

# Introduction to GIS

Jaber Belkhiria, DVM, MPVM, PhD  
University of California Davis, School of Veterinary Medicine  
One Health Institute

[https://github.com/jaberbel/Pasteur\\_workshop](https://github.com/jaberbel/Pasteur_workshop)

- Download the folder on your Desktop

github.com/jaberbel/Pasteur\_workshop

GIS workshop for Pasteur institute

Manage topics

Edit

8 commits 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

jaberbel Add files via upload Latest commit a3ba3cb 1 hour ago

World\_Countries Add files via upload 1 hour ago

README.md Update README.md 1 hour ago

novel-coronavirus - data\_cities\_cn.csv Add files via upload 1 hour ago

novel-coronavirus.csv Add files via upload 1 hour ago

novel-coronavirus.xlsx Add files via upload 1 hour ago

README.md

## Pasteur\_workshop

GIS workshop for Pasteur institute This workshop is an intro to using R and GIS. Please download the folder in your Desktop. You will find all the code documents needed and the powerpoint presentation

R Packages:

- library (sp) : <https://cran.r-project.org/web/packages/sp/sp.pdf>



# Overview

- What is GIS ?
- Who uses GIS ?
- What can I do with GIS?
- Data in GIS
- Software's
- 
- What should be in a map?

# What is GIS?

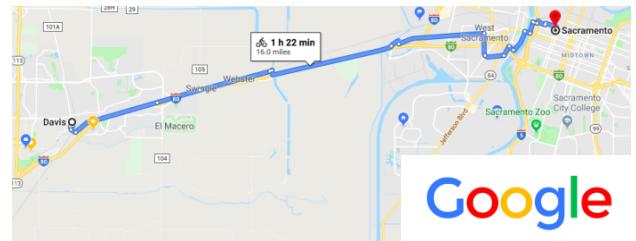
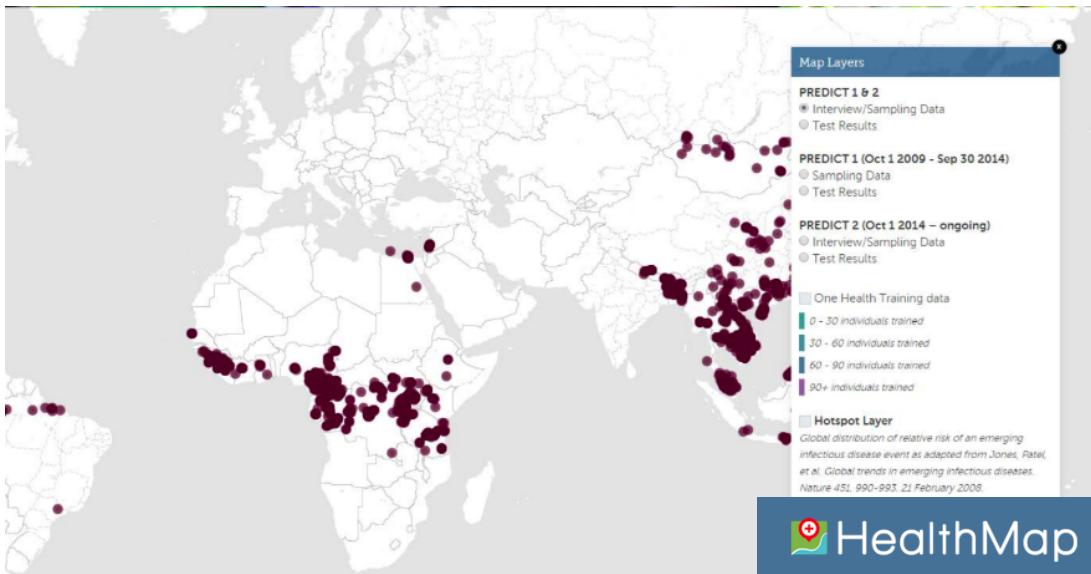
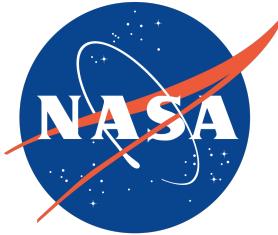
Geographic  
Information  
System



*“Geographic Information Systems is a computer-based tool that analyzes, stores, manipulates and visualizes geographic information, usually in a map. ”*

*In other words: Use a software to connect data to a map in order to visualize and analyse it*

# Who uses GIS?



Google



# What can I do with GIS?

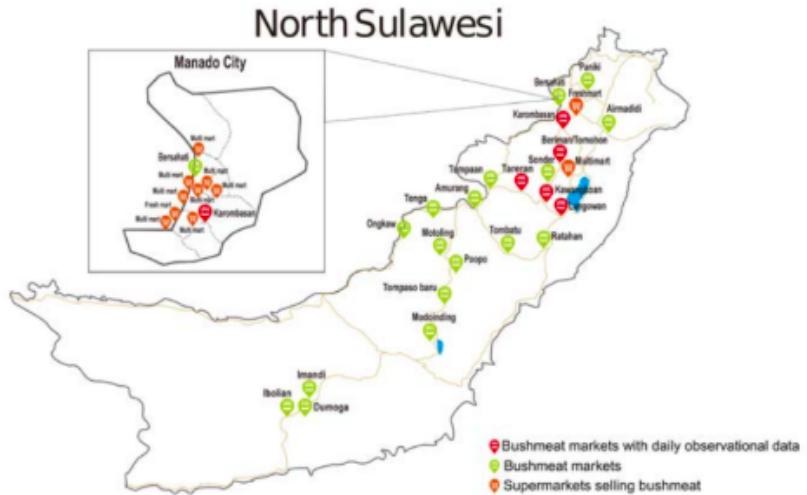
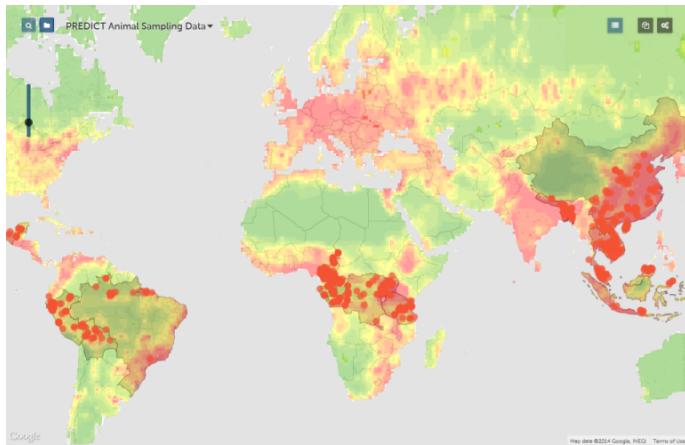


FIGURE 2. Map of the wildlife trade/value chain in North Sulawesi, Indonesia. Green markers indicate bushmeat markets surveyed by PREDICT, and red denotes markets where daily observational data were recently collected. Supermarkets where bushmeat is also available are marked in orange.

VIEW DATA AND CREATE MAPS

# What can I do with GIS?



## PERFORM STATISTICAL ANALYSIS

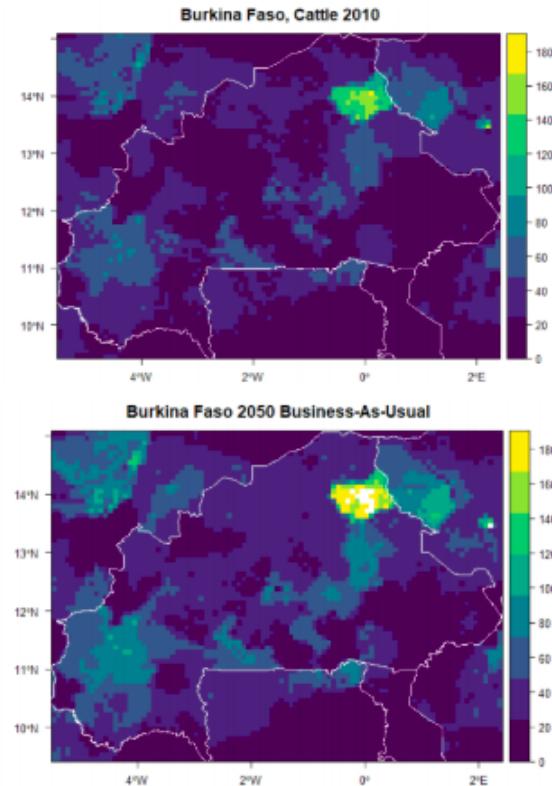
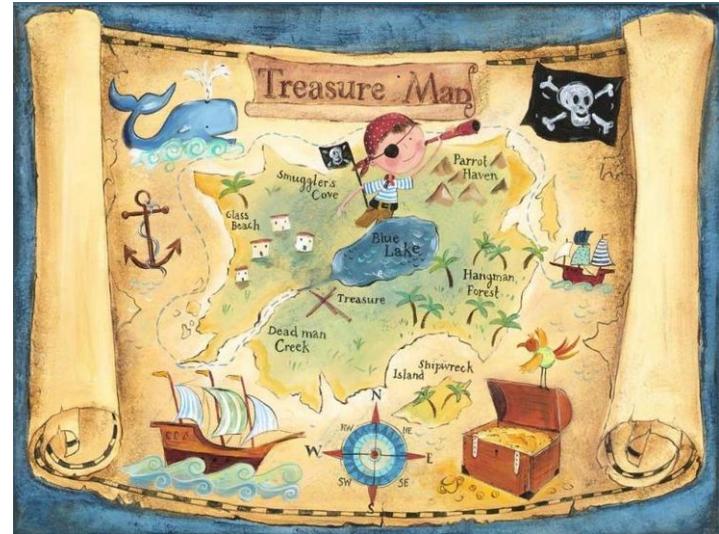


FIGURE 5. Forecasted changes in cattle density for Burkina Faso from 2010 to 2050 under a business-as-usual scenario. Maps produced by PREDICT as part of the Africa Sustainable Livestock 2050 (ASL2050) project in collaboration with FAO.

