

Test__data

Valentin Lauret

05/10/2020

Je compare les données SAMM que Matthieu a récemment envoyé, avec les données SAMM que Matthieu a précédemment envoyé et que j'ai utilisé pour l'étude Occupancy.

Les données récentes

```
load("20180914_SAMM_data_LauretValentin.RData")
load("trSAMM.rdata") # donnees d'effort utilisées pour l'occupancy
```

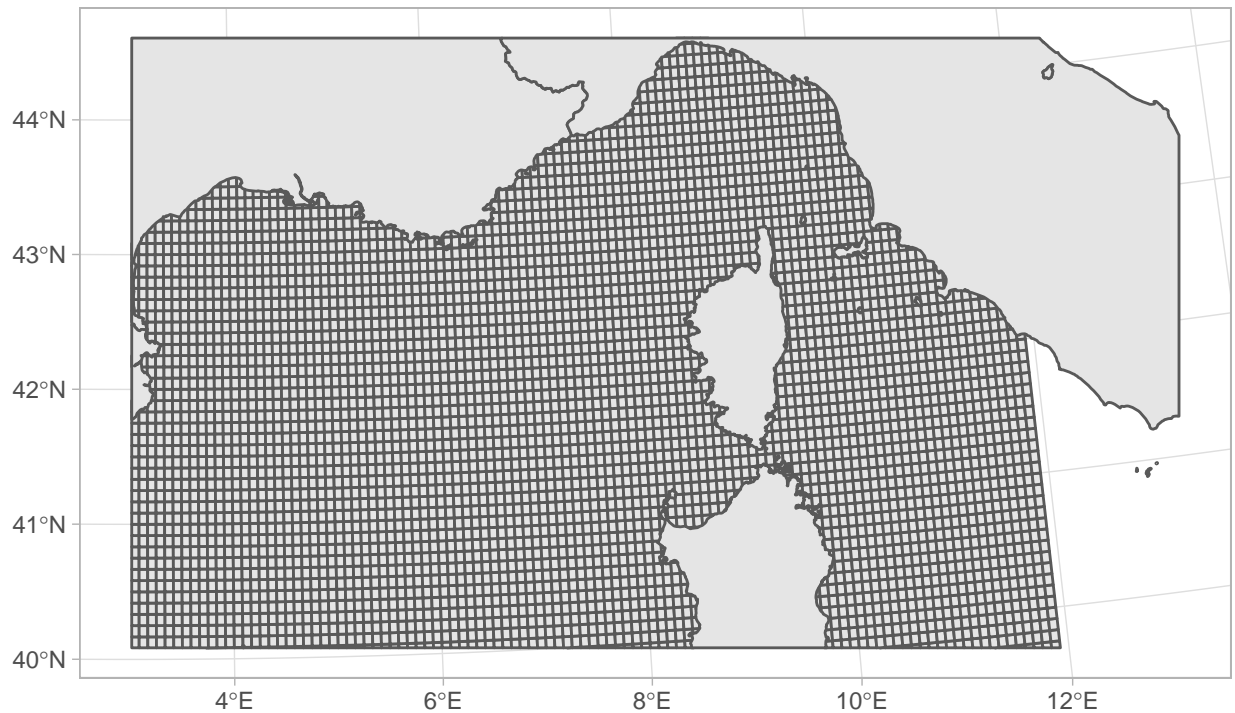
L'effort Les données transect uniquement.

La grille.

```
load("pays.rdata")
grid <- st_read("Grid/grid.shp")
```

```
## Reading layer `grid' from data source `/Users/valentinlauret/Google Drive/These/Work/Multispecies Oc
## Simple feature collection with 4356 features and 3 fields
## geometry type: POLYGON
## dimension: XY
## bbox: xmin: 701000 ymin: 5886622 xmax: 1467639 ymax: 6390000
## epsg (SRID): NA
## proj4string: +proj=lcc +lat_1=44 +lat_2=49 +lat_0=46.5 +lon_0=3 +x_0=700000 +y_0=6600000 +ellps=GRS80
```

```
grid %>%
  ggplot() +
  geom_sf() +
  geom_sf(data= pays)
```

