

Introduction

Dans ce tutoriel nous allons discuter de la création de cartes sur R et de la cartographie de données dessus à partir d'un dataset.

.1 Ressources utilisées

.1.1 R

R est un langage de programmation et un logiciel libre destiné aux statistiques et à la science des données.

Le langage R est largement utilisé par les statisticiens, les data miners, data scientists pour le développement de logiciels statistiques et l'analyse des données. [2]

.1.2 Algeria shapefiles

Le shapefile, ou « fichier de formes » est un format de fichier pour les systèmes d'informations géographiques (SIG) [3] Ce dernier contient les coordonnées géographiques du pays pour pouvoir y cartographier des données. Le notre à été téléchargé à partir du GADM (Global Administrative Areas) [1] (Les notre ce trouvent dans le dossier "SHP"). Note : L'utilisation de ces cartes est limitée à un usage non commercial!

.1.3 Dataset

Le dataset est nul autre que celui fournit par notre professeur du module de Statistique. La feuille choisie étant la seconde.

.1.4 Packages

Pour les packages(Library) utilisés nous en avons deux. Le premier étant "tmap", qui sert à configurer nos cartes et la second étant "sf" qui permet la lecture des données géographique contenu dans notre shapefile.

Mise en pratique

.1 Modification du shapefile

La toute première étape étant de modifier le shapefile (dans notre cas c'est le fichier "gadm36_DZA_1.dbf") et d'y rajouter le contenu de notre dataset pour avoir un shapefile de la sorte :

TYPE	ENGTYP	CO	HASC	1.Q.WILAYA2.C.18	D12.	MAP1	MAP2	MAP3	MAP4
Wilaya	Province	1	DZ.AR	Adrar	4690	78	25	46	31
Wilaya	Province	44	DZ.AD	Ain Defla	88	710	433	272	223
Wilaya	Province	46	DZ.AI	Ain Timouchent	0	325	105	218	166
Wilaya	Province	16	DZ.AL	Alger	0	2078	988	1079	849
Wilaya	Province	23	DZ.AN	Annaba	0	543	265	278	222
Wilaya	Province	5	DZ.BT	Batna	1041	506	358	147	115
Wilaya	Province	8	DZ.BC	Bechar	951	106	51	54	51
Wilaya	Province	6	DZ.BJ	Bejaia	121	479	225	253	240
Wilaya	Province	7	DZ.BS	Biskra	6219	154	92	62	52
Wilaya	Province	9	DZ.BL	Blida	0	1000	585	404	329
Wilaya	Province	34	DZ.BB	Bordj Bou Arreridj	662	467	348	119	95
Wilaya	Province	10	DZ.BU	Bouira	223	476	258	214	188
Wilaya	Province	35	DZ.BM	Boumerdes	0	418	192	222	162
Wilaya	Province	2	DZ.CH	Chlef	52	748	398	349	258
Wilaya	Province	25	DZ.CO	Constantine	0	532	349	182	154
Wilaya	Province	17	DZ.DJ	Djelfa	3136	569	419	150	138
Wilaya	Province	32	DZ.EB	El Bayadh	2172	94	52	39	31
Wilaya	Province	39	DZ.EO	El Oued	5345	105	56	48	36
Wilaya	Province	36	DZ.ET	El Tarf	0	190	76	114	103
Wilaya	Province	47	DZ.GR	Ghardaia	2746	103	70	33	26

FIGURE .1 – Aperçu du shapefile

Une fois ceci fait on peut passer au R !

