

La plus belle feuille de la série cartographique au 1:25 000 du PORTUGAL nous invite à quelques exercices de manipulation d'objets spatiaux

P^r Jean R. LOBRY

En prenant comme prétexte la reproduction de la *folha* n° 612 de la série cartographique M888 de l'*Instituto Geográfico do Exército* du PORTUGAL, la fiche propose quelques exercices de projections géographiques et de manipulations d'objets spatiaux.

Table des matières

1	Introduction	2
2	Obtenir le fond de carte du Portugal continental	2
2.1	Les données administratives du PORTUGAL	2
2.2	Sélection des données continentales	5
2.3	Fusion des régions	6
2.4	Fond de carte à basse résolution	7
3	Reproduction de la <i>folha</i> n° 612	7
3.1	Recherche de l'EPSG	7
3.2	Les coins de la <i>folha</i> n° 612	10
3.3	Re-cr�ation de la <i>folha</i> n° 612	11
3.4	V�rification avec une carte interactive	11
4	Reproduction du <i>Diagrama de localiza�o</i>	12
4.1	G�n�ration du maillage par translation	12
4.2	Intersection d'objets spatiaux	15
4.3	Relations entre objets spatiaux	17
4.4	V�rification avec une carte interactive	19
4.5	Nomenclature des cartes	19
4.6	Optimisation du d�coupage	22
	R�f�rences	32

1 Introduction

SI vous souhaitez découvrir le PORTUGAL à vélo, je vous recommande vivement de vous procurer les feuilles de la *Carta Militar de Portugal* (voir la figure 15 page 31) correspondant à la région d'intérêt (voir le *diagrama de localização* de la figure 13 page 29). Il s'agit de l'équivalent de ce que l'on appelle couramment en FRANCE, un peu par abus de langage, les cartes d'état-major au 1 :25 000, commercialisées par l'IGN sous le nom de TOP 25. C'est l'échelle idéale pour randonner à bicyclette, explorer les chemins de traverse et dénicher des sites toujours plus enchanteurs. Vous pouvez commander ces cartes directement, pour une somme plus que raisonnable, sur le site de l'« *Instituto Geográfico do Exército*¹. »

C'EST à l'occasion d'une excursion dans le sud du PORTUGAL, dans la région de l'ALGARVE, que je suis tombé par hasard sur l'extraordinaire carte n° 612 (figure 1 page 3). Elle est pour moi d'une très grande poésie, très surréaliste. Elle me fait penser à un dessin du chat de PHILIPPE GELUCK où on le voit, en plein désert, contempler sur un panneau une carte d'orientation vierge de toute inscription sauf du fameux point « vous êtes ici. » C'est un peu la même chose ici, sauf que nous avons un objet concret, existant dans la vraie vie, et non issu de l'imagination féconde de GELUCK.

JE tiens à préciser qu'il n'y a ici *strictement* aucune ironie dans mes propos. Les cartes hors-norme m'ont toujours fascinées, la dernière en date étant celle du réseau routier d'une morne plaine dans le COLORADO (voir la figure 14 page 30). Dès lors que l'on pose un quadrillage aux mailles suffisamment fines sur un polygone irrégulier (figure 13 page 29) il n'y a rien d'extraordinaire à ce que certaines d'icelles soient pauvres en contenu. La poésie vient, à mon sens, du contraste entre la richesse de la légende générique et du vide du contenu légendé. Pour rendre hommage à ce *bel objet*, nous allons tenter de le re-créeer sous .

2 Obtenir le fond de carte du Portugal continental

2.1 Les données administratives du Portugal

LES données administratives du PORTUGAL sont récupérées directement dans  sous la forme d'un objet de type `SpatVector` grâce au paquet `geodata` [2]. La fonction `plot()` du paquet `terra` [1] permet ensuite de représenter le fond de carte.

```
library(geodata)
country_codes("Portugal")[ , 1:4]
  NAME ISO3 ISO2 NAME_ISO
182 Portugal PRT PT PORTUGAL

prt <- gadm("PRT", path = "Portugal")
library(terra)
plot(prt, main = "Les données du Portugal")
```

¹<https://www.igeoe.pt>

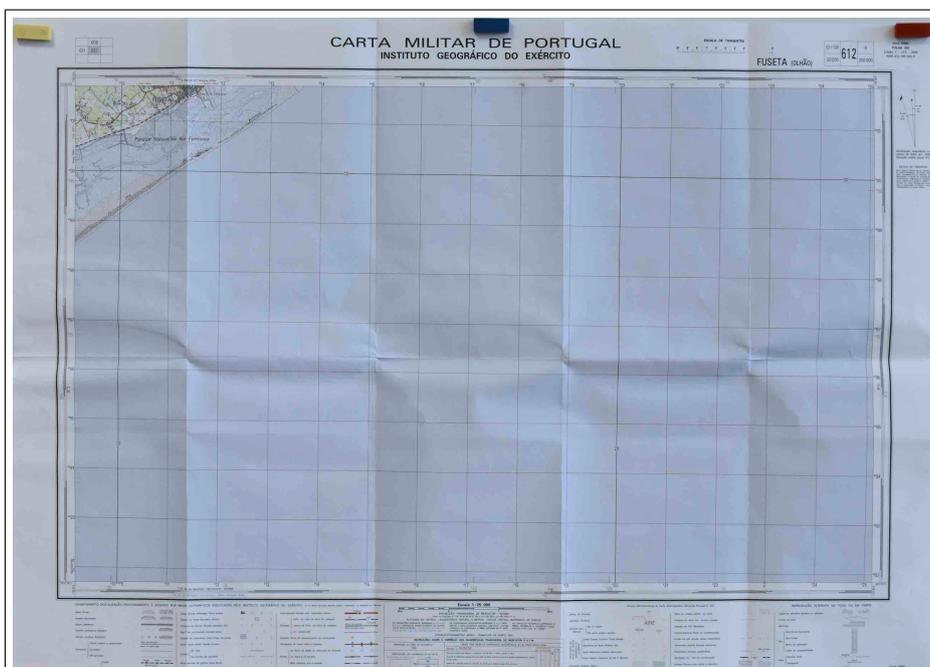


FIGURE 1 : Photographie à base résolution de la plus belle feuille de la carte d'état-major du PORTUGAL dont il est question ici. Il s'agit de la *folha* n° 612, dite FUSETA (OLHÃO), de la série M888 à l'échelle 1:25 000, soit 1 cm sur la carte pour 250 m sur le terrain. La carte *stricto sensu* fait 64 cm par 40 cm, ce qui correspond à 16 km par 10 km *in situ*. En haut à gauche (détail dans la figure 2 page 4) on trouve une partie du charmant village de FUSETA qui a la particularité de posséder pas moins de deux gares de chemin de fer sur la ligne de l'ALGARVE avec FUSETA-MONCARAPACHO et FUSETA-A. Vous pouvez ainsi devenir pour une somme modique l'heureux détenteur d'un objet que n'auraient pas renié les surréalistes. Le fait qu'il ne soit pas unique est même complètement dans leur esprit puisqu'ils étaient opposés à la notion même d'unicité des œuvres d'art, et ont beaucoup joué autour de ce thème. Je déconseille de l'encadrer pour l'exposer dans votre cabinet de curiosités : l'effet est plus saisissant quand vos amis déplieront eux-mêmes la carte.

Crédit photo : François DÉBIAS.

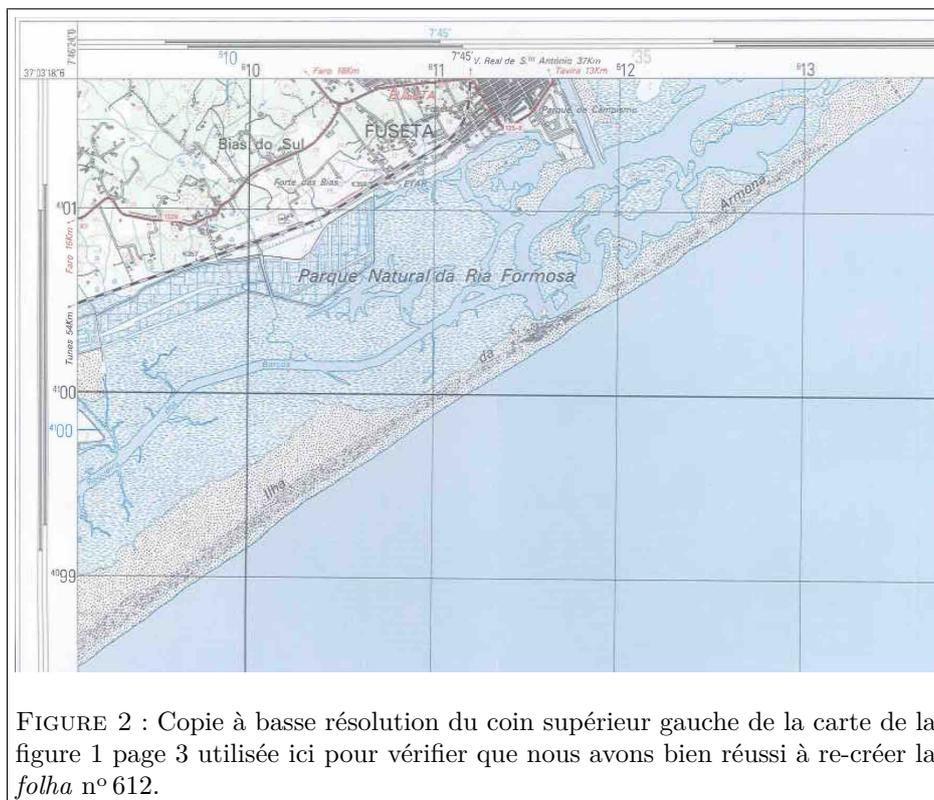
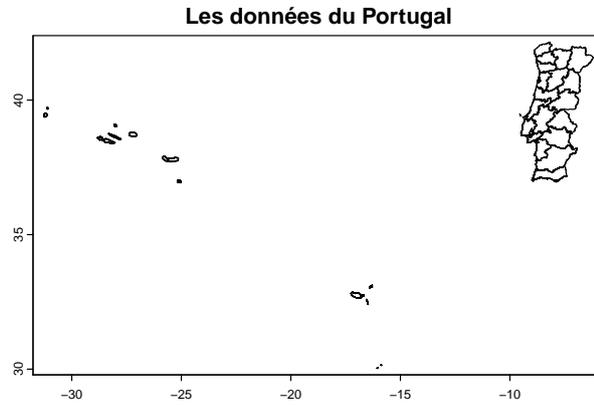


