Homework 1 - Divij Sanjanwala

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

Question 1

Use the `World` data set available in the R package `tmap`.

Question 1.1

What type of spatial object is the `World` data set? What is the CRS?

```r
# What type of spatial object is the World data set?
data("World")
class(World)
```

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

```r
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
## 1     AFG          Afghanistan          Afghanistan               Asia
## 2     AGO               Angola               Angola             Africa
## 3     ALB              Albania              Albania             Europe
## 4     ARE United Arab Emirates United Arab Emirates               Asia
## 5     ARG            Argentina            Argentina      South America
## 6     ARM              Armenia              Armenia               Asia
```

```
## 7      ATA             Antarctica          Antarctica              Antarctica
## 8      ATF Fr. S. Antarctic Lands              France Seven seas (open ocean)
## 9      AUS              Australia           Australia                 Oceania
## 10     AUT                Austria             Austria                  Europe
##                    area  pop_est pop_est_dens                    economy
## 1    652860.000 [km^2] 28400000 4.350090e+01  7. Least developed region
## 2   1246700.000 [km^2] 12799293 1.026654e+01  7. Least developed region
## 3     27400.000 [km^2]  3639453 1.328268e+02       6. Developing region
## 4     71252.172 [km^2]  4798491 6.734519e+01       6. Developing region
## 5   2736690.000 [km^2] 40913584 1.495003e+01    5. Emerging region: G20
## 6     28470.000 [km^2]  2967004 1.042151e+02       6. Developing region
## 7  12259213.973 [km^2]     3802 3.101341e-04       6. Developing region
## 8      7257.455 [km^2]      140 1.929051e-02       6. Developing region
## 9   7682300.000 [km^2] 21262641 2.767744e+00 2. Developed region: nonG7
## 10    82523.000 [km^2]  8210281 9.949082e+01 2. Developed region: nonG7
##                  income_grp gdp_cap_est life_exp well_being footprint inequality
## 1             5. Low income    784.1549   59.668        3.8      0.79 0.42655744
## 2    3. Upper middle income   8617.6635       NA         NA        NA         NA
## 3    4. Lower middle income   5992.6588   77.347        5.5      2.21 0.16513372
## 4   2. High income: nonOECD  38407.9078       NA         NA        NA         NA
## 5    3. Upper middle income  14027.1261   75.927        6.5      3.14 0.16423830
## 6    4. Lower middle income   6326.2469   74.446        4.3      2.23 0.21664810
## 7   2. High income: nonOECD 200000.0000       NA         NA        NA         NA
## 8   2. High income: nonOECD 114285.7143       NA         NA        NA         NA
## 9      1. High income: OECD  37634.0832   82.052        7.2      9.31 0.08067825
## 10     1. High income: OECD  40132.6093   81.004        7.4      6.06 0.07129351
##         HPI                       geometry
## 1  20.22535 MULTIPOLYGON (((61.21082 35...
## 2        NA MULTIPOLYGON (((16.32653 -5...
## 3  36.76687 MULTIPOLYGON (((20.59025 41...
## 4        NA MULTIPOLYGON (((51.57952 24...
## 5  35.19024 MULTIPOLYGON (((-65.5 -55.2...
## 6  25.66642 MULTIPOLYGON (((43.58275 41...
## 7        NA MULTIPOLYGON (((-59.57209 -...
## 8        NA MULTIPOLYGON (((68.935 -48....
## 9  21.22897 MULTIPOLYGON (((145.398 -40...
## 10 30.47822 MULTIPOLYGON (((16.97967 48...
```

The object `World` has the class `SpatialPolygonsDataFrame`. This is a spatial object that contains a data frame and a spatial polygon object. The data frame contains the data for each country, and the spatial polygon object contains the geometry for each country. The CRS for the object `World` is `+proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0`.

Question 1.2:

Using the projection argument of tm_shape(), create maps of the World data set with two other projections.