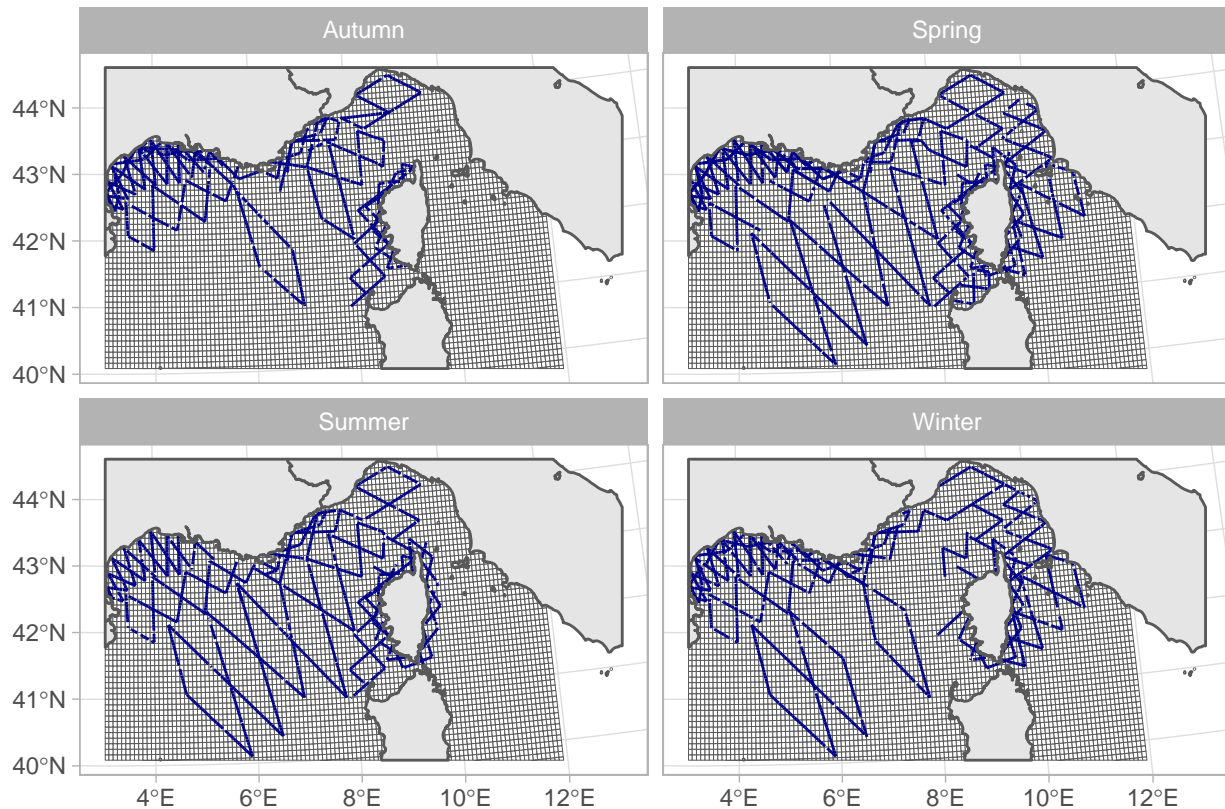


Visualisation des transects par mois. Avec les données qui datent de l'occupancy. Ca a l'air d'être les même transect.

```
load("pays.rdata")

tr2 <- tr %>% mutate(season = case_when(
  m %in% c(1,2,3) ~ "Winter",
  m %in% c(4,5,6) ~ "Spring",
  m %in% c(7,8,9) ~ "Summer",
  m %in% c(10,11,12) ~ "Autumn"
))

grid %>%
  ggplot() +
  geom_sf(fill = "white", lwd = 0.1) +
  geom_sf(data= tr2, color = "darkblue")+
  geom_sf(data= pays) +
  facet_wrap(~season, ncol = 2)
```



Les observations

Les données utilisées pour l'occupancy : 130 detections.

```
load("tt_obs.rdata")
tt_obs %>% filter(suivi == "SAMM") %>% nrow()
```

```
## [1] 130
```

Nombre de cellules de la grille avec detection de dauphin, d'après les données d'occupancy.

```
int <- tt_obs %>% filter(suivi == "SAMM") %>%
  st_intersects(grid) %>% unlist()
length(unique(int))
```

```
## [1] 87
```