FIGURE 2 : Copie à basse résolution du coin supérieur gauche de la carte de la figure 1 page 3 utilisée ici pour vérifier que nous avons bien réussi à re-cr er la *folha* n  612.

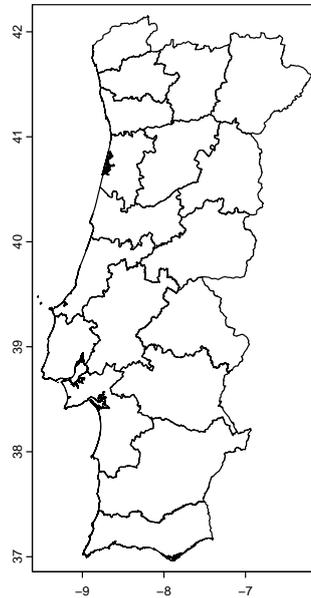


2.2 Sélection des données continentales

DANS les données du PORTUGAL nous avons les régions autonomes des AÇORES et de MADÈRE qui ne nous intéressent pas ici. On les retire donc du fond de carte.

```
prtc <- prt[prt$ENGTYP1 != "'Autonomous Region'", ]  
plot(prtc, main = "Les données du Portugal continental")
```

Les données du Portugal continental

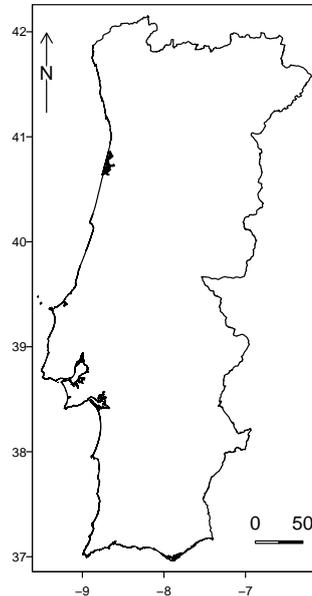


2.3 Fusion des régions

ON agrège les données pour ne plus avoir la représentation des limites entre les régions.

```
prtcf <- aggregate(prtc)
plot(prtcf, main = "Les données du Portugal continental", las = 1)
sbar(50, "bottomright", type = "bar")
north("topleft")
```

Les données du Portugal continental



2.4 Fond de carte à basse résolution

ON construit un fond de carte à basse résolution en demandant à ce qu'il ait au moins une seconde d'arc (un soixantième de mille marin, soit environ 30 mètres) entre deux points.

```
fdc <- simplifyGeom(prtcf, tolerance = 1/60)
```

DANS l'objet `prtcf` le contour du PORTUGAL est défini avec 133447 points, alors que dans l'objet `fdc` il n'en reste plus que 2132. On utilisera donc ce dernier avec avantage quand on aura besoin de faire figurer les frontières du PORTUGAL qu'à titre illustratif, par exemple dans le rendu final du *Diagrama de localização* (section 4 page 12), mais certainement pas pour des opérations de manipulation fine des objets spatiaux.

3 Reproduction de la *folha* n° 612

3.1 Recherche de l'EPSG

LA projection de la surface d'un patatoïde sur un plan induisant nécessairement des distorsions, il y a énormément de systèmes de projections qui ont été utilisés en fonction des objectifs poursuivis. Ils peuvent être définis dans `terra` avec une chaîne de caractères répondant à une syntaxe précise². Heureusement pour nous, il existe une base de données dite EPSG (acronyme de *European Petroleum Survey Group*) qui donne une clef d'identification aux projections les plus courantes (il y en a quand même plusieurs milliers de définies).

²Type : `"+proj=lcc +lat_0=38.9072 +lon_0=-77.0369 +lat_1=33 +lat_2=45 +ellps=GRS80"`

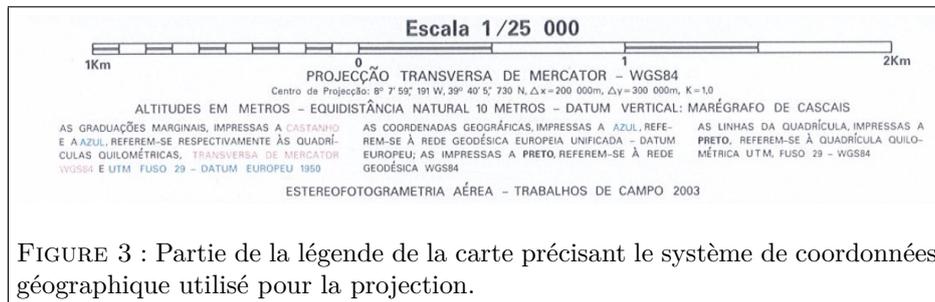


FIGURE 3 : Partie de la légende de la carte précisant le système de coordonnées géographique utilisé pour la projection.

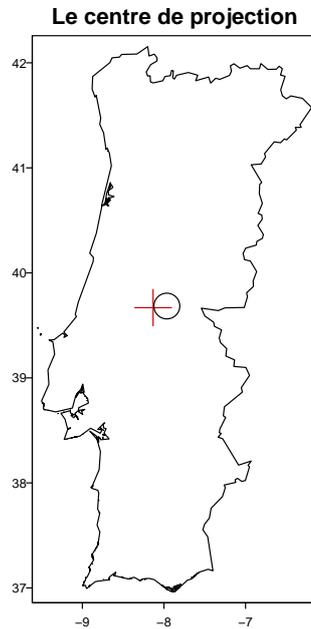
```
PROJCS["ETRS89 / Portugal TM06",
  GEOGCS["ETRS89",
    DATUM["European_Terrestrial_Reference_System_1989",
      SPHEROID["GRS 1980",6378137,298.257222101,
        AUTHORITY["EPSG","7019"]],
      TOWGS84[0,0,0,0,0,0],
      AUTHORITY["EPSG","6258"]],
    PRIMEM["Greenwich",0,
      AUTHORITY["EPSG","8901"]],
    UNIT["degree",0.0174532925199433,
      AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4258"]],
  PROJECTION["Transverse_Mercator"],
  PARAMETER["latitude_of_origin",39.6682583333333],
  PARAMETER["central_meridian",-8.13310833333333],
  PARAMETER["scale_factor",1],
  PARAMETER["false_easting",0],
  PARAMETER["false_northing",0],
  UNIT["metre",1,
    AUTHORITY["EPSG","9001"]],
  AXIS["Easting",EAST],
  AXIS["Northing",NORTH],
  AUTHORITY["EPSG","3763"]]
```

FIGURE 4 : Copie d'écran des informations disponibles pour EPSG:3763 sur le site <https://epsg.io/3763>

Il suffit d'utiliser par exemple la notation EPSG:4326 pour faire référence au système de projection utilisé par les GPS.

DANS la légende de la carte (figure 3 page 8) le centre de projection est donné comme étant 8°7'59",191 W de longitude et 39°40'5",730 N de latitude. Convertissons en degrés décimaux :

```
xlon <- -1*(8 + 7/60 + 59.191/3600)
dput(xlon)
-8.13310861111111
ylat <- 39 + 40/60 + 5.730/3600
dput(ylat)
39.6682583333333
cdp <- vect(cbind(xlon, ylat), crs = "EPSG:4326")
plot(fdc, main = "Le centre de projection", las = 1)
points(cdp, cex = 3, pch = 3, col = "red3")
points(centroids(fdc), cex = 3, pch = 1)
```



LE centre de projection (croix rouge) se trouve au centre du PORTUGAL, assez proche du centre de gravité (cercle noir). Ceci permet de minimiser les distorsions sur la carte. En entrant dans un moteur de recherche « 39.6682583333333 » j'ai trouvé que l'EPSG qui nous intéresse est vraisemblablement le 3763 (figure 4 page 8). Si tel est bien le cas, les coordonnées du centre de projection devraient être nulles ou presque, vérifions :

```
test <- project(cdp, "EPSG:3763")
geom(test)[1, 3:4]
-2.383491e-02  3.540577e-09
```

C'EST parfait pour la latitude, mais pas tout à fait pour la longitude. J'ai trouvé qu'en enlevant un millième de seconde d'arc on avait une valeur plus proche de zéro :

```
xlon2 <- -1*(8 + 7/60 + 59.190/3600)
dput(xlon2)
-8.13310833333333
cdp2 <- vect(cbind(xlon2, ylat), crs = "EPSG:4326")
test2 <- project(cdp2, "EPSG:3763")
geom(test2)[1, 3:4]
-4.093644e-10  3.540577e-09
```

CETTE fois la longitude est quasi-nulle et on retrouve la valeur donnée sur le site de l'EPSG (figure 4 page 8). La distance entre ces deux points n'est que de l'ordre de 2.4 cm :

```
distance(cdp, cdp2, unit = "m")[1, 1]
[1] 0.02383491
```

JE ne sais pas si cet écart est très pertinent et mérite que l'on s'y attarde plus avant. Dans la suite je vais considérer que je suis bien en EPSG:3763.

