

title: "SpatialdataVisualization"

output:

html_document: default

This is used to provide an accessible introduction to techniques for handling, analysing and visualizing spatial data in R. It has a wide range of packages which provide additional functionality for handling spatial data and performing complex spatial analysis operations. GIS and geovisualization allows for more interactive maps, including the ability to explore different layers of the maps, to zoom in or out and to change the visual appearance of the map, usually on computer display.

```
library(sf)
```

```
## Linking to GEOS 3.6.1, GDAL 2.2.3, PROJ 4.9.3
```

```
library(raster)
```

```
## Loading required package: sp
```

```
library(dplyr)
```

```
##  
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:raster':  
##  
## intersect, select, union
```

```
## The following objects are masked from 'package:stats':  
##  
## filter, lag
```

```
## The following objects are masked from 'package:base':  
##  
## intersect, setdiff, setequal, union
```

```
library(spData)
```

```
## To access larger datasets in this package, install the spDataLarge  
## package with: `install.packages('spDataLarge',  
## repos='https://nowosad.github.io/drat/', type='source')`
```

```
library(tmap)      # for static and interactive maps  
library(leaflet)   # for interactive maps  
library(mapview)   # for interactive maps  
library(ggplot2)   # tidyverse data visualization package  
library(shiny)     # for web applications
```

1.Import the inbuilt spatial data file and Display the shape of Newzealand

```
# Add fill layer to nz shape  
tm_shape(nz) +  
  tm_fill()
```



Visualization of Newzealand data

2. Add border layer to the shape

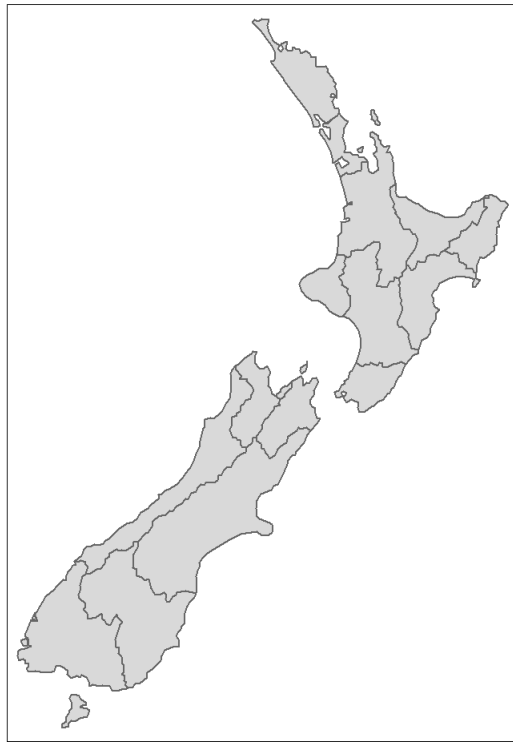
```
# Add border layer to nz shape  
tm_shape(nz) +  
  tm_borders()
```



Borders are added to the NewZealand data

3. How to fill the layers inside the border ?

```
# Add fill and border layers to nz shape  
tm_shape(nz) +  
  tm_fill() +  
  tm_borders()
```



