

# Estimation du taux de couvert en chênes tempérés sur des mailles de $8 \times 8$ km en France continentale

Pr Jean R. LOBRY

---

Dans cette fiche on cherche à estimer le taux de couverture en chêne sessile (*Quercus petrae*) et chêne pédonculé (*Q. robur*) sur des mailles carrées de 8 km de coté définissant un pavage régulier de la France continentale. Ce sont les mailles SAFRAN pour lesquelles on dispose de données météorologiques. C'est l'occasion de manipuler des données géographiques de type bitmap (*raster*) et vectoriel (*shapefile*) avec le paquet `terra` et ses amis.

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Méthode d'extrapolation</b>	<b>2</b>
2.1	La table <code>mailles</code> . . . . .	2
2.2	La table <code>allIFN</code> . . . . .	4
2.3	Extrapolation du taux de couvert en chênes . . . . .	7
2.4	Sauvegarde des résultats . . . . .	15
<b>3</b>	<b>Annexes</b>	<b>16</b>
3.1	Obtention de la table <code>mailles</code> . . . . .	16
3.2	Obtention de la table <code>allIFN</code> . . . . .	16
3.3	Paquets utiles . . . . .	35
3.4	Les EPSG utiles . . . . .	36
	<b>Références</b>	<b>36</b>

## 1 Introduction

COMME pour tout travail d'extrapolation, les résultats présentés ici doivent être pris *cum grano salis*. Il y a tout d'abord une extrapolation spatiale puisque l'on passe d'un petit point d'inventaire de 7 ares à une maille de  $1 \times 1$  km, ce qui peut sembler *a priori* pour le moins osé (figure 1 page 4). Ce n'est pas complètement irrationnel dans le sens où les forêts en France continentale sont en général de grandes zones homogènes : dans le bilan patrimonial des forêts domaniales 2015<sup>1</sup> on peut voir page 13 que 99 % des surfaces boisées sont couvertes par des forêts de plus de 2 km<sup>2</sup>. De plus les points d'inventaires ne sont pas posés au hasard mais après examen par des spécialistes de photos aériennes [6], on peut donc espérer qu'ils soient représentatifs. Il y a ensuite une extrapolation temporelle : les données satellitaires ont été acquises en 2018 alors que les données des points d'inventaires couvrent la période 2006-2022, soit 17 ans. Comme nous nous intéressons aux chênes, dont l'âge de la coupe est en générale supérieure à 100 ans, on peut espérer que le couvert n'a pas dramatiquement changé pendant cette période.

LES fichiers utilisés ici sont lus à partir du répertoire local TCACPS. Une copie est disponible en ligne, pour y accéder il faut adapter le code, par exemple :

```
# Au lieu de load("TCACPS/mailllesTCACPS.Rda")
chmin <- "http://pbil.univ-lyon1.fr/R/donnees/TCACPS/"
load(url(paste0(chmin, "mailllesTCACPS.Rda")))
```

## 2 Méthode d'extrapolation

LE détail de comment les données ont été obtenues sera détaillé en annexe. On dispose de deux jeux de données : `maillles` qui contient des informations relatives aux mailles SAFRAN de  $8 \times 8$  km et `allIFN` qui contient des informations sur les mailles de  $1 \times 1$  km de l'IFN (inventaire forestier national) incluses dans les mailles SAFRAN.

```
load("TCACPS/mailllesTCACPS.Rda")
load("TCACPS/allIFN_COORD.Rda")
nrow(allIFN) == 64*nrow(maillles)
[1] TRUE
```

ON a vérifié ici que l'on avait exactement  $8 \times 8 = 64$  fois plus de mailles IFN que de mailles SAFRAN.

### 2.1 La table mailles

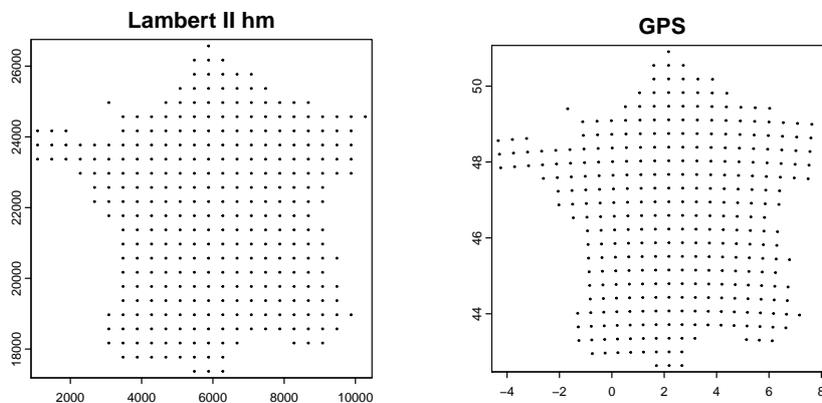
Les variables disponibles dans la table `maillles` sont les suivantes :

```
head(maillles)
  LAMBIIX_hm LAMBIYY_hm LAT_DG  LON_DG altitude i  j IDij TCA_CPS n
4          760      24010 48.3953 -4.74612    54 1  98 1-98 23.00000 4
5          760      23930 48.3237 -4.73641    33 1  97 1-97  0.00000 0
7          840      24170 48.5450 -4.65775    37 2 100 2-100  0.00000 4
8          840      24090 48.4734 -4.64814    70 2  99 2-99 41.75000 8
9          840      24010 48.4017 -4.63855    76 2  98 2-98 37.00000 6
12         840      23610 48.0435 -4.59101    50 2  93 2-93 17.77778 9
```

<sup>1</sup><https://www.onf.fr/onf/+/f1::bilan-patrimonial-des-forets-domaniales-2015.html>

LES quatre premières colonnes sont les coordonnées du centre de la maille SAFRAN exprimées en LAMBERT II (EPSG:27572) et en GPS (EPSG:4326). Voir en annexe section 3.4 page 36 les principaux systèmes de projection utilisés ici. Dans la figure ci-après on affiche une maille sur 5 dans les deux systèmes de projection. On voit que les points sont parfaitement alignés en LAMBERT II mais pas en GPS.

```
with(subset(mailles, i %% 5 == 0 & j %% 5 == 0), {
  par(mfrow = c(1, 2))
  pts <- terra::vect(cbind(LAMBIIX_hm, LAMBIIY_hm), crs = "EPSG:27572")
  terra::plot(pts, main = "Lambert II hm", pch = 19, cex = 0.2)
  pts <- terra::vect(cbind(LON_DG, LAT_DG), crs = "EPSG:4326")
  terra::plot(pts, main = "GPS", pch = 19, cex = 0.2)
})
```

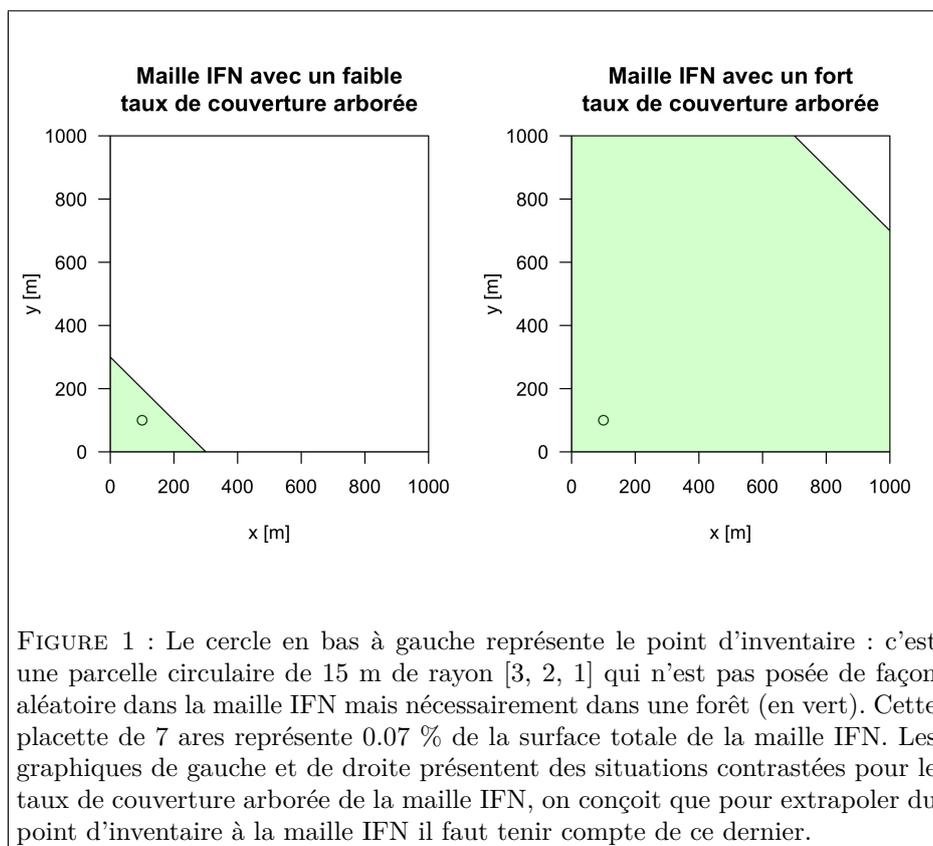


LA colonne `altitude` donne l'altitude moyenne de la maille SAFRAN exprimée en mètres. Les colonnes `i` et `j` sont les rangs des mailles utilisés pour construire par concaténation la clef d'identification `IDij` de la maille. Le principe de calcul est illustré dans le code ci-après. Les valeurs de 760 et de 16250 sont les valeurs minimales de longitude et de latitude pour toutes les mailles SAFRAN (Corse incluse), la valeur de 80 est la distance en hectomètres entre les mailles, et on ajoute 1 pour avoir un indice de valeur minimale égale à l'unité.

```
with(mailles, {
  stopifnot(all( (LAMBIIX_hm - 760)/80 + 1 == i ))
  stopifnot(all( (LAMBIIY_hm - 16250)/80 + 1 == j ))
  stopifnot(all(IDij == paste(i, j, sep = "-")))
})
```

L'INTÉRÊT de la clef d'identification `IDij` est de pouvoir faire facilement des opérations de jointures avec d'autres tables portant des informations sur les mailles SAFRAN, par exemple des données météorologiques. Les deux dernières colonnes donnent une information très grossière (du type présence-absence) sur le taux de couvert en chêne sessiles et pédonculés<sup>2</sup>. La colonne `n` donne le nombre

<sup>2</sup>Plus de détails dans la fiche « À quelle altitude trouve-t-on des chênes sessiles et pédonculés en France métropolitaine? » à <http://pbil.univ-lyon1.fr/R/pdf/altichene.pdf>



de mailles IFN avec un point d'inventaire documenté qui ont été rattachées à la maille SAFRAN sur la base de la distance la plus courte. C'est un peu différent de ce qui a été fait ici où on ne considère que les 64 mailles IFN incluses dans la maille SAFRAN. La petite différence ne concerne que les mailles IFN en périphérie qui ne sont pas dans une maille SAFRAN. La colonne TCA\_CPS est la moyenne des taux de couverture arborée des chênes sessiles et pédonculés des *points d'inventaire* des  $n$  mailles IFN documentées. C'est très grossier parce que le point d'inventaire ne représente qu'une petite fraction de la surface de la maille IFN et que l'on ne tient pas compte du taux de couverture arborée de la maille IFN (voir la figure 1 page 4).

## 2.2 La table allIFN

```
head(allIFN)
  XLII_m  YLII_m IDij IDkl TCD      XGPS      YGPS      XL93      YL93 dataIFN TCA_CPS
1  72200 2397600 1-98 1-1   0 -4.794015 48.36167 123631.5 6835290 FALSE   NA
2  72200 2398600 1-98 1-2   0 -4.795238 48.37063 123639.4 6836290 FALSE   NA
3  72200 2399600 1-98 1-3   0 -4.796461 48.37958 123647.4 6837289 FALSE   NA
4  72200 2400600 1-98 1-4   0 -4.797684 48.38853 123655.4 6838288 FALSE   NA
5  72200 2401600 1-98 1-5   0 -4.798908 48.39749 123663.3 6839288 FALSE   NA
6  72200 2402600 1-98 1-6   0 -4.800132 48.40644 123671.3 6840287 FALSE   NA
```

CETTE table contient des informations sur toutes les mailles IFN incluses dans les maille SAFRAN. Les coordonnées du centre de la maille sont don-