# What can I do with GIS?

Estimate areas, distances....



.... And More !

# Satellite imagery



## Shapefile

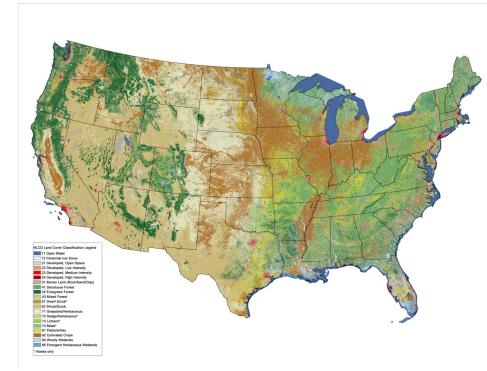


A group of vectors

+

Database linked to it

## Raster



A data matrix

# From a Satellite image to GIS

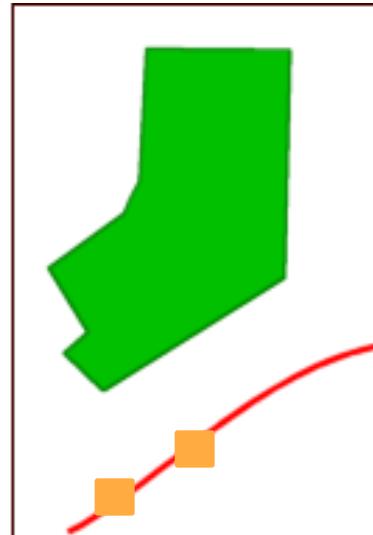


**Real World**

# From a Satellite image to GIS



Real World

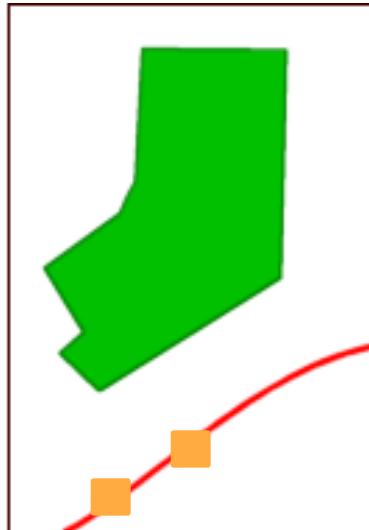


Vector

# From a Satellite image to GIS



Real World

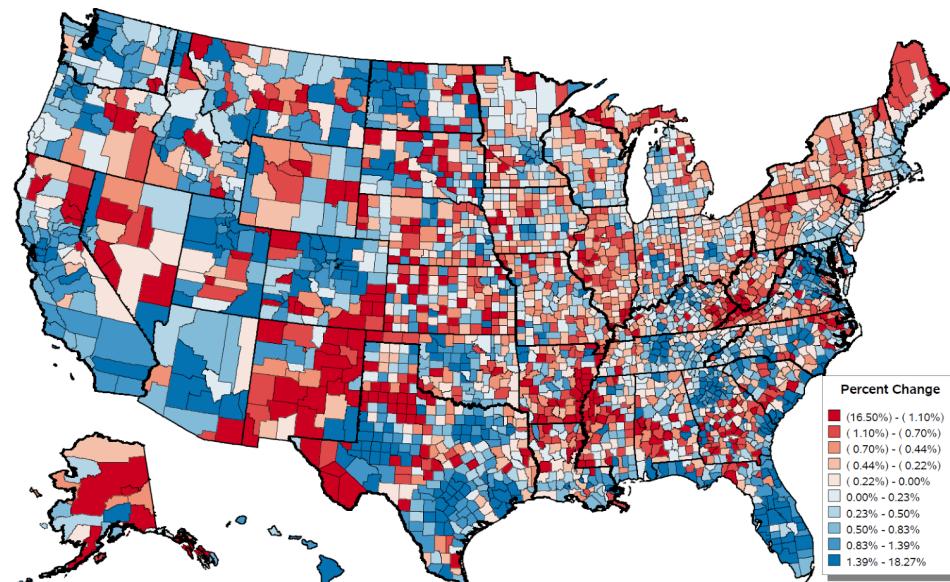


Vector

Field ID	Type	Label	Type
1	Polygon	Hospital	Charles N.
2	Polygon	Car	BMW
3	Polygon	Car	Lada
4	Line	Road	Highway