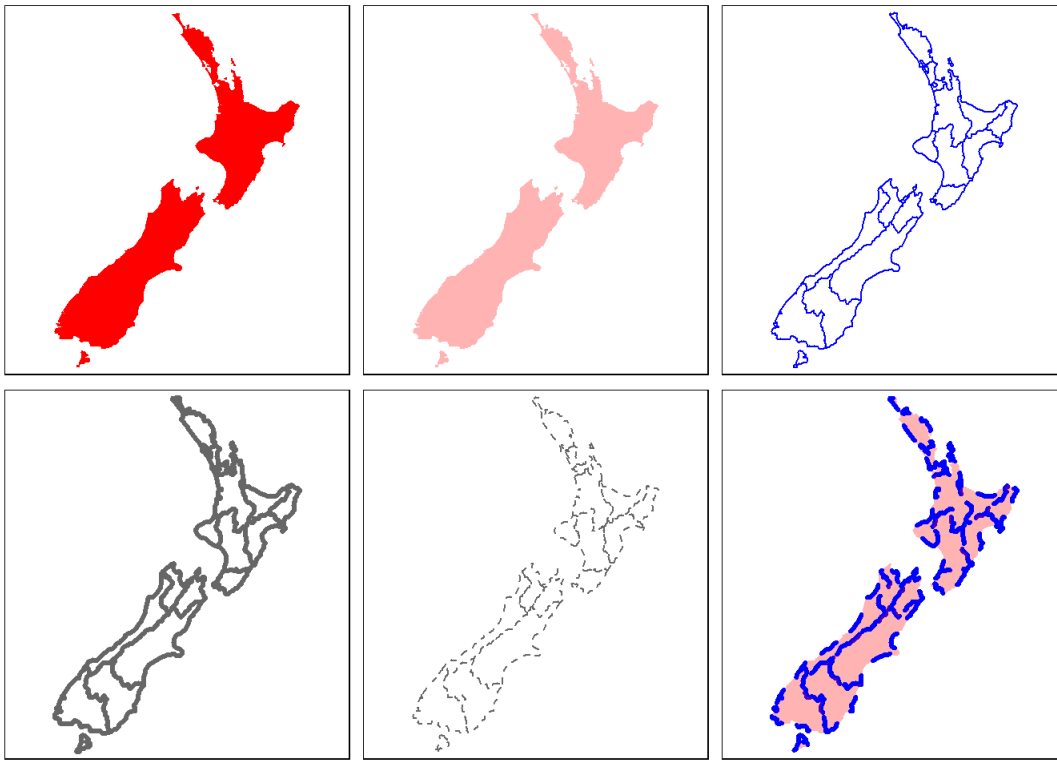
```
map_nz = tm_shape(nz) + tm_polygons()
class(map_nz)
```

```
## [1] "tmap"
```

Data is filled with layers and borders for the country.

4. How to make different visualizations with the nz data?

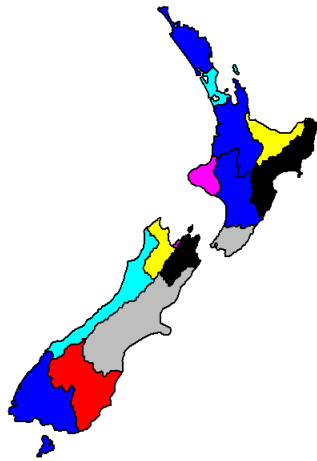
```
ma1 = tm_shape(nz) + tm_fill(col = "red")
ma2 = tm_shape(nz) + tm_fill(col = "red", alpha = 0.3)
ma3 = tm_shape(nz) + tm_borders(col = "blue")
ma4 = tm_shape(nz) + tm_borders(lwd = 3)
ma5 = tm_shape(nz) + tm_borders(lty = 2)
ma6 = tm_shape(nz) + tm_fill(col = "red", alpha = 0.3) +
  tm_borders(col = "blue", lwd = 3, lty = 2)
tmap_arrange(ma1, ma2, ma3, ma4, ma5, ma6)
```



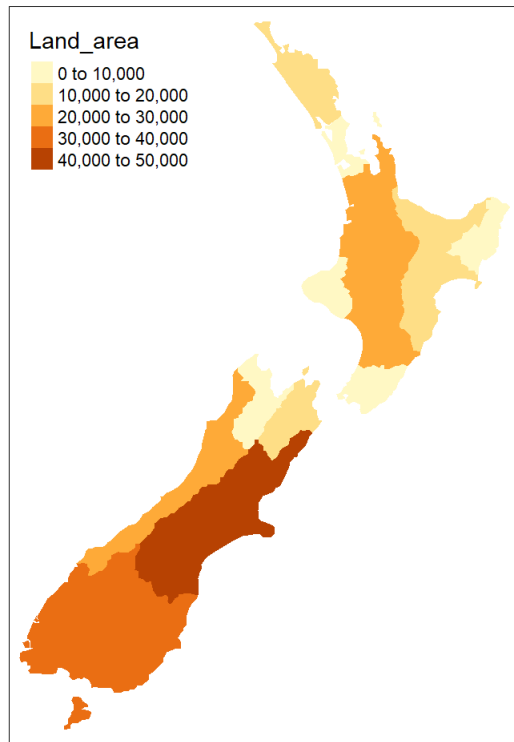
Data is visualized with different patterns and colours.

