

**REPUBLIQUE DU SENEGAL**

**Un peuple-Un but-Une foi**



\*\*\*\*\*

**MINISTRE DE L'ECONOMIE, DES FINANCES ET DU PLAN**

\*\*\*\*\*



**Agence Nationale de la Statistique et de la Démographie**



**Ecole Nationale de la Statistique et de l'Analyse Economique Pierre Ndiaye**

**STATISTIQUES SPATIALES :**

## **PRESENTATION DES PRECEDENTS TP<sub>s</sub>**

**Rédigé par**

FOGWOUNG DJOUFACK Sarah-Laure

NIASS Ahmadou

NGUEMFOUO NGOUMTSA Céline

SENE Malick

*(Elèves en ISE\_Cycle long)*

**Sous la supervision**

M. Aboubacar HEMA

**Année académique : 2024-2025**

## TP 1

### Groupe 4

**Logiciel : GEE**

**Pays : Sénégal**

#### 1. Consigne

Importer et visualiser des données shapefiles.

#### 2. Inputs

Fichiers shapefiles des différentes divisions administratives du Sénégal : niveaux national, régional, départemental et communal.

#### 3. Etapes suivies et méthodes utilisées

- **Présentation de Google Earth Engine :** GEE est une plateforme cloud conçue pour l'analyse et la visualisation de données spatiale. GEE est utilisé dans divers domaines, tels que l'agriculture, la gestion des ressources naturelles, la planification urbaine, la recherche scientifique, et la réponse aux catastrophes.
- **Importation des données shapefiles :** L'importation se fait dans l'onglet Asset, ensuite New, puis sous Table Upload, choisir l'option Shape files.
- **Définition des paramètres de visualisation :** Les paramètres de visualisation **color**, **fillColor**, **width** ont été définis pour styliser les couches géographiques. Ils permettent de spécifier la couleur des contours, la couleur de remplissage, et l'épaisseur des contours des polygones.
- **Visualisation des couches :** La méthode **Map.addLayer** a été utilisée pour ajouter les couches à la carte, et **Map.centerObject** pour centrer la carte sur chaque couche.

#### 4. Outputs

Cartes des différents niveaux administratifs du Sénégal (niveau national, régions, départements et communes).

La carte ci-dessous est la visualisation pour le niveau régions :

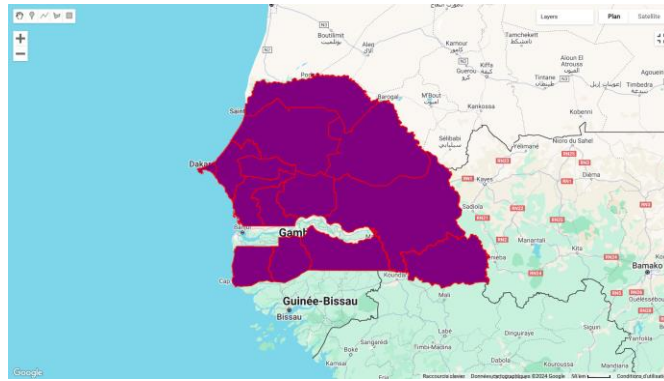


Figure 1 : Visualisation des shapefiles pour le niveau régions

## 5. Défis rencontrés

- Nouveauté du logiciel
- Difficulté à visualiser les labels (noms des régions ...)
- Difficulté à changer de couleurs par polygone dans un shapefile

## TP 2

### Groupe 4

#### Logiciel : GEE

#### Pays : Madagascar

## 1. Consigne

- **Section 1 : Données Vectorielles**
  - Importer les shapefiles ;
  - Calculs statistiques : nombre de géométries suivant les niveaux, superficie de chaque feature, projection, CRS, extent, centroides ;
  - Visualisation.
- **Section 2 : Raster**
  - Importer les rasters et visualiser ;
  - Calcul de moyenne, médiane, écart type, minimum, maximum, et retourner une seule image pour chaque statistique ;
  - Visualiser des données shapefiles.

## 2. Inputs

- Fichiers shapefiles des niveaux administratifs de Madagascar (pays, régions, provinces, arrondissements, communes).
- Images raster représentant les taux de malaria pour les années 2000 à 2022.