# Shapefile example

2013 to 2014 Population Change

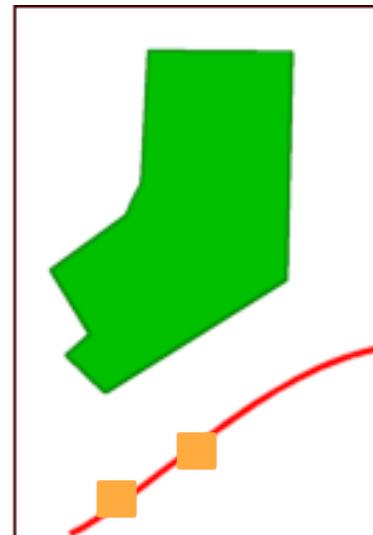


Source: US Census Bureau, 2014 Population Estimates

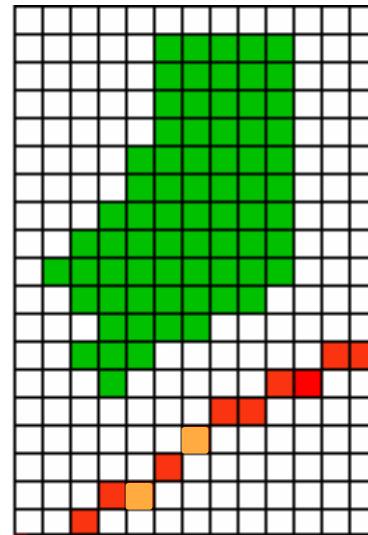
# From a Satellite image to GIS



Real World

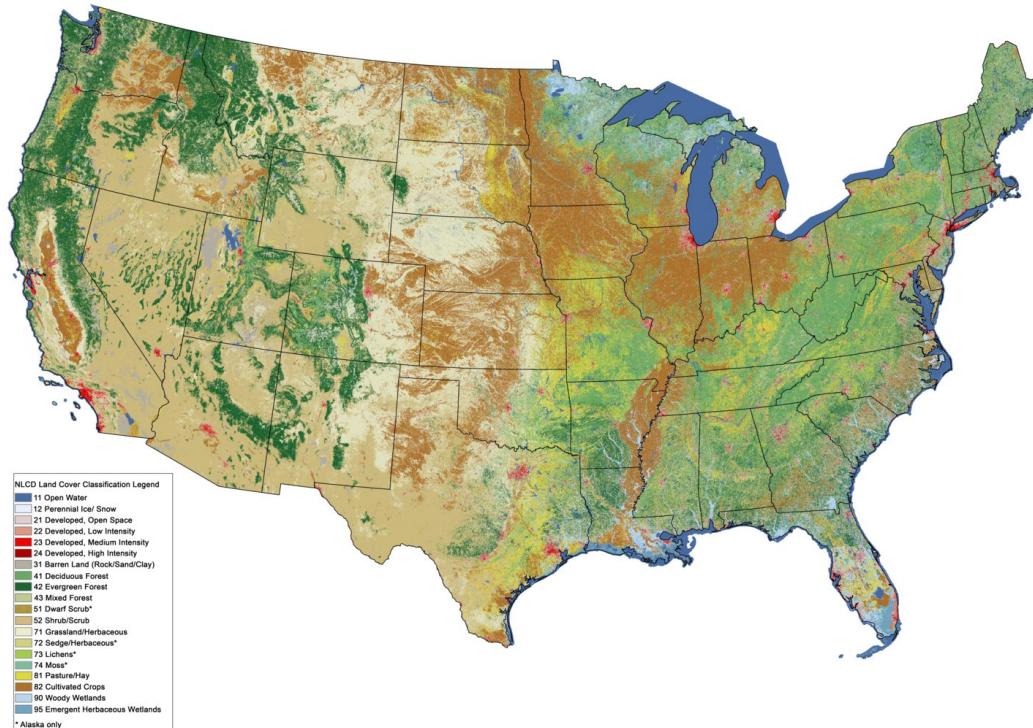


Vector



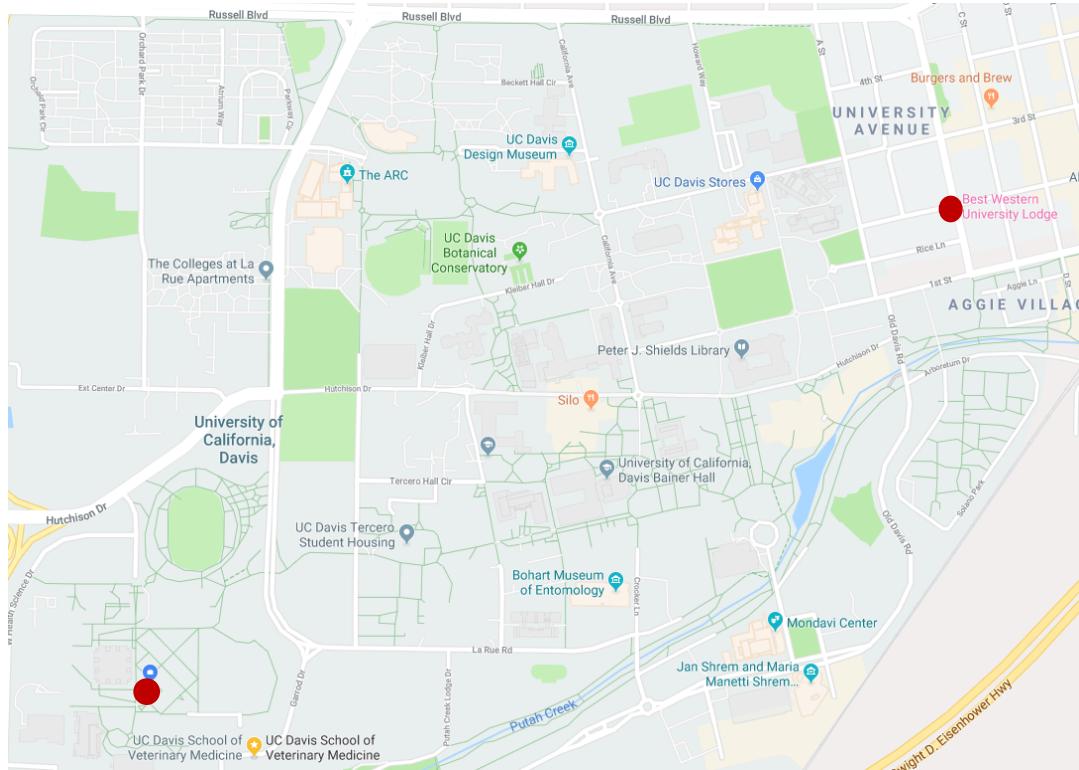
Raster

# Raster Example



# Vector Data

- Points

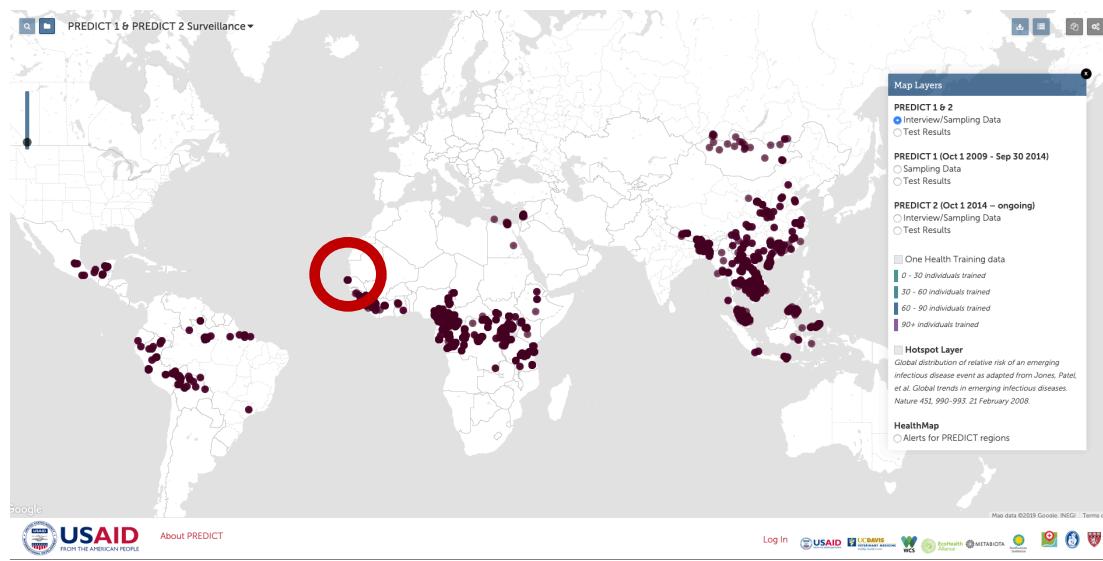
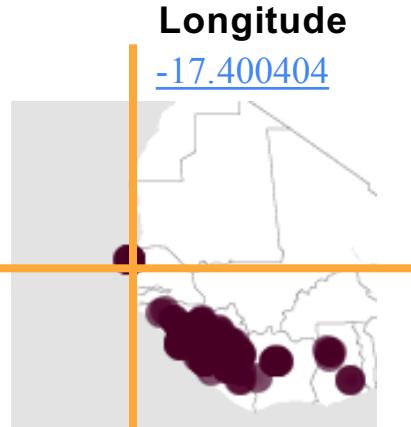


# Vector Data: Point

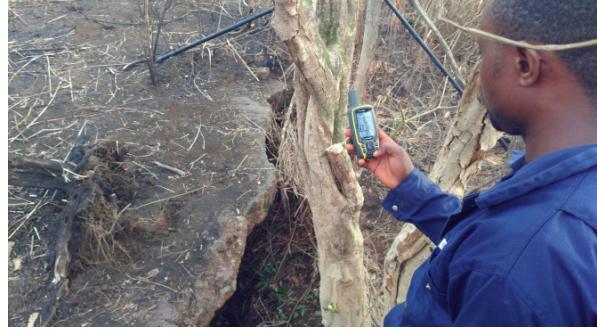
A point is defined by its coordinates: **Latitude and Longitude**

There are different units for coordinates.

Both latitude and longitude have to be in the same unit



# Vector Data: Point



Can you identify what is the location of this point without using your computer?

**38.532194, -121.761469**

# Vector Data: Point

Latitude: - 90 to 90

Longitude : -180 to 180

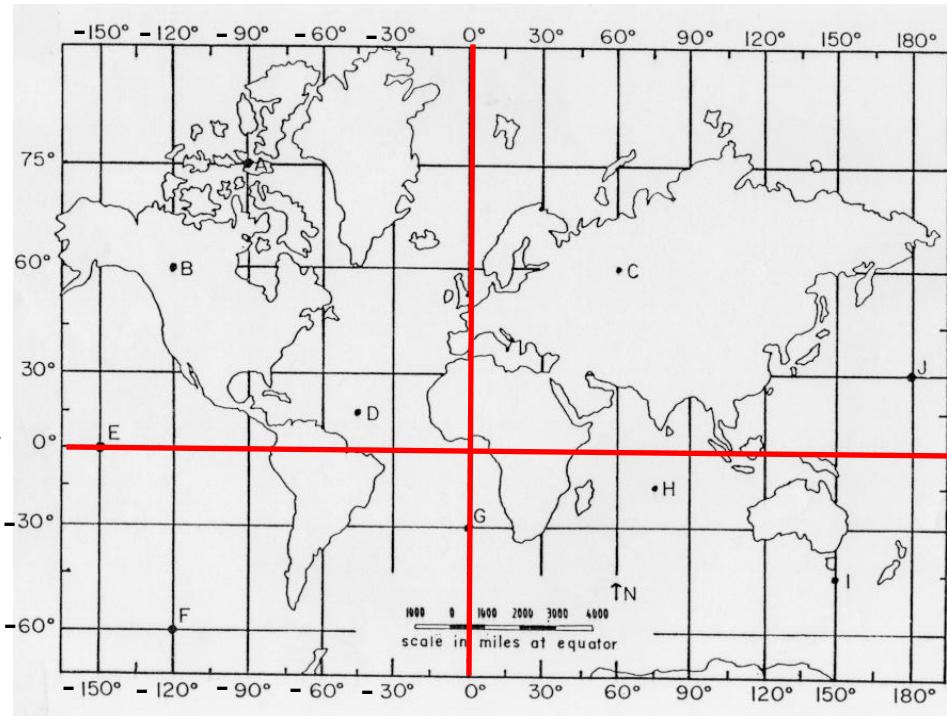
**38.532194, -121.761469**

Thus !!