.2 Partie R

.2.1 Importation des librairies

Première chose que nous allons faire c'est d'inclure nos packages citées précédemment comme cela :

```
library(tmap)  
library(sf)
```

FIGURE .2 – "sf" et "tmap"

.2.2 Lecture du shapefile

Ceci consiste à taper une seule commande qui reçoit comme paramètre le répertoire du shapefile ainsi que son nom :

```
shape <- read_sf(dsn = "./shp", layer = "gadm36_DZA_1")
```

FIGURE .3 – Lecture du shapefile

.2.3 Création de la map

Tout est désormais prêt pour créer notre map. On peut dès maintenant voir un aperçu du shapefile en tapant la commande "qtm(shape)"



FIGURE .4 – Carte retournée par "qtm(shape)"

La fonction qtm nous permet de créer des cartes simples sans grande possibilité de personnalisation. La meilleure façon de faire et de rajouter les différentes fonction proposer par la librairie "tmap". L'une des première

fonctions est la fonction "tm_shape()" qui reçoit comme argument la variable contenant le shapefile (shape dans notre cas)

```
tm_shape(shape)
```

FIGURE .5 – tm_shape(shape)

La seconde, "tm_fill()", permet de sélectionner la colonne à cartographier sur la map, le titre de la légende ainsi que ses classes(avec les valeurs manquantes si y'en a)

```
tm_fill("MAP1", title = "Legende", style = "fixed",  
       breaks = c(0, 150, 400, 600, 1000, 2500),  
       textNA = "Donnee inconnue") |
```

FIGURE .6 – tm_fill(shape)

"tm_text()" permet d'afficher le nom de chaque wilaya, avec 3 arguments, le premier étant la colonne contenant les noms de wilayas, le second étant la taille (AREA dans notre cas pour que le nom soit adapté à la taille de la wilaya en question) le dernier est "root" qui indique l'arborescence pour l'affichage des noms sur la wilaya

```
tm_text("NAME_1", size="AREA", root=10)
```

FIGURE .7 – tm_text()

tm_style() nous permet de choisir la palette de couleur de notre carte ("bw"), la couleur d'arrière plan ainsi que la personnalisation de la légende de la carte.

```
tm_style("bw", legend.frame = TRUE, bg.color="lightblue", legend.bg.color="lightblue")
```

FIGURE .8 – tm_style()

"tm_compass()" nous permet l'affichage du compas en bas à droite de la carte. "tm_credit()" permet l'affichage d'un petit credit en bas à droite de la map aussi par défaut. "tm_layout()" nous permet de configurer le titre et la légende, en choisissant leur police, taille, couleur,..etc "tm_scale_bar()"

permet l'ajout de barre à d'échelle notre carte. On peut stocker le résultat de toutes ses fonctions dans une variable qu'on appellera "tm", on obtiendra ceci :

```
tm <- tm_shape(shape) +  
tm_fill("MAP1", title = "Legende", style = "fixed",  
        breaks = c(0, 150, 400, 600, 1000, 2500),  
        textNA = "Donnee inconnue") +  
tm_borders() +  
tm_text("NAME_1", size="AREA", root=10)+  
tm_style("bw", legend.frame = TRUE, bg.color="lightblue", legend.bg.color="lightblue")+  
tm_compass(type = "rose", show.labels=2) +  
tm_credits("Algerie", size=1.3, fontfamily="Z003") +  
tm_layout("Year 2011 Map1",  
          fontfamily="Z003",  
          title.size=1.8,  
          legend.title.size = 1.5,  
          legend.text.size = 1.3,  
          legend.position = c("left", "bottom"),  
          legend.bg.color = "white",  
          legend.bg.alpha = 0.7)+  
tm_scale_bar()
```

FIGURE .9 – Code

.2.4 Suvegarde

Et voila ! notre carte est terminer et reste plus qu'à la sauvegarder et l'admirer ! Pour sauvegarder notre map rien de plus simple, ceci est la commande à taper :

```
tmap_save(tm, "map1.png", width=4000, height=3000)
```

FIGURE .10 – Sauvegarde

les arguments étant la variable précédemment déclarée, le nom du fichier tel qu'il sera sauvegarder ainsi que la résolution de notre image.