## 3. Etapes suivies et méthodes utilisées

#### a. Section 1 : Données vectorielles

- **Importation des shapefiles** : elle s'est fait de manière manuelle, toujours avec l'onglet Assets comme dit plus haut.
- **Visualisation des shapefiles** : Les méthodes **Map.addLayer** et **Map.centerObject** ont été utilisées pour respectivement ajouter des couches, et centrer la carte.
- **Nombre de géométries** : elle a été calculée par la méthode **size**. Madagascar contient 22 régions, 119 provinces, 1579 arrondissements et 17465 communes d'après les données recueillies dans les shapefile.
- **Superficie** : pour calculer les aires des géométries, c'est la méthode **geometry().area** qui a été utilisée. Elle méthode retourne l'aire en mètres carrés. Elle a retourné 590284 km<sup>2</sup> comme superficie du pays.
- **CRS** : La méthode **geometry().projection** a été utilisée pour obtenir le Système de Référence de Coordonnées (CRS) d'une couche. Tous nos shaprfiles avaient pour CRS EPSG 4326.
- **Extent** : La méthode **geometry().bounds** qui a été utilisée retourne un rectangle englobant toutes les géométries de la couche.
- **Centroïdes** : la méthode qui a utilisée ici est **geometry().centroid**.

#### b. Section 2 : Raster

- **Importation et visualisation des rasters** : l'importation des rasters s'est aussi faite de manière manuelle comme celle des shapefiles, et leur visualisation a été effectuée par la méthode **Map.addLayer**.
- **Projection** : la méthode **projection** a été utilisée pour obtenir le CRS qui est également EPSG 4326 comme pour les shapefiles.
- **Statistiques descriptives** : la méthode **reduceRegion** a été utilisée pour calculer la moyenne, la médiane, l'écart type, le minimum, et le maximum des valeurs raster.
- **Création d'une collection d'images et calcul des statistiques** : la méthode **ee.ImageCollection** a été utilisée pour créer une collection d'images à partir des rasters importés. Ensuite, la méthode **reduce** a été utilisée pour calculer les statistiques sur cette collection, puis elles ont été visualisées en utilisant la méthode **Map.addLayer**, chaque statistique étant affichée comme une couche séparée sur la carte.

#### 4. Outputs

- Cartes interactives des différents niveaux administratifs de Madagascar ;
- Résultats des calculs des superficies et des centroïdes pour chaque niveau administratif ;
- Résultats des calculs des statistiques descriptives sur les données raster (moyenne, médiane, écart type, minimum, maximum) ;
- Affichage et exportation des résultats statistiques sous forme d'images raster.

Ci-dessus la couche des centroïdes des régions de Madagascar et le raster de la moyenne des rasters.



Figure 3 : Shapefile des centroïdes des régions de Madagascar

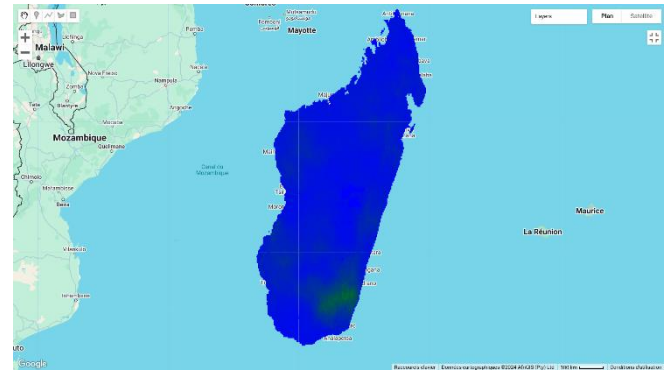


Figure 2 : Raster de la moyenne des rasters

## 5. Défis rencontrés

- Présence des valeurs manquantes
- Problèmes avec les paramètres min et max pour la visualisation, ce qui faisait qu'on avait des problèmes de visualisation.

## TP 3

### Groupe 4

Logiciel : Python

Pays : Madagascar

## 1. Consigne

Calculer le taux de malaria pour le pays, chaque région, chaque département, etc.

## 2. Inputs

- Fichiers raster des taux de malaria de Madagascar pour les années 2000 à 2022 ;
- Fichiers shapefiles des niveaux administratifs de Madagascar (pays, régions, provinces, arrondissements, communes).