**38.532194** is a Latitude

**-121.761469** is a Longitude

Prime Meridian



# Vector Data: Point

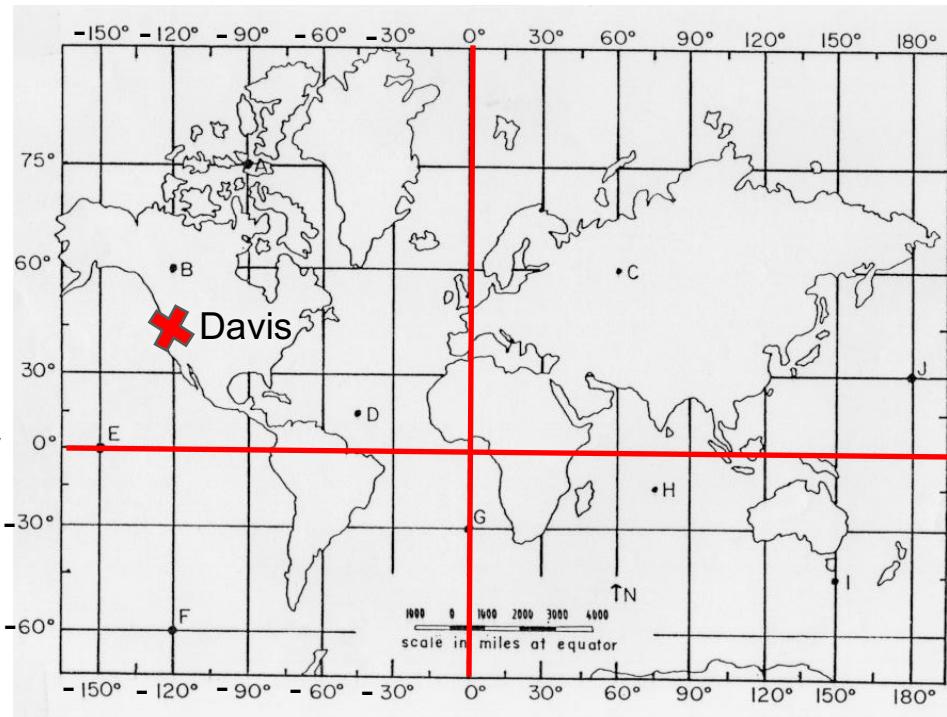
**38.532194, -121.761469**



Equator



Prime Meridian

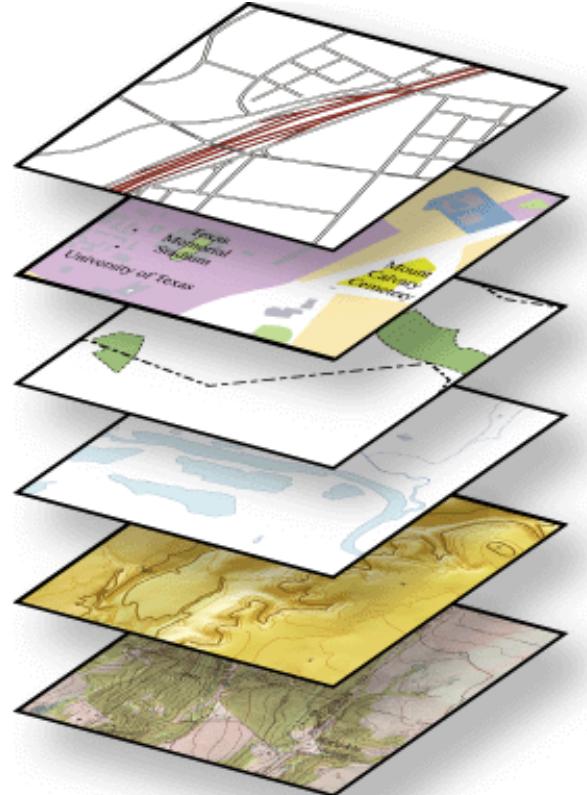
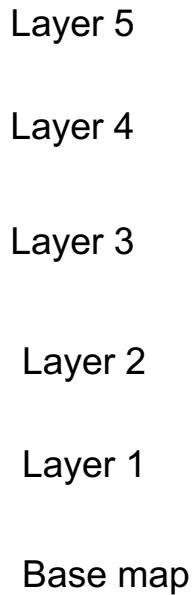
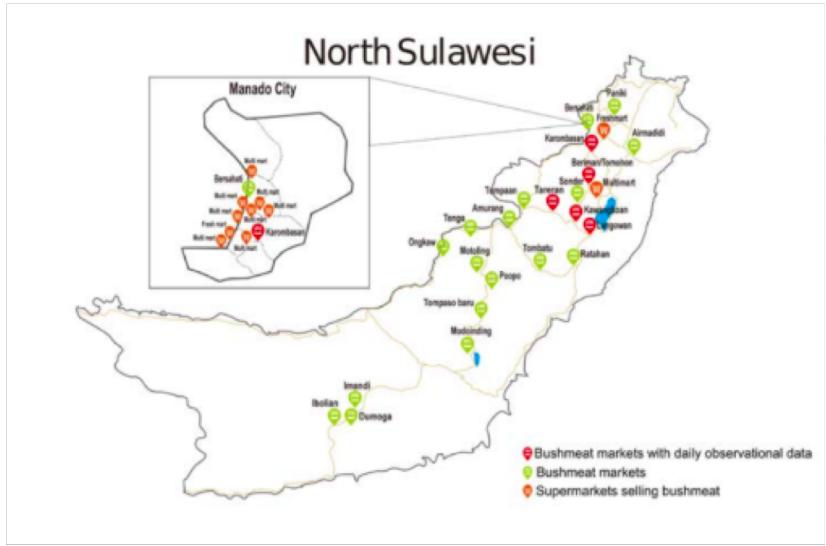


## HUMAN QUESTIONNAIRE LIVING LOCATIONS

This map displays a pin for the GPS location entered for Q9 - Where do you live? of the human questionnaire.  
Grey pins indicate the GPS location was not provided for the participant(s).



# Concept of layering in GIS



# What are the GIS software's?

Free



You have to pay



ArcGIS

Other GIS softwares

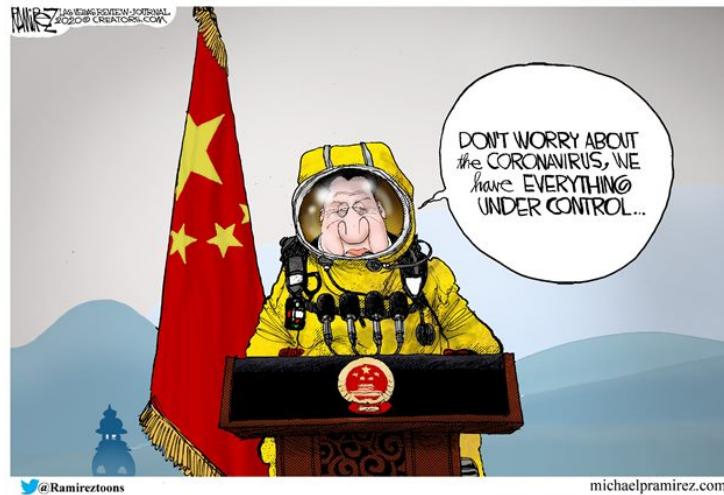
Google Earth





# CORONA VIRUS INVESTIGATION !

- What ingredients will be necessary to start our Spatial investigation?
  - Outbreak locations
  - Number of cases / Number of deaths
  - Map of the world
  - Map of china??



[https://github.com/jaberbel/Pasteur\\_workshop](https://github.com/jaberbel/Pasteur_workshop)

- Download the folder on your Desktop



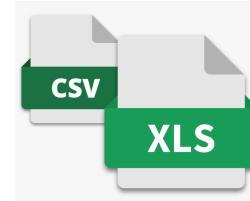
```
#libraries  
install.packages('sp')  
install.packages('raster')  
install.packages('rgdal')  
install.packages('dplyr')  
install.packages('rgeos')  
  
library (sp)  
library(raster)  
library (rgdal)  
library(dplyr)  
library(rgeos)
```

```
# Set working directory  
setwd("~/Desktop/Pasteur_workshop")
```

## Data

- Outbreak locations
- Number of cases /  
Number of deaths
- Map of the world
- Map of china??