State what aspect(s) are preserved with the projections you use. Comment on the visual differences and biases across the three maps.

```
# Using the projection argument of tm_shape(), create maps of the World data set with two other project

# 1st Map using projection = WGS84 (The standard projection of the default in the World Data set) ie: L
```

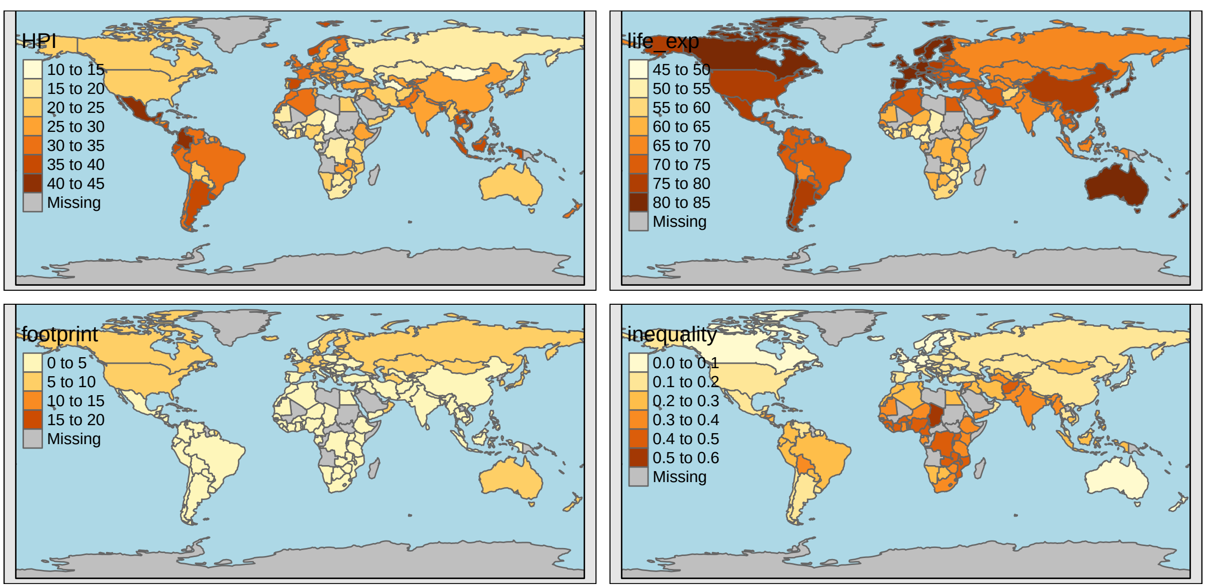
```r
map1 <- tm_shape(World) + tm_polygons() + tm_layout("Eckhart IV projection. Recommended in statistical m

# 2nd Map using projection = eck4 ie: Latitude/Longitude
# Eckert IV is an equal-area pseudocylindrical map projection for world maps. The lateral meridians are
map2 <- tm_shape(World, projection="+proj=eck4") + tm_polygons() + tm_layout("Eckhart IV projection. Rec

# 3rd Map using robin ie: Latitude/Longitude
# It is an arithmetic mean of projected coordinates of Aitoff and equidistant cylindrical projections.
map3 <- tm_shape(World, projection="+proj=robin") + tm_polygons() + tm_layout("Winkel-Tripel projection
```

State what aspect(s) are preserved with the projections you use. Comment on the visual differences and biases across the three maps.

Question 1.3

The 'World' data set contains a column labeled "HPI" that stands for Happy Planet Index. Information about this metric can be found here: http://happyplanetindex.org Make a four panel map that plots: HPI, life expectancy, footprint and inequality.

```r
tm_shape(World) +
    tm_polygons(c("HPI", "life_exp", "footprint", "inequality")) +
    tm_facets(sync = TRUE, nrow = 2, ncol = 2) +
    tm_layout(bg.color="lightblue",
            earth.boundary = TRUE,
            space.color="grey90")
```



Question 2

Altitude data for the USA can be obtained using the raster R package:

```
usa_elev <- raster::getData('alt', country='USA', level=1)
oahu <- crop(usa_elev[[4]], extent(-158.330079, -157.646180, 21.241392, 21.711673))
oahu
```

```
## class      : RasterLayer
## dimensions : 56, 82, 4592  (nrow, ncol, ncell)
## resolution : 0.008333333, 0.008333333  (x, y)
## extent     : -158.3333, -157.65, 21.24167, 21.70833  (xmin, xmax, ymin, ymax)
## crs        : +proj=longlat +ellps=WGS84
## source     : memory
## names      : USA4_msk_alt
## values     : 1, 1164  (min, max)
```

Select the altitude data for Oahu, Hawaii, only.