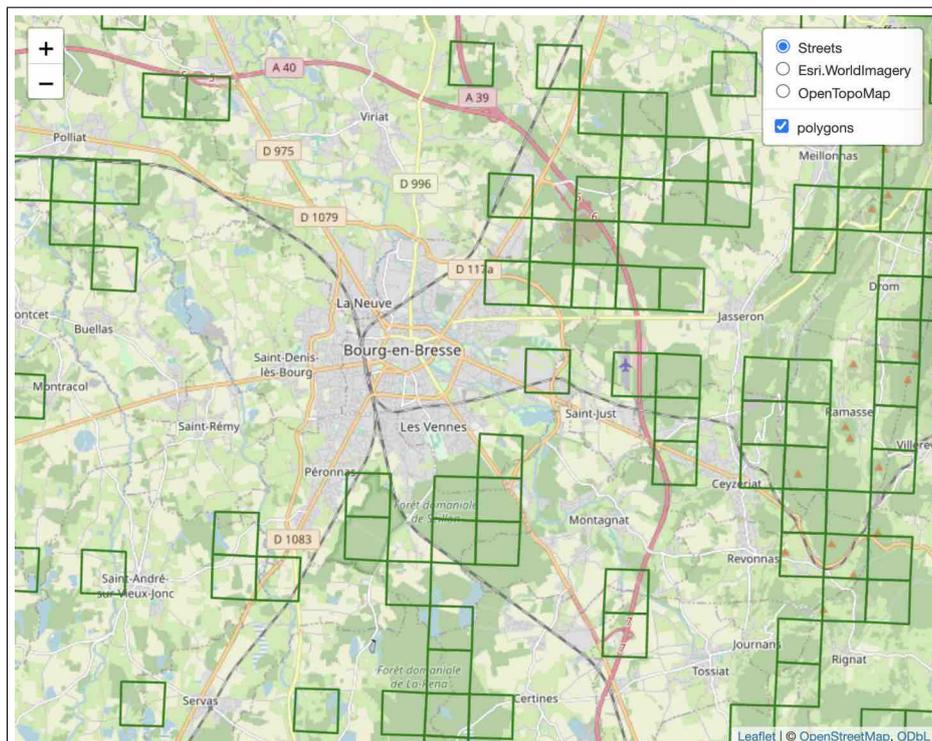
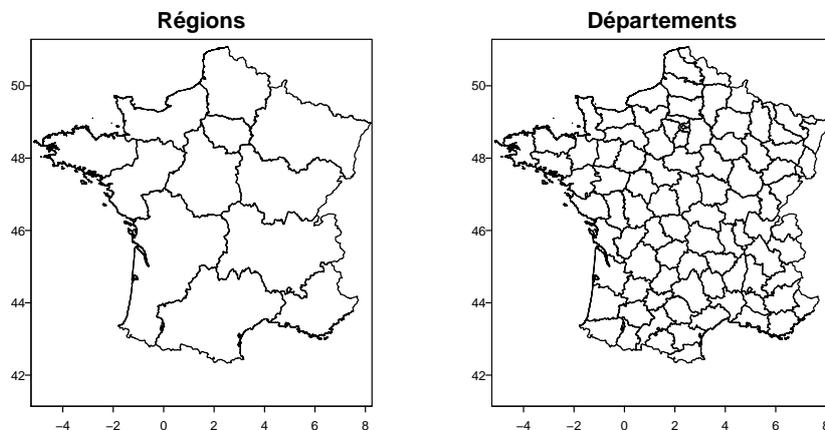


FIGURE 2 : Illustration de l'hétérogénéité du taux de couverture arborée des mailles IFN. La maille IFN en bas à gauche au voisinage de SAINT-ANDRÉ SUR VIEUX-JONC a un taux de couvert arborée bien plus faible celles au Sud de BOURG-EN-BRESSE dans les forêts domaniales de SEILLON et de LA RENA. Le code utilisé pour produire cette carte est donné en annexe section 3.3 page 35.

nées dans 3 systèmes de projection différents dans les colonnes X\* et Y\*. La clef d'identification IDij permet de savoir à quelle maille SAFRAN on se rattache. La clef d'identification IDk1 précise la position de la maille IFN dans la maille SAFRAN, de 1-1 pour la maille en bas à gauche à 8-8 pour celle en haut à droite. La colonne TCD, pour *Tree Cover Density*, et le taux de couverture arborée moyen de la maille IFN, exprimé en pourcentage. Il est obtenu à partir de données satellitaires d'une résolution de  $100 \times 100$  m, elles-mêmes issues de données avec une résolution de  $10 \times 10$  m. Un dessin valant mieux qu'un long discours, nous allons illustrer ceci avec une carte. Avec la fonction `gadm()` du paquet `geodata` [9] on récupère les données des régions administratives françaises au niveau des régions et des départements pour pouvoir faire des fonds de carte.

```
FR1 <- geodata::gadm(country = "France", level = 1, path = ".")
terra::writeVector(FR1, "TCACPS/FR1.shp", overwrite=TRUE)
FR2 <- geodata::gadm(country = "France", level = 2, path = ".")
terra::writeVector(FR2, "TCACPS/FR2.shp", overwrite=TRUE)

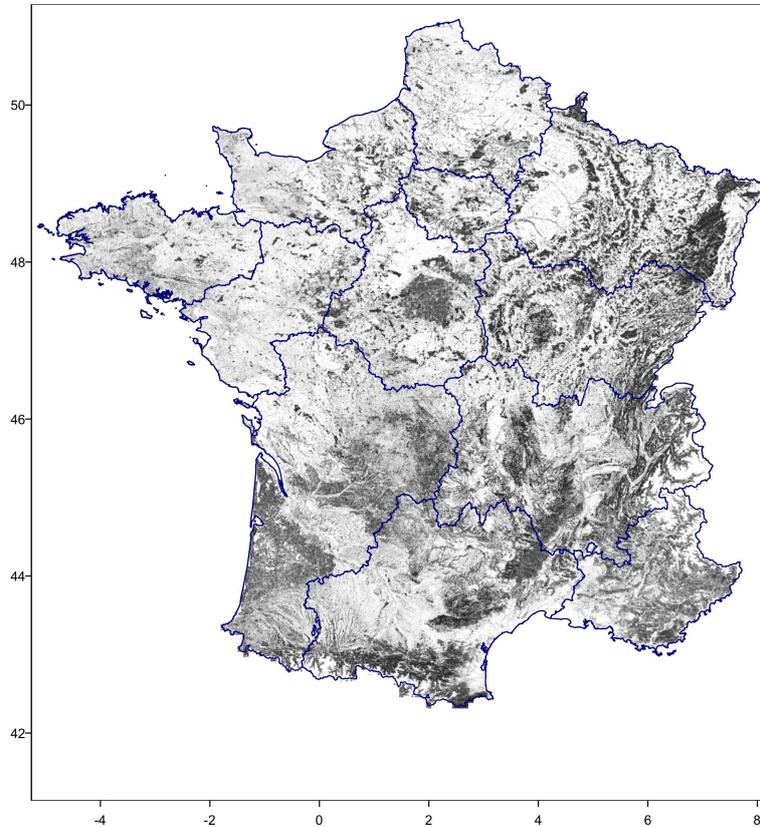
par(mfrow = c(1, 2))
FR1 <- terra::vect("TCACPS/FR1.shp")
terra::plot(FR1, xlim = c(-5, 8), las = 1, main = "Régions")
FR2 <- terra::vect("TCACPS/FR2.shp")
terra::plot(FR2, xlim = c(-5, 8), las = 1, main = "Départements")
```



AVEC une résolution de  $1 \times 1$  km, quand on représente de taux de couvert arborée, on commence à distinguer des éléments de reliefs.

```
with(allIFN, {
  terra::plot(FR1, xlim = c(-5, 8), las = 1, main = "Taux de couverture arborée des mailles IFN")
  pts <- terra::vect(cbind(XGPS, YGPS), crs = "EPSG:4326")
  terra::points(pts, col = grey(1 - TCD/100), cex = 0.1)
  terra::lines(FR1, col = "blue4")
})
```

### Taux de couverture arborée des mailles IFN



LES deux dernières colonnes contiennent les données de l'IFN. La variable `dataIFN` est à `TRUE` si la maille IFN a un point d'inventaire. La variable `TCA_CPS` donne alors le taux de couvert (pourcentage) en chênes sessiles et pédonculés du point d'inventaire.

```
head(subset(allIFN, dataIFN)[ , 10:11])
  dataIFN TCA_CPS
26    TRUE      40
32    TRUE      10
34    TRUE       0
133   TRUE       0
187   TRUE       0
196   TRUE      64
```

## 2.3 Extrapolation du taux de couvert en chênes

NOUS allons extrapoler progressivement pour mettre en évidence l'évolution des résultats. La première étape ne sert pas à grand chose, sauf à vérifier que nous retrouvons les résultats antérieurs.

### 2.3.1 Moyenne des taux des points d'inventaire

C'EST le degré 0 de l'extrapolation où on ne tient pas compte du taux de couvert arborée de la maille IFN. Cela permet quand même de débroussailler

les choses et en particulier de préciser ce que l'on fait s'il n'y a pas de maille IFN documentée dans la maille SAFRAN : c'est une zone où il y a très peu de forêts, on considère donc que le taux de couvert en chênes est nul. L'idée est de compléter la table `mailles` ligne par ligne. La variable `the_IDij` contient la clef d'identification de la maille SAFRAN en cours, on sélectionne dans `allIFN` toutes les mailles IFN documentées pour cette maille SAFRAN pour calculer la moyenne.

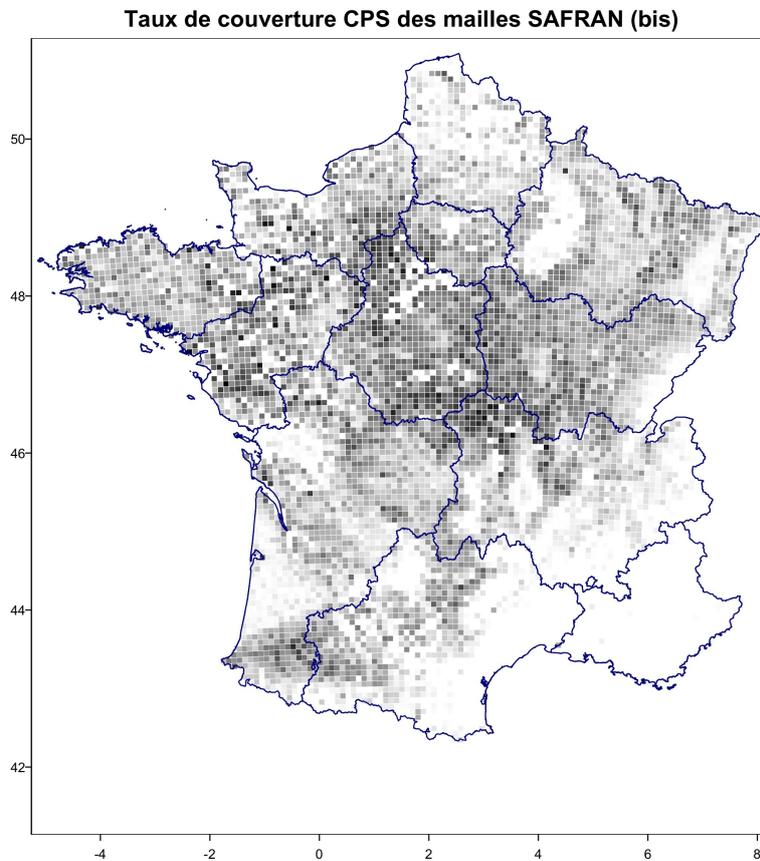
```
for(i in seq_len(nrow(mailles))){
  the_IDij <- mailles[i, "IDij"]
  with(subset(allIFN, IDij == the_IDij & dataIFN),{
    nplac <- length(TCA_CPS)
    mailles[i, "nbis"] <<- nplac
    if(nplac == 0L){
      mailles[i, "TCA_CPSbis"] <<- 0
    } else {
      mailles[i, "TCA_CPSbis"] <<- mean(TCA_CPS)
    }
  })
}
save(mailles, file = "TCACPS/maillesbis.Rda")
```

IL Y A des petites différences pour le nombre de mailles IFN documentées par maille SAFRAN que j'attribue à la différence de méthodologie utilisée pour affecter les mailles IFN aux mailles SAFRAN (la méthode utilisée ici me semble plus rigoureuse). Mais quand les nombres de mailles IFN documentées sont égaux on obtient toujours exactement le même taux de couvert pour les chênes :

```
load("TCACPS/maillesbis.Rda")
with(subset(mailles, n == nbis), {
  all(TCA_CPS == TCA_CPSbis)
})
[1] TRUE
```

Faisons une petite représentation graphique.

```
with(mailles, {
  terra::plot(FR1, xlim = c(-5, 8), las = 1, main = "Taux de couverture CPS des mailles SAFRAN (bis)")
  pts <- terra::vect(cbind(LON_DG, LAT_DG), crs = "EPSG:4326")
  terra::points(pts, col = grey(1 - TCA_CPSbis/max(TCA_CPSbis)), cex = 0.6, pch = 15)
  terra::lines(FR1, col = "blue4")
})
```



CETTE carte doit s'interpréter comme la représentation de données en présence-absence. Ce que l'on voit surtout ici c'est que l'on ne rencontre pas de chênes sessiles et pédonculés dans certaines régions, par exemple dans le pourtour méditerranéen.