## 3. Etapes suivies et méthodes utilisées

- **Librairies utilisées :**
  - **rasterio** : permet de manipuler des fichiers raster ;
  - **geopandas** : pour manipuler des fichiers vecteurs ;
  - **matplotlib.pyplot** : afin de générer des graphiques et d'autres visualisations ;
  - **folium** : utilisée pour la création de cartes interactives ;
  - **os** : permet d'interagir avec le système d'exploitation ;
  - **numpy** : pour des calculs numériques et la manipulation de tableaux ;

- **pandas** : utilisée pour l'analyse de données dans des tableaux (DataFrames).
- **Importation et préparation des données** : tout d'abord, la fonction **os.chdir** a été utilisée pour définir le répertoire de travail qui contient tous les fichiers utilisés. Ensuite, une liste de fichiers .tiff a été créée pour les raster du taux de malaria de Madagascar des années 2000 à 2022. Enfin, la fonction **gpd.read\_file** a été utilisée pour lire les shapefiles représentant les différentes divisions administratives de Madagascar.
- **Géométrie des shapefiles** : la propriété **shape** de **GeoDataFrame** a été utilisée pour compter le nombre de formes distinctes (polygones, points ou lignes) dans chaque shapefile. Les résultats confirment qu'il y a 22 régions, 119 provinces, 1579 arrondissements et 17465 communes.
- **Caractéristiques des shapefiles** : avec la propriété **crs** de **GeoDataFrame (gdf)**, on trouve que le système de coordonnées pour les shapefiles est EPSG 4326.
- **Caractéristiques des rasters** : en utilisant la bibliothèque **rasterio** et les méthodes telles que **src.count**, **src.height**, **src.width**, **src.dtypes**, **src.transform**, **src.nodata** et **src.bounds**, diverses informations ont été extraites :
  - Les rasters contiennent deux bandes ;
  - Chaque raster mesure 329 pixels en hauteur et 176 pixels en largeur ;
  - Les types de données sont en float32. Les transformations affines des données raster ont été calculées ;
  - Les valeurs manquantes ont été identifiées avec une valeur codée de -9999.0 ;
  - Les coordonnées délimitant l'étendue spatiale des rasters ont été déterminées.

Ensuite, d'autres caractéristiques ont été calculées comme le taux de valeurs manquantes dans chacune des bandes (26.2% de valeurs manquantes dans chacune des bandes). La taille des pixels a ensuite été calculée avec la méthode **src.transform** qui nous a donné comme résultat 0.04167 degré en longitude et 0.04165 en latitude.

- **Création du raster moyenne** : pour ce faire, la fonction **zonal\_stats** de la bibliothèque **rasterstats** a été utilisée. Elle permet de calculer des statistiques sur des zones géographiques définies dans un shapefile en fonction des valeurs de pixels d'un raster. Dans le but de trouver la valeur adéquate du paramètre **all\_touched** de la fonction (savoir celui qui fait la moyenne arithmétique suivant la pondération par des pixels), la fonction a été testée avec les deux valeurs du paramètre, un code manuel a été effectué. En comparant les résultats obtenus, il s'est avéré que la valeur du paramètre qui correspond à la prise en compte des pondérations est *False*.
- **Visualisation** : Seul les rasters des taux de malaria par région et province ont été visualisés.
  - *Visualisation du taux moyen de malaria par région*: la fonction **plot** de **GeoPandas** a été utilisée, avec un dégradé de couleur allant du jaune au rouge (**colormap YlOrRd**) pour représenter les variations de taux. Les

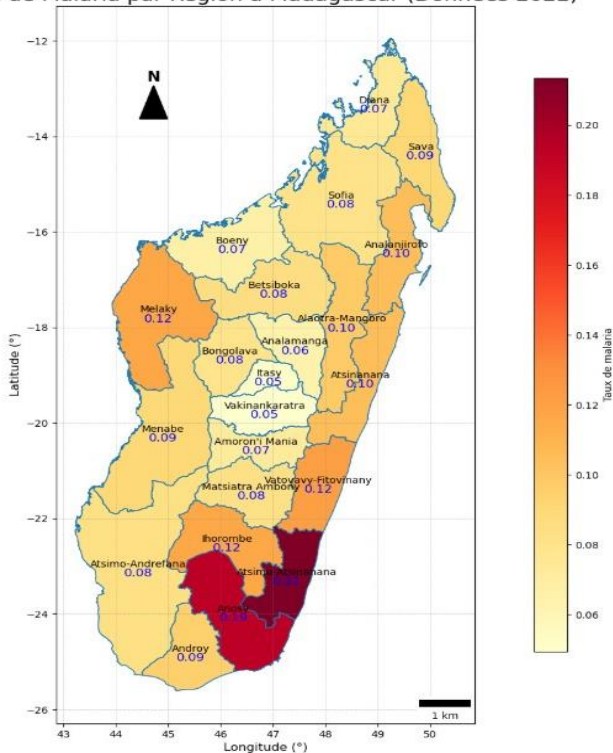
régions sans données étaient colorées en gris. Des étiquettes ont ensuite été ajoutées. Les taux ont été arrondis à deux décimales et affichés en bleu, tandis que les noms étaient affichés en noir. Les valeurs manquantes étaient marquées comme "N/A". Ensuite, la carte a été personnalisée (titre général, des étiquettes pour les axes (longitude et latitude), une flèche et l'échelle).