3.2 Les coins de la *folha* n° 612

LES coordonnées GPS des quatre coins de la carte sont indiquées en marge de celle-ci, voir la figure 2 page 4 par exemple pour le coin supérieur gauche. Convertissons ces coordonnées en degrés décimaux et projetons les sur la carte :

```
yt1 <- 37 + 3/60 + 18.6/3600
xt1 <- -1*(7 + 46/60 + 24/3600)
t1 <- vect(cbind(xt1, yt1), crs = "EPSG:4326")
tlb <- project(t1, "EPSG:3763")
geom(tlb)[1, 3:4]

      x      y
32001.08 -290001.46
```

LES coordonnées cartésiennes du coin supérieur gauche sont donc très proches de $x = 32\ 000$ m et $y = -290\ 000$ m. Ce sont ces coordonnées arrondies que je vais utiliser dans la suite. Pour le coin supérieur droit j'ai :

```
ytr <- 37 + 3/60 + 16.2/3600
xtr <- -1*(7 + 35/60 + 36.5/3600)
tr <- vect(cbind(xtr, ytr), crs = "EPSG:4326")
trb <- project(tr, "EPSG:3763")
geom(trb)[1, 3:4]

      x      y
47999.8 -289999.8
distance(tlb, trb, unit = "km")[1, 1]
[1] 15.99872
```

LES coordonnées cartésiennes du coin supérieur droit sont donc très proches de $x = 48\ 000$ m et $y = -290\ 000$ m. L'ordonnée est la même que pour le coin supérieur gauche, comme attendu. La distance entre les deux coins supérieurs est proche de 16 km, ce qui est cohérent avec les dimensions de la carte (figure 1 page 3). Pour le coin inférieur droit nous avons :

```
ybr <- 36 + 57/60 + 51.8/3600
xbr <- -1*(7 + 35/60 + 38.8/3600)
br <- vect(cbind(xbr, ybr), crs = "EPSG:4326")
brb <- project(br, "EPSG:3763")
terra::geom(brb)[1, 3:4]

      x      y
47999.61 -300000.56
distance(trb, brb, unit = "km")[1, 1]
[1] 10.00078
```

LES coordonnées cartésiennes du coin inférieur droit sont donc très proches de $x = 48\ 000$ m et $y = -300\ 000$ m. L'abscisse est la même que pour le coin supérieur droit, comme attendu. La distance entre les deux coins à droite est proche de 10 km, ce qui est cohérent avec les dimensions de la carte (figure 1 page 3). Enfin, pour le coin inférieur gauche nous avons :

```
ybl <- 36 + 57/60 + 54.3/3600
xbl <- -1*(7 + 46/60 + 25.6/3600)
bl <- vect(cbind(xbl, ybl), crs = "EPSG:4326")
blb <- project(bl, "EPSG:3763")
geom(blb)[1, 3:4]

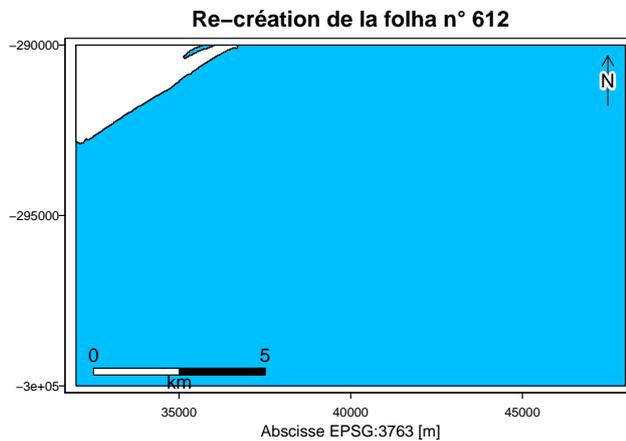
      x      y
31999.29 -299998.92
distance(blb, brb, unit = "km")[1, 1]
[1] 16.00032
distance(blb, tlb, unit = "km")[1, 1]
[1] 9.997457
```

Tout semble cohérent ici également.

3.3 Re-création de la *folha* n° 612

ON utilise les valeurs arrondies en EPSG:3763 pour définir le polygone (ici un rectangle) correspondant à la *folha* n° 612, puis on fait l'intersection avec les données du PORTUGAL à haute résolution pour re-créer la carte.

```
x1 <- 32000 ; xr <- 48000 ; yt <- -290000 ; yb <- -300000
f612 <- vect(cbind(c(x1, xr, xr, x1), c(yt, yt, yb, yb)),
            type = "polygon", crs = "EPSG:3763")
PRT <- terra::project(prtcf, "EPSG:3763")
#writeVector(PRT, "Portugal/PRT", overwrite = TRUE)
plot(f612, col = "deepskyblue", main = "Re-création de la folha n° 612", las = 1,
     xlab = "Abscisse EPSG:3763 [m]")
polys(intersect(PRT, f612), col = "white")
north()
sbar(5000, "bottomleft", type = "bar", scaleby = 1000, below = "km", xpd = NA)
```



LE résultat semble cohérent, au moins à cette échelle, avec la carte de la figure 1 page 3. Pour vérifier que nous sommes bien correctement positionnés nous allons utiliser deux stratégies : zoomer avec une carte interactive (section 3.4 page 11) et essayer de reproduire la grille de la figure 13 page 29 pour vérifier que la *folha* n° 1 tout au nord est correcte (section 4 page 12).

```
jpeg("Portugal.jpg", 480/4, 480/6)
plot(f612, col = "deepskyblue", las = 1, axes = FALSE, mar = c(0, 0, 0, 0))
polys(intersect(PRT, f612), col = "white")
dev.off()
```

3.4 Vérification avec une carte interactive

LE code suivant permet de produire une carte interactive dont l'état initial est donné dans la figure 5 page 12. En zoomant au maximum sur la ligne de chemin de fer au nord (figure 6 page 13) et à l'ouest (figure 7 page 14) on

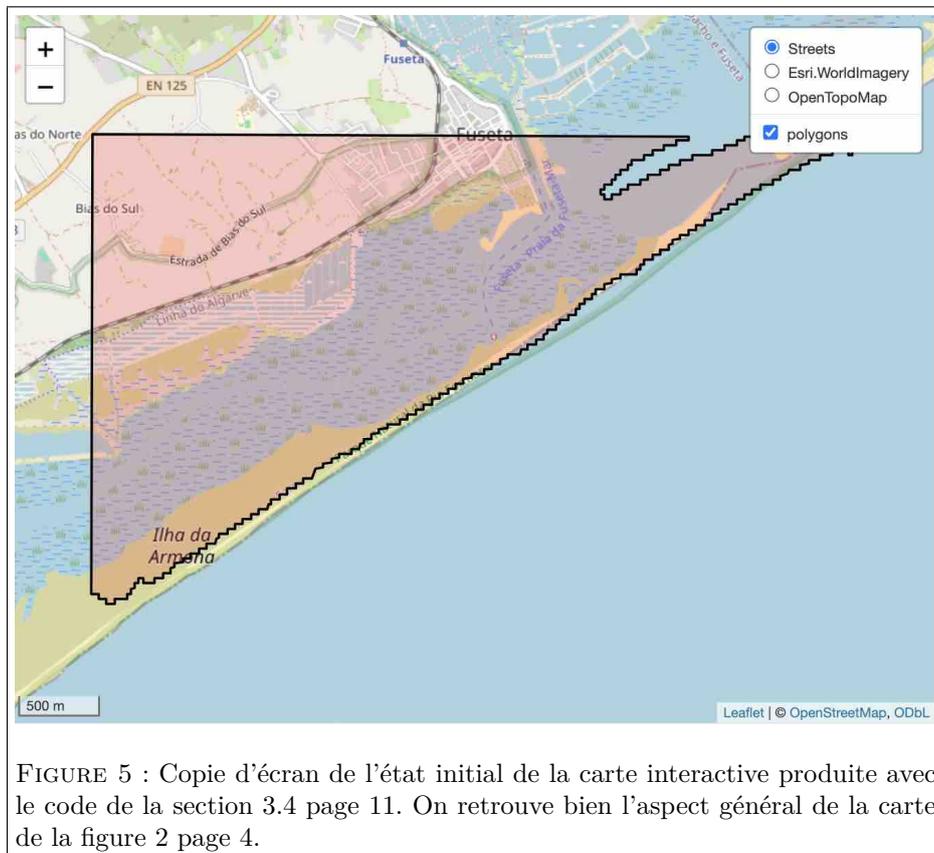


FIGURE 5 : Copie d'écran de l'état initial de la carte interactive produite avec le code de la section 3.4 page 11. On retrouve bien l'aspect général de la carte de la figure 2 page 4.