Les données fraîches

En prenant en compte la colonne `n` du fichier `$segdata`.

```

dauphins_summer <- summer
dauphins_winter <- winter

transect_summer <- dauphins_summer$segdata %>%
  as_tibble() %>%
  select(date = date,
         transect = Transect.Label,
         eastings = X,
         northings = Y,
         counts = n,
         effort = Effort,
         Sample.Label = Sample.Label) %>%
  add_column(season = "summer")

transect_winter <- dauphins_winter$segdata %>%
  as_tibble() %>%
  select(date = date,
         transect = Transect.Label,
         eastings = X,
         northings = Y,
         counts = n,
         effort = Effort,
         Sample.Label = Sample.Label) %>%
  add_column(season = "winter")

transect <- bind_rows(transect_summer, transect_winter)

transect %>%
  count(transect, wt = counts, sort = TRUE) %>%
  select(n) %>%
  sum()

```

```
## [1] 105
```

Où alors en regardant le fichier \$obsdata

```

## ou
bind_rows(winter$obsdata, summer$obsdata) %>%
  nrow()

```

```
## [1] 105
```

```
# 105 groups
```

Mais si on prend le nombre de lignes avec une detection, on obtient 78.

```

transect %>% filter(counts > 0) %>%
  nrow()

```

```
## [1] 78
```

Si on cherche le nb de cellule avec une detection.

```
int2 <- transect %>% filter(counts > 0) %>%
  st_as_sf(coords = c("eastings", "northings"), crs = st_crs(grid)) %>%
  st_intersects(grid) %>% unlist()

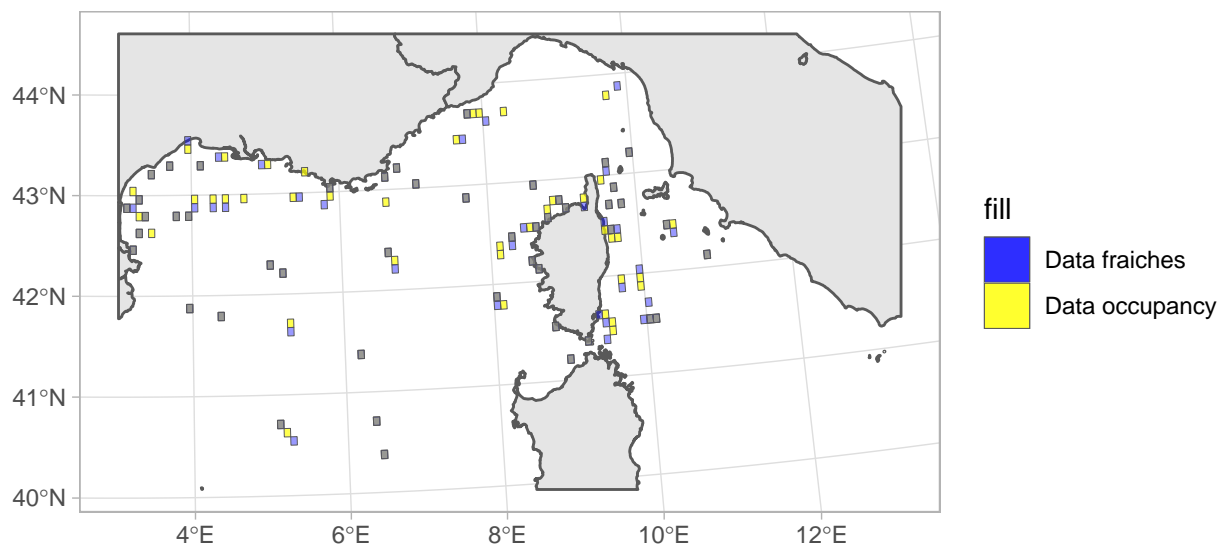
length(unique(int2))
```

```
## [1] 77
```

A partir de là, il y a un soucis... On a 87 cellules avec dauphin pour mes anciennes données et 77 cellules maintenant.