Question 2.1:

Make the altitude data for Oahu into a spatial object with geometry type points. Include R code used and output of the data set to show that it is in fact now a spatial object. Do not forget to select a CRS.

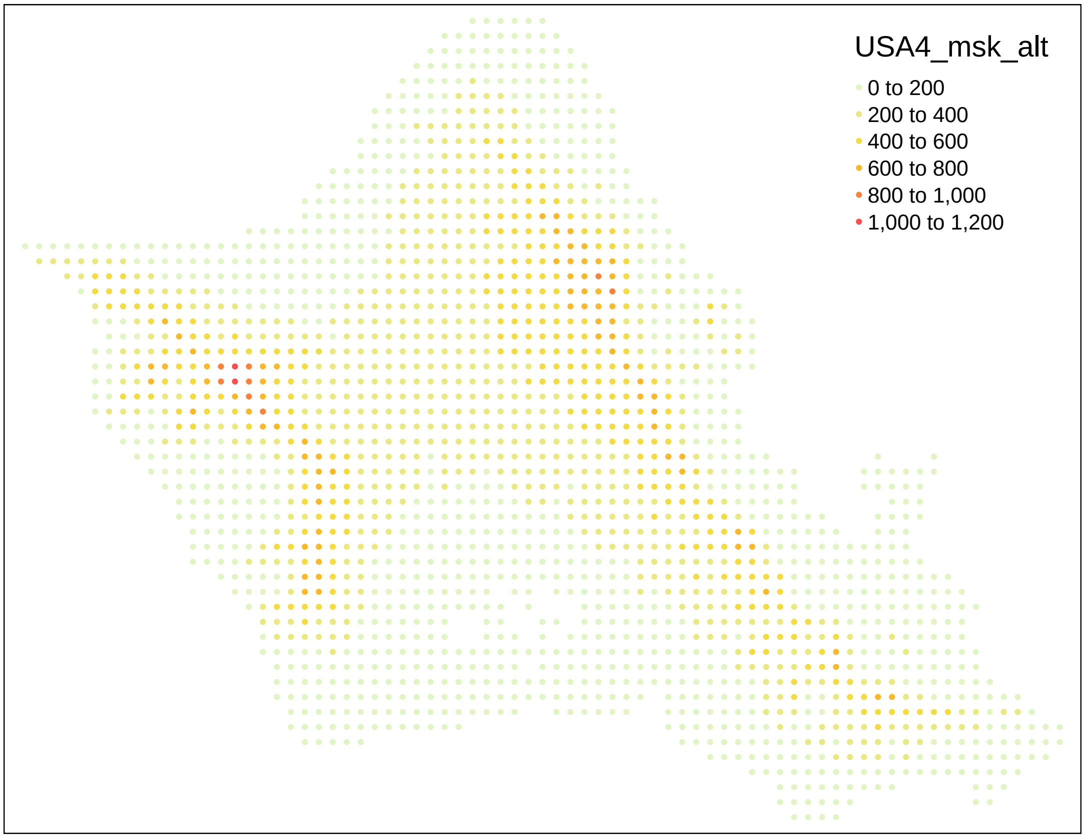
```
oahu <- rasterToPoints(oahu)
dsp <- sf::st_as_sf(data.frame(oahu), coords=c("x", "y"),
crs=4326) %>% st_transform()
dsp
```

```
## Simple feature collection with 1982 features and 1 field
## Geometry type: POINT
## Dimension:     XY
## Bounding box:  xmin: -158.2708 ymin: 21.2625 xmax: -157.6542 ymax: 21.70417
## Geodetic CRS:  WGS 84
## First 10 features:
##    USA4_msk_alt                   geometry
## 1             2 POINT (-158.0042 21.70417)
## 2             7 POINT (-157.9958 21.70417)
## 3             6 POINT (-157.9875 21.70417)
## 4             9 POINT (-157.9792 21.70417)
## 5             5 POINT (-157.9708 21.70417)
## 6             4 POINT (-157.9625 21.70417)
## 7             2 POINT (-158.0208 21.69583)
## 8             6 POINT (-158.0125 21.69583)
## 9             9 POINT (-158.0042 21.69583)
## 10           10 POINT (-157.9958 21.69583)
```