5.Display the land area of the spatial data

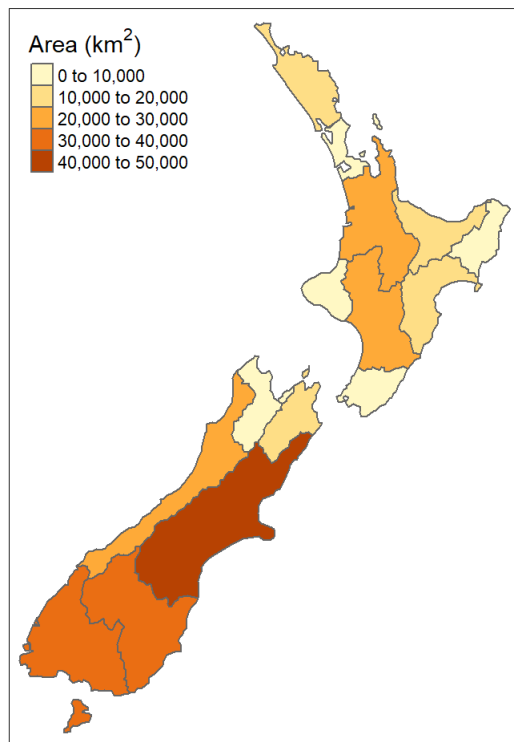
```
plot(st_geometry(nz), col = nz$Land_area)
```



```
tm_shape(nz) + tm_fill(col = "Land_area")
```



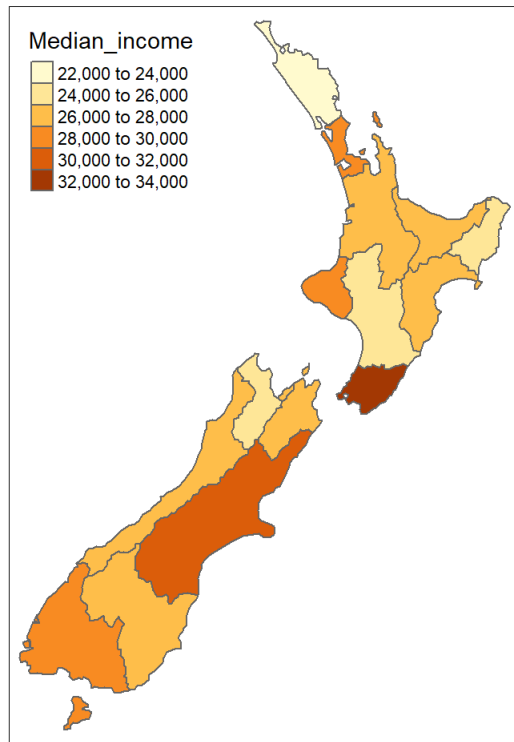
```
legend_title = expression("Area (km2*)")
map_nza = tm_shape(nz) +
  tm_fill(col = "Land_area", title = legend_title) + tm_borders()
map_nza
```



The land area of the country are display by showing variations in the color with geometrical function.

6. How to display the distribution of income in map ?

```
tm_shape(nz) + tm_polygons(col = "Median_income")
```

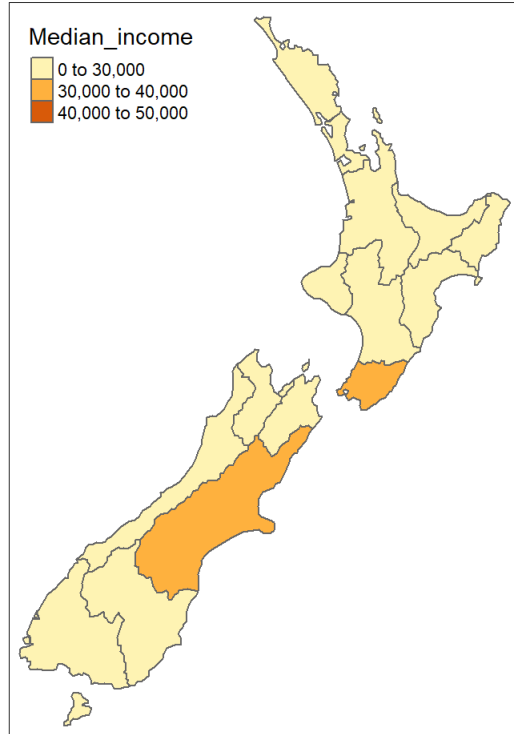


```
breaks = c(0, 3, 4, 5) * 10000
```