peut constater que notre reproduction est conforme à la carte originelle avec une précision, au doigt mouillé, de l'ordre du mètre.

```
library(leaflet)
mamap <- plet(terra::intersect(PRT, f612))
mamap <- addScaleBar(mamap, "bottomleft", scaleBarOptions(imperial = FALSE))
mamap
```

4 Reproduction du *Diagrama de localização*

4.1 Génération du maillage par translation

ON translate avec la fonction `shift()` autant de fois que nécessaire le rectangle de la *folha* n° 612 de façon à générer une grille qui couvre l'ensemble du territoire du PORTUGAL continental. Comme la *folha* fait 16×10 km, on fait des pas élémentaires horizontaux de 16 km et verticaux de 10 km. J'ai déterminé empiriquement les bornes des variables `x` et `y` ci-dessous, par essai-erreur jusqu'à couvrir l'ensemble du PORTUGAL. Pour les longitudes nous avons besoin de 19 grilles, soit 304 km d'est en ouest, et pour les latitudes de 58 grilles, soit 580 km



FIGURE 6 : Copie d'écran d'un agrandissement au nord de la figure 5 page 12. Le bord supérieur de la carte passe au milieu du pont de l'avenue MARIA BARROSO SOARES au dessus de la ligne de chemin de fer de l'ALGARVE, comme dans la figure 2 page 4. On a donc une précision de l'ordre du mètre pour la latitude.



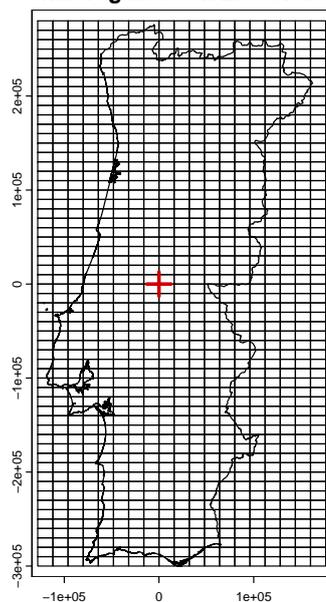
FIGURE 7 : Copie d'écran d'un agrandissement à l'ouest de la figure 5 page 12. Le trait pointillé en bleu est la piste cyclable de l'ECOVIA, une portion de l'EUROVELO 1 longue de 11 150 km entre le CAP NORD en NORVÈGE et CAMINHA au PORTUGAL. On reconnaît bien le dessin caractéristique des salines en bleu hachuré sur le bord gauche de la carte comme dans la figure 2 page 4, mais il vaut mieux avoir la carte originelle pour s'en convaincre. On a donc une précision de l'ordre du mètre pour la longitude.

du nord au sud³. Le couple (x, y) est une clef d'identification de chaque maille, nous l'utiliserons dans la section 4.3 page 17 pour des opérations de jointure, au sens de l'algèbre relationnelle, entre objets spatiaux.

```
magrid <- f612 ; magrid$x <- magrid$y <- NA
magrid[1, "x"] <- magrid[1, "y"] <- 0
for(x in -10:8){
  for(y in 0:57){
    if(x == 0 && y == 0) next # déjà fait
    magrid <- rbind(shift(f612, dx = 16000*x, dy = 10000*y), magrid)
    magrid[1, "x"] <- x
    magrid[1, "y"] <- y
  }
}
writeVector(magrid, file = "Portugal/magrid")

magrid <- vect("Portugal/magrid")
plot(PRT, ext = ext(magrid), main = "Grille régulière en EPSG:3763")
polys(magrid)
points(project(cdp, "EPSG:3763"), cex = 2, pch = 3, col = "red3", lwd = 3)
```

Grille régulière en EPSG:3763



Le centre de projection indiqué par la croix rouge est à l'intersection de quatre mailles. Il y a plein de rectangles qui ne servent à rien car complètement hors zone. Nous allons voir comment pallier ce problème de différentes façons.

4.2 Intersection d'objets spatiaux

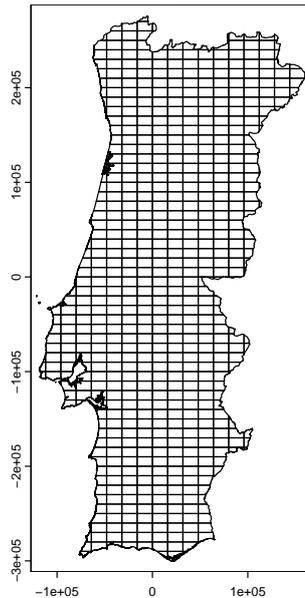
Une première approche est de calculer l'intersection entre les rectangles de la grille et le PORTUGAL avec la fonction `intersect()` du paquet `terra`. Les rectangles hors zone auront une intersection nulle et seront donc éliminés.

³Si on souhaitait assembler toutes les feuilles de la série M888 il faudrait donc disposer d'une surface de $19 \times 64 \text{ cm} = 12.16 \text{ m}$ par $58 \times 40 \text{ cm} = 23.2 \text{ m}$.

```
inter <- intersect(PRT, magrid)
writeVector(inter, "Portugal/inter")
```

```
inter <- vect("Portugal/inter")
plot(inter, main = "Intersection de la grille et du Portugal")
```

Intersection de la grille et du Portugal

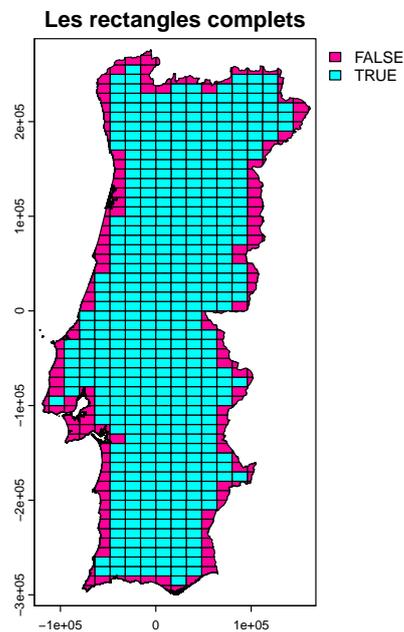


ÉVIDEMMENT, si on compare avec la figure 13 page 29, ce n'est pas *exactement* ce que l'on veut pour reproduire le « *Diagrama de localização* ». Mais l'opération présente quand même un intérêt puisqu'elle va nous permettre de calculer la surface « utile » de chaque rectangle : les rectangles au centre du PORTUGAL seront complets avec une surface utile de 160 km^2 tandis que ceux en périphérie se verront amputer d'une partie de leur superficie. La fonction `expanse()` nous permet de calculer ces surfaces.

```
inter$surf <- expanse(inter, unit = "km")
head(inter$surf)
[1] 125.33856 51.64511 159.86733 159.98759 159.98759 159.99364
tail(sort(table(round(inter$surf))))
  0  2 147 156  1 160
  6  6  6  6  8 455
```

À cause des imprécisions numériques les rectangles complets de font pas exactement 160 km^2 , on arrondissant à l'unité on voit que nous avons 455 rectangles complets sur 654, soit 70 %. La représentation ci-après permet de vérifier que les rectangles incomplets sont bien en périphérie.

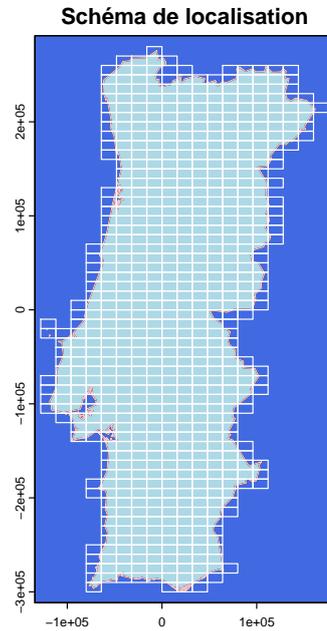
```
inter$complet <- ifelse(round(inter$surf) >= 160, TRUE, FALSE)
plot(inter, "complet", main = "Les rectangles complets")
```