### 2.3.2 Prise en compte du taux de couvert arborée de la maille IFN

ON cherche à traiter le problème illustré par la figure 1 page 4, à savoir que quand on calcule la moyenne du taux de couverture en chêne au niveau de la maille SFRAN il ne faut pas donner le même poids aux mailles IFN avec un faible ou un fort taux de couverture arborée. L'idée est simplement de multiplier le taux de couverture en chêne du point d'inventaire (TCA\_CPS) par le taux de couverture arborée de la maille IFN (TCD/100).

```
for(i in seq_len(nrow(mailles))){
  the_IDij <- mailles[i, "IDij"]
  # Mailles IFN documentées
  with(subset(allIFN, IDij == the_IDij & dataIFN),{
    nplac <- length(TCA_CPS)
    mailles[i, "n2"] <- nplac
    if(nplac == 0L){
      mailles[i, "TCA_CPS2"] <- 0
    } else {
      mailles[i, "TCA_CPS2"] <- mean(TCA_CPS*TCD/100)
    }
  })
}
```

```

} })
save(mailles, file = "TCACPS/mailles2.Rda")

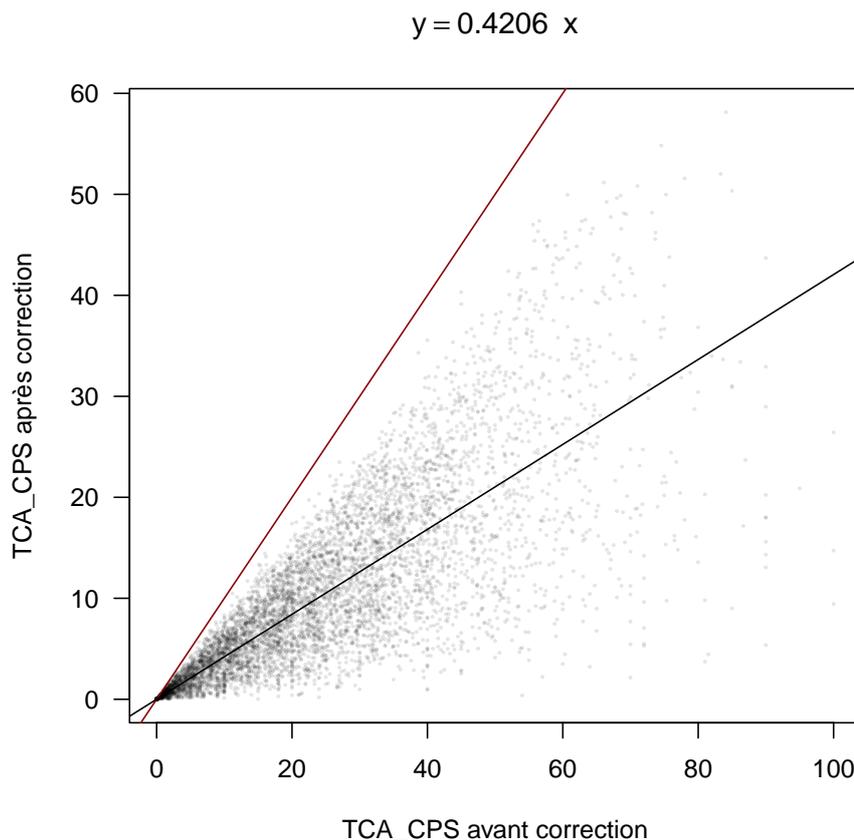
```

DANS la figure ci-dessous on compare les valeurs du taux de couverture en chêne des mailles SAFRAN avant et après correction. La première bissectrice est en rouge, tous les points sont en dessous, c'est normal car il n'y a que dans le cas où toutes les mailles IFN auraient un taux de couvert arborée de 100 % que les valeurs non nulles seraient inchangées après correction. La ligne noire est la droite de régression par l'origine, en moyenne les valeurs sont divisées approximativement par 2.

```

load("TCACPS/mailles2.Rda")
with(mailles, {
  stopifnot(all(n2 == nbis)) # Sanity check
  plot(TCA_CPSbis, TCA_CPS2, las = 1,
       xlab = "TCA_CPS avant correction",
       ylab = "TCA_CPS après correction",
       pch = 19, cex = 0.2, col = rgb(0,0,0,0.1))
  abline(c(0, 1), col = "red4")
  abline(lm1 <- lm(TCA_CPS2~TCA_CPSbis - 1))
  pente <- signif(lm1$coef[1], 4)
  title(main = bquote(y == .(pente)*~x))
})

```



SI on calcule le taux de couverture arborée moyen des mailles IFN documentées on voit qu'il est de l'ordre de 50 %, il est donc normal que les valeurs après

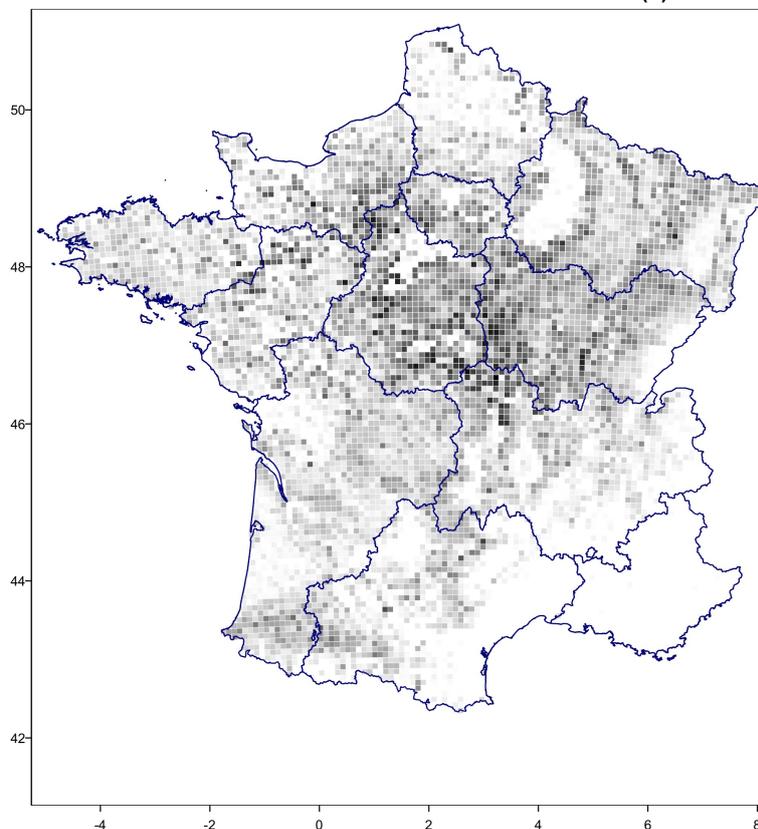
correction soient divisées par 2.

```
with(subset(allIFN, dataIFN), mean(TCD))
[1] 48.95251
```

P RAPPORT à la carte précédente, on voit que ce sont surtout les régions de l'Ouest avec une forte emprise agricole qui voient (de façon relative parce que l'échelle de gris a été recalée sur la valeur maximale) leur valeur de taux de couvert en chêne diminuer.

```
with(mailles, {
  terra::plot(FR1, xlim = c(-5, 8), las = 1, main = "Taux de couverture CPS des mailles SAFRAN (2)")
  pts <- terra::vect(cbind(LON_DG, LAT_DG), crs = "EPSG:4326")
  terra::points(pts, col = grey(1 - TCA_CPS2/max(TCA_CPS2)), cex = 0.6, pch = 15)
  terra::lines(FR1, col = "blue4")
})
```

Taux de couverture CPS des mailles SAFRAN (2)



### 2.3.3 Prise en compte du taux de couvert arborée de la maille SAFRAN

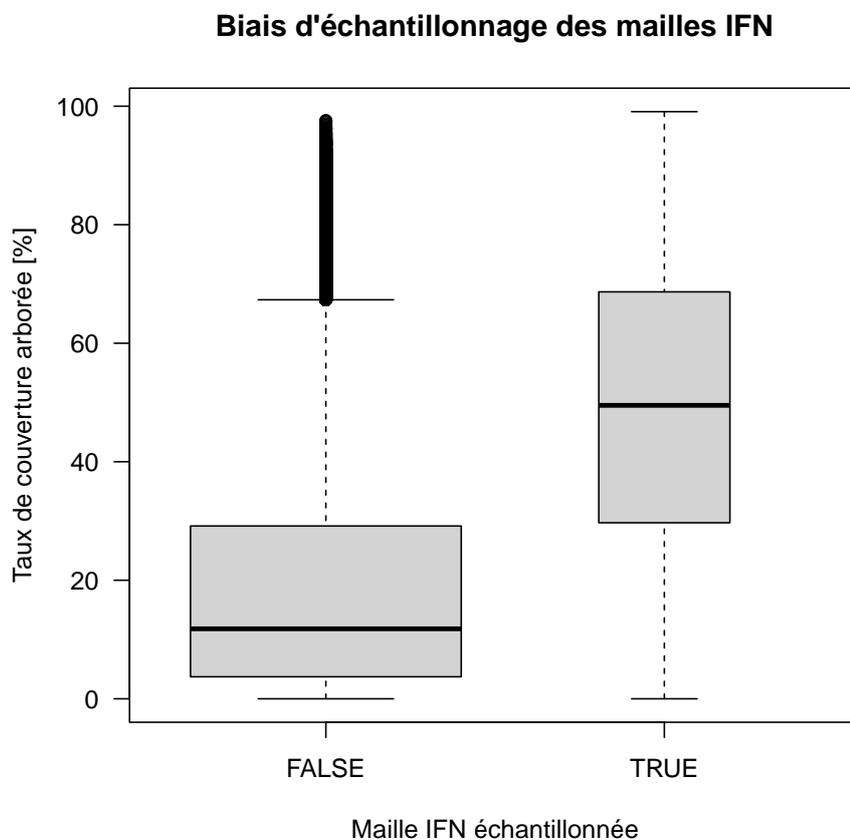
LA méthode utilisée dans le paragraphe précédent serait satisfaisante si les  $n$  mailles IFN documentées dans les 64 mailles IFN possibles dans la maille SAFRAN étaient le fruit d'un échantillonnage aléatoire simple. Mais ce n'est absolument pas le cas : on a un problème analogue à celui de la figure 1 page 4, mais

cette fois à l'échelle de la maille SAFRAN. Pour prendre un exemple extrême, une maille IFN entièrement urbanisée ne sera *jamaïs* échantillonnée puisqu'elle ne comporte pas de forêt. Avec la méthode utilisée dans le paragraphe précédent nous avons un biais systématique de surestimation du taux de couverture en chêne. La source du biais d'échantillonnage est illustrée dans le graphique suivant :

```

boxplot(TCD-dataIFN, allIFN, varwidth = TRUE, las = 1,
        xlab = "Maille IFN échantillonnée",
        ylab = "Taux de couverture arborée [%]",
        main = "Biais d'échantillonnage des mailles IFN")

```



Le taux de couverture arborée est de l'ordre de 50 % dans les mailles IFN échantillonnées contre seulement 10 % dans celles qui ne le sont pas. Le *dernier* quartile des mailles non-échantillonnées est à 30 % alors que c'est la valeur du *premier* quartile des mailles échantillonnées. Pas besoin de faire un test statistique pour affirmer que la différence est extrêmement significative puisque notre échantillon est exhaustif. Que faire pour pallier ce problème ? Nous allons commencer par imputer les valeurs manquantes en prenant la moyenne des points d'inventaire disponibles dans la maille SAFRAN. L'idée est de dire que si il y avait de la forêt dans la maille non-échantillonnée, alors elle aurait la même composition que ce qui a été observé au voisinage immédiat.

```
moy <- function(x){
  if(length(x) == 0L) return(0) # Pour éviter NaN si pas de données
  return(mean(x))
}
for(i in seq_len(nrow(mailles))){
  the_IDij <- mailles[i, "IDij"]
  allIFN[allIFN$IDij == the_IDij & !allIFN$dataIFN, "TCA_CPS"] <-
    with(subset(allIFN, IDij == the_IDij & dataIFN), moy(TCA_CPS))
}
save(allIFN, file = "TCACPS/allIFN_imputed.Rda")
```

L'AVANTAGE de travailler avec la moyenne quand on fait de l'imputation c'est que la valeur de la moyenne est inchangée que l'on intègre les valeurs imputées ou non :

```
load("TCACPS/allIFN_imputed.Rda")
atlestone <- with(allIFN, tapply(dataIFN, IDij, \(x) sum(x) >= 1))
all.equal(with(subset(allIFN, IDij %in% atlestone),
  tapply(TCA_CPS, IDij, mean)),
  with(subset(allIFN, IDij %in% atlestone & allIFN$dataIFN),
  tapply(TCA_CPS, IDij, mean)),
  check.attributes = FALSE)
```

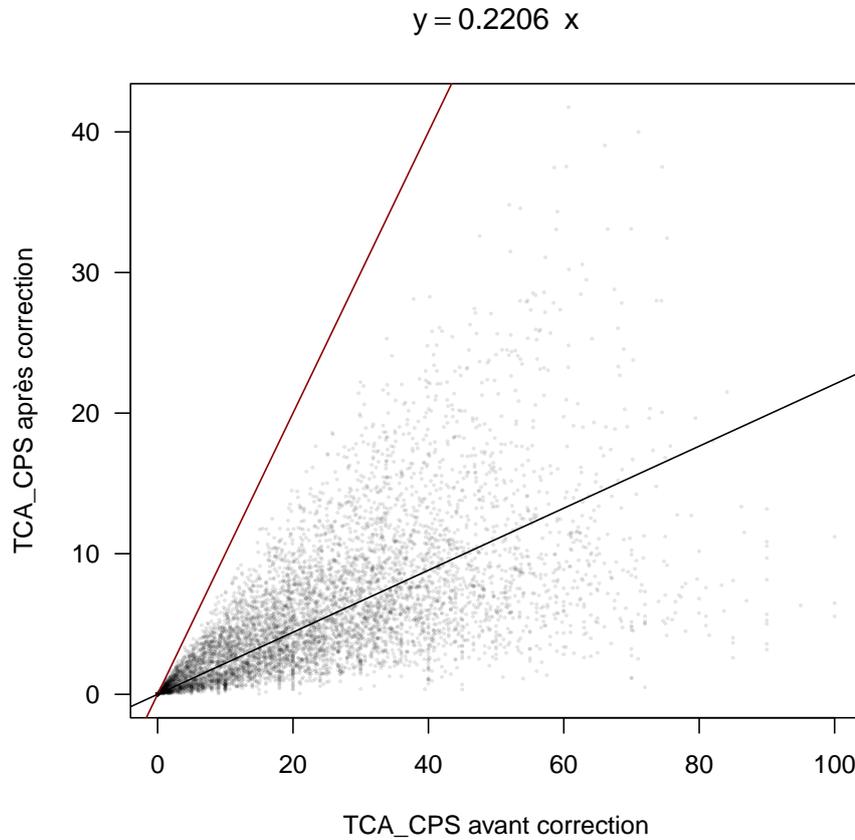
[1] TRUE

IL suffit maintenant de recalculer la moyenne comme dans la section précédente, mais cette fois avec toutes les mailles IFN de la maille SAFRAN.

```
for(i in seq_len(nrow(mailles))){
  the_IDij <- mailles[i, "IDij"]
  # Toutes les mailles IFN
  with(subset(allIFN, IDij == the_IDij),{
    nplac <- sum(dataIFN) # Que les documentées
    mailles[i, "n3"] <- nplac
    if(nplac == 0L){
      mailles[i, "TCA_CPS3"] <- 0
    } else {
      mailles[i, "TCA_CPS3"] <- mean(TCA_CPS*TCD/100)
    }
  })
}
save(mailles, file = "TCACPS/mailles3.Rda")
```

DANS la figure ci-dessous on compare les valeurs du taux de couverture en chêne des mailles SAFRAN avant et après correction. La première bissectrice est en rouge, tous les points sont en dessous, comme attendu. La ligne noire est la droite de régression par l'origine, en moyenne les valeurs sont divisées approximativement par 5.