Visualisation

```
pays %>% ggplot() + geom_sf() +
  geom_sf(data = grid[unique(int),], aes(fill = "Data occupancy"), alpha = 0.7, lwd = 0.1)+
  geom_sf(data = grid[unique(int2),], aes(fill = "Data fraiches"), alpha = 0.4, lwd = 0.1)+
  scale_fill_manual(values = c("Data occupancy" = "yellow", "Data fraiches" = "blue"))
```



Il y a quelques données en plus dans le jeu 'occupancy' et un léger décalage sur certains sites...

On va au bout de la construction des matrices pour l'occupancy

Les activités humaines. On fait un `left_join` avec les tables `$segdata` correspondantes à la saison.

```
load("20200928_SAMM_data_Pressure.RData")

#winter
winter_fishingactivities$obsdata$Sample.Label <- as_factor(winter_fishingactivities$obsdata$Sample.Label)

fish_w <- left_join(transect, winter_fishingactivities$obsdata, by = "Sample.Label")

#summer
summer_fishingactivities$obsdata$Sample.Label <- as_factor(summer_fishingactivities$obsdata$Sample.Label)

fish_s <- left_join(transect, summer_fishingactivities$obsdata, by = "Sample.Label")

# bind both
fish <- bind_rows(fish_w, fish_s)
```

Nombre de cellules avec des chalutiers

```
int_f <- fish %>% filter(what == "Bateau chalutier") %>%
  st_as_sf(coords = c("eastings", "northings"), crs = st_crs(grid)) %>%
  st_intersects(grid) %>% unlist()

length(unique(int_f))
```

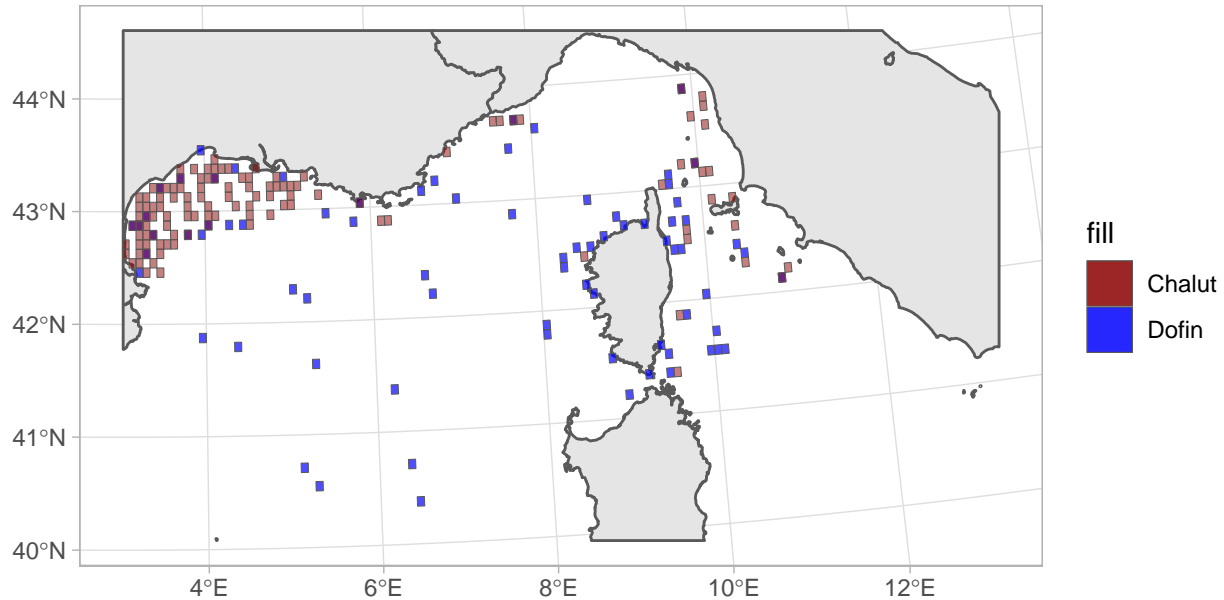
```
## [1] 115
```

Visualisation

```
pays %>% ggplot() + geom_sf() +
  geom_sf(data = grid[unique(int2),], aes(fill = "Dofin"), alpha = 0.7, lwd = 0.1) +
  geom_sf(data = grid[unique(int_f),], aes(fill = "Chalut"), alpha = 0.5, lwd = 0.1) +
  scale_fill_manual(values = c("Dofin" = "blue", "Chalut" = "darkred")) +
  labs(title = "Carte de co-occurrence Tursiops et chalutiers", subtitle = "Données SAMM")
```