Question 2.2:

Using a basemap of your choosing, plot the altitude points onto a map of Oahu. Allow the points to change colors by altitude. Include R code used.

```r
# plot dsp using tm_shape()
tm_shape(dsp) +
  tm_dots(c("USA4_msk_alt"),
          border.alpha = 0,
          palette=c("#E1F5C4", "#EDE574", "#F9D423", "#FC913A", "#FF4E50"),
          stretch.palette = TRUE)
```

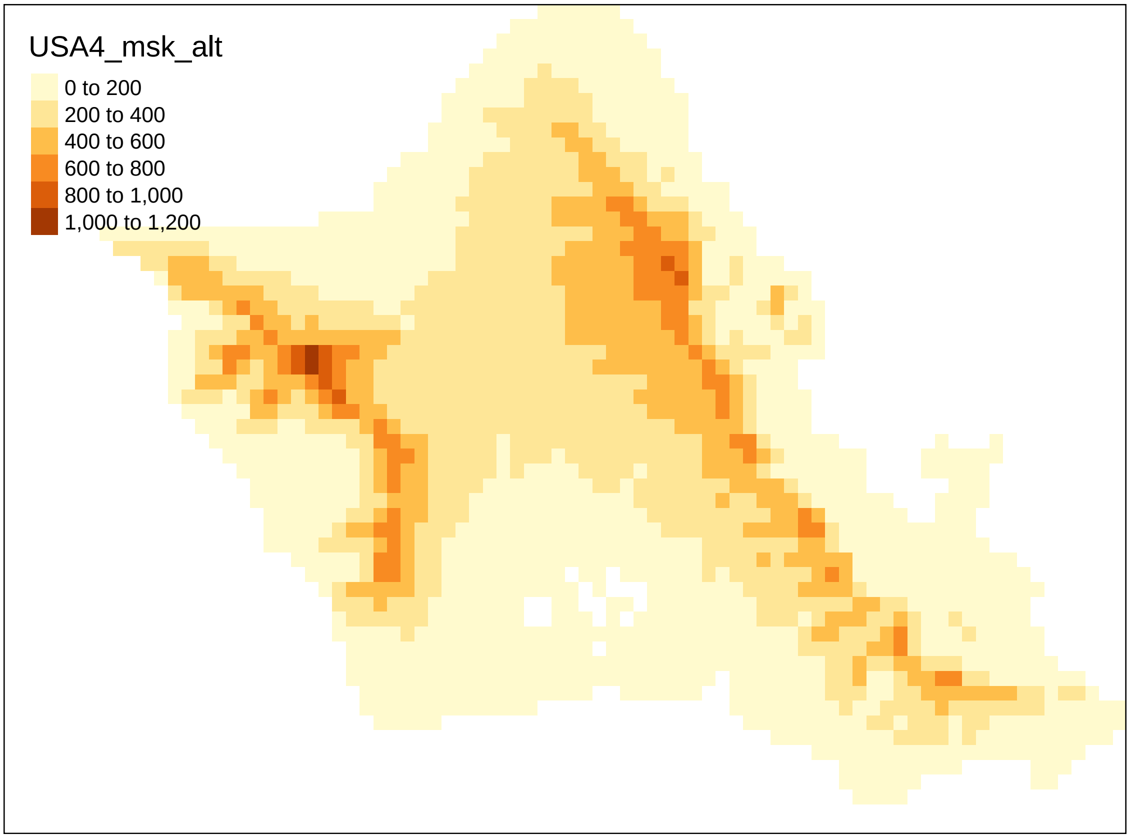


Question 2.3:

Convert the spatial object into a raster, and plot the raster. Include R code and show that the object is now a raster.

```r
# Convert points to sp (assumes that the sf object is called example_points)
oahu_points <- as(dsp, "Spatial")
# create an empty raster layer
empty_raster <- raster(nrow = 56,
                       ncol = 82,
                       xmn = -158.330079,
                       xmx = -157.646180,
                       ymn = 21.241392,
                       ymx = 21.711673)

oahu_raster <- rasterize(oahu_points, empty_raster)
# Taking only the useful dimension of USA4_msk_alt, ID not required.
```