Réalisation

Et pour finir, ci-dessous, les quarts cartes développées :

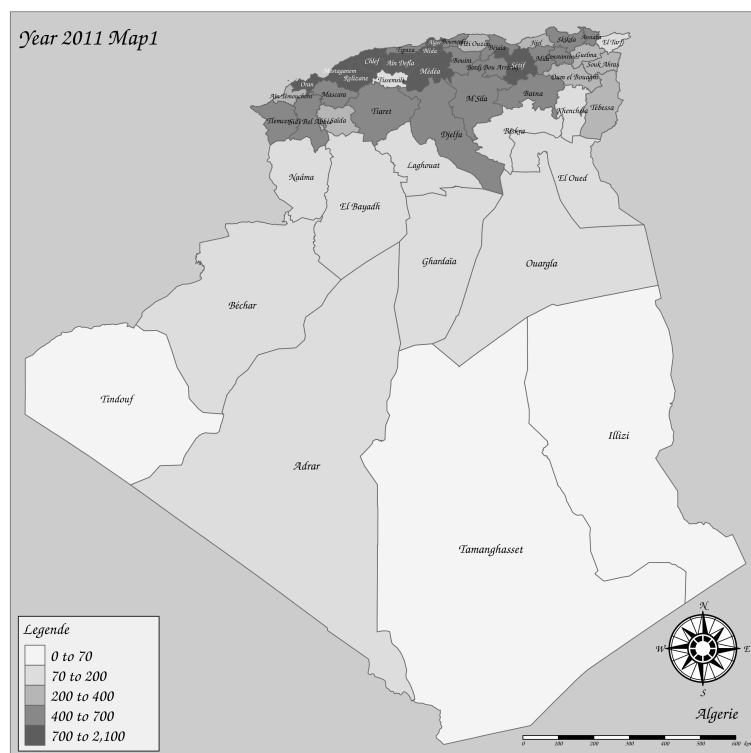


FIGURE .1 – Première carte

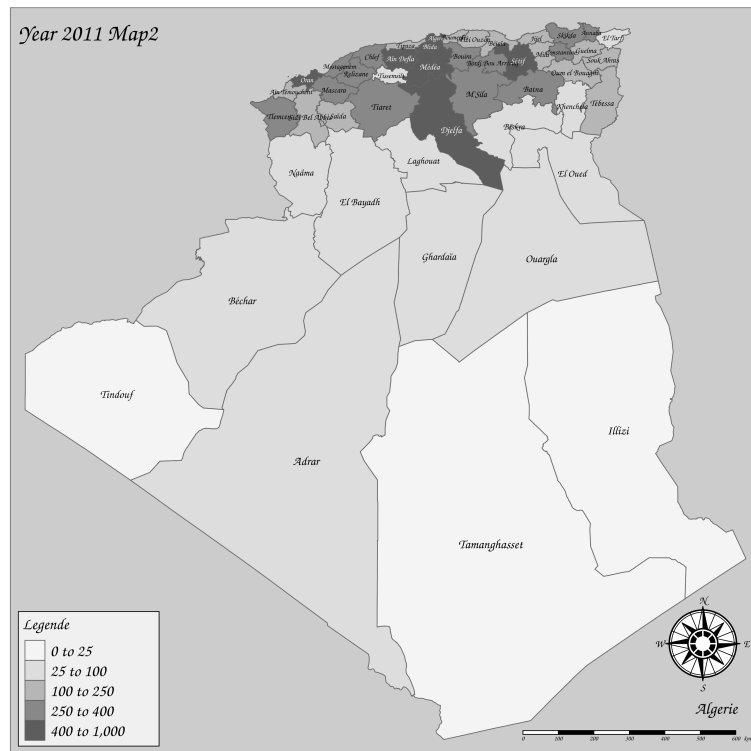


FIGURE .2 – Seconde carte

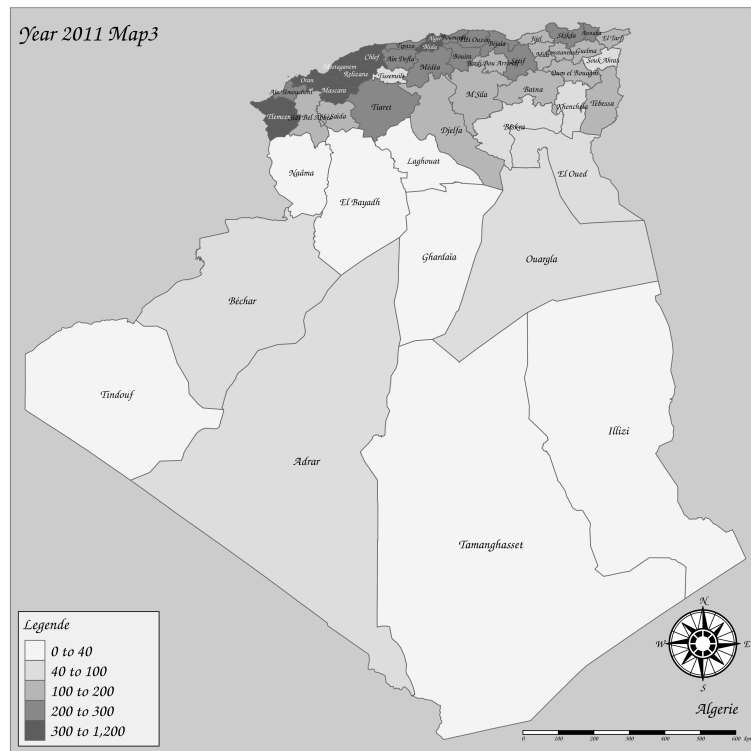


FIGURE .3 – Troisième carte