- *Visualisation du taux moyen de malaria par province*: le même cheminement a été utilisé pour visualiser le taux de malaria par province, à la différence que les labels n'ont pas été affichés et les couleurs utilisées vont du bleue au blanc.

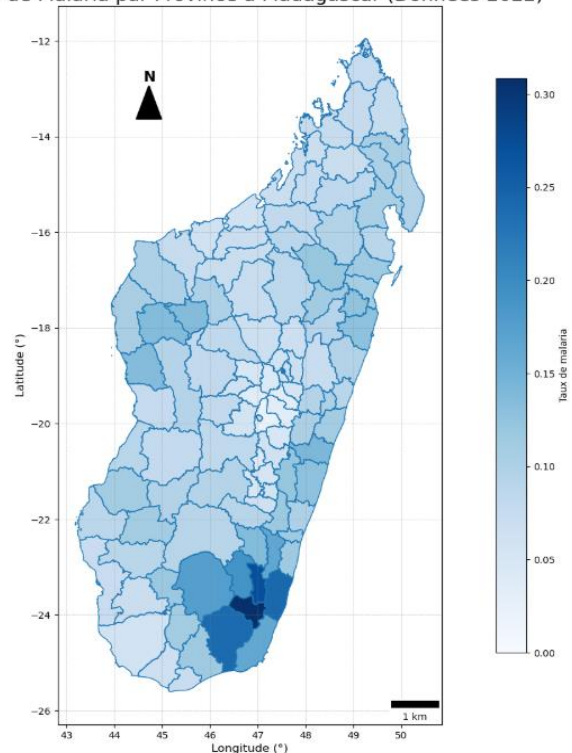
#### 4. Outputs

- Fichier Excel des taux de malaria par région à Madagascar (2022)
- Taux de malaria par région à Madagascar (2022)
- Taux de malaria par province à Madagascar (2022)

Taux de Malaria par Région à Madagascar (Données 2022)



Taux de Malaria par Province à Madagascar (Données 2022)



#### 5. Défis rencontrés

- Nouveauté du logiciel
- Recommencer avec le TP 2 afin d'obtenir les résultats à utiliser pour le TP 3
- Difficulté pour représenter à flèche
- Incompréhension au niveau de l'échelle de la carte



## TP 4

### Groupe 4

Logiciel : Python

Pays : Madagascar

#### 1. Consigne

L'objectif principal du TP était d'analyser la prévalence historique et récente du paludisme à Madagascar en utilisant des données raster. Plus précisément, il s'agissait de :

- **Analyser les tendances historiques** de la prévalence du paludisme en calculant des statistiques (moyenne et écart-type) à partir de cartes de prévalence multi-annuelles.
- **Classer les niveaux de risque** pour les années récentes (2020-2022) en identifiant les zones normales, d'alerte, et critiques selon des seuils définis par les statistiques historiques.
- **Produire des visualisations claires** pour interpréter les données et inclure des éléments cartographiques tels que les limites administratives, une rose des vents, et une barre d'échelle.

#### 2. Inputs

Les rasters représentant le risque de paludisme pour différentes zones à Madagascar de 2000 à 2022, les shapefiles des différents découpages administratifs (Pays, régions, départements, communes).

#### 3. Étapes suivies et méthodes utilisées

**a. Lecture et traitement des données raster historiques :** les cartes de prévalence du paludisme, représentant plusieurs années, ont été extraites et combinées dans un tableau tridimensionnel comprenant les dimensions (années, hauteur, largeur). Cela a permis de créer une base de données structurée pour l'analyse spatiale et temporelle.

