Fertility Models Fitted

Loïc Pages

2025-02-28

Introduction

```
rm(list=ls())
library(knitr)
library(spaMM)
```

```
## Registered S3 methods overwritten by 'registry':
##   method               from
##   print.registry_field proxy
##   print.registry_entry proxy
```

```
## spaMM (Rousset & Ferdy, 2014, version 4.5.30) is loaded.
## Type 'help(spaMM)' for a short introduction,
## 'news(package='spaMM')' for news,
## and 'citation('spaMM')' for proper citation.
## Further infos, slides, etc. at https://gitlab.mbb.univ-montp2.fr/francois/spamm-ref.
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(splines)
library(patchwork)
library(SplinesUtils)

setwd("/media/loic/Commun/0Travail/Stage 2025 ISEM/Models")
```

```
centauree_data <- read.csv("donnesIPM_short.csv")
centauree_data_complet <- read.csv("donnesIPM.csv")


#Supprimer plantes dont l'age est inconnu
centauree_data <- centauree_data[!is.na(centauree_data$age0), ]
centauree_data$age1 <- ifelse(centauree_data$Stage1=="V",centauree_data$age0+1,NA)

#Forcer l'age maximal à 8
length(centauree_data$age0[centauree_data$age0 >= 8])
```

```
## [1] 93
```

```
centauree_data$age0[centauree_data$age0 > 8] <- 8

spaMM.options(separation_max=70)
```

```
annees <- 1995:2022
populations <- c("Po","Au","Pe","E1","E2","Cr")
taille_range <- seq(0.5, 25, by = 0.5)
age_range <- 1:8

fake_data <- expand.grid(
  year = annees,
  Pop = populations,
  SizeOMars = taille_range,
  age0 = age_range
)

fake_data <- fake_data %>%
  mutate(Nrw = row_number())
```

BIC

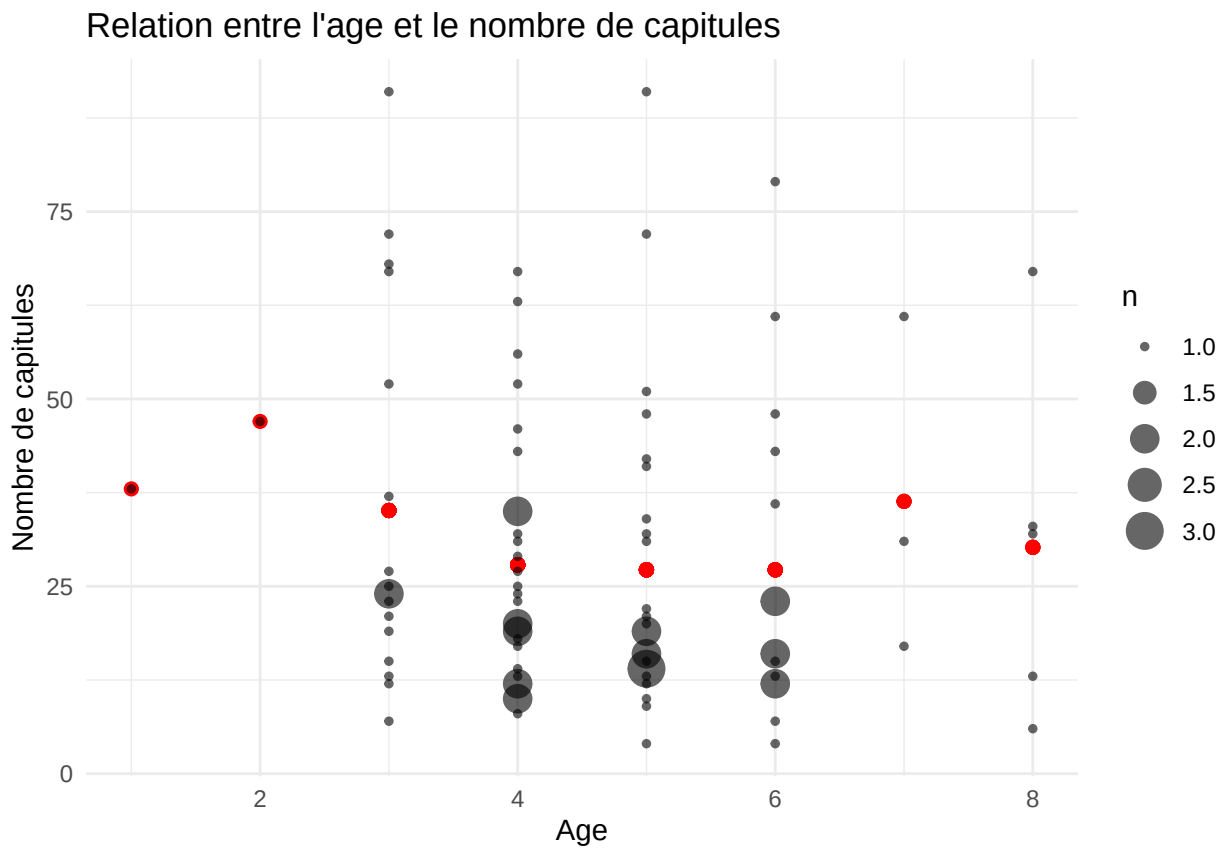
```
extractBIC <- function(fit, n){
  extractAIC(fit)[[2]]+(log(n)-2)*DoF(fit)[[3]]
}
```