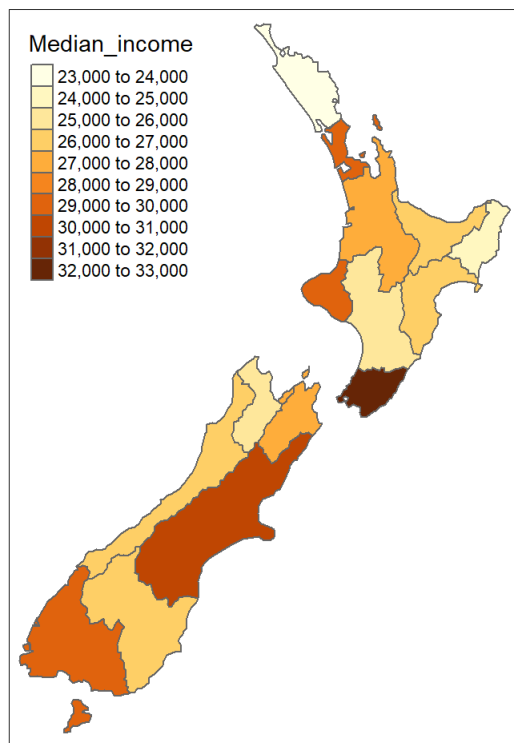
The map is distributed with respect to the median range of the income of the people

7. Display the average of the median income

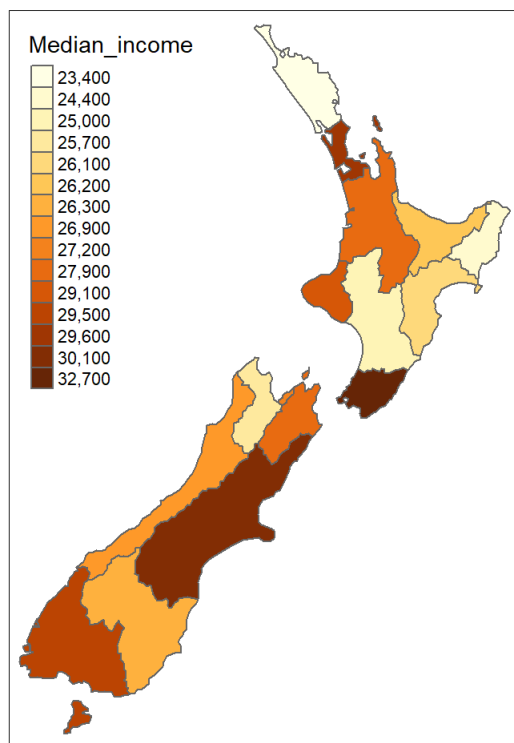
```
tm_shape(nz) + tm_polygons(col = "Median_income", breaks = breaks)
```



```
tm_shape(nz) + tm_polygons(col = "Median_income", n = 10)
```



```
tm_shape(nz) + tm_polygons(col = "Median_income", style="cat", n=5)
```