4.3 Relations entre objets spatiaux

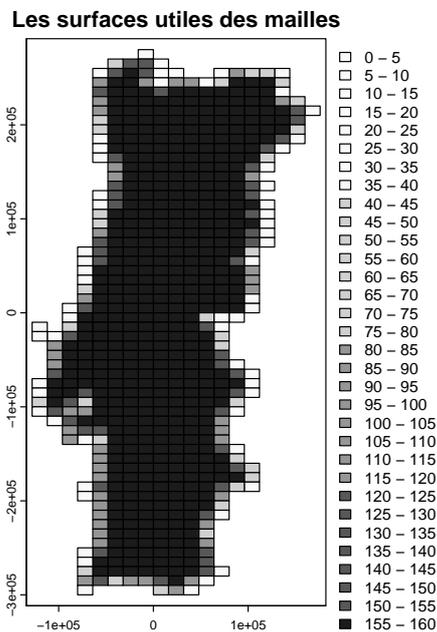
ON aimerait conserver tous les rectangles qui contiennent au moins un petit bout de PORTUGAL. La fonction `is.related()` du paquet `terra` va nous permettre de construire un vecteur de booléens indicateur des rectangles que l'on souhaite conserver. Cette fonction est très puissante car riche de nombreuses relations envisageables entre des objets spatiaux (*viz. intersects, touches, crosses, overlaps, within, contains, covers, coveredby, disjoint*). Nous allons utiliser ici la relation `intersects` pour conserver les rectangles d'intersection non nulle.

```
keepme <- is.related(magrid, PRT, relation = "intersects")
mg2 <- magrid[keepme, ]
PRTlow <- project(fdc, "EPSG:3763")
plot(PRTlow, ext = ext(mg2), main = "Schéma de localisation",
      background = "royalblue", col = "lightblue")
polys(PRTlow, border = "pink")
polys(mg2, border = "white")
```



CETTE fois nous avons quelque chose de bien plus ressemblant (mais pas parfaitement identique) au « *Diagrama de localização* » de la figure 13 page 29. Comme les couples (x, y) sont des clefs d'identification des mailles, nous pouvons récupérer les surfaces utiles par une opération de jointure avec l'objet spatial `inter` calculé dans la section 4.2 page 15.

```
mg3 <- merge(mg2, inter, by.x = c("x", "y"), by.y = c("x", "y"))
nseq <- 32
plot(mg3, "surf", col = rev(hcl.colors(5, "Grays")),
     main = "Les surfaces utiles des mailles",
     breaks = seq(0, 160, le = nseq + 1))
```



LES mailles les plus claires, avec les surfaces utiles les plus faibles, se retrouvent fort logiquement en périphérie.

4.4 Vérification avec une carte interactive

J'AI utilisé le code ci-après pour produire une carte interactive et aller zoomer tout au nord du pays sur la *folha* n° 1 (figure 8 page 20). Je n'ai malheureusement pas cette carte entre les mains pour pouvoir faire des vérifications fines. Tout ce que l'on peut dire c'est que c'est cohérent avec la carte de présentation générale de l'« *Instituto Geográfico do Exército* » avec la ligne de chemin de fer qui plonge pile dans le coin inférieur gauche de la carte (figure 9 page 21). En zoomant plus on voit qu'il y a un décalage d'environ 10 m en longitude (figure 10 page 21), il faudra que j'attende d'avoir la *folha* n° 1 sous les yeux pour vérifier que c'est cohérent avec une précision de l'ordre du mètre comme dans la section 3.4 page 11.

```
tmp <- rbind(PRT, mg3)
mamap <- plet(tmp, col = "transparent")
library(leaflet)
mamap <- addScaleBar(mamap, "bottomleft", scaleBarOptions(imperial = FALSE))
mamap
```

4.5 Nomenclature des cartes

LES cartes de la série M888 sont numérotées régulièrement en suivant le sens de lecture mais avec quelques exceptions qui portent une ou deux lettres en suffixe pour les cartes de faible surface utile (figure 13 page 29). Je me suis longtemps demandé quel pouvait bien être l'intérêt d'un tel schéma de numérotation. Mon interprétation actuelle est que c'est pour assurer la stabilité

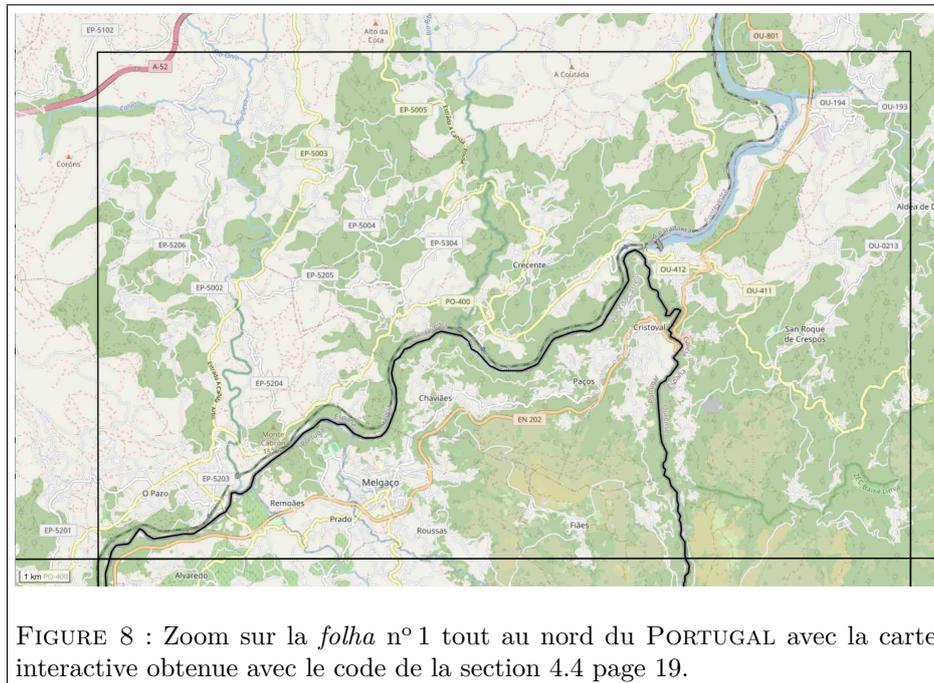


FIGURE 8 : Zoom sur la *folha* n° 1 tout au nord du PORTUGAL avec la carte interactive obtenue avec le code de la section 4.4 page 19.

à long terme de la nomenclature des carte avec une grande surface utile. Si le trait littoral vient à changer, ou si on décide de modifier légèrement le centre de projection, des cartes à faible surface utile peuvent apparaître ou disparaître. En les sortant de la série générale on assure ainsi la stabilité de la nomenclature des cartes les plus utiles. J'ai essayé de reproduire cette nomenclature en sortant les cartes de moins de 16 km² de surface utile de la série générale :

```

mg3$ID <- paste(mg3$x, mg3$y, sep = "-")
mg3$name <- NA ; mg3$regular <- FALSE
ctrname <- 1 ; ctr <- 1
previous <- logical(0)
for(y in max(mg3$y):min(mg3$y)){
  for(x in min(mg3$x):max(mg3$x)){
    my_ID <- paste(x, y, sep = "-")
    ii <- which(mg3$ID == my_ID)
    if(length(ii) == 0) next # n'existe pas
    if(mg3$surf[ii] < 16){ # pas assez de surface utile
      lastregular <- max((1:length(previous))[previous])
      shift <- ctr - lastregular
      mg3[ii, "regular"] <- FALSE
      mg3[ii, "name"] <- paste0(ctrname - 1, LETTERS[shift])
      previous <- c(previous, FALSE)
    } else {
      mg3[ii, "name"] <- as.character(ctrname)
      mg3[ii, "regular"] <- TRUE
      previous <- c(previous, TRUE)
      ctrname <- ctrname + 1
    }
    ctr <- ctr + 1
  }
}
writeVector(mg3, "Portugal/mg3")

mg3 <- vect("Portugal/mg3")
plot(mg3)

```

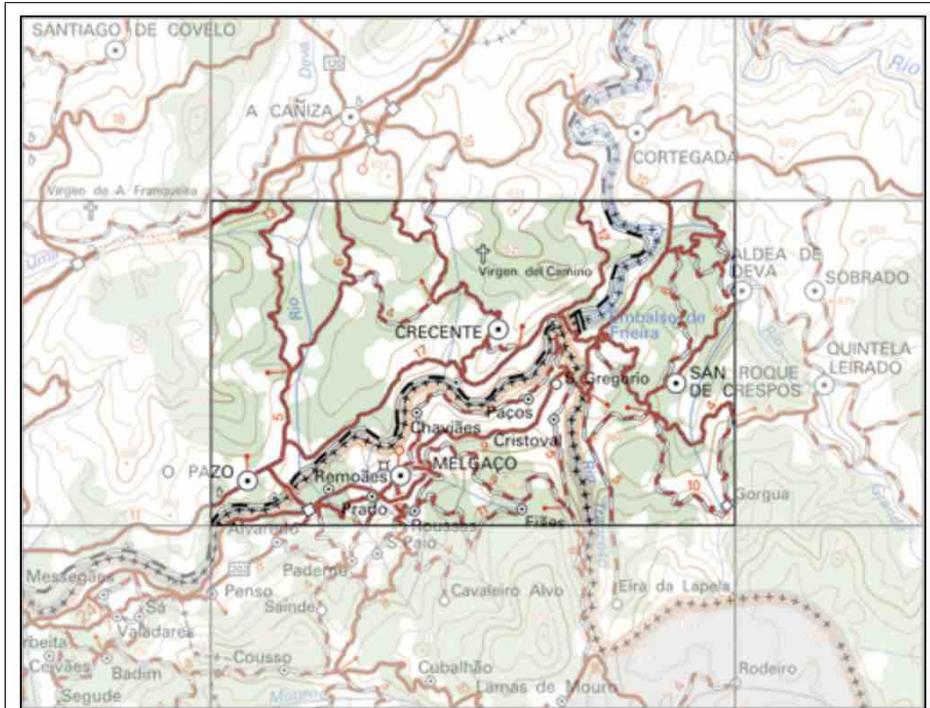


FIGURE 9 : Copie d'écran de la présentation générale de la *folha* n° 1 tout au nord du PORTUGAL que l'on peut trouver sur le site de l'« *Instituto Geográfico do Exército* » (<https://www.igeoe.pt>).