Nombre de capitules

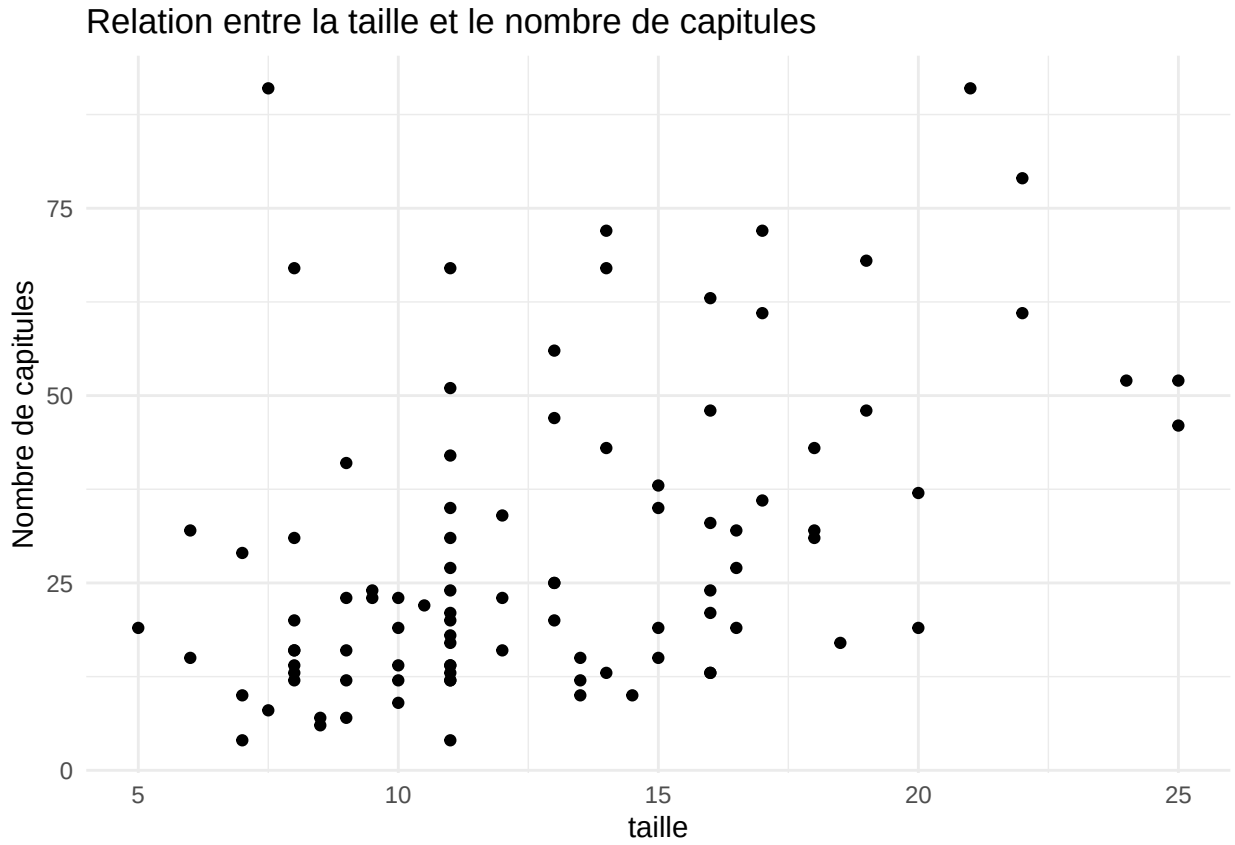
```
cptldata <- centauree_data[!is.na(centauree_data$Cptl0),]
cptldata <- cptldata[!cptldata$Flowering0==0,]
```

```
# Nombre de capitules moyen / age
capidata <- cptldata %>%
  group_by(age0) %>%
  mutate(meancptl=mean(Cptl0))

capidata%>%
```