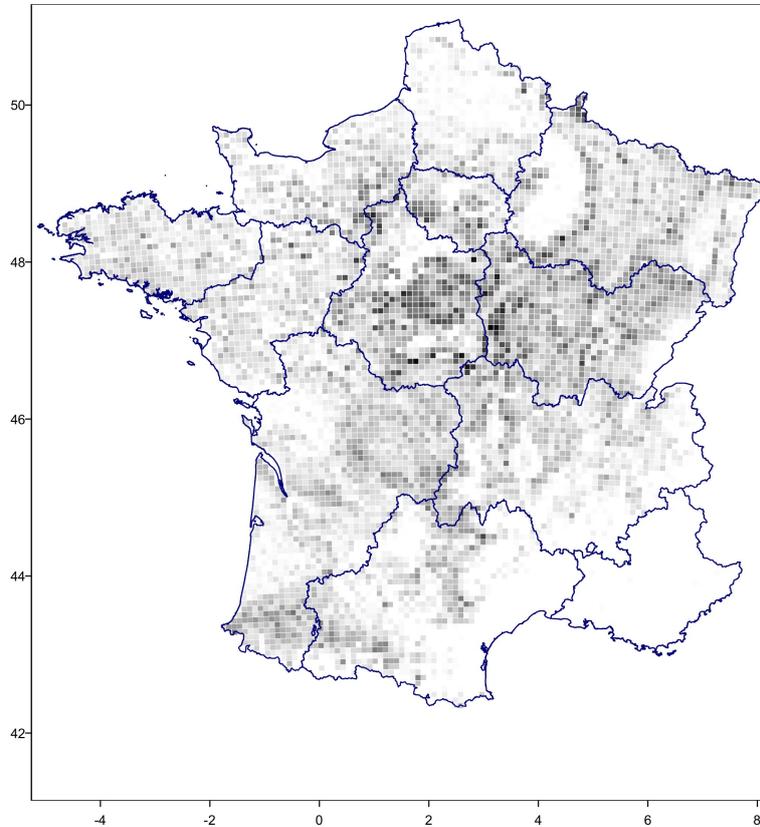
```
load("TCACPS/mailles3.Rda")
with(mailles, {
  stopifnot(all(n3 == nbis)) # Sanity check
  plot(TCA_CPSbis, TCA_CPS3, las = 1,
    xlab = "TCA_CPS avant correction",
    ylab = "TCA_CPS après correction",
    pch = 19, cex = 0.2, col = rgb(0,0,0,0.1))
  abline(c(0, 1), col = "red4")
  abline(lm1 <- lm(TCA_CPS3~TCA_CPSbis - 1))
  pente <- signif(lm1$coef[1], 4)
  title(main = bquote(y == .(pente)*~x))
})
```



PRAPPORT à la carte précédente, on a augmenté le contraste avec les régions de l'Ouest encore plus pâlichonnes. La grande virgule vide en haut à droite correspond à la sylvoécocorégion B43 (CHAMPAGNE crayeuse) réputée pour ses grandes plaines, le gros massif au centre à la sylvoécocorégion B70 (SOLOGNE-ORLÉANAIS) réputée pour ses forêts.

```
with(mailles, {
  terra::plot(FR1, xlim = c(-5, 8), las = 1, main = "Taux de couverture CPS des mailles SAFRAN (3)")
  pts <- terra::vect(cbind(LON_DG, LAT_DG), crs = "EPSG:4326")
  terra::points(pts, col = grey(1 - TCA_CPS3/max(TCA_CPS3)), cex = 0.6, pch = 15)
  terra::lines(FR1, col = "blue4")
})
```

### Taux de couverture CPS des mailles SAFRAN (3)



## 2.4 Sauvegarde des résultats

ON conserve les données qui nous intéressent, un peu de renommage, et on sauvegarde le tout. La signification des variables est la suivante :

**i, j et ID<sub>ij</sub>** : les rangs des mailles SAFRAN et la clef d'identification d'icelles comme expliqué dans la section 2.1 page 2.

**XLII<sub>hm</sub> et YLII<sub>hm</sub>** : les coordonnées du centre de la maille SAFRAN en LAMBERT II, EPSG:27572 dans la section 3.4 page 36.

**XGPS et YGPS** : les coordonnées du centre de la maille SAFRAN en coordonnées GPS, EPSG:4326 dans la section 3.4 page 36.

**alt<sub>m</sub>** : l'altitude moyenne de la maille SAFRAN en mètres.

**TCA<sub>CPS</sub>** : le taux de couvert arborée de la maille SAFRAN en chênes sessiles et pédonculés tels que calculé dans la section 2.3.3 page 11.

**n** : le nombre de mailles IFN documentées (avec un point d'inventaire) dans la maille SAFRAN.

```
load("TCACPS/maillles3.Rda")
maillles <- maillles[, c("i", "j", "IDij", "LAMBIIIX_hm", "LAMBIIY_hm",
                        "LON_DG", "LAT_DG", "altitude", "TCA_CPS3", "n3")]
colnames(maillles) <- c("i", "j", "IDij", "XLII_hm", "YLII_hm", "XGPS", "YGPS",
                       "alt_m", "TCA_CPS", "n")
head(maillles)
  i   j IDij XLII_hm YLII_hm   XGPS   YGPS alt_m TCA_CPS n
4 1  98 1-98    760  24010 -4.74612 48.3953   54 1.375219 3
5 1  97 1-97    760  23930 -4.73641 48.3237   33 0.000000 0
7 2 100 2-100   840  24170 -4.65775 48.5450   37 0.000000 2
8 2  99 2-99    840  24090 -4.64814 48.4734   70 6.595246 6
9 2  98 2-98    840  24010 -4.63855 48.4017   76 4.551756 4
12 2  93 2-93   840  23610 -4.59101 48.0435   50 3.187872 6

comment(maillles) <- paste("Généré le :", Sys.time())
save(maillles, file = "TCACPS/TCACPSmaillles.Rda")
```

## 3 Annexes

### 3.1 Obtention de la table mailles

ON utilise cette table pour avoir les coordonnées du centre des mailles SAFRAN [14]. On trouvera dans la section 2 d'un précédent document<sup>3</sup> tous les détails utiles. Je ne détaille pas le calcul du taux de couvert en chêne sessiles et pédonculés (TCA\_CPS) parce qu'il a été fait avec une méthode différente et que l'on ne l'utilisera pas directement ici. Je vais simplement censurer les mailles en Corse pour gagner de la place sur les cartes et qu'il n'y a pas de chênes sessiles et pédonculés en Corse.

```
load("TCACPS/mailllesTCA.Rda")
maillles <- subset(maillles, i < 125) # Sans la Corse
comment(maillles) <- paste("Mailles SAFRAN hors Corse. Généré le :", Sys.time())
save(maillles, file = "TCACPS/mailllesTCACPS.Rda")
```

### 3.2 Obtention de la table allIFN

J'AI besoin de générer la grille de toutes les mailles IFN possibles dans toutes les mailles SAFRAN, soit  $8462 \times 64 = 541568$  grilles de  $1 \times 1$  km, pour pouvoir calculer le taux de couvert arborée à partir des données satellitaires.

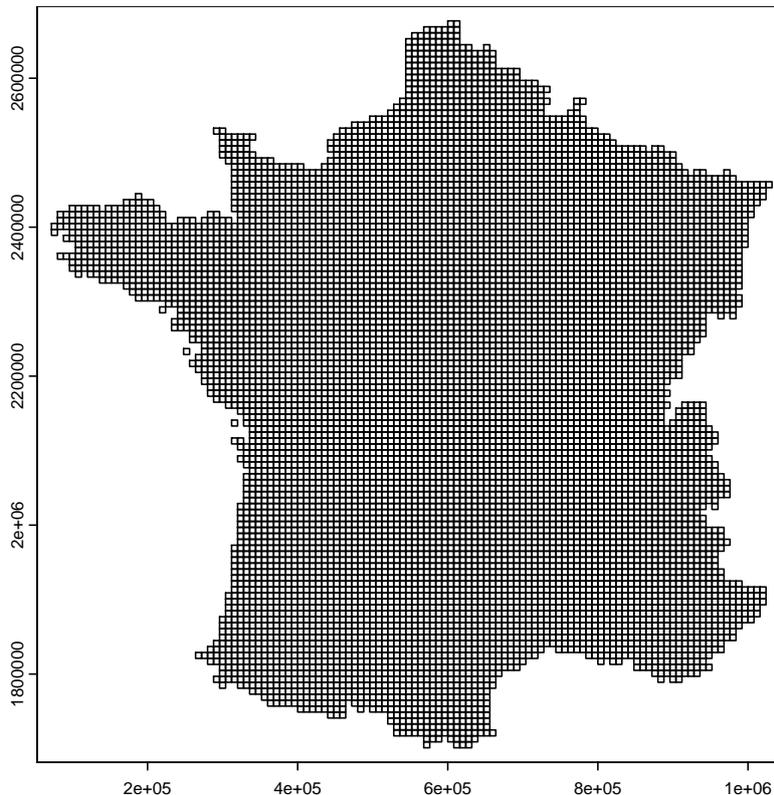
#### 3.2.1 Relation entre les mailles SAFRAN et IFN

ON va travailler en LAMBERT II pour avoir des mailles bien carrées. Dans mailles j'ai les coordonnées en hectomètres, je multiplie par un facteur `shm` pour les avoir en mètres. On veut construire une liste de polygones (des carrés en fait) avec la fonction `terra::vect()` à qui on donne à manger la matrice `lonlat`. La première colonne est une clef d'identification du polygone, comme dans un carré il y a 4 sommets, avec `rep(1:n, each = 4)` le tour est joué. La deuxième colonne sert quand on a des polygones multiples, pas besoin ici, je mets tout à 1. Les deux dernières colonnes contiennent les coordonnées x et y des points définissant les polygones. Je commence par mettre les sommets aux coordonnées du centre de la grille, puis je les décale de 4 km pour atteindre les coins du carré.

<sup>3</sup>« À quelle altitude trouve-t-on des chênes sessiles et pédonculés en France métropolitaine? » à <http://pbil.univ-lyon1.fr/R/pdf/altichene.pdf>

```
load("TCACPS/mailllesTCACPS.Rda")
n <- nrow(maillles) ; id <- rep(1:n, each = 4) ; part <- rep(1, n)
# Calcul des coordonnées des sommets des carrés des mailles
shm <- 100
lon <- shm*rep(maillles[1:n, "LAMBIIIX_hm"], each = 4)
lon <- lon + shm*40*c(-1, +1, +1, -1)
lat <- shm*rep(maillles[1:n, "LAMBIIY_hm"], each = 4)
lat <- lat + shm*40*c(+1, +1, -1, -1)
lonlat <- cbind(id, part, lon, lat)
# Création d'un SpatVector en Lambert II
safgridLII <- terra::vect(lonlat, type = "polygons", crs = "EPSG:27572")
terra::plot(safgridLII, main = "Les mailles SAFRAN")
```

Les mailles SAFRAN



C'EST parfait, nous avons un pavage avec des carrés de 8 km de coté de la FRANCE continentale correspondant aux mailles SAFRAN. Je veux faire exactement la même chose avec les mailles IFN documentées. Je récupère les coordonnées de ces dernières dans la table `plachene` et je ne conserve que celles en première visite.

```
load("TCACPS/placheneIDij.Rda")
# Que première visite
plachene <- subset(plachene, VISITE == 1)
head(plachene)
```

	CAMPAGNE	IDP	VISITE	XL	YL	TCA_CHP	TCA_CHS	TCA_CPS	IDij
1	2005	19	1	903872.3	6804836	0	0	0	98-95
2	2005	23	1	1004318.3	6863931	0	0	0	111-102
3	2005	27	1	849382.0	6743346	0	0	0	91-87
4	2005	41	1	959775.1	6334750	0	0	0	106-36
5	2005	45	1	393417.1	6221638	0	0	0	35-21
6	2005	47	1	561071.3	6681826	0	0	0	55-79

LES coordonnées XL et YL sont en LAMBERT 93, première étape convertir ces coordonnées en Lambert II (en mètres) pour compléter la table `plachene`. Il y a plusieurs façon de faire cela, je vais essayer de n'utiliser que des fonctions du paquet `terra`.

```
ptsL93 <- with(plachene, terra::vect(cbind(XL, YL), crs = "EPSG:9794"))
ptsLII <- terra::project(ptsL93, "EPSG:27572")
plachene$XLII_m <- round(terra::geom(ptsLII)[ , "x"])
plachene$YLII_m <- round(terra::geom(ptsLII)[ , "y"])
head(plachene)
```

	CAMPAGNE	IDP	VISITE	XL	YL	TCA_CHP	TCA_CHS	TCA_CPS	IDij	XLII_m
1	2005	19	1	903872.3	6804836	0	0	0	98-95	853200
2	2005	23	1	1004318.3	6863931	0	0	0	111-102	953200
3	2005	27	1	849382.0	6743346	0	0	0	91-87	799200
4	2005	41	1	959775.1	6334750	0	0	0	106-36	913200
5	2005	45	1	393417.1	6221638	0	0	0	35-21	347200
6	2005	47	1	561071.3	6681826	0	0	0	55-79	511200