- **Actions menées :**
  - L'outil **rasterio** a été utilisé pour ouvrir les fichiers raster et accéder aux bandes principales contenant les données de prévalence.
  - Les données extraites ont été empilées en utilisant **numpy**, ce qui a permis de regrouper les années en une seule matrice multidimensionnelle. Cela a aussi permis de gérer les valeurs manquantes avec les fonctions **np.nanmean** et **np.nanstd**.
- **Résultat :**

À l'issue de cette étape, les moyennes et les écarts-types historiques de la prévalence du paludisme ont été calculés pour chaque pixel. Ces résultats ont été exportés sous forme de fichiers **GeoTIFF** par la suite.

**b. Analyse des années récentes (2020-2022)**

L'objectif de cette étape était de comparer les cartes de prévalence des années 2020 à 2022 avec les statistiques historiques afin d'identifier les zones présentant des niveaux de risque différents.

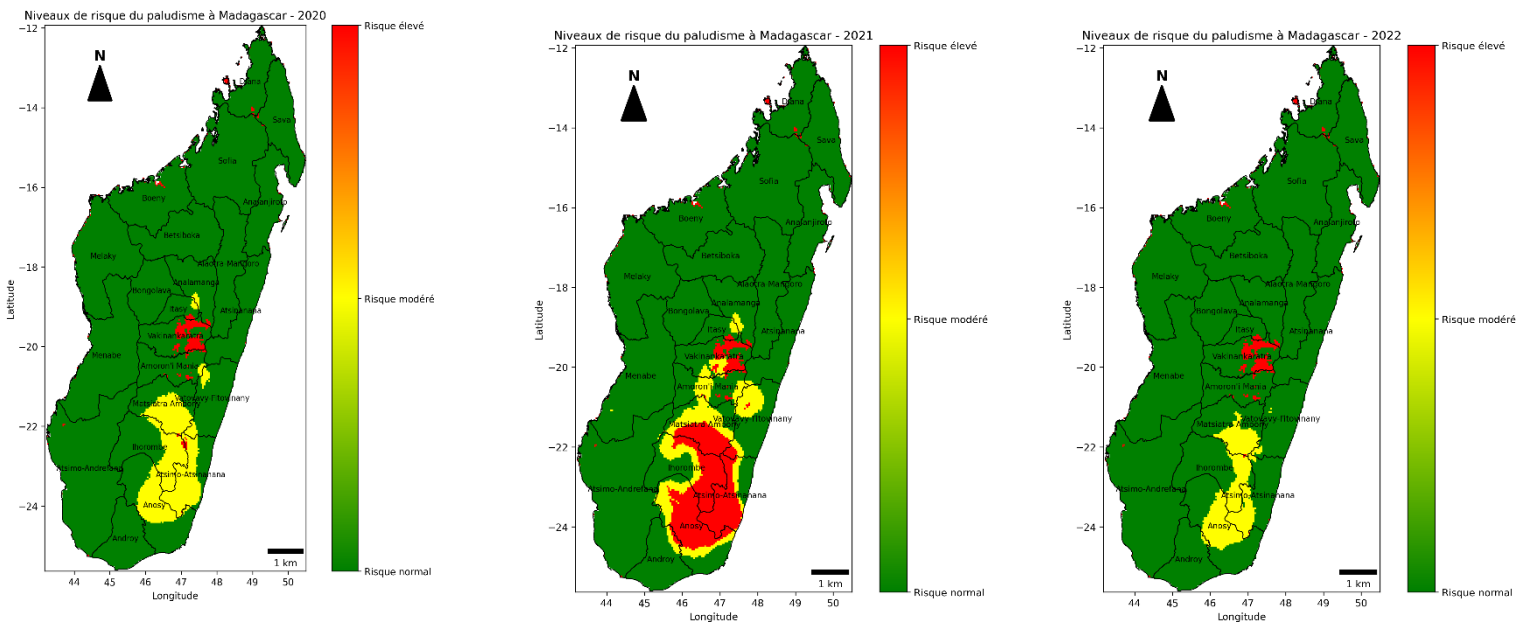


- **Actions menées :**

- Les cartes de prévalence pour 2020, 2021, et 2022 ont été extraites et limitées géographiquement à Madagascar à l'aide des limites administratives. Cette opération a été réalisée en combinant **rasterio** et son module **mask** pour restreindre l'analyse aux pixels d'intérêt.
- Une classification des zones a été effectuée en utilisant les seuils basés sur la moyenne et l'écart-type historique :
  - ✓ **Risque normal** : les pixels dont les valeurs sont inférieures à (moyenne+écart-type).
  - ✓ **Risque modéré** : les pixels dont les valeurs se situent entre (moyenne+écart-type) et (moyenne + 2\*écart-type).
  - ✓ **Risque élevé** : les pixels ayant une valeur supérieure ou égale à la moyenne plus 2 écarts-types.

