

# Cartographie en R & Introduction au SIG

AMMnet Hackathon — Session "Mapping in R"

---

Dr. Ousmane Diao & équipe facilitatrice

28 Octobre 2025

Ifakara Health Institute / MAP EA Node / Dar es salaam

## 1. Introduction aux données géospatiales et au SIG (~15 min)

- Qu'est-ce qu'une donnée spatiale ?
- Qu'est-ce qu'un SIG ?
- QGIS vs R

## 2. Session pratique (~75 min)

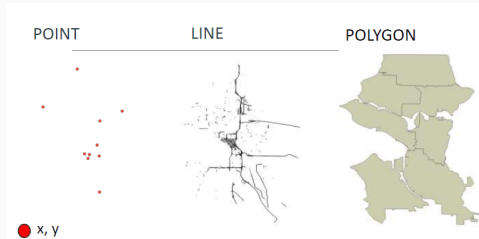
- Démarrer RStudio & découverte de l'interface
- Chargement des données (vecteur & raster)
- Symbolologie & jointures attributaires
- Traitement raster de base
- Exportation de cartes

## Qu'est-ce qu'une donnée (géo)spatiale ?

- Information qui identifie l'emplacement géographique d'objets ou de limites sur Terre (naturels ou construits).
- Des données qui peuvent être cartographiées — elles ajoutent une dimension supplémentaire aux analyses.
- Ex. *non spatiales* : tableau par hôpital et année ; *spatiales* : latitude/longitude, polygones administratifs, etc.

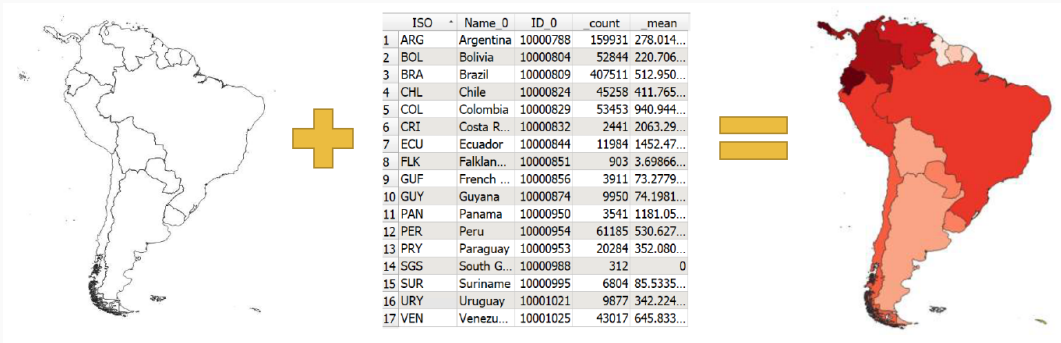
Non-geographical data				Geographic information	
Location	Year	Disease	Conf cases	Latitude	Longitude
Hospital A	2019	TB	30	38.80182	9.02207
Hospital B	2019	TB	245	38.80277	9.02308
Hospital C	2019	TB	72	38.81129	9.01217
Hospital A	2020	TB	102	38.80182	9.02207
Hospital B	2020	TB	324	38.80277	9.02308

- **Point** : localisation unique (ex. GPS).
- **Ligne** : séquence de points connectés (ex. routes, rivières).
- **Polygone** : surface délimitée (ex. districts, pays). Peut contenir des trous.
- Chaque entité possède une **géométrie** (forme/position) et des **attributs** (ex. couleur, population, prévalence).



(Schéma point / ligne / polygone)

## Données vecteur - Attributs



## Types de données spatiales — Raster

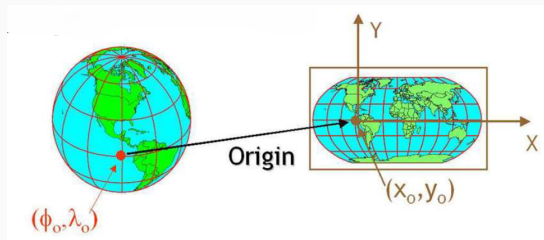
- Grille (lignes/colonnes) de **pixels** (cellules) ; la taille du pixel détermine la résolution.
- Chaque pixel représente une zone géographique et contient une valeur (p. ex. élévation, pluie, population).
- Idéal pour **surfaces continues** (pluie, altitude, densité) ou certaines classes discrètes (occupation du sol).