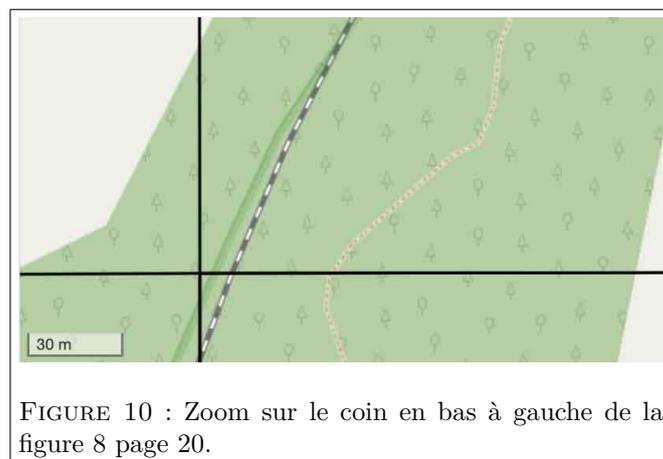
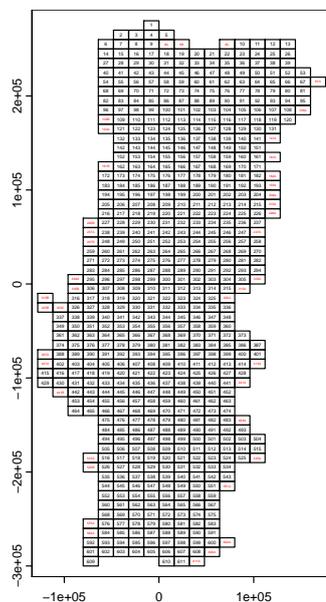


FIGURE 10 : Zoom sur le coin en bas à gauche de la figure 8 page 20.

```
mycex <- ifelse(mg3$regular, 0.25, 0.2)
mycol <- ifelse(mg3$regular, "black", "red")
text(mg3, mg3$name, cex = mycex, col = mycol)
```



Le résultat ressemble, mais n'est pas parfaitement identique, à celui de la figure 13 page 29. Par exemple dans le cas de la carte n° 600, on s'attendrait à avoir une carte 600A à droite et une carte 608A en dessous. Mais ce n'est pas le cas parce que la carte n° 600 débord légèrement du cadre habituel à droite (figure 11 page 23) pour couvrir ce qui serait la carte 600A et en bas (figure 12 page 23) pour couvrir ce qui serait la carte 608A. Les cartes 600A et 608A ne sont donc plus nécessaires, et de fait n'existent pas (figure 13 page 29). Ces ajustements manuels, au cas par cas, font qu'il n'est pas possible de reproduire automatiquement la nomenclature utilisée.

Mais au fait, si on suit la nomenclature, notre fameuse *folha* n° 612 ne devrait-elle pas s'appeler la *folha* n° 611A ? En bien non, comme c'est la dernière de la série, même si elle venait à disparaître, elle ne perturberait pas la numérotation de la série principale. C'est une exception à l'exception, qui n'est pas sans lui ajouter encore un peu plus de charme.

4.6 Optimisation du découpage

Nous avons ici un problème d'optimisation intéressant : comment choisir le centre de projection⁴ pour minimiser le nombre de cartes avec une faible surface utile ? Ce problème n'est pas parfaitement défini puisque, comme nous l'avons vu à la section précédente, il y a des ajustements manuels difficilement

⁴C'est un raccourci de langage : le centre de projection géographique est inchangé, on se contente ici de traduire le point d'origine des cartes. Pour être tout à fait rigoureux il faudrait changer le centre de projection, c'est possible, mais cela dépasse les compétences d'un simple utilisateur d'EPSG comme moi.

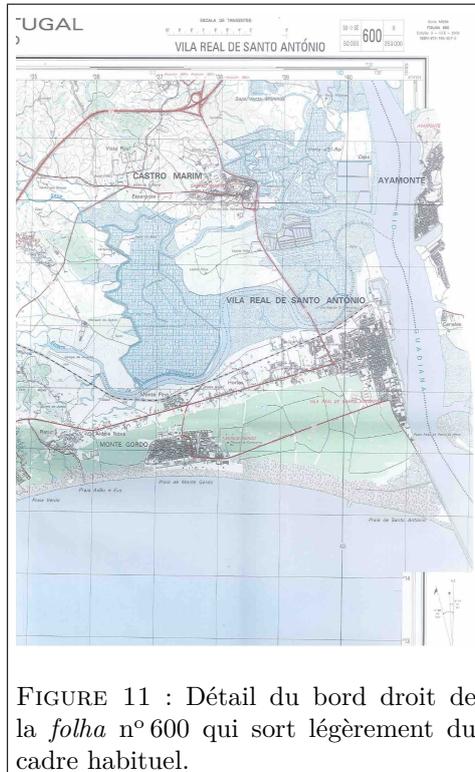


FIGURE 11 : Détail du bord droit de la *folha* n° 600 qui sort légèrement du cadre habituel.

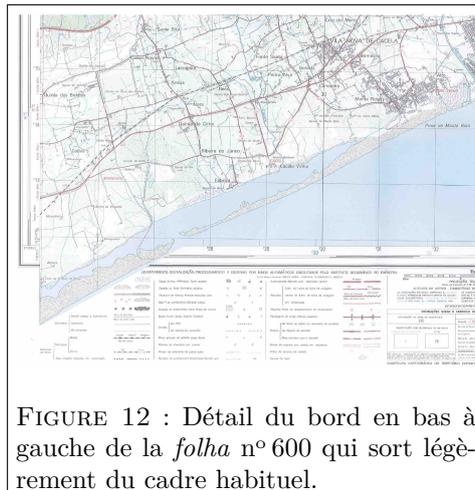
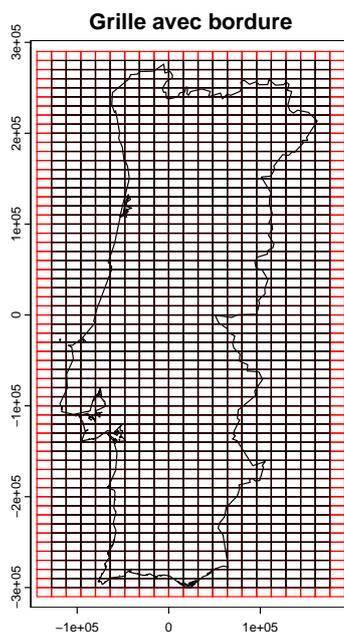


FIGURE 12 : Détail du bord en bas à gauche de la *folha* n° 600 qui sort légèrement du cadre habituel.

formalisables qui sont opérés. Mais on peut quand même par curiosité essayer de trouver une solution qui pourrait fournir un bon point de départ aux ajustements faits à la marge. Ma première tentative en ce sens fut un échec complet, j'avais des résultats incohérents. Avec un pavage périodique du plan on s'attend à avoir les mêmes résultats avec par exemple un décalage de ($dx = -8$ km, $dy = -5$ km) et ($dx = +8$ km, $dy = +5$ km), or tel n'était pas le cas. J'ai fini par comprendre qu'il fallait *absolument* rajouter une bordure à ma grille pour être certain de ne rater aucune carte.

```
magrid2 <- f612 ; magrid2$x <- magrid2$y <- NA
magrid2[1, "x"] <- magrid2[1, "y"] <- 0
for(x in -11:9){
  for(y in -1:58){
    if(x == 0 && y == 0) next
    magrid2 <- rbind(shift(f612, dx = 16000*x, dy = 10000*y), magrid2)
    magrid2[1, "x"] <- x
    magrid2[1, "y"] <- y
  }
}
writeVector(magrid2, file = "Portugal/magrid2")
```

```
magrid2 <- vect("Portugal/magrid2")
plot(PRTlow, ext = ext(magrid2), main = "Grille avec bordure")
polys(magrid2, border = "red")
polys(magrid)
```



```
library(terra)
magrid2 <- vect("Portugal/magrid2")
PRT <- vect("Portugal/PRT")
xseq_km <- seq(-8, 8, by = 0.1) ; yseq_km <- seq(-5, 5, by = 0.1)
nx <- length(xseq_km) ; ny <- length(yseq_km)
cnames <- c("x_shift_km", "y_shift_km", "n_small_poly")
tabopt <- as.data.frame(matrix(NA, nrow = nx*ny, ncol = length(cnames)))
colnames(tabopt) <- cnames
threshold <- 16 # Petit si moins de 16 km2
```

```
irow <- 1
for(i in seq_len(nx)){
  for(j in seq_len(ny)){
    tabopt[irow, "x_shift_km"] <- xseq_km[i]
    tabopt[irow, "y_shift_km"] <- yseq_km[j]
    shifted_grid <- shift(magrid2, dx = 1000*xseq_km[i], dy = 1000*yseq_km[j])
    inter <- intersect(PRT, shifted_grid)
    inter$surf <- expanse(inter, unit = "km")
    tabopt[irow, "n_small_poly"] <- sum(inter$surf < threshold)
    irow <- irow + 1
    save(tabopt, file = "Portugal/tabopt.Rda")
  }
}
```