### CSV files



- cases latitude/ longitude
- Name of the cities
- Number of deaths/ cases

```
Corona_data <- read.csv("novel-coronavirus.csv")
View(Corona_data)
```

52	United States of America	5210	Arizona	34.20000	-111.5500	
53	Vietnam	3352	Ho Chi Minh City	10.77620	106.8245	
54		NA		NA	NA	
55		NA		NA	NA	
56		NA		NA	NA	
57		NA		NA	NA	
58		NA		NA	NA	

```
Corona_data <- read.csv("novel-coronavirus.csv")
View(Corona_data)
```

```
|
names(Corona_data)
summary(Corona_data )
```

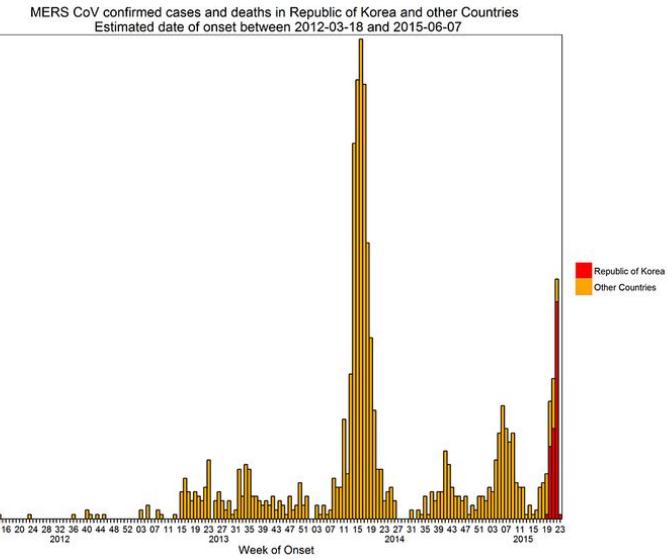
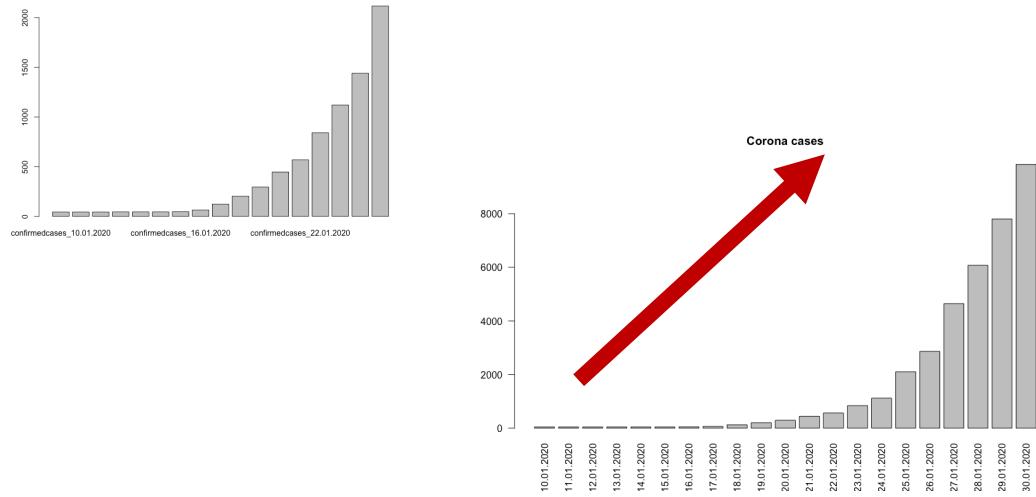
I'm curious to know the evolution of the number of cases per day

```
# Checking on number of cases per day
Cases_data = Corona_data %>%
  select(country, contains('confirmedcases')) %>%
  group_by(country) %>%
  summarise_all(funs(sum))
```

From **Cases data**, **Select** the columns 'Country' and all columns **containing** the word 'confirmedcases'. Then I want you to **group** the data by country and show me **the sum of all** cases per day and per country

```
View(Cases_data)
```

```
barplot(colSums(Cases_data[,2:21]))
barplot(colSums(Cases_data[,2:length(Cases_data)]), las=2, main='Corona cases') # making it horz
```



I'm curious to know the evolution of the **deaths** per day??

From **Cases** data, **Select** the columns 'Country' and all columns **containing** the word '**'XXXXX'**. Then I want you to **group** the data by country and show me **the sum of all cases per day and per country**

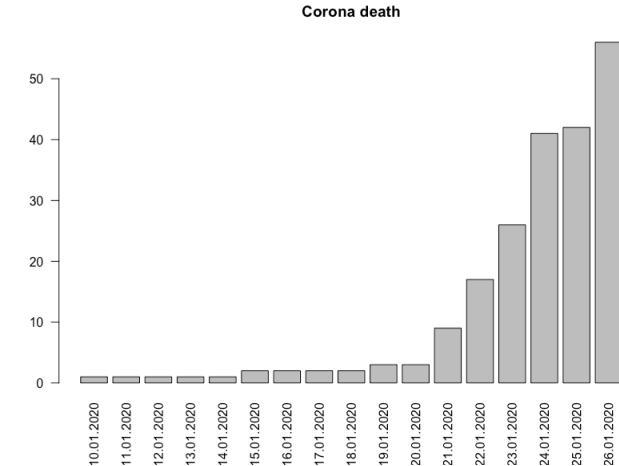
```
# Checking on number of cases per day
Corona_data = Corona_data %>%
  select(country, contains('XXXXXX')) %>%
  group_by(country) %>%
  summarise_all(funs(sum))
```

I'm curious to know the evolution of the **deaths** per day??

From **Cases** data, **Select** the columns 'Country' and all columns **containing** the word '**'death'**'. Then I want you to **group** the data by country and show me **the sum of all cases** per day and per country

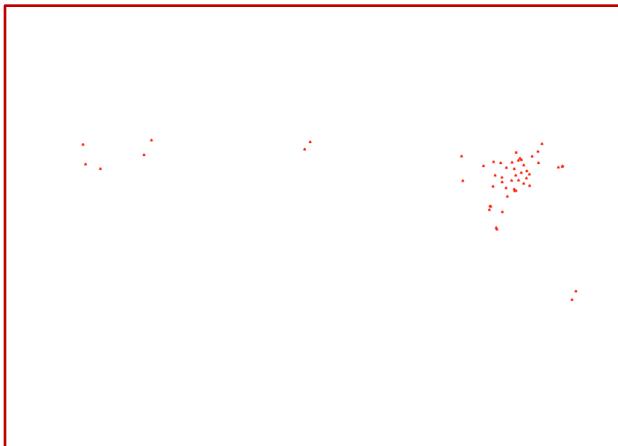
```
# Checking on number of death per day  
Death_data = Corona_data %>%  
  select(country, contains('death')) %>%  
  group_by(country) %>%  
  summarise_all(funs(sum))
```

#Checking deaths specifically for china ????



# Plotting

```
#plotting  
#Creating spatial points  
Corona_locations<- subset(Corona_data, select=c('country','location',"longitude", "latitude"))  
coordinates(Corona_locations) <- c("longitude", "latitude")  
plot(Corona_locations,col='red', pch=2, cex=0.2,lwd=2)
```



Try to change col= 'green'

Try to change pch= 1

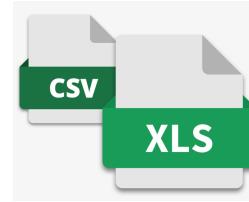
Try to change cex= 10

Try to change lwd= 10

# Data

- Outbreak locations
- Number of cases / Number of deaths
- Map of the world
- Map of china??

## CSV files



- cases latitude/ longitude
- Name of the cities
- Number of deaths/ cases

## Shapefiles



- World shapefile
- China shpefile

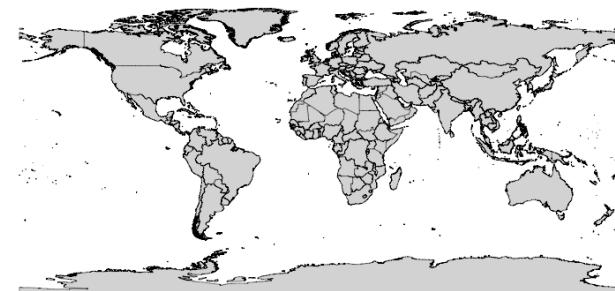
## Adding a shapefile

```
# World shapefile  
World_Countries <- shapefile("World_Countries/World_Countries.shp")  
World_Countries  
  
plot(World_Countries , border='black', col='light grey', lwd=0.5)
```

border='green', col=black', lwd=1

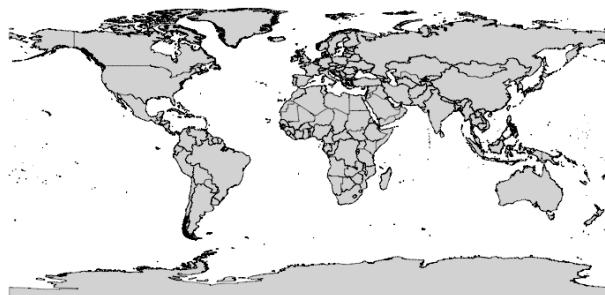
```
> World_Countries  
class       : SpatialPolygonsDataFrame  
features    : 252  
extent     : -180, 180, -90, 83.6236  (xmin, xmax, ymin, ymax)  
crs        : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0  
variables   : 1  
names      : COUNTRY  
min values : Afghanistan  
max values : Zimbabwe
```

World\_Countries\$COUNTRY

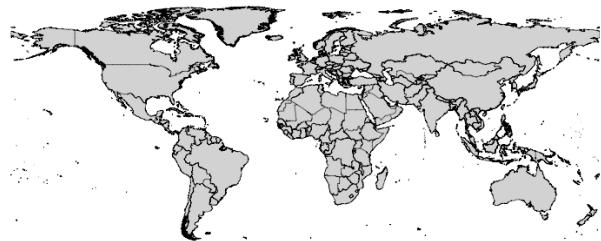


I would like to remove Antarctica !!

```
#removing Antarctica  
World_Countries<-World_Countries[World_Countries$COUNTRY != "Antarctica",]  
plot(World_Countries, border='black', col='light grey', lwd=0.5,main = "Coronavirus cases in the world")
```



