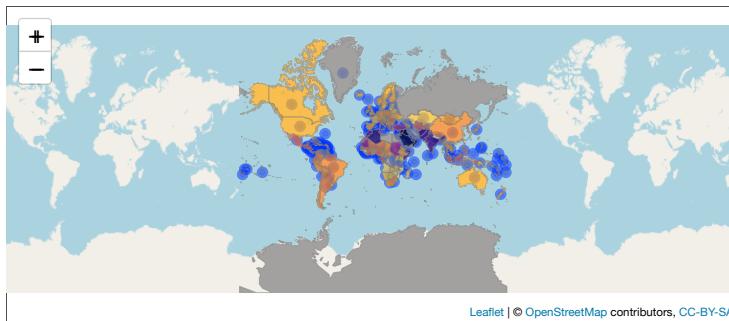




• Labels

Adding label to highlight the percentage value for each country on the map:

```
>leaflet()>%>
  addCircles(data=spatial_dataframe,
    lat = longitude,
    lon = latitude,
    radius=3)>%
  addPolygons(data=spatial_dataframe,
    weight = 1,
    fillOpacity = 0.7,
    smoothFactor = 0.5,
    fillColor = variance(range),
    label = range,
    color = "grey")
```



• Legend

A legend to get an idea of the color ranges and the associated data value:

```
1.legend position
>position <- "bottomleft"
2.Opacity
>opacity <- 15
3.Colorscale
>scale <- variance (from)
4.Associated data range
>value <- range (from)
5.Heading
>title <- "Percentage of women in workforce"
```

Now build the leaflet code based on the above parameters:

```
>leaflet()>%>
  addCircles(data=spatial_dataframe,
    lat = longitude,
    lon = latitude,
    radius=3)>%
  addPolygons(data=spatial_dataframe,
    weight = 1,
    fillOpacity = 0.7,
    smoothFactor = 0.5,
    fillColor = variance(range),
    label = range,
    color = "grey")>%
  addLegend(position,
    scale,
    value,
    title=title,
    opacity=opacity)
```