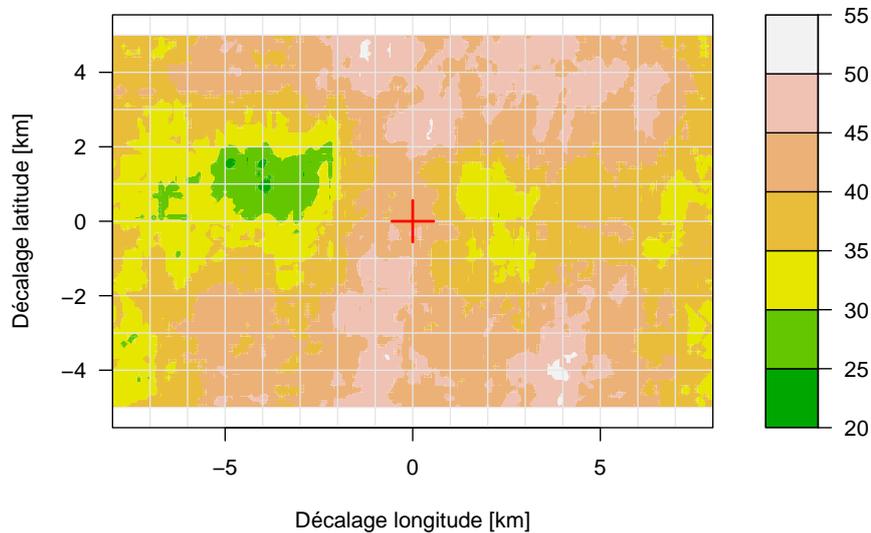
ON commence par un petit *sanity check* pour vérifier la périodicité des résultats, on a en fait ici la topologie d'un tore.

```
load("Portugal/tabopt.Rda")
all(subset(tabopt, x_shift_km == -8)$n_small_poly ==
     subset(tabopt, x_shift_km == +8)$n_small_poly)
[1] TRUE
all(subset(tabopt, y_shift_km == -5)$n_small_poly ==
     subset(tabopt, y_shift_km == +5)$n_small_poly)
[1] TRUE
```

TOUT va bien. On peut proposer la représentation graphique suivante, qui est une sorte de méta-carte puisque l'on cartographie des données de cartographie!

```
mkxyzlist <- function(df, xname = "x", yname = "y", zname = "z"){
  x <- unique(sort(df[, xname])) ; nx <- length(x)
  y <- unique(sort(df[, yname])) ; ny <- length(y)
  z <- matrix(NA, nrow = nx, ncol = ny)
  for(i in seq_len(nrow(df))){
    ii <- which(x == df[i, xname])
    jj <- which(y == df[i, yname])
    z[ii, jj] <- df[i, zname]
  }
  return(list(x = x, y = y, z = z))
}
xyzlist <- mkxyzlist(tabopt, "x_shift_km", "y_shift_km", "n_small_poly")
filled.contour(xyzlist, asp = 1, color.palette = terrain.colors, nlevels = 7,
               xlim = c(-8, 8), ylim = c(-5, 5),
               plot.title = title(main = "Optimisation du centre de projection",
                                   xlab = "Décalage longitude [km]",
                                   ylab = "Décalage latitude [km]"),
               plot.axes = { axis(1); axis(2)
                             mycol <- grey(0.9)
                             abline(v = -8:8, col = mycol)
                             abline(h = -5:5, col = mycol)
                             points(0, 0, pch = 3, cex = 3, col = "red", lwd = 2)})
```

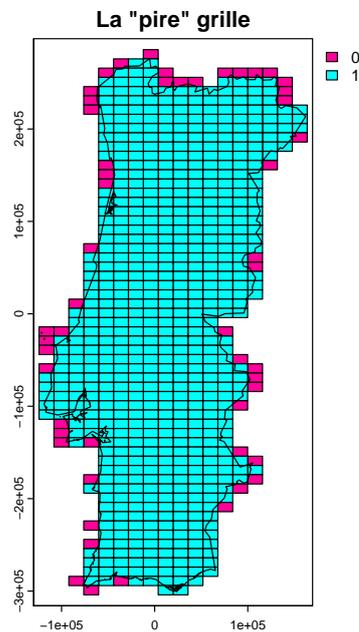
Optimisation du centre de projection



Le choix du point d'origine a donc un impact puisque l'on se promène entre 23 et 52 cartes avec une faible zone utile. On voit qu'il y a plusieurs minimums (et maximums) locaux, un algorithme exploitant le gradient pourrait se faire piéger ici. Illustrons les cas extrêmes. Le « pire » cas avec 52 cartes est trouvé pour un décalage $dx = +3.8$ km et $dy = -4.1$ km.

```
badgrid <- shift(magrid2, dx = +3800, dy = -4100)
badinter <- intersect(PRT, badgrid)
badinter$surf <- expanse(badinter, unit = "km")
keepme <- is.related(badgrid, PRT, relation = "intersects")
badgrid <- badgrid[keepme, ]
badgrid <- merge(badgrid, badinter, by.x = c("x", "y"), by.y = c("x", "y"))
badgrid$big <- ifelse(badgrid$surf <= 16, FALSE, TRUE)
writeVector(badgrid, file = "Portugal/badgrid")
```

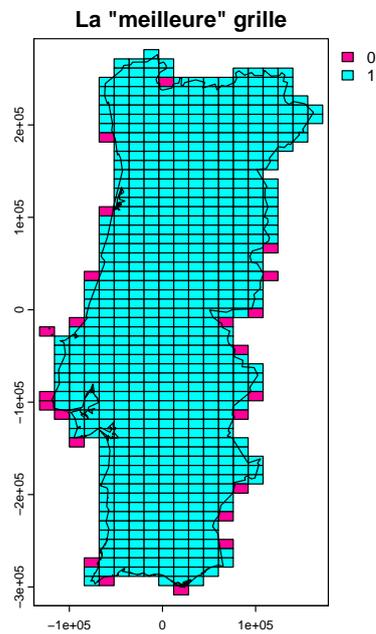
```
badgrid <- vect("Portugal/badgrid")
plot(badgrid, "big", main = 'La "pire" grille')
polys(PRTflow)
```



Il y a effectivement beaucoup de feuilles de faible surface utile, surtout au nord du pays, ce qui serait difficile à rattraper manuellement parce qu'il faudrait alors mordre en haut sur le titre général des feuilles. Le « meilleur » des cas avec 23 feuilles est trouvé pour un décalage $dx = -4.0$ et $dy = +1.5$.

```
goodgrid <- shift(magrid2, dx = -4000, dy = +1500)
goodinter <- intersect(PRT, goodgrid)
goodinter$surf <- expanse(goodinter, unit = "km")
keepme <- is.related(goodgrid, PRT, relation = "intersects")
goodgrid <- goodgrid[keepme, ]
goodgrid <- merge(goodgrid, goodinter, by.x = c("x", "y"), by.y = c("x", "y"))
goodgrid$big <- ifelse(goodgrid$surf <= 16, FALSE, TRUE)
writeVector(goodgrid, file = "Portugal/goodgrid")

goodgrid <- vect("Portugal/goodgrid")
plot(goodgrid, "big", main = 'La "meilleure" grille')
polys(PRTflow)
```



Il y a beaucoup moins feuilles de faible surface utile, mais il n'est pas certain que ce soit un très bon point de départ pour une optimisation manuelle parce qu'il reste encore beaucoup de feuilles qui ne dépassent que de peu notre seuil arbitraire de 16 km^2 . L'intérêt ici est surtout didactique pour montrer l'intérêt qu'il y a à faire de la cartographie sous **R** : nous bénéficions d'un environnement de programmation complet et ne sommes pas limités aux opérations pré-établies dans un logiciel basé sur une interface graphique.