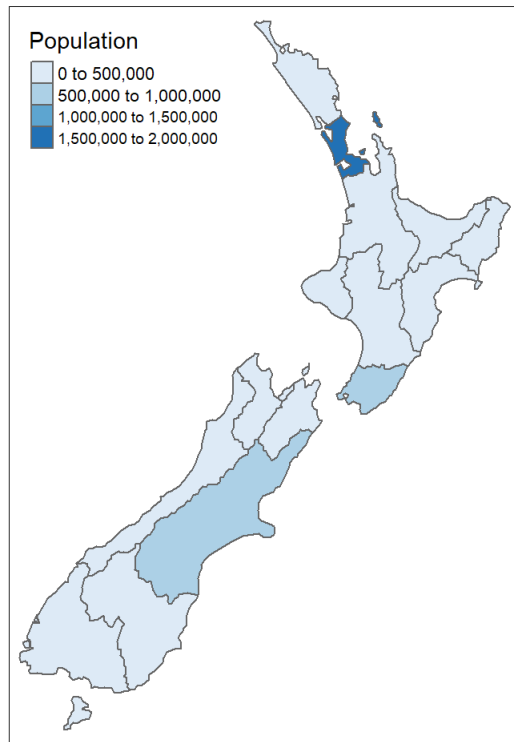


The map is displayed with the average of the median income that has been distributed over a particular area and different styles can be used for its visualization.

8. How to display the population distribution?

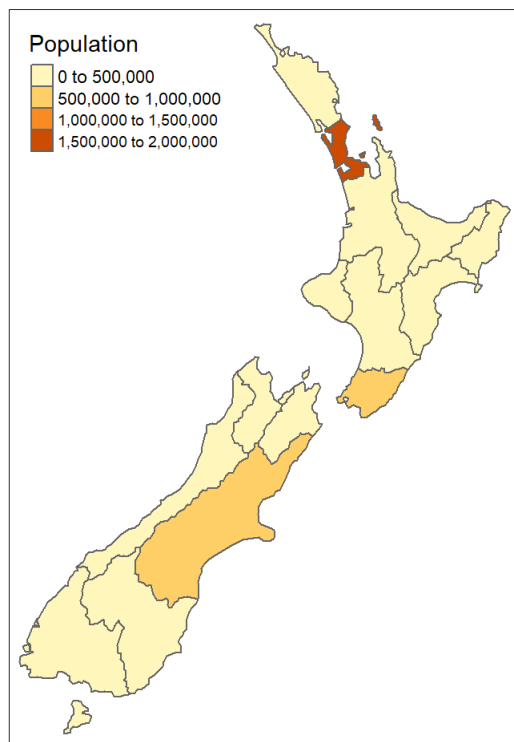
```
tm_shape(nz) + tm_polygons("Population", palette = "Blues")
```

```
## Some legend labels were too wide. These labels have been resized to 0.64, 0.59, 0.59. Increase legend.wid  
th (argument of tm_layout) to make the legend wider and therefore the labels larger.
```



```
tm_shape(nz) + tm_polygons("Population", palette = "YlOrBr")
```

```
## Some legend labels were too wide. These labels have been resized to 0.64, 0.59, 0.59. Increase legend.wid  
th (argument of tm_layout) to make the legend wider and therefore the labels larger.
```



Population of the country has been distributed with representation of colors ranging from light to dark as lowest population to the highest.

9. How to add a compass to show the direction of the graph?

```
map_nz +  
  tm_compass(type = "8star", position = c("left", "top")) +  
  tm_scale_bar(breaks = c(0, 100, 200), text.size = 1)
```



The compass is added to the visualization that shows the direction of the map.

10. Add the water layout to the border of the country

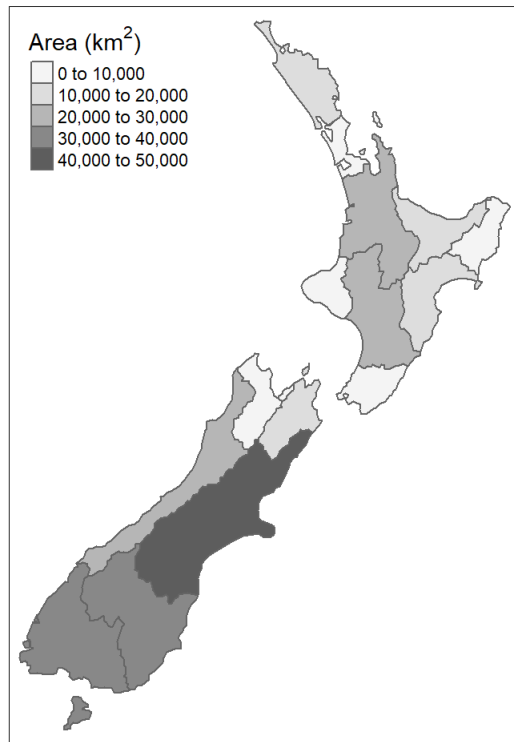
```
map_nz + tm_layout(title = "New Zealand",scale = 3,bg.color = "lightblue")
```