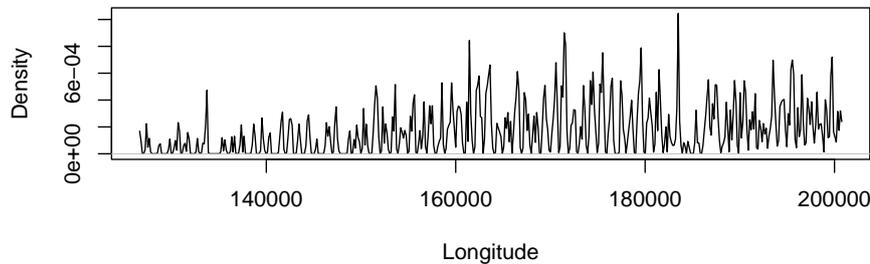
```
YLII_m
1 2373600
2 2433600
3 2311600
4 1903600
5 1785600
6 2247600

save(plachene, file = "TCACPS/plachene11var.Rda")
```

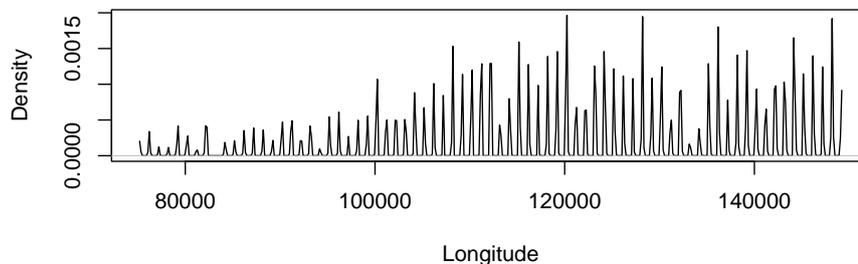
Le graphique ci-après montre que les coordonnées des mailles IFN sont régulièrement espacées en LAMBERT II. Autrement dit, même si elles sont publiées en LAMBERT 93, sous le capot on travaille en LAMBERT II pour avoir des mailles bien alignées.

```
with(subset(plachene, XLII_m < 1.5*10^5), {
  par(mfrow = c(2, 1))
  plot(density(XL, adjust = 0.001), main = "Lambert 93", xlab = "Longitude")
  plot(density(XLII_m, adjust = 0.001), main = "Lambert II", xlab = "Longitude")
})
```

### Lambert 93



### Lambert II

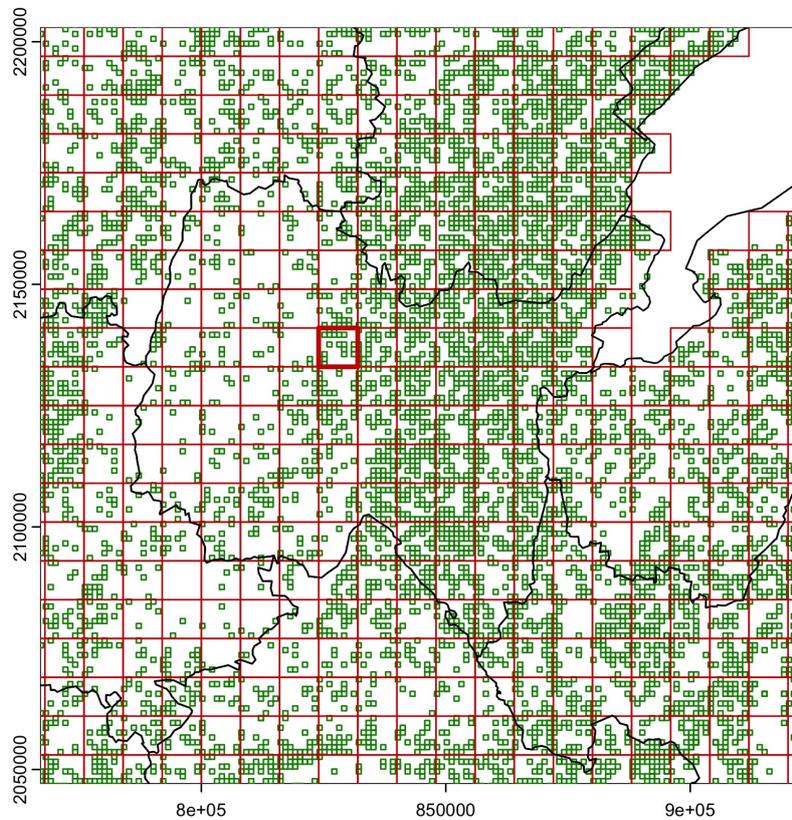


ON fait maintenant la même opération que pour les mailles SAFRAN mais cette fois en s'écartant de plus ou moins 500 m pour générer les coordonnées des coins des carrés.

```
n <- nrow(plachene)
lon <- rep(plachene[1:n, "XLII_m"], each = 4)
lon <- lon + 500*c(-1, +1, +1, -1)
lat <- rep(plachene[1:n, "YLII_m"], each = 4)
lat <- lat + 500*c(+1, +1, -1, -1)
lonlat <- cbind(id = rep(1:n, each = 4), part = 1, lon, lat)
IFNgridLII <- terra::vect(lonlat, type = "polygons", crs = "EPSG:27572")
```

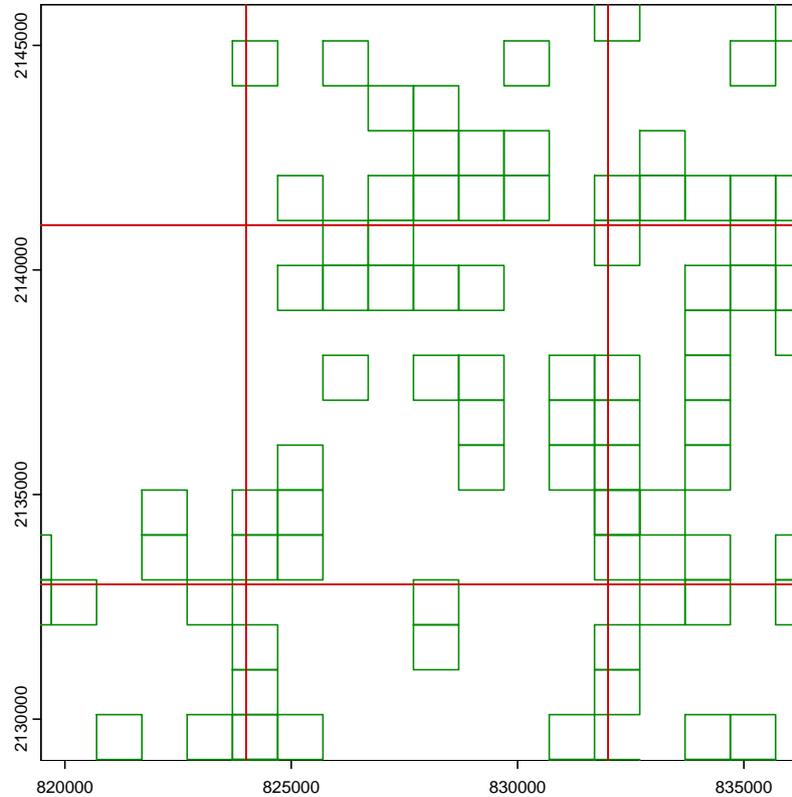
COMME il y a beaucoup de mailles IFN (114976) on fait un zoom sur le département de l'AIN. Les mailles SAFRAN sont en rouge et les mailles IFN en vert. Il n'y a pas de mailles en haut à droite parce que c'est en SUISSE. Le densité de mailles IFN est faible à l'Ouest du département, zone des étangs de LA DOMBES, et forte à l'Est dans le massif du JURA. La maille SAFRAN en gras est celle à proximité de la forêt domaniale de SEILLON sur laquelle on va zoomer plus encore après.

```
FR2 <- terra::vect("TCACPS/FR2.shp")
FR2LII <- terra::project(FR2, "EPSG:27572")
terra::plot(FR2LII, xlim = 10^6*c(0.77, 0.92), ylim = 10^6*c(2.05, 2.2))
terra::lines(IFNgridLII, col = "green4")
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII)
xySeillon <- 100*subset(mailles, IDij == "95-65")[ , c("LAMBIIIX_hm", "LAMBIIY_hm")]
SeillonGridLII <- terra::vect(cbind(rep(1, 4), rep(1, 4),
  xySeillon[1, 1] + 4000*c(-1, +1, +1, -1), xySeillon[1, 2] + 4000*c(+1, +1, -1, -1)),
  type = "polygons", crs = "EPSG:27572")
terra::lines(SeillonGridLII, col = "red3", lwd = 3)
```



POUR zoomer sur la maille d'intérêt j'ai simplement utilisé la fonction de base `locator()` pour récupérer interactivement en cliquant sur la carte précédente les coordonnées qui m'intéressaient.

```
terra::plot(FR2LII, xlim = c(819790.9, 835975.1), ylim = c(2145581, 2129397))
terra::lines(IFNgridLII, col = "green4")
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII)
```



COMME on peut le constater, l'alignement entre les mailles SAFRAN et les mailles IFN n'est pas parfait. Par rapport aux mailles SAFRAN, les mailles IFN sont décalées de 100 m vers le Nord et de 300 m vers l'Ouest. Pour le vérifier on considère les coordonnées des mailles en hectomètres :

```
head(unique(sort(round(plachene$XLII_m/100))), n = 10)
[1] 752 762 772 782 792 802 812 822 842 852
head(unique(sort(round(plachene$XLII_m/100))) %% 10, n = 10)
[1] 2 2 2 2 2 2 2 2 2 2
head(unique(sort(mailles$LAMBIIX_hm)))
[1] 760 840 920 1000 1080 1160
head(unique(sort(mailles$LAMBIIX_hm)) %% 10)
[1] 0 0 0 0 0 0
```

POUR les longitudes coordonnées des centres des mailles IFN sont en 2 modulo 10 et celles de SAFRAN en 0 modulo 10. Les coordonnées des coins des mailles SAFRAN sont à  $\pm 40$  hm des centres, donc toujours en 0 modulo 10. Les coordonnées des coins des mailles IFN sont à  $\pm 5$  hm des centres, donc en 7 modulo 10, d'où le décalage horizontal de 3 hm.

```
head(unique(sort(round(plachene$YLII_m/100))), n = 10)
[1] 16216 16236 16246 16256 16276 16286 16296 16316 16326 16336
```

```
head(unique(sort(round(plachene$YLII_m/100))) % 10, n = 10)
[1] 6 6 6 6 6 6 6 6 6 6
head(unique(sort(mailles$LAMBIIY_hm)))
[1] 17050 17130 17210 17290 17370 17450
head(unique(sort(mailles$LAMBIIY_hm)) % 10)
[1] 0 0 0 0 0 0
```

POUR les latitudes les coordonnées des centres des mailles IFN sont en 6 modulo 10 et celles de SAFRAN en 0 modulo 10. Les coordonnées des coins des mailles SAFRAN sont à  $\pm 40$  hm des centres, donc toujours en 0 modulo 10. Les coordonnées des coins des mailles IFN sont à  $\pm 5$  hm des centres, donc en 1 modulo 10, d'où le décalage vertical de 1 hm.

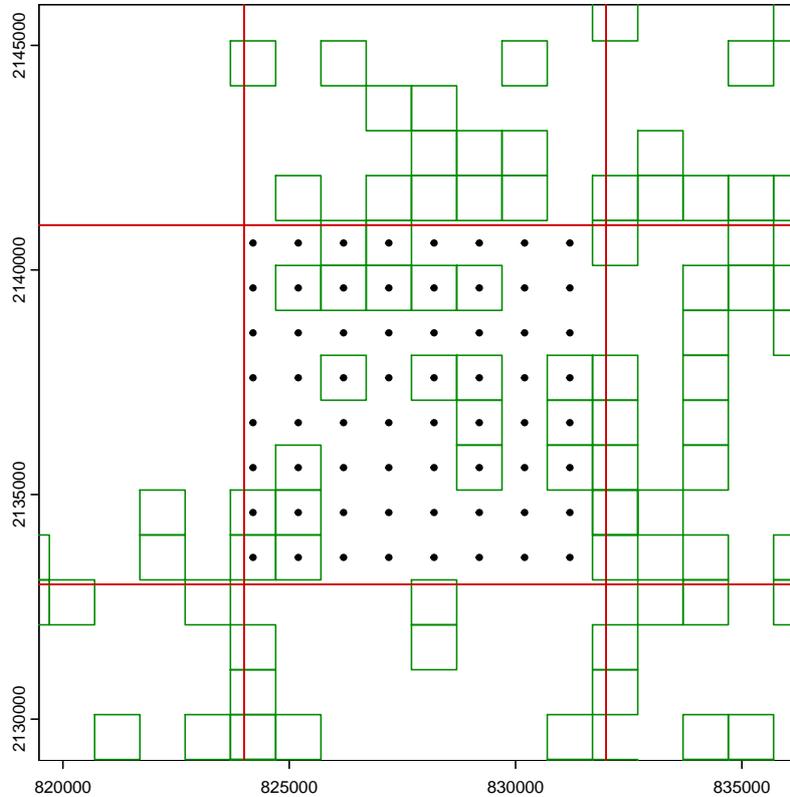
### 3.2.2 Génération de toutes les mailles IFN