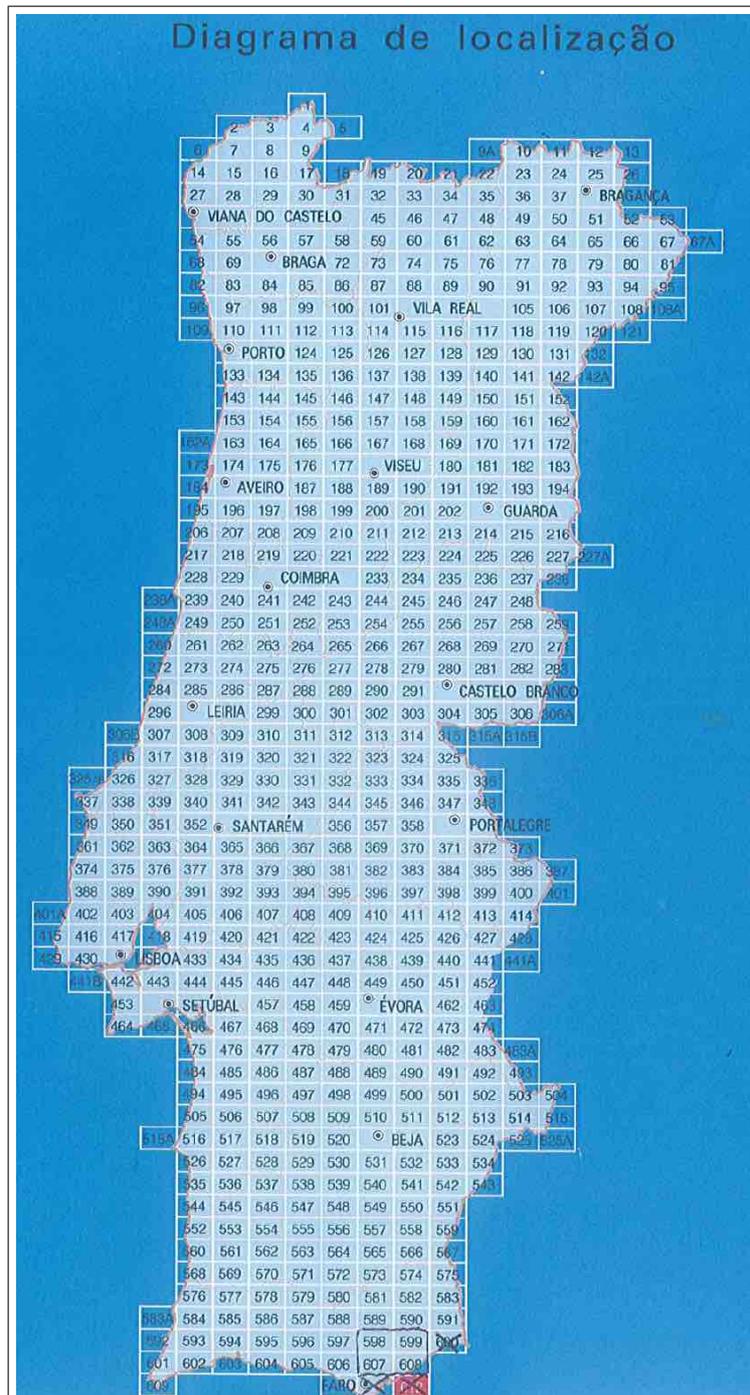


FIGURE 13 : Recto de la « Carta Militar de Portugal » pour la folha n° 612. Les cartes sont numérotées de haut en bas et de gauche à droite avec 20 exceptions : 9A, 67A, 108A, 142A, 162A, 227A, 238A, 248A, 306A, 306B, 315A, 315B, 325AB, 401A, 441A, 441B, 483A, 515A, 525A et 583A.

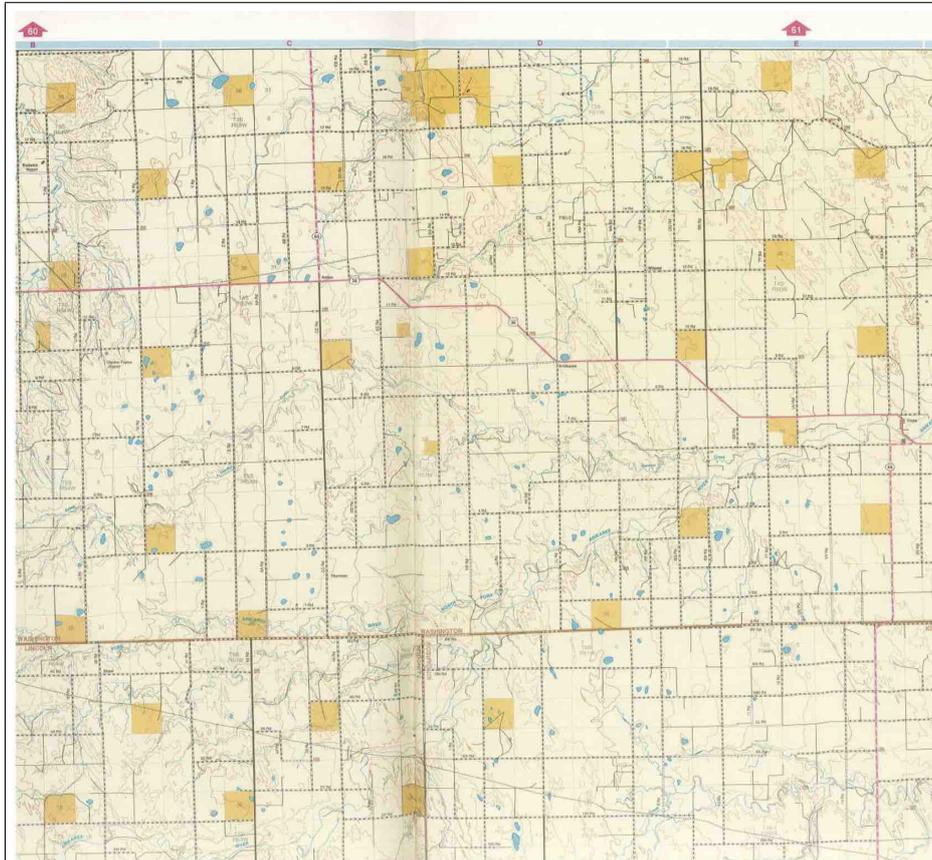


FIGURE 14 : Copie à basse résolution d'une partie de la carte n° 76-77 de l'atlas routier du COLORADO (*The roads of Colorado*) © 1996 by Shearer Publishing. ISBN 0-940672-59-6. Les carrés élémentaires font 1 miles (1.6 km) de coté. Les droites sont des voies de circulation. La couleur orange signifie que c'est un « *State Land* », la couleur jaune que c'est un « *Private Land* ». Les lignes rouges continues sont des « *U.S. Highway* » (ici la n° 36 faisant 2276 km depuis le parc national de ROCKY MOUNTAIN dans le COLORADO jusqu'à UHRICHSVILLE dans l'OHIO), les lignes rouges discontinues des « *State Highway* ». L'« échelle » est donnée comme étant « *1 inch = 2.5 miles* », ce qui pour un nombre sans dimensions frôle l'hérésie en mélangeant des unités différentes, mais on voit bien l'idée. Ces unités vernaculaires sont heureusement rattachées au système métrique avec 1 pouce = 2.54 cm et 1 miles = 1609,344 m, d'où on déduit que la véritable échelle, grandeur dans dimension, est de 1:158400, soit 1 cm pour 1.6 km environ.

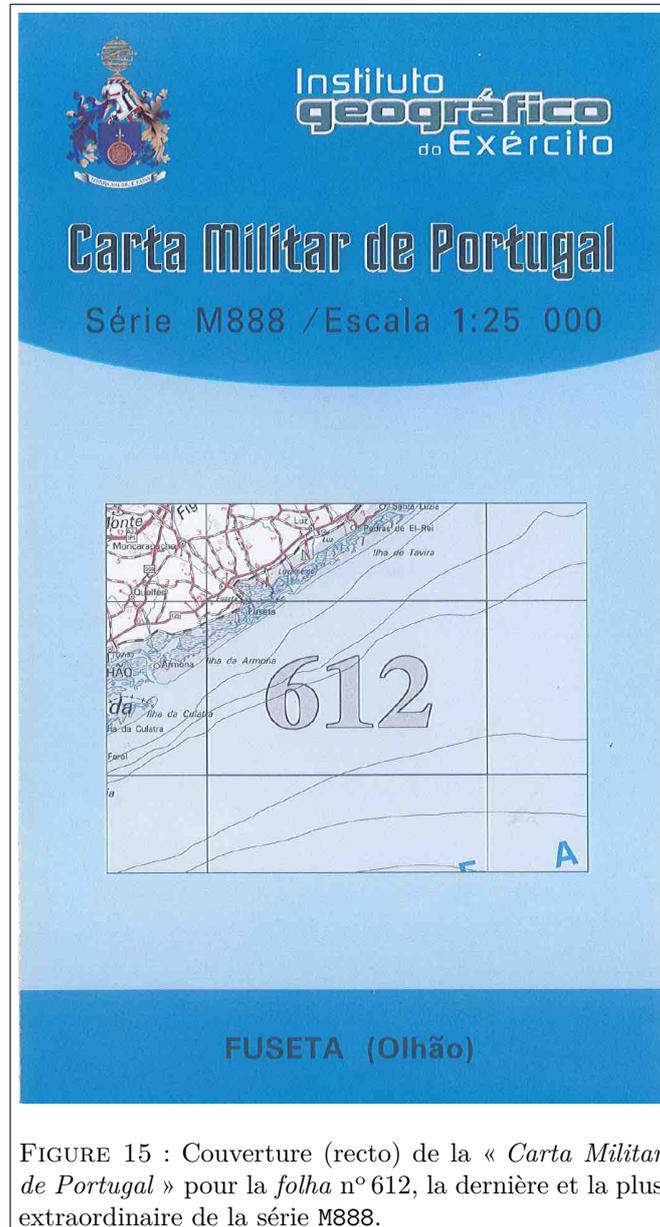


FIGURE 15 : Couverture (recto) de la « *Carta Militar de Portugal* » pour la *folha* n° 612, la dernière et la plus extraordinaire de la série M888.

Références

- [1] R.J. Hijmans. *terra : Spatial Data Analysis*, 2024. R package version 1.7-71.
- [2] R.J. Hijmans, M. Barbosa, A. Ghosh, and A. Mandel. *geodata : Download Geographic Data*, 2023. R package version 0.5-9.