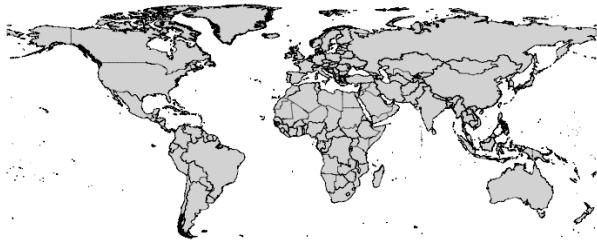
Coronavirus cases in the world



Just for you we can change the projection of the

```
laea <- CRS("+proj=laea +lat_0=52 +lon_0=10 +x_0=4321000 +y_0=3210000 +ellps=GRS80 +units=m +no_defs")  
# Lambert Azimuthal Equal Area  
World_Countries_laea <- spTransform(World_Countries,laea) # project  
plot(World_Countries_laea) # new projection
```

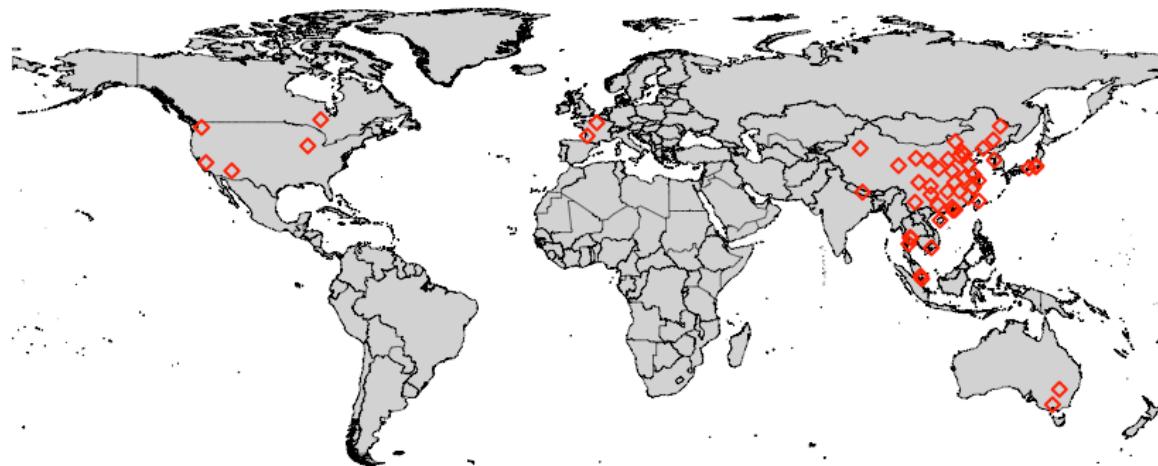
Coronavirus cases in the world



## Adding points

```
#Adding the points  
plot(World_Countries, border='black', col='light grey', lwd=0.5, main = "Coronavirus cases in the world")  
plot(Corona_locations, col='red', pch=5, cex=1, lwd=2, add=TRUE)
```

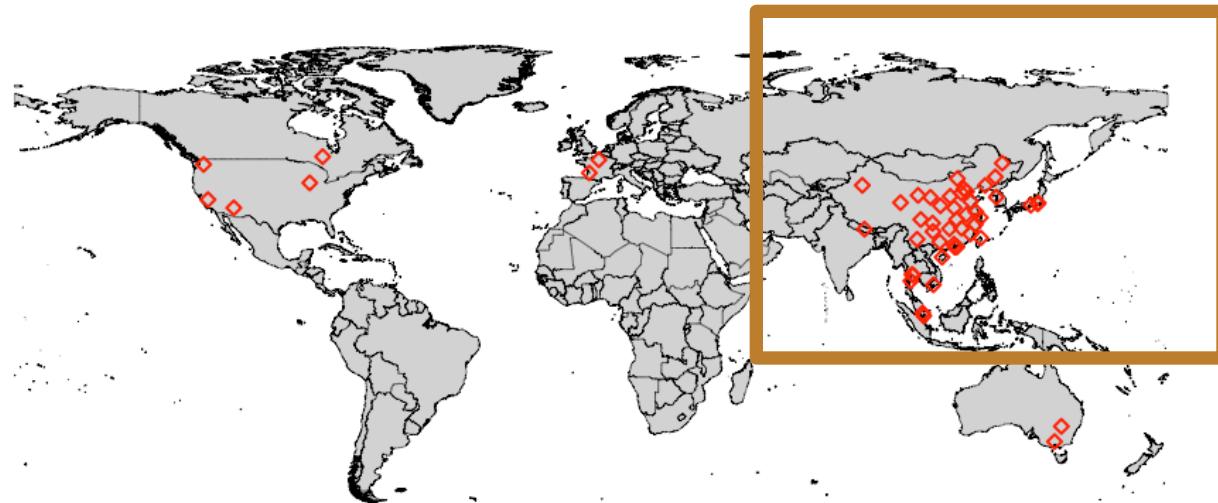
Coronavirus cases in the world



## Adding points

```
#Adding the points  
plot(World_Countries, border='black', col='light grey', lwd=0.5, main = "Coronavirus cases in the world")  
plot(Corona_locations, col='red', pch=5, cex=1, lwd=2, add=TRUE)
```

Coronavirus cases in the world



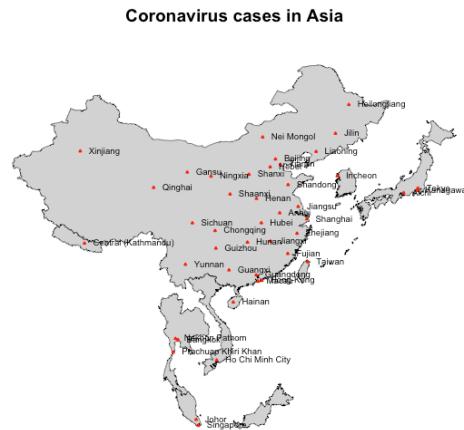
I would like to work specifically on Asian countries

```
#Subsetting multiple countries
Asian_countries<- c("China", "Taiwan", "Japan", 'Nepal', 'South Korea', 'Singapore',
                     'Thailand', 'Vietnam', 'Macau', 'Hong-Kong', 'Malaysia')
Asia<-World_Countries[World_Countries$COUNTRY %in% Asian_countries,]
Corona_asia<-Corona_locations[Corona_locations$country%in% Asian_countries,]
plot(Asia, border='black', col='light grey', lwd=0.5, main='Coronavirus cases in Asia')
plot(Corona_asia, col='red', pch=2, cex=0.2, lwd=2, add= TRUE)
text(Corona_asia, Corona_asia$location, pos = 4, cex= 0.6)
```

Please select only Asian countries from the column 'country' in shapefile world countries

Please select only points in Asian

Please add the location as a text with each point



I would like to work anywhere I want ( after all I want to be free !! )

```
# Cropping a shapefile and points
plot(World_Countries , border='black', col='light grey', lwd=0.5)
drawExtent()
```

```
> drawExtent()
class      : Extent
xmin       : 76.19885
xmax       : 111.9394
ymin       : 23.58612
ymax       : 54.28483
```



Locator active (Esc to finish)

Finish

X max  
Y max

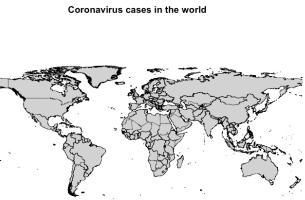


```
e<-extent(xmin,xmax ,ymin,ymax)
World_Countries_crop<- crop (World_Countries,e)
Corona_locations_crop<- crop (Corona_locations,e)
plot(World_Countries_crop, border='black', col='light grey', lwd=0.5,main = "Coronavirus cases in this part of the world")
plot(Corona_locations_crop,col='red', pch=2, cex=0.2,lwd=2, add= TRUE)
text(Corona_locations_crop,Corona_locations_crop$location, pos = 4,cex= 0.6)
```

# Choropleth

I want a map where **countries** change colors by **number of cases**

World\_Countries shapefile



```
> World_Countries
class      : SpatialPolygonsDataFrame
features   : 251
extent    : -180, 180, -58.49861, 83.6236 (xmin, xmax, ymin, ymax)
crs       : +proj=longlat +datum=WGS84 +no_defs +ellps=WGS84 +towgs84=0,0,0
variables  : 1
names     : COUNTRY
min values: Afghanistan
max values: Zimbabwe
```