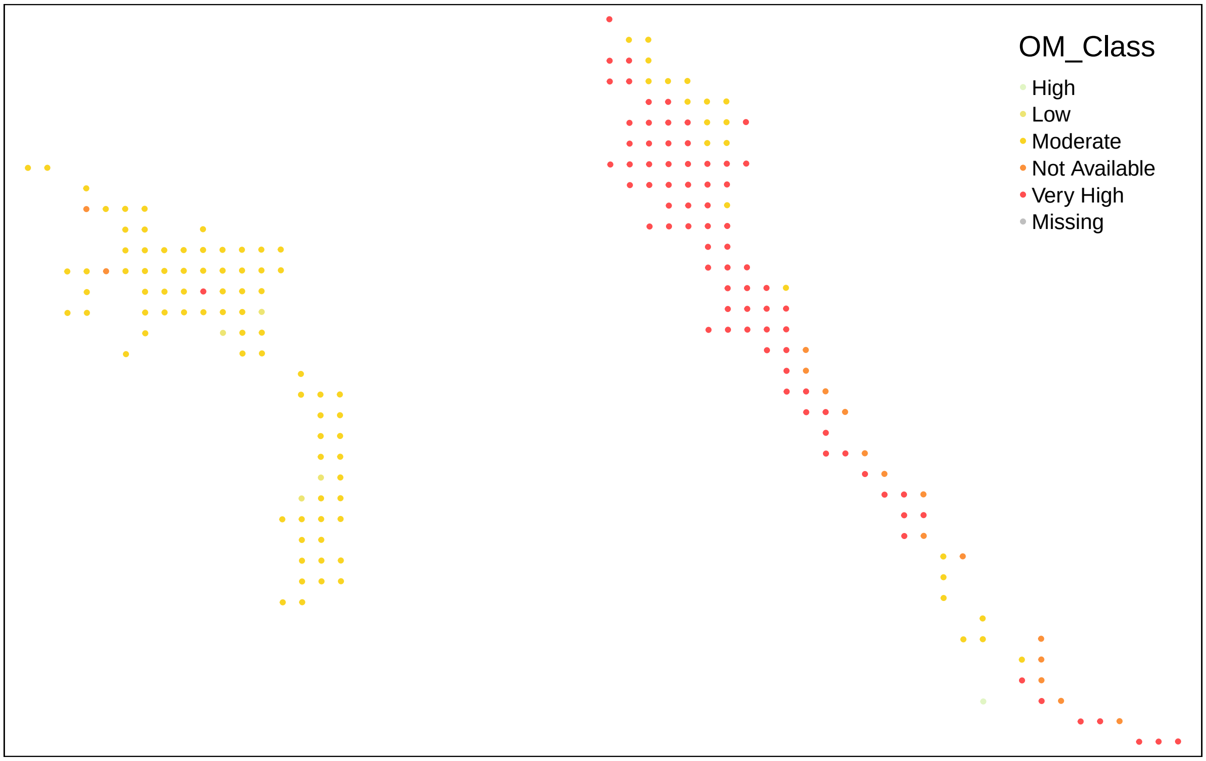
```r
oahu_raster <- oahu_raster$USA4_msk_alt
# Plotting the raster!
tm_shape(oahu_raster) + tm_raster()
```



Question 2.4:

Download the data for organic matter at https://gis.ctahr.hawaii.edu/SoilAtlas#downloads. What soil orders of organic matters exist at altitudes of > 500 m? Make a map of the organic matter soil orders for altitudes > 500 m. Include R code.

```r
organic <- read_sf('./OrganicMatter/OrganicMatter_State.shp')
# filter dsp for altitude > 500
dsp_500 <- dsp %>% filter(USA4_msk_alt > 500)
# oahu_points_organic <- as(dsp_500, "Spatial")
dsp_500 <- sf::st_transform(dsp_500, crs=st_crs(organic))
oahuorganic <- sf::st_join(x = dsp_500, y=organic[dsp_500, c("OM_Class")], join=st_intersects)[1+2*(0:1

tm_shape(oahuorganic) +
  tm_dots(c("OM_Class"),
          border.alpha = 0,
          palette=c("#E1F5C4", "#EDE574", "#F9D423", "#FC913A", "#FF4E50"),
          stretch.palette = TRUE)
```



Question 3.1:

Using the R packages tmap or mapview, make the plot in Question 1.4 interactive.