JE commence par calculer les centres des grilles IFN en LAMBERT II. Pour ce faire je considère toutes les mailles SAFRAN et je génère à chaque fois les 64 mailles IFN correspondantes.

```
cnames <- c("XLII_m", "YLII_m", "IDij", "IDk1") ; nc <- length(cnames)
nall <- 64*nrow(mailles) # 541 568
allIFN <- as.data.frame(matrix(NA, nrow = nall, ncol = nc))
colnames(allIFN) <- cnames
allIFN$IDij <- rep(mailles$IDij, each = 64)
for(i in seq(1, nall, by = 64)){
  imaille <- which(mailles$IDij == allIFN[i, "IDij"])
  xlon_m <- mailles[imaille, "LAMBIIIX_hm"]*100
  ylon_m <- mailles[imaille, "LAMBIIY_hm"]*100
  ii <- 0
  for(j in seq_len(8)){
    for(k in seq_len(8)){
      allIFN[i + ii, "IDk1"] <- paste0(j, "-", k)
      allIFN[i + ii, "XLII_m"] <- xlon_m - 4000 + 200 + (j - 1)*1000
      allIFN[i + ii, "YLII_m"] <- ylon_m - 4000 + 600 + (k - 1)*1000
      ii <- ii + 1
    }
  }
}
save(allIFN, file = "TCACPS/allIFN4var.Rda")
```

JE fais une petite vérification pour la maille SAFRAN de la forêt de Seillon en posant un point au centre des mailles IFN générées. Il y en a bien 64 et elles sont bien au centre des carrés des mailles IFN.

```
load("TCACPS/allIFN4var.Rda")
terra::plot(FR2LII, xlim = c(819790.9, 835975.1), ylim = c(2145581, 2129397))
terra::lines(IFNgridLII, col = "green4")
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII)
with(subset(allIFN, IDij == "95-65"), points(XLII_m, YLII_m, pch = 19, cex = 0.5))
```

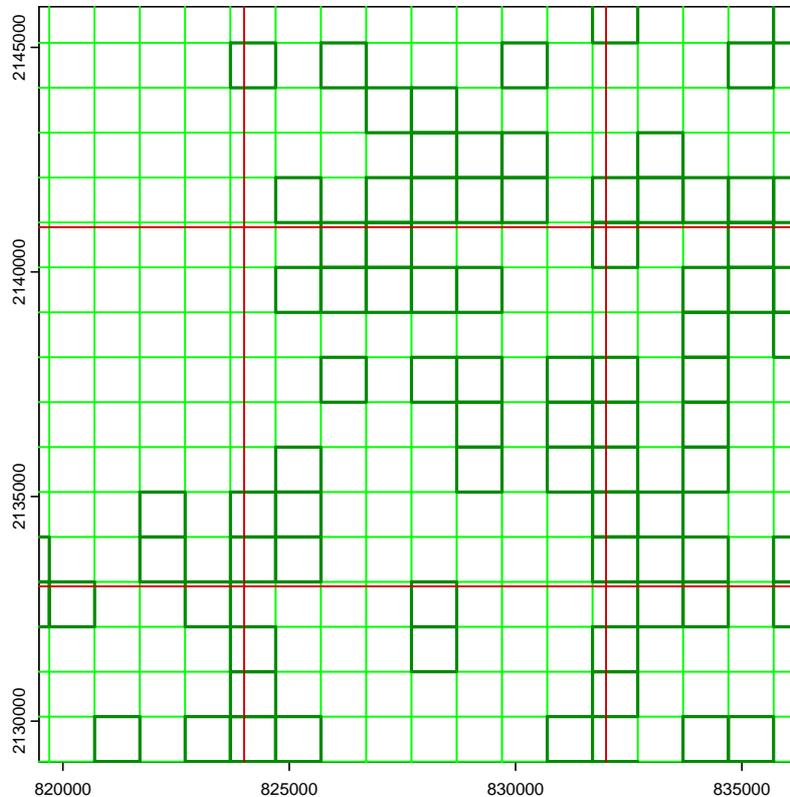


Je peux générer maintenant toutes les carrés des mailles IFN possibles.

```
n <- nrow(allIFN)
lon <- rep(allIFN[1:n, "XLII_m"], each = 4)
lon <- lon + 500*c(-1, +1, +1, -1)
lat <- rep(allIFN[1:n, "YLII_m"], each = 4)
lat <- lat + 500*c(+1, +1, -1, -1)
lonlat <- cbind(id = rep(1:n, each = 4), part = 1, lon, lat)
allIFNgridLII <- terra::vect(lonlat, type = "polygons", crs = "EPSG:27572")
```

PETITE vérification ultime avec en vert clair toutes les grilles IFN possibles et en vert foncé les mailles IFN documentées :

```
terra::plot(FR2LII, xlim = c(819790.9, 835975.1), ylim = c(2145581, 2129397))
terra::lines(allIFNgridLII, col = "green")
terra::lines(IFNgridLII, col = "green4", lwd = 2)
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII)
```



### 3.2.3 Calcul du taux de couvert arborée des mailles IFN

LES données<sup>4</sup> satellitaires proviennent du service Copernicus de surveillance des terres de l'union européenne (CORINE Land Cover : CLC<sup>5</sup>) fournit des informations sur le taux de couvert arborée (TCD : *Tree Cover Density*) à une résolution de  $100 \times 100$  m et varie de 0 % (zones sans arbres) à 100 % (zones entièrement couvertes d'arbres). Ce sont des données déjà agrégées à partir d'images satellites d'une résolution de  $10 \times 10$  m. La densité du couvert forestier est définie comme la projection verticale des cimes des arbres sur une surface terrestre horizontale. La production de cet ensemble de données a été coordonnée par l'Agence européenne pour l'environnement (AEE) dans le cadre du programme européen Copernicus. Les coordonnées sont en EPSG :3035 :

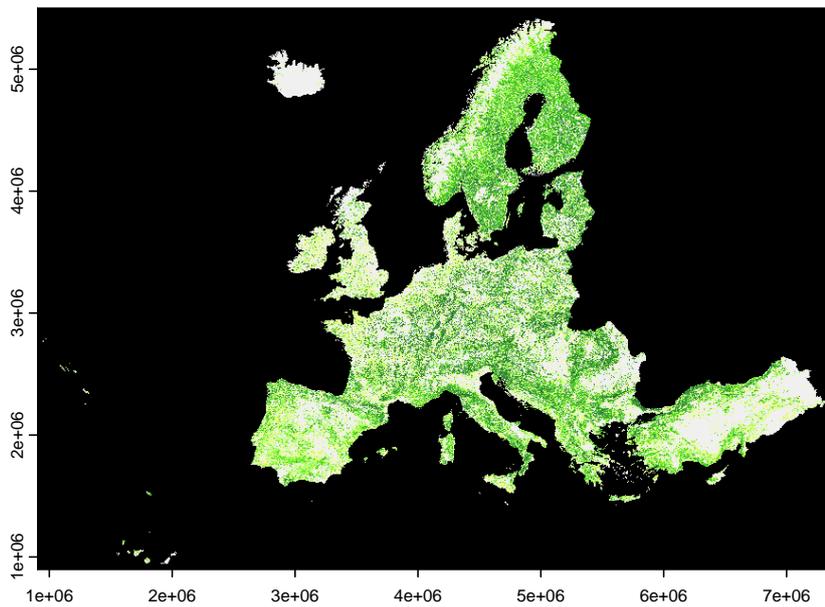
```
tcd <- terra::rast("TCACPS/TCD_2018_100m_eu_03035_V2_0.tif")
tcd
class      : SpatRaster
dimensions : 46000, 65000, 1 (nrow, ncol, nlyr)
resolution : 100, 100 (x, y)
extent     : 9e+05, 7400000, 9e+05, 5500000 (xmin, xmax, ymin, ymax)
coord. ref.: ETRS89-extended / LAEA Europe (EPSG:3035)
```

<sup>4</sup><https://doi.org/10.2909/c7bf34ea-755c-4dbd-85b6-4efc5fd302a2>

<sup>5</sup><https://land.copernicus.eu/en/products/corine-land-cover>

```
source      : TCD_2018_100m_eu_03035_V2_0.tif
color table : 1
categories  : Class_Name, AREA_KM2, AREA_PERC
name        : Class_Name
min value   : all non-tree covered areas
max value   : outside area
```

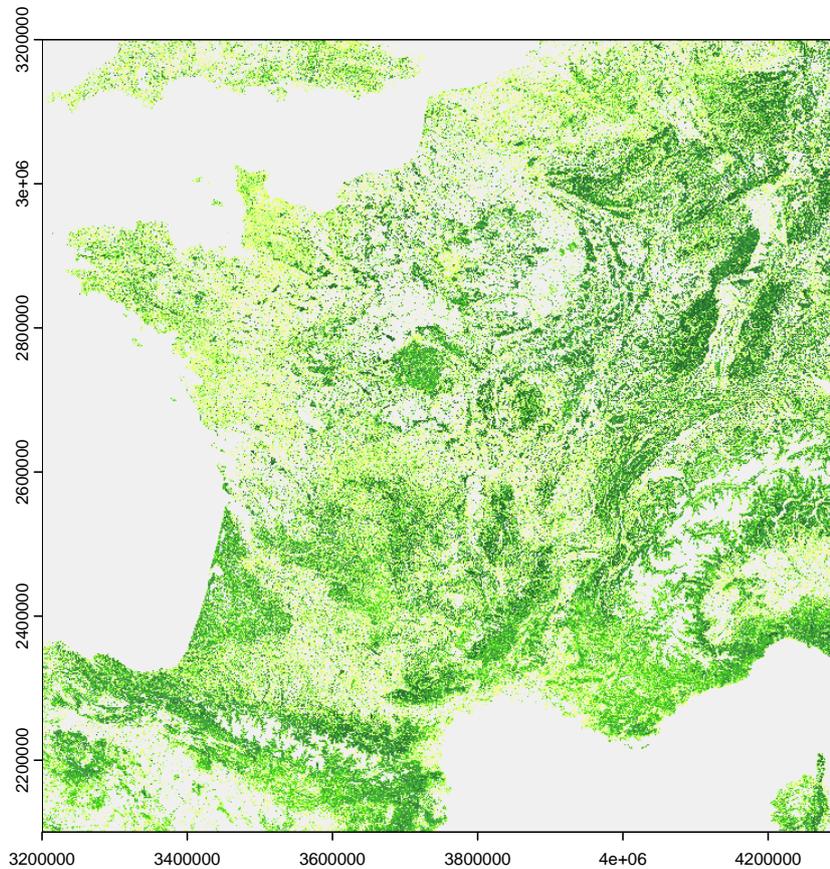
```
terra::plot(tcd, legend = FALSE)
```



COMME ce sont des données assez volumineuses, j'ai commencé par extraire avec `terra::crop()` la région de la FRANCE continentale qui m'intéresse :

```
e <- terra::ext(3.2*10^6, 4.3*10^6, 2.1*10^6, 3.2*10^6)
tcdF <- terra::crop(tcd, e)
terra::writeRaster(tcdF, "TCACPS/tcdF.tif", overwrite = TRUE,
                  datatype = "INT1U",
                  filetype = "GTiff",
                  NAflag = NA,
                  gdal = c("COMPRESS=DEFLATE", "TFW=YES"))
```

```
tcdF <- terra::rast("TCACPS/tcdF.tif")
terra::plot(tcdF, legend = FALSE)
```

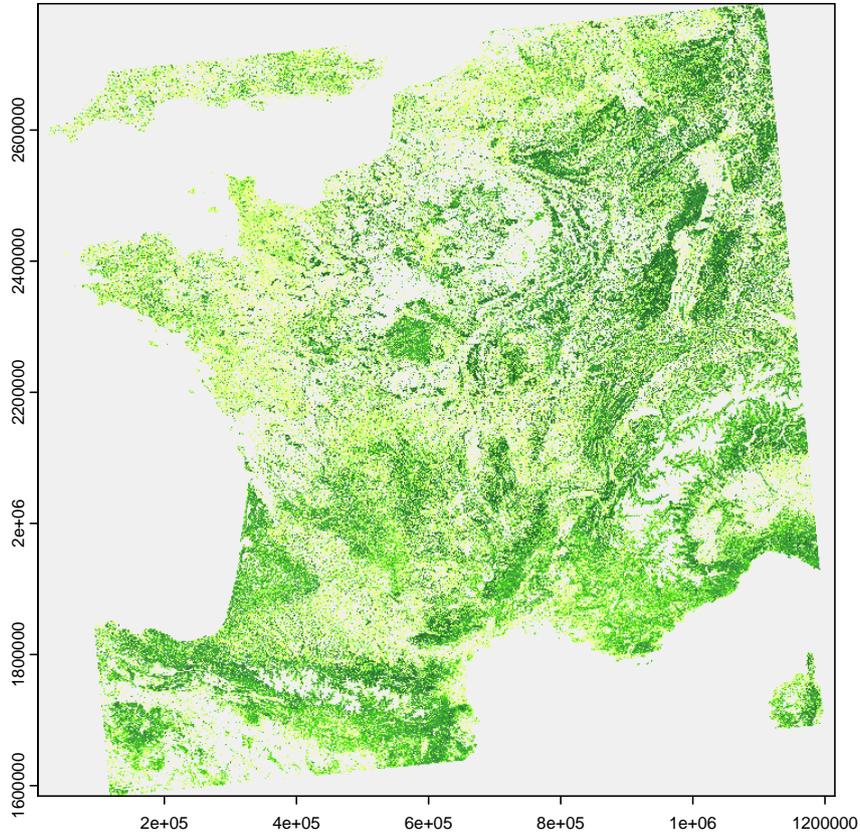


Je passe les coordonnées en LAMBERT II pour être cohérent avec celles des mailles IFN.

```
tcdFLII <- terra::project(tcdF, "EPSG:27572")
terra::writeRaster(tcdFLII, "TCACPS/tcdFLII.tif", overwrite = TRUE,
  datatype = "INT1U",
  filetype = "GTiff",
  NAflag = NA,
  gdal = c("COMPRESS=DEFLATE", "TFW=YES"))
```