Case data



country	confirmedcases_10.01.2020	confirmedcases_11.01.2020	confirmedcases_12.01.2020	confirmedcas
1 Australia	0	0	0	0
2 Canada	0	0	0	0
3 China	44	44	44	44
4 France	0	0	0	0
5 Hong-Kong	0	0	0	0
6 Japan	0	0	0	0
7 Macau	0	0	0	0
8 Malaysia	0	0	0	0
9 Nepal	0	0	0	0
10 Singapore	0	0	0	0
11 South Korea	0	0	0	0
12 Taiwan	0	0	0	0

We need to add the cases data to the World\_Country shapefile

MERGING

مرجع

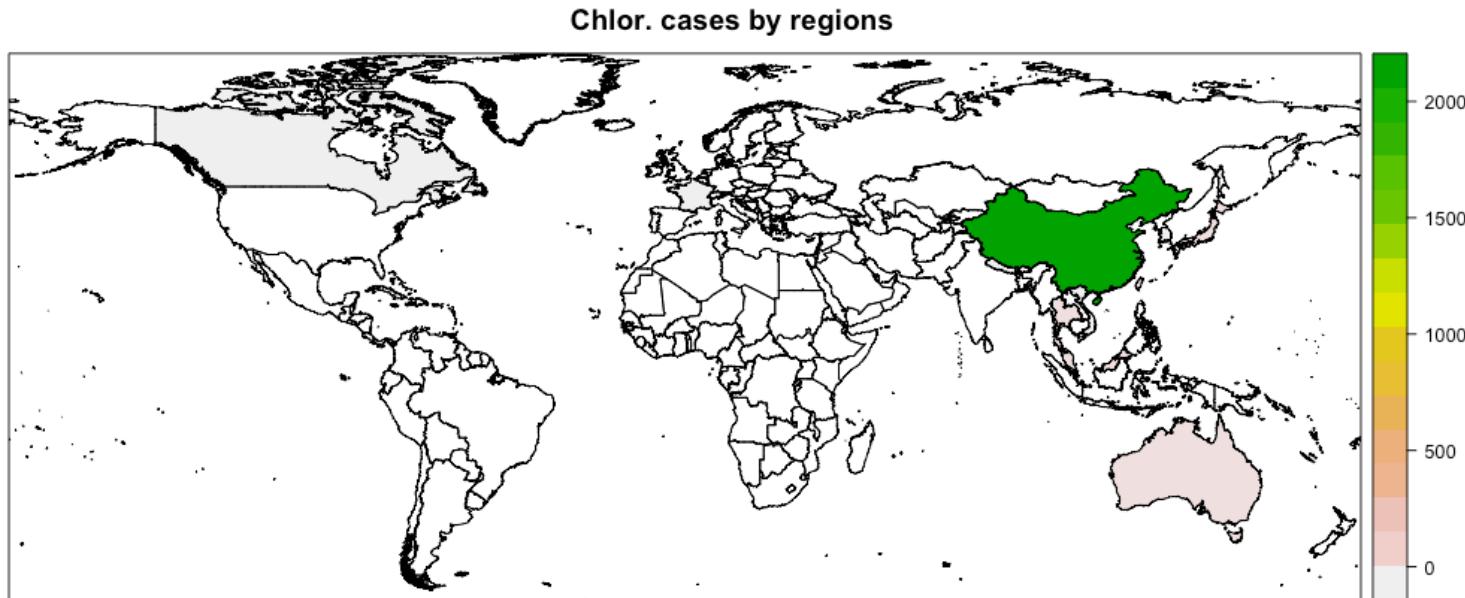
```
World_Countries_cases<- merge(World_Countries, Cases_data, by.x='COUNTRY', by.y='country')
```

World\_Country\_cases is an **excel csv** or a **shapefile**??

# Choropleth

I want a map where **countries** change colors by **number of cases**

```
spplot( World_Countries_cases,'confirmedcases_26.01.2020',col.regions= rev(terrain.colors(25) ),main='Chlor. cases by regions')
```



What if I want to work on **deaths**??

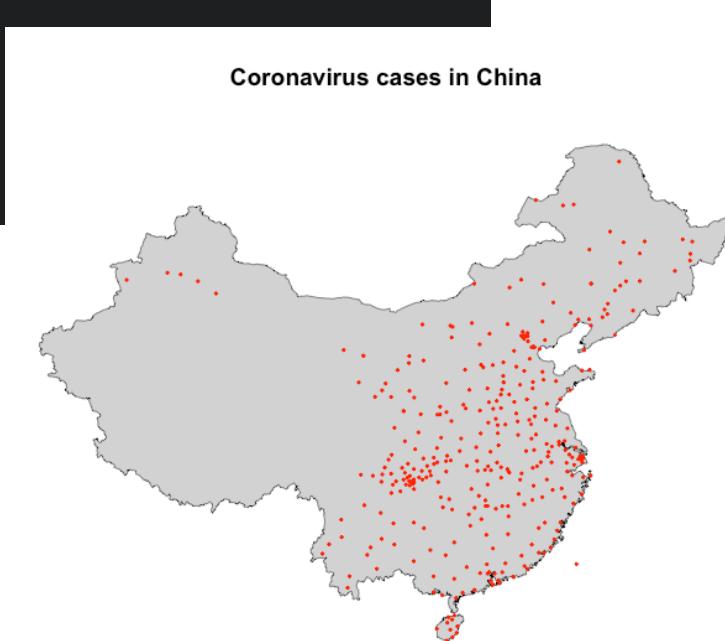
What if I want to work on another date??

# How are things in China?

```
#Working specifically on China
```

```
China<-World_Countries[World_Countries$COUNTRY == "China",]  
China_data <- read.csv('novel-coronavirus - data_cities_cn.csv')
```

```
plot(China, border='black', col='light grey', lwd=0.5, main='Coronavirus cases in China')  
points(China_data $longitude, China_data $latitude,  
       cex = 0.4,  
       pch=20,  
       ylab = "Latitude", xlab="Longitude",  
       col="red") # too big|
```



# How are things in China?

```
plot(China, border='black', col='light grey', lwd=0.5, main='Coronavirus cases in China')
points(China_data $longitude, China_data $latitude,
       cex = China_data$confirmedcases_28.01.2020,
       pch=20,
       ylab = "Latitude", xlab="Longitude",
       col="red") # too big
```

Coronavirus cases in China



# How are things in China?

```
plot(China, border='black', col='light grey', lwd=0.5, main='Coronavirus cases in China')
points(China_data $longitude, China_data $latitude,
       cex = China_data$confirmedcases_28.01.2020/100,
       pch=20,
       ylab = "Latitude", xlab="Longitude",
       col="red")
```

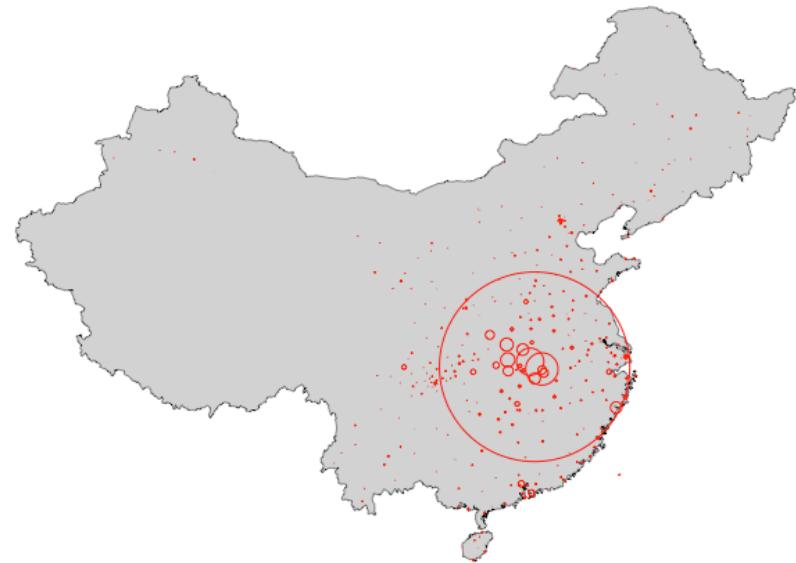
Coronavirus cases in China



# How are things in China?

```
plot(China, border='black', col='light grey', lwd=0.5,main='Coronavirus cases in China')
points(China_data $longitude, China_data $latitude,
       cex = China_data$confirmedcases_28.01.2020/100,
       pch=1,
       ylab = "Latitude", xlab="Longitude",
       col="red")
```

Coronavirus cases in China



# I want a Choropleth of number of cases in China by city?

First create the data

```
China_data <- read.csv('novel-coronavirus - data_cities_cn.csv')
```

```
#Checking specifically for china
China_Cases= Corona_data %>%
  filter(country=='China')%>%
  select(location, contains('confirmedcases')) %>%
  group_by(location) %>%
  summarise_all(funs(sum))
```

# I want a Choropleth of number of cases in China by city?

First create the data

```
china<- read.csv('novel-coronavirus - data_cities_cn.csv')
```

```
#Checking specifically for china
China_Cases= Corona_data %>%
  filter(country=='China')%>%
  select(location, contains('confirmedcases')) %>%
  group_by(location) %>%
  summarise_all(funs(sum))
```