Carte de co-occurrence Tursiops et chalutiers

Données SAMM



Construction des chroniques de détection/non-détections des sites

Ajoute le système de coordonnées

```
names(obs)
obs <- activ %>%
  mutate(month = month(date)) %>%
  select(eastings, northings, date, dolphins, peche, what, season, id, transect, month) %>%
  as.data.frame() %>%
  st_as_sf(coords = c("eastings", "northings"),
            crs = st_crs(grid))

obs %>% count(dolphins) # 87 obs de dauphins
```

Effort

Intersection des transects avec la grille et calcul de la longueur de chaque morceau de transect dans les cellules

```
head(tr2)
# intersection of transects with grid cells
rp <- st_intersection(tr2, grid)

rp$length <- st_length(rp)
```

```
head(rp)
```

Il y a plusieurs morceaux de transect par cellule et par occasion. Donc on regroupe tout pour avoir une seule donnée d'effort par cellule et par occasion.

```
rp2 <- rp %>% group_by(objectid, season) %>%  
  summarise(eff = sum(length)) # calculate length transect  
head(rp2)
```

On ajoute chaque valeur d'effort à la cellule correspondante dans la grille.

```
effort <- grid %>% mutate(Autumn = NA,  
                          Winter = NA,  
                          Spring = NA,  
                          Summer = NA)  
  
for(i in 1:nrow(rp2)){  
  index <- which(data$objectid == rp2$objectid[i])  
  effort[index,rp2$season[i]] <- rp2$eff[i]  
}
```

Visualisation

```
p1 <- effort %>%  
  ggplot() +  
  geom_sf(aes(fill = Autumn), lwd = 0.1) +  
  geom_sf(data= pays)  
  
p2 <- effort %>%  
  ggplot() +  
  geom_sf(aes(fill = Winter), lwd = 0.1) +  
  geom_sf(data= pays)  
  
p3 <- effort %>%  
  ggplot() +  
  geom_sf(aes(fill = Spring), lwd = 0.1) +  
  geom_sf(data= pays)  
  
p4 <- effort %>%  
  ggplot() +  
  geom_sf(aes(fill = Summer), lwd = 0.1) +  
  geom_sf(data= pays)  
  
plot_grid(p1,p2,p3,p4, ncol = 2, nrow = 2)
```

Observations

Intersection entre les observations et la grille


```
int <- st_intersection(obs, grid)
head(int)
```

Ajoute les observation à la grille

```
df <- effort %>% mutate(fishing = 0,
                        what = NA,
                        dolphin = 0)

for(i in 1:nrow(int)){
  index <- which(grid$objectid == int$objectid[i])

  df$fishing[index] <- int$peche[i]
  df$what[index] <- int$what[i]
  df$dolphin[index] <- int$dolphins[i]
}

df <- df %>% mutate( obs = case_when(
  dolphin == 0 & fishing == 0 ~ 0,
  dolphin == 1 & fishing == 0 ~ 1,
  dolphin == 0 & fishing == 1 ~ 2,
  dolphin == 1 & fishing == 1 ~ 3,
))

# garde seulement les chalutiers
df <- df %>% mutate( obs = case_when(
  dolphin == 0 & what != "Bateau chalutier" ~ 0,
  dolphin == 1 ~ 1,
  dolphin == 0 & what == "Bateau chalutier" ~ 2,
  dolphin == 1 & what == "Bateau chalutier" ~ 3,
))

df %>% count(obs)
```

Seulement 4 sites avec des observations de dauphin ET de pêche (toute pêche confondue). Et seulement 2 sites avec observations dauphins ET chalutier.

Visualisation

```
df[df$obs>0,] %>% ggplot() + geom_sf(aes(fill = as_factor(obs))) +
  scale_fill_manual(name = "Observation", labels = c("dolphin", "peche", "both"), values = c("#000a39",
  geom_sf(data = pays, lwd = 0.4)
```