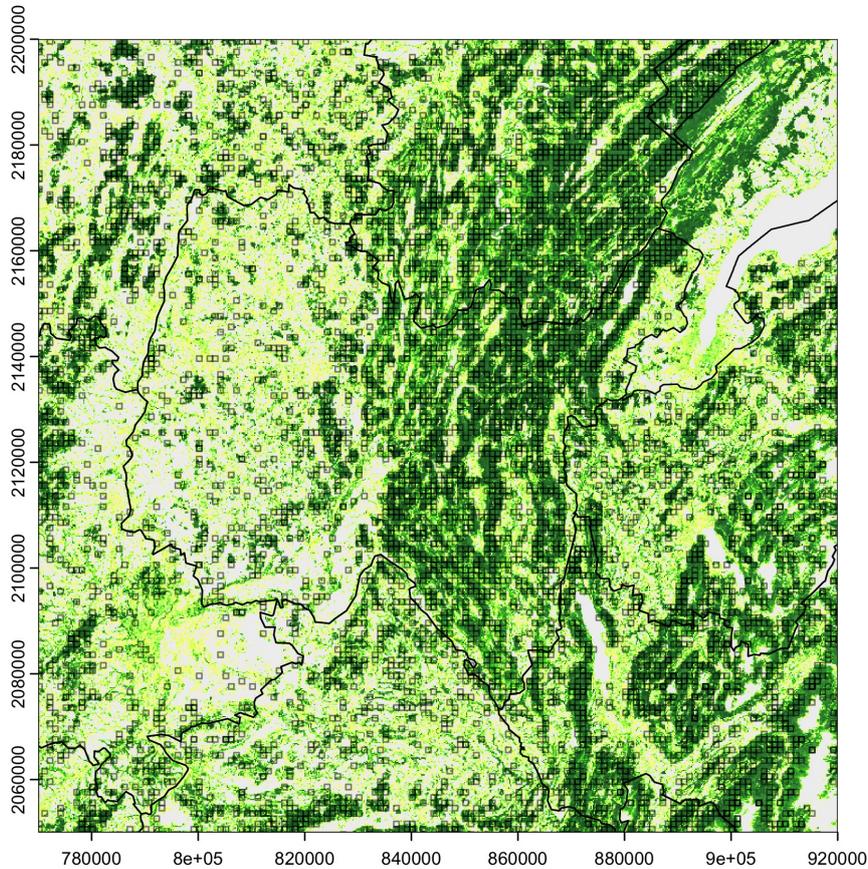
Globalement, je n'ai pas trop de distorsions :

```
tcdFLII <- terra::rast("TCACPS/tcdFLII.tif")
terra::plot(tcdFLII, legend = FALSE)
```



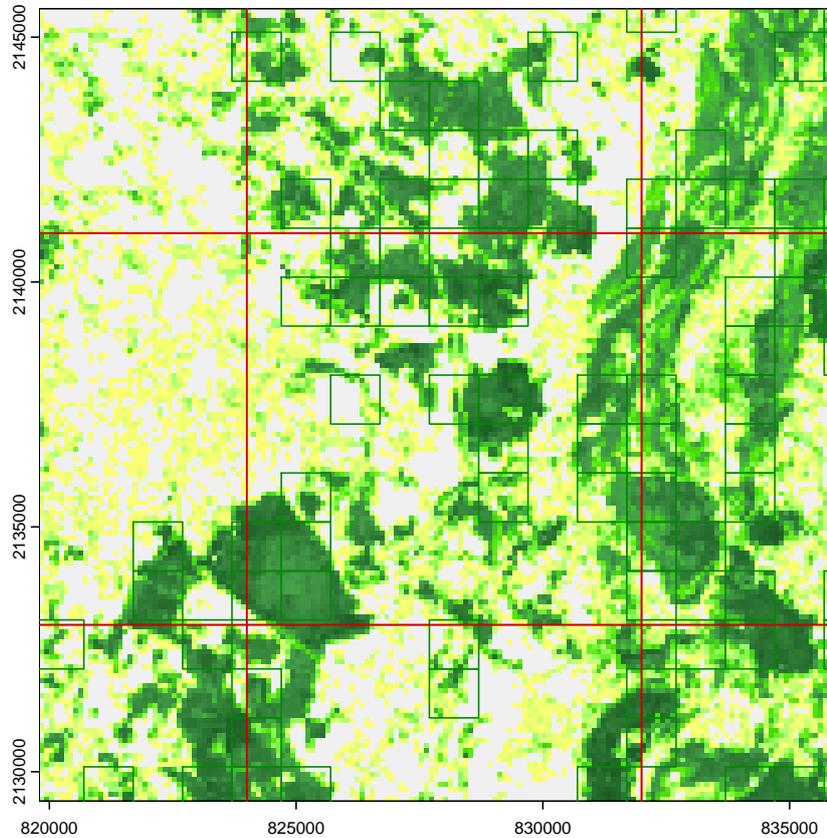
JE fais un zoom sur le département de l'AIN, tout semble être cohérent, on retrouve la plus grande densité de grilles IFN documentées à l'Est du département sur le sud du massif du JURA.

```
terra::plot(tcdFLII, xlim = 10^6*c(0.77, 0.92), ylim = 10^6*c(2.05, 2.2), legend = FALSE)
terra::lines(IFNgridLII, col = rgb(0,0,0,0.5))
terra::lines(FR2LII)
```



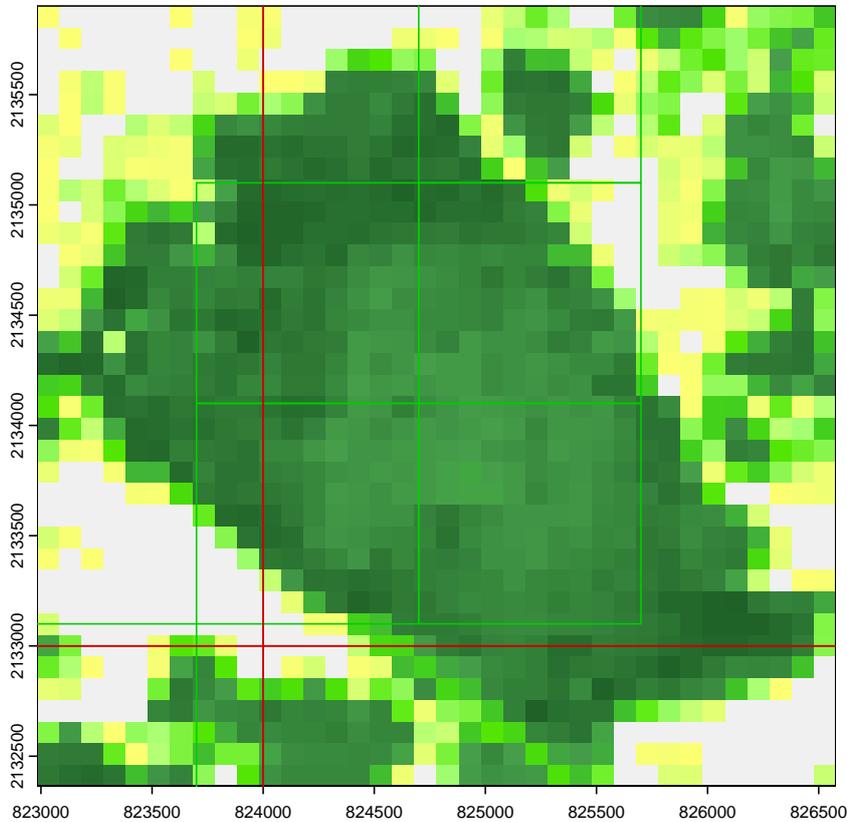
JE zoome encore sur la maille SAFRAN qui contient la forêt domaniale de SEILLON, c'est parfait, on reconnaît bien (pour qui connaît) la forme caractéristique de la forêt de SEILLON en bas à gauche de la maille.

```
terra::plot(tcdFLII, xlim = c(819790.9, 835975.1), ylim = c(2145581, 2129397), legend = FALSE)
terra::lines(IFNgridLII, col = "green4")
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII)
```



Je zoome encore pour mieux voir les pixels. L'alignement entre les pixels CLC et les mailles IFN n'est pas parfait, mais pas si mal que ça. Il y a bien  $10 \times 10$  pixels par maille IFN comme attendu. Cela peut sembler un peu pixelisé ainsi, mais il ne faut pas oublier que chaque pixel ici est le résumé de 100 pixels de  $10 \times 10$  m, voir la figure 3 page 31 pour visualiser les données originelles.

```
terra::plot(tcdFLII, xlim = c(822982.4, 826576.7), ylim = c(2135902, 2132365), legend = FALSE)
terra::lines(IFNgridLII, col = "green3")
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII)
```



NOUS pouvons maintenant le taux de couverture arborée moyen pour toutes les mailles IFN possibles. C'est la fonction `terra::extract()` qui va faire le travail pour nous. Il faut lui préciser avec l'argument `fun` comment on veut calculer la moyenne. Je commence par exclure toutes les valeurs non comprises entre 0 et 100 % (la valeur 255 code pour les valeurs non documentées). Je vérifie aussi que mon vecteur n'est pas vide pour éviter que la fonction `mean()` ne me renvoie un `NaN`. Je complète la table `allIFN` avec la colonne `TCD` pour le taux de couvert arborée moyen de la maille IFN.

```
moy <- function(x){
  x[x > 100] <- NA # outside area
  x <- x[!is.na(x)]
  if(length(x) == 0L) return(0) # Pas de données
  return(mean(x))
}
allIFNtcd <- terra::extract(tcdFLII, allIFNgridLII, fun = moy)
allIFN$TCD <- allIFNtcd[, 2]
save(allIFN, file = "TCACPS/allIFN_TCD.Rda")

load("TCACPS/allIFN_TCD.Rda")
summary(allIFN$TCD)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.00   5.11   16.52   25.11  40.40   99.08
```

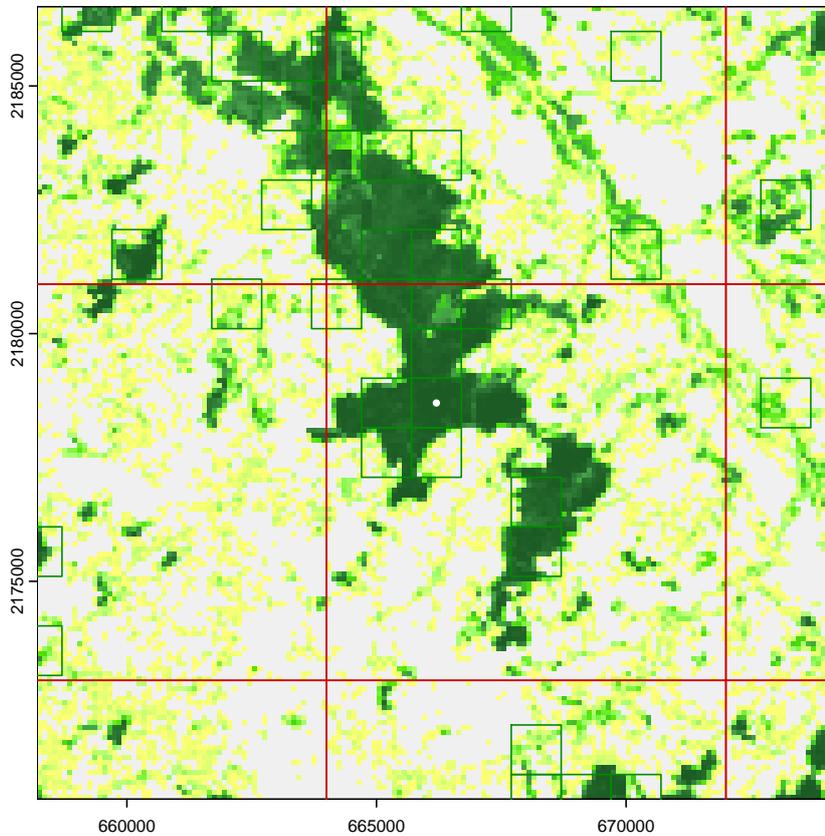
ON est quand même curieux de savoir où est la maille IFN avec le record de taux de couverture arborée avec son 99 %. C'est au Nord du département



FIGURE 3 : Cette image de la forêt de Seillon avec des pixels de  $10 \times 10$  m donne une idée de la résolution originelle des données satellitaires utilisées pour estimer le taux de couverture arborée.

de l'ALLIER, au Nord-Ouest de MOULINS, dans la forêt domaniale des PRIEURÉ-BAGNOLET. Dans la carte ci-dessous c'est la maille IFN au centre avec le point blanc, elle est effectivement bien verte.

```
imax <- which.max(allIFN$TCD)
allIFN[imax, ]
      XLII_m YLII_m IDij IDk1 TCD
326102 666200 2178600 75-70 3-6 99.08
xx <- allIFN[imax, "XLII_m"] ; yy <- allIFN[imax, "YLII_m"]
delta <- 8000
FR2 <- terra::vect("TCACPS/FR2.shp")
FR2LII <- terra::project(FR2, "EPSG:27572")
terra::plot(tcdFLII, xlim = c(xx - delta, xx + delta),
            ylim = c(yy - delta, yy + delta), legend = FALSE)
terra::lines(IFNgridLII, col = "green4")
terra::lines(safgridLII, col = "red3")
terra::lines(FR2LII, lwd = 2)
points(xx, yy, pch = 19, col = "white", cex = 0.5)
```