Next Adding the shapefile

```
China_shp<- shapefile('CHN_adm/CHN_adm2.shp')
```

# I want a Choropleth of number of cases in China by city?

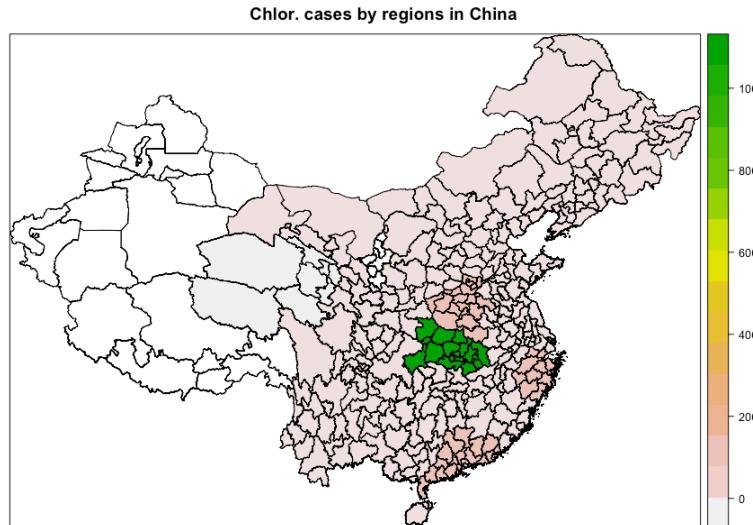
MERGING

مرجع

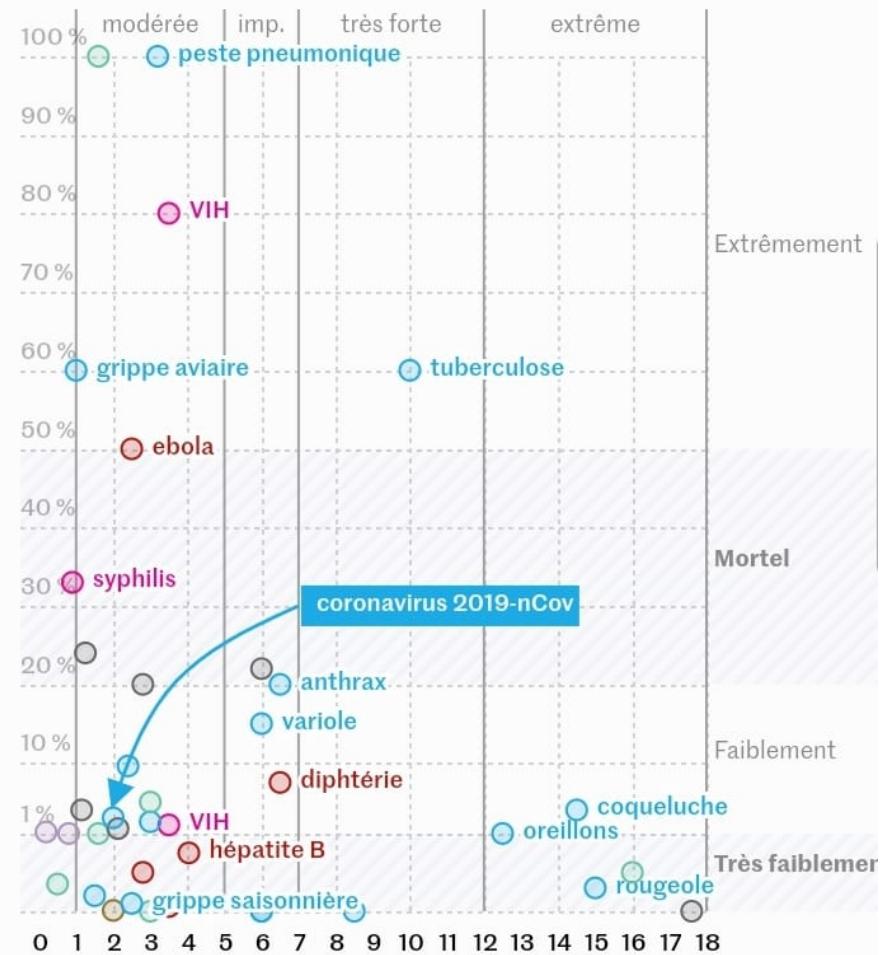
```
China_shp_Cases<- merge(China_shp, China_Cases, by.x='NAME_1', by.y='location')
```

Finally plotting

```
spplot( China_shp_Cases,'confirmedcases_26.01.2020',col.regions= rev(terrain.colors(25) ),  
main='Chlor. cases by regions in China')
```



## Contagiosité





# Jaber Belkhiria

•II 13.66 · DVM MPVM PhD · [Edit](#)

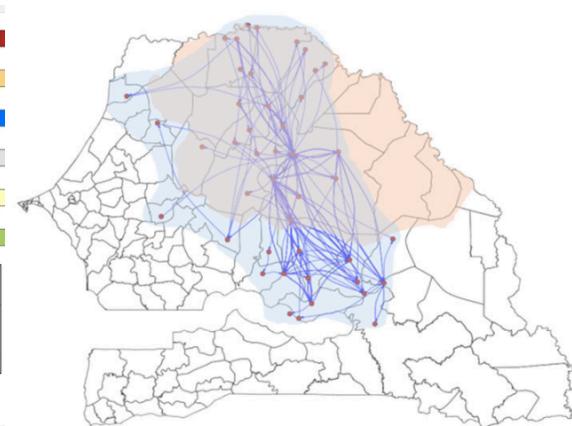
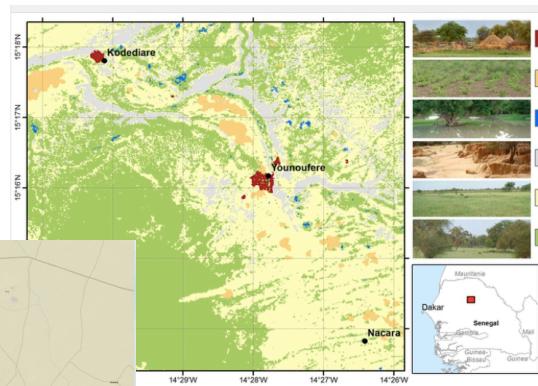
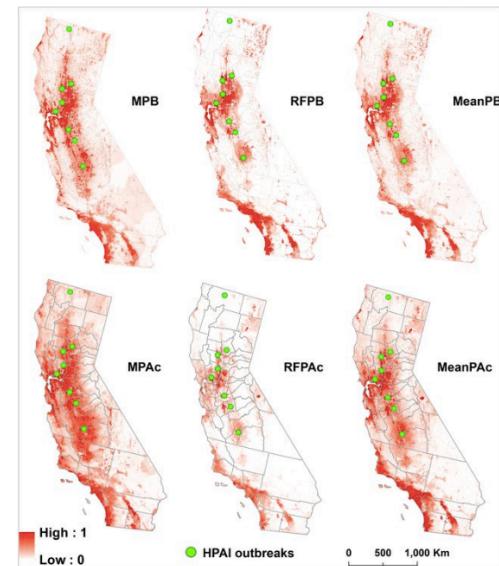
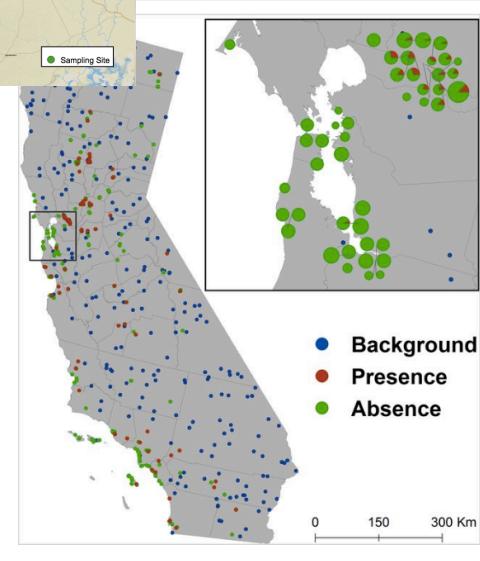
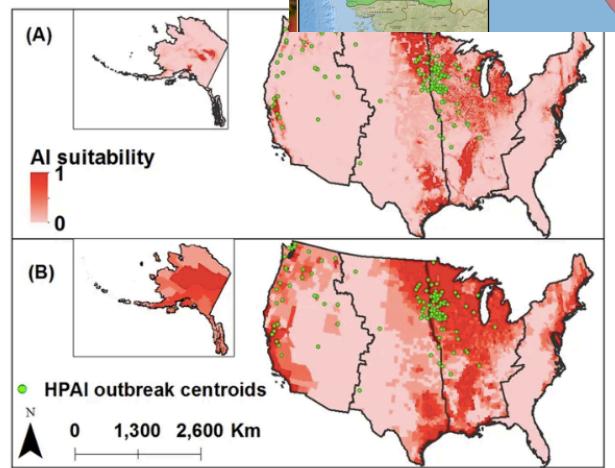


Figure 3





**THANK YOU**