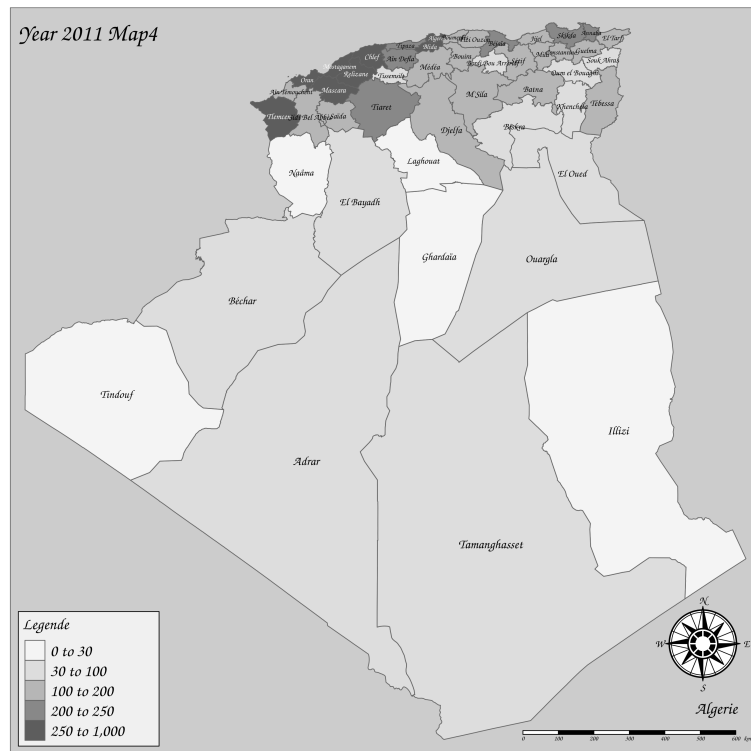


FIGURE .4 – Quatrième carte

Bibliographie

- [1] *GADM*. URL : <https://gadm.org/>.
- [2] *R (langage)*. URL : [https://fr.wikipedia.org/wiki/R_\(langage\)](https://fr.wikipedia.org/wiki/R_(langage)).
- [3] *Shapefile*. URL : <https://fr.wikipedia.org/wiki/Shapefile>.