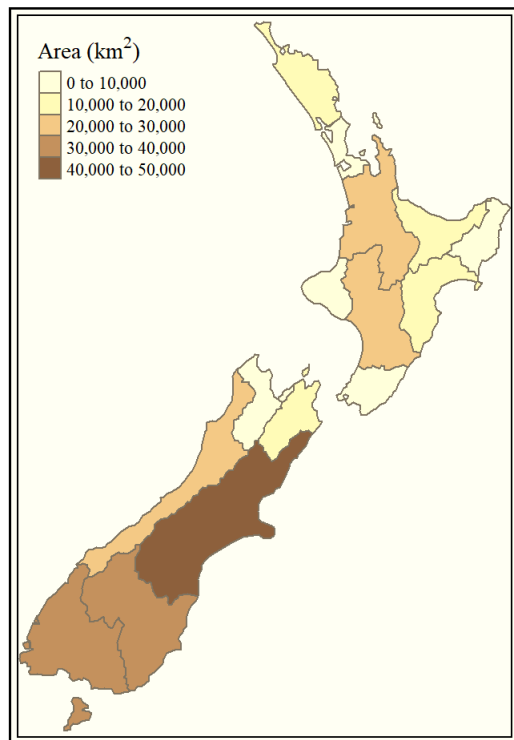
The country name has been added as title and the borders are shown with oceans represented in light blue color.

11.What are the different inbuild styles used for the visualization ?

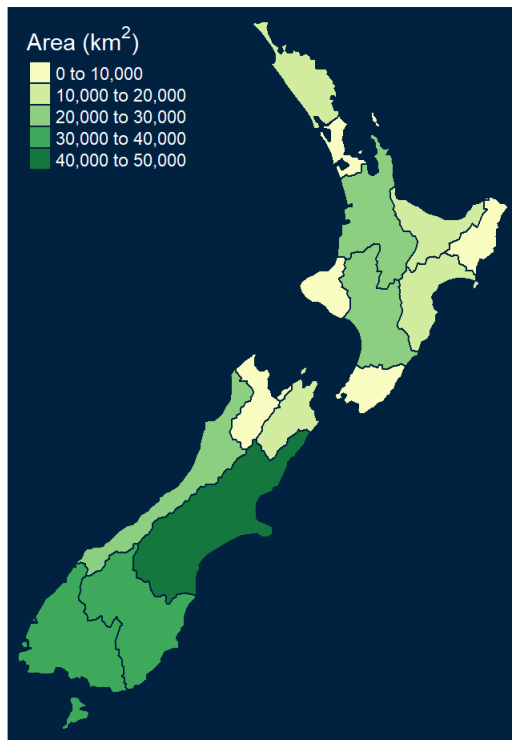
```
map_nza + tm_style("bw")
```



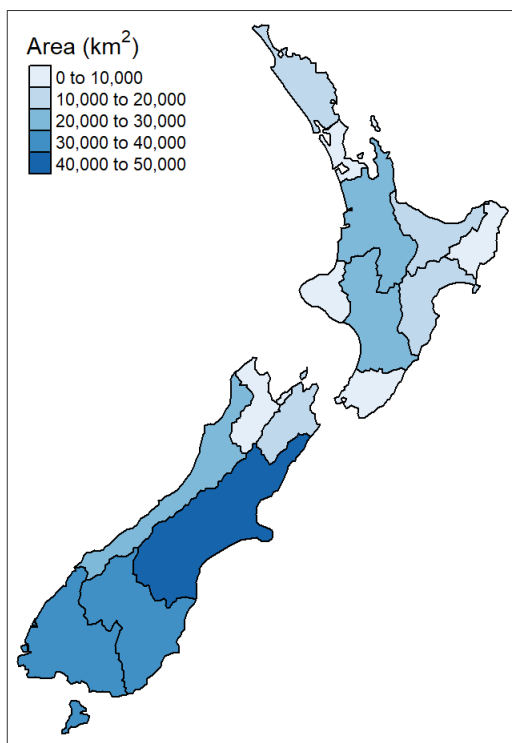
```
map_nza + tm_style("classic")
```



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



```
map_nza + tm_style("colblind")
```



The different styles of visualization of the map are represented.