- **Résultat :**

Cette analyse a permis de produire une carte de classification des risques pour chaque année récente, montrant la répartition des zones normales, d'alerte, et critiques qu'on a visualisé par la suite en ajoutant des éléments cartographiques tels qu'une rose des vents, une barre d'échelle, et des étiquettes d'axes.



## Défis rencontrés

La rose de vents a été construite manuellement étant donné qu'on n'a pas de fonctions au préalable sous python qui le fait déjà.

## TP 5

### Groupe 4

Logiciel : GEE-Python

Pays : Niger

#### 1. Consigne :

L'objectif de ce TP était d'importer les fichiers nécessaires, visualiser le raster WorldPop Niger, puis calculer et exporter le nombre de personnes par zone administrative au format .csv. Par la suite, il s'agissait de reprojeter le raster population à une résolution de 5 km, de le visualiser avec une palette de couleurs adaptée, et de créer un nouveau raster représentant le nombre d'enfants de 0 à 12 ans (en multipliant la population par 0,1 %). À partir de ce raster, il fallait calculer le nombre d'enfants dans chaque classe de malaria (0-3), ainsi que le nombre et le taux d'enfants atteints de malaria par zone administrative, et exporter les résultats.

#### 2. Inputs :

Les rasters représentant le risque de paludisme pour différentes zones à Madagascar de 2000 à 2022, les shapefiles des différents découpages administratifs (Pays, régions, départements, communes) ; le Worldpop.

#### 3. Etapes suivies et méthodes utilisées

- **Importation et Préparation des Données**

- **Données raster** : Importation des données raster pour chaque année (2020, 2021, 2022) afin de représenter la situation de santé par pixel.
- **Données shapefiles** : Importation pour les différents niveaux administratifs (pays, régions, départements, communes).

- **Représentation des Situations**

Classification des pixels suivant trois niveaux (pareil qu'au TP4) :

- **1 : Aucune situation grave**
- **2 : Situation modérée**
- **3 : Crise**

Création de **cartes interactives** pour visualiser les situations par année avec une palette de couleurs (vert, jaune, rouge) correspondant aux niveaux de gravité.

- **Taux d'enfants Affectés par Situation**

- Calcul du raster représentant les enfants de 2 à 10 ans (qu'on a supposé qu'ils représentaient 0,001%) ;
- Calculer la population d'enfants pour chaque situation en multipliant les taux pour chacune des catégories par la population d'enfants ;
- Ramener la population d'enfants pour chaque situation pour chacun des découpages administratifs ;
- Exportation des données sous format .csv.

#### 4. Défis rencontrés

- Recommencer le travail qui a été fait au TP4 (celui-ci étant une continuité du TP4) du fait que le pays avec lequel on travaillait a été modifié du fait de la disponibilité des données (on passe de Madagascar au Niger)
- Difficulté de ramener la taille des pixels à 5000m. Quelque soit le type d'interpolation pour l'agrégateur reducersum utilisé (interpolation bilinéaire, cubiques et proche en proche), nous avons de très grandes pertes de données

### TP 6

#### Groupe 4

**Logiciel :** GEE-Python

**Pays :** Niger

#### 1. Consigne

Le TP consiste à importer le fichier CSV **Points\_data** disponible sur GitHub, puis à compter le nombre de points présents dans chaque unité administrative. Ensuite, il s'agit de créer un raster qui regroupe ces différents points, avant de visualiser ce raster en le classant par catégories.

#### 2. Inputs :

Un fichier csv Points\_data

#### 3. Etapes suivies et méthodes utilisées

- Importation et Préparation des Données (Installation du fichier csv Points\_data dans l'environnement de travail et presentation)
- Calcul du nombre de points suivant les différents niveaux administratifs (pays, région, département et commune) ;
- Création d'un raster de 5km pour représenter tous les événements de la feuille .csv (Chaque événement était représenté sous forme de points) en utilisant la fonction reduceToImage() ;
- Visualisation des données de densité d'événements sous forme de catégories en fonction des terciles calculés directement à partir des valeurs min et max. Pour cela, nous faisons cette division et les terciles constitueront les bornes de séparation ;
- Visualisation de l'évolution du nombre d'événements suivant les années de 2000 à 2005.

4. Outputs