# Systèmes de coordonnées de référence (CRS)

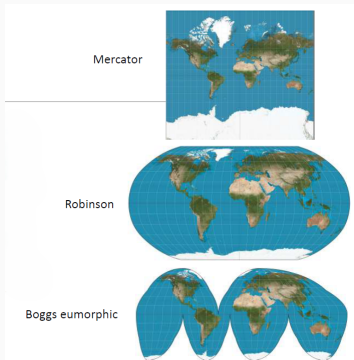
- Un **CRS** définit la manière de localiser un point à la surface de la Terre.
- Les systèmes **géographiques** utilisent la latitude/longitude (ex. WGS84 : EPSG 4326).
- Les systèmes **projetés** (ex. UTM) expriment des coordonnées en mètres, utiles pour mesures/analyses locales.

Bonnes pratiques : Conserver un CRS cohérent entre données ; reprojeter au besoin (`st_transform()`) avant opérations métriques.



# Projections cartographiques

- On projette l'ellipsoïde (Terre) sur un plan 2D, inévitablement avec **distorsions** (surface, distance, forme).
- Choisir une projection selon la zone d'intérêt et les distorsions acceptables (Mercator, Robinson, etc.).
- Démonstrateur utile : <https://www.thetruesize.com>





## Qu'est-ce qu'un SIG ?

- Un **Système d'Information Géographique** permet de manipuler, analyser, gérer et présenter des données spatiales.
- Composants : matériel, logiciels (QGIS, ArcGIS, R), données (vecteur & raster), personnes, méthodes d'analyse.

# QGIS vs R : quand utiliser quoi ?

## QGIS

- Interface intuitive, exploration rapide
- Publication de cartes « clé en main »
- Plugins & outils interactifs
- Pas de code requis (mais possible)

## R

- **Reproductibilité** de bout en bout
- Nettoyage, analyse, carto dans un même flux
- Faceting (séries temporelles, multiples cartes)
- Écosystème de packages (sf, terra, ggplot2, etc.)

## Données vecteur dans R avec sf

- sf (simple features) : lecture/écriture rapides, intégration fluide avec le **tidyverse**.
- Les objets sf se comportent comme des data frames ; toutes les fonctions commencent par st\_.

### Exemples

```
library(sf); library(dplyr)
nga <- st_read("NGA_LGAs.shp") %>%
  mutate(adm1 = snakecase::to_snake_case(State),
         adm2 = snakecase::to_snake_case(LGA))
# Reprojection si besoin
nga_utm <- st_transform(nga, 32632) # UTM zone d'exemple

# Jointure attributaire
metrics <- read.csv("incidence_by_LGA.csv")
nga_map <- left_join(nga, metrics, by = c("adm2" = "lga"))
```

## Conversion $sp \leftrightarrow sf$

```
# sp -> sf
sf_obj <- st_as_sf(sp_obj)
# sf -> sp
sp_obj <- as(sf_obj, "Spatial")
```

- Conseillé : rester en `sf` si possible ; convertir seulement lorsque requis par un package.

- sf : <https://r-spatial.github.io/sf/>
- terra : <https://rspatial.org/terra/>
- ggplot2 : <https://ggplot2.tidyverse.org/>
- Projections/CRS : <https://epsg.io>

Merci !

✉ Contact : [odiao@ihi.or.tz](mailto:odiao@ihi.or.tz)

- Préférer des palettes perceptuelles (ex. viridis) pour valeurs continues.
- Limiter le nombre de classes (5–7) pour les choroplèthes.
- Toujours indiquer unité/échelle, source de données et période.
- Vérifier cohérence CRS, et valeurs manquantes (NA).