12. Display the geometrical decription of the maps?

```
nz_region = st_bbox(c(xmin = 1340000, xmax = 1450000,
                      ymin = 5130000, ymax = 5210000),
                    crs = st_crs(nz_height)) %>%
  st_as_sf()
nz_region
```

```
## Geometry set for 1 feature
## geometry type: POLYGON
## dimension: XY
## bbox: xmin: 1340000 ymin: 5130000 xmax: 1450000 ymax: 5210000
## epsg (SRID): 2193
## proj4string: +proj=tmerc +lat_0=0 +lon_0=173 +k=0.9996 +x_0=1600000 +y_0=10000000 +ellps=GRS80 +towgs84=0,0,0,0,0,0 +units=m +no_defs
```

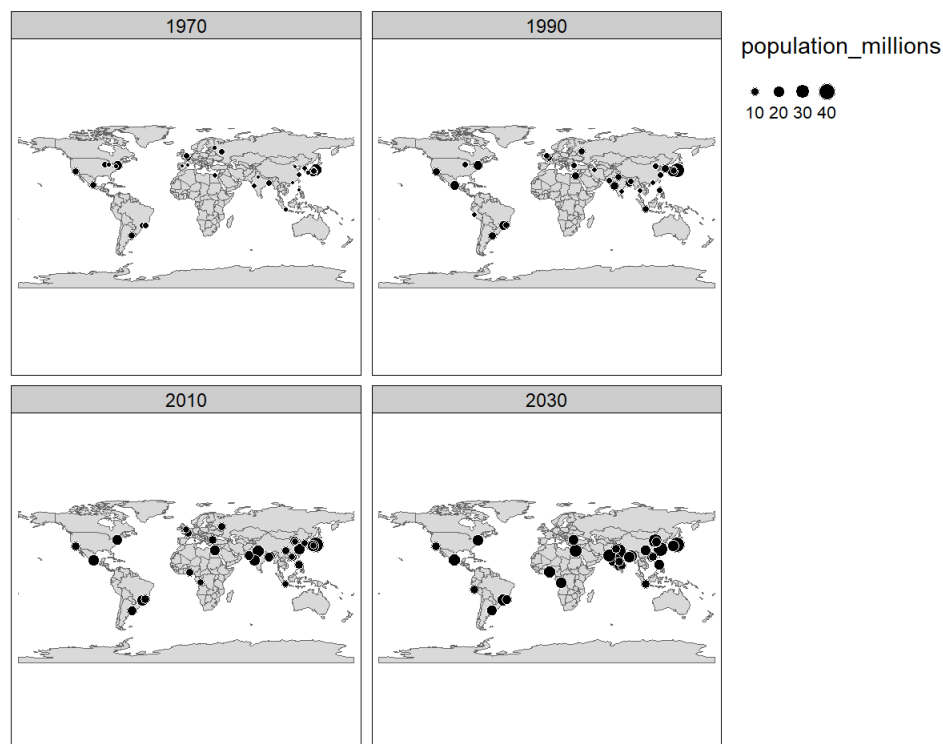
```
## POLYGON ((1340000 5130000, 1450000 5130000, 145...
```

The geometric set for the given x minimum and maximum and the same for y are described.

13. Difference in the world population for the last 50 years ?

```
urb_1970_2030 = urban_agglomerations %>%
  filter(year %in% c(1970, 1990, 2010, 2030))

tm_shape(world) +
  tm_polygons() +
  tm_shape(urb_1970_2030) +
  tm_symbols(col = "black", border.col = "white", size = "population_millions") +
  tm_facets(by = "year", nrow = 2, free.coords = FALSE)
```



The populations are visualized with a interval of 20 years starting from 1970 and it is reprinted in terms of millions.

Conclusion:

Various techniques to display and analyze the spatial data has been visualized.