### 3.2.4 Projection coordonnées Lambert 93 et GPS

CETTE section ne sert pas à grand chose, mais j'avais envie d'avoir dans un coin un exemple complet de code `R` pour changer de système de coordonnées en utilisant uniquement le paquet `terra` [8]. Il y a sans doute plus élégant. Les EPSG utilisés sont décrits dans la section 3.4 page 36.

```
ptsLII <- with(allIFN, terra::vect(cbind(XLII_m, YLII_m), crs = "EPSG:27572"))
ptsGPS <- terra::project(ptsLII, "EPSG:4326")
allIFN$XGPS <- terra::geom(ptsGPS)[, "x"]
allIFN$YGPS <- terra::geom(ptsGPS)[, "y"]
ptsL93 <- terra::project(ptsLII, "EPSG:9794")
allIFN$XL93 <- terra::geom(ptsL93)[, "x"]
allIFN$YL93 <- terra::geom(ptsL93)[, "y"]
save(allIFN, file = "TCACPS/allIFN8var.Rda")
```

```
load("TCACPS/allIFN8var.Rda")
head(allIFN)
```

	XLII_m	YLII_m	IDij	IDkl	TCD	XGPS	YGPS	XL93	YL93
1	72200	2397600	1-98	1-1	0	-4.794015	48.36167	123631.5	6835290
2	72200	2398600	1-98	1-2	0	-4.795238	48.37063	123639.4	6836290
3	72200	2399600	1-98	1-3	0	-4.796461	48.37958	123647.4	6837289
4	72200	2400600	1-98	1-4	0	-4.797684	48.38853	123655.4	6838288
5	72200	2401600	1-98	1-5	0	-4.798908	48.39749	123663.3	6839288
6	72200	2402600	1-98	1-6	0	-4.800132	48.40644	123671.3	6840287

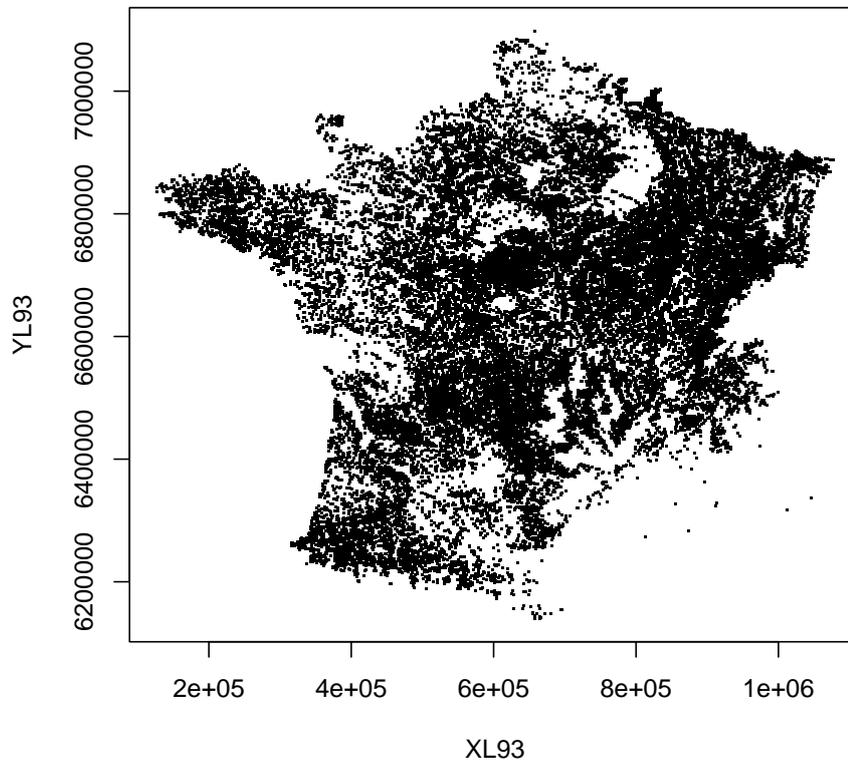
### 3.2.5 Rattachement des données des points d'inventaire IFN

ON veut compléter la table `allIFN` avec les données des mailles IFN documentées. La variable booléenne `dataIFN` sera un indicateur de la présence de données. La variable `TCA_CPS` donne le taux de couverture du point d'inventaire en chênes sessiles et pédonculés. On se base sur les coordonnées des mailles en LAMBERT II arrondies à l'hectomètre pour faire les rattachements.

```
load("TCACPS/allIFN8var.Rda")
load("TCACPS/plachene11var.Rda")
allIFN$dataIFN <- FALSE # Faux par défaut
plachene$found <- TRUE # Vrai par défaut
for(i in seq_len(nrow(plachene))){
  the_X <- round(plachene[i, "XLII_m"]/100)
  the_Y <- round(plachene[i, "YLII_m"]/100)
  the_IDij <- plachene[i, "IDij"]
  # Identification via les coordonnées
  ii <- which(round(allIFN$XLII_m/100) == the_X &
             round(allIFN$YLII_m/100) == the_Y)
  # Cas où la maille IFN est trouvée
  if(length(ii) == 1L){
    # Sanity check sur l'identité de la maille SAFRAN
    stopifnot(allIFN[ii, "IDij"] == the_IDij)
    allIFN[ii, "dataIFN"] <- TRUE
    allIFN[ii, "TCA_CPS"] <- plachene[i, "TCA_CPS"]
    plachene[i, "found"] <- TRUE
  } else {
    # Cas où la maille IFN n'est pas trouvée
    plachene[i, "found"] <- FALSE
  }
}
save(plachene, file = "TCACPS/plachene_COORD.Rda")
save(allIFN, file = "TCACPS/allIFN_COORD.Rda")
```

ON vérifie avec une petite carte en présence-absence qu'il n'y a pas de bug majeur ici.

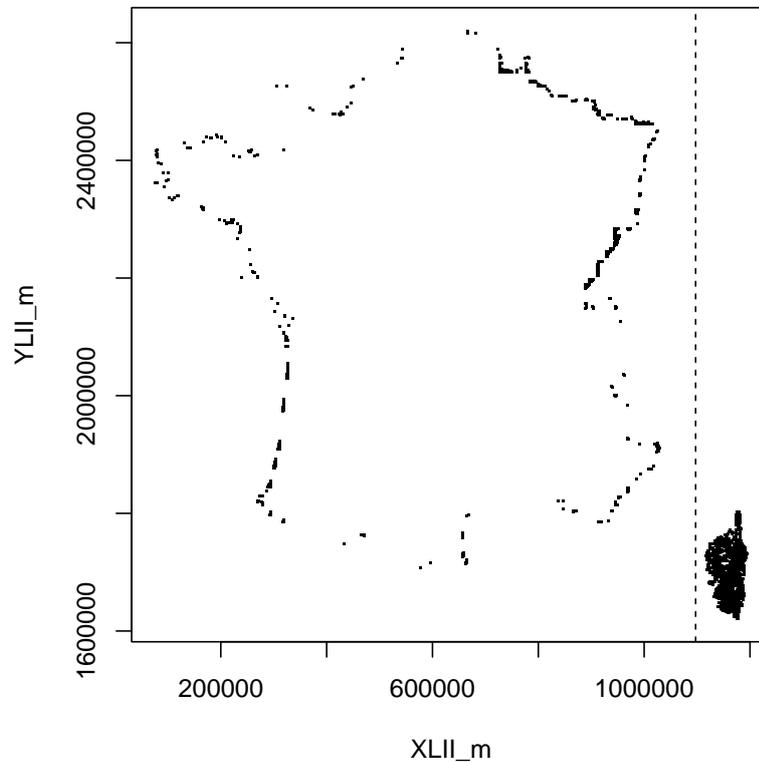
```
load("TCACPS/allIFN_COORD.Rda")
with(subset(allIFN, dataIFN & TCA_CPS > 0), plot(XL93, YL93, pch = "."))
```



### 3.2.6 Les mailles IFN perdues

NOUS avons perdu 2633 mailles IFN dans la bataille, dont 1702 en CORSE et 931 en FRANCE continentale. Pour les mailles en CORSE c'est normal puisque dans `allIFN` nous n'avons généré que les mailles IFN en FRANCE continentale. En FRANCE continentale ce sont des mailles IFN périphériques qui tombent en dehors d'une maille SAFRAN. Elle ne représentent que 0.82 % des mailles continentales.

```
load("TCACPS/plachene_COORD.Rda")
par(pty="s")
with(subset(plachene, found == FALSE), {
  plot(XLII_m, YLII_m, pch = ".")
  abline(v = 1097252, lty = 2)
})
```



```
with(subset(plachene, found == FALSE), sum(!found))
[1] 2633
with(subset(plachene, found == FALSE & XLII_m > 1097252), sum(!found))
[1] 1702
with(subset(plachene, found == FALSE & XLII_m <= 1097252), sum(!found))
[1] 931
with(subset(plachene, XLII_m <= 1097252), length(XLII_m))
[1] 113274
```

### 3.3 Paquets utiles

LES paquets  permettant de faire des représentations cartographiques font parti d'un écosystème riche et évolutif. Le paquet `sf` [10, 11] devrait remplacer le paquet `sp` [12, 4] à terme. Le paquet `terra` [8] est le successeur du paquet `raster` [7] et a une très bonne documentation en ligne<sup>6</sup>. J'ai essayé de n'utiliser que le paquet `terra` ici. Les données sur les frontières administratives ont été récupérées avec le paquet `geodata` [9].

<sup>6</sup><https://rspatial.org/>

UNE possibilité intéressante est d'utiliser la fonction `terra::plet()` pour produire une carte interactive prise en charge par le paquet `leaflet` [5]. Attention, pour être compatible avec les fonds de cartes importés d'`OpenStreetMap` il faut d'abord passer en coordonnées GPS. Le code suivant a été utilisé pour produire la figure 2 page 5.

```
IFNgripGPS <- terra::project(IFNgridLII, "EPSG:4326")
m <- terra::plet(IFNgripGPS, col = "transparent", border = "green")
library(leaflet)
m
```

### 3.4 Les EPSG utiles

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. Il peuvent être définis dans `terra` avec une chaîne de caractères répondant à une syntaxe précise<sup>7</sup>. 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). Il suffit d'utiliser par exemple la notation "EPSG:4326" pour faire référence au système de projection utilisé par les GPS. Voici la liste des codes EPSG utilisés ici.

**4326** : WGS 84 - WGS84 - World Geodetic System 1984. C'est celui utilisé par les GPS. Il est incontournable et on en aura besoin tôt où tard. Par exemple ici pour récupérer le fond de carte de la figure 2 page 5.

**3035** : ETRS89-extended / LAEA Europe. C'est celui utilisé par l'agence environnementale européenne [13] pour la diffusion des données satellitaires sur le taux de couvert arborée.

**27572** : NTF (Paris) / LAMBERT II Carto (Centre) dit LAMBERT II étendue. C'est celle qui est utilisée par SAFRAN. Mais attention dans les fichiers SAFRAN ils sont donnés en hectomètres.

**2154** : RGF93 v1 / LAMBERT 93 - France. Ancienne projection LAMBERT 93 utilisée au XX<sup>e</sup> siècle.

**9794** : RGF93 v2b / LAMBERT 93 - France. C'est la projection officielle actuelle pour la France métropolitaine. C'est celle utilisée par l'IFN.

## Références

- [1] Anonymous. Documentation des données brutes de l'inventaire forestier mises en ligne sur DATAIFN. fichier : `COUVERT.csv`. Technical report, Service de l'information statistique forestière et environnementale and Institut National de l'Information Géographique et Forestière, 2022. 3 pp. Version 2.0.

<sup>7</sup>Type : "+proj=lcc +lat\_0=38.9072 +lon\_0=-77.0369 +lat\_1=33 +lat\_2=45 +ellps=GRS80"

- [2] Anonymous. Documentation des données brutes de l'inventaire forestier mises en ligne sur DATAIFN. fichier : `PLACETTE.csv`. Technical report, Service de l'information statistique forestière et environnementale and Institut National de l'Information Géographique et Forestière, 2022. 12 pp. Version 2.0.
- [3] Anonymous. Données brutes de l'inventaire forestier mises en ligne sur DATAIFN. Technical report, Service de l'information statistique forestière et environnementale and Institut National de l'Information Géographique et Forestière, 2022. 8 pp. Version 2.0.
- [4] R.S. Bivand, E. Pebesma, and V. Gomez-Rubio. *Applied spatial data analysis with R, Second edition*. Springer, NY, 2013.
- [5] J. Cheng, B. Schloerke, B. Karambelkar, and Y. Xie. *leaflet : Create Interactive Web Maps with the JavaScript 'Leaflet' Library*, 2024. R package version 2.2.2.
- [6] J.-C. Hervé, S. Wurpillot, C. Vidal, and B. Roman-Amat. L'inventaire des ressources forestières en france : un nouveau regard sur de nouvelles forêts. *Revue forestière française*, 66 :247–260, 2014.
- [7] R.J. Hijmans. *raster : Geographic Data Analysis and Modeling*, 2021. R package version 3.4-8/r3420.
- [8] R.J. Hijmans. *terra : Spatial Data Analysis*, 2024. R package version 1.7-71.
- [9] R.J. Hijmans, M. Barbosa, A. Ghosh, and A. Mandel. *geodata : Download Geographic Data*, 2023. R package version 0.5-9.
- [10] E. Pebesma. Simple Features for R : Standardized Support for Spatial Vector Data. *The R Journal*, 10(1) :439–446, 2018.
- [11] E. Pebesma and R. Bivand. *Spatial Data Science : With applications in R*. Chapman and Hall/CRC, 2023.
- [12] E.J. Pebesma and R.S. Bivand. Classes and methods for spatial data in R. *R News*, 5(2) :9–13, November 2005.
- [13] H. Peifer. About the EEA reference grid. Technical report, European Environment Agency, 2011. 2 pp.
- [14] J.-P. Vidal, E. Martin, L. Franchistéguy, M. Baillon, and J.-M. Soubeyrou. A 50-year high-resolution atmospheric reanalysis over France with the safran system. *International Journal of Climatology*, 30(11) :1627–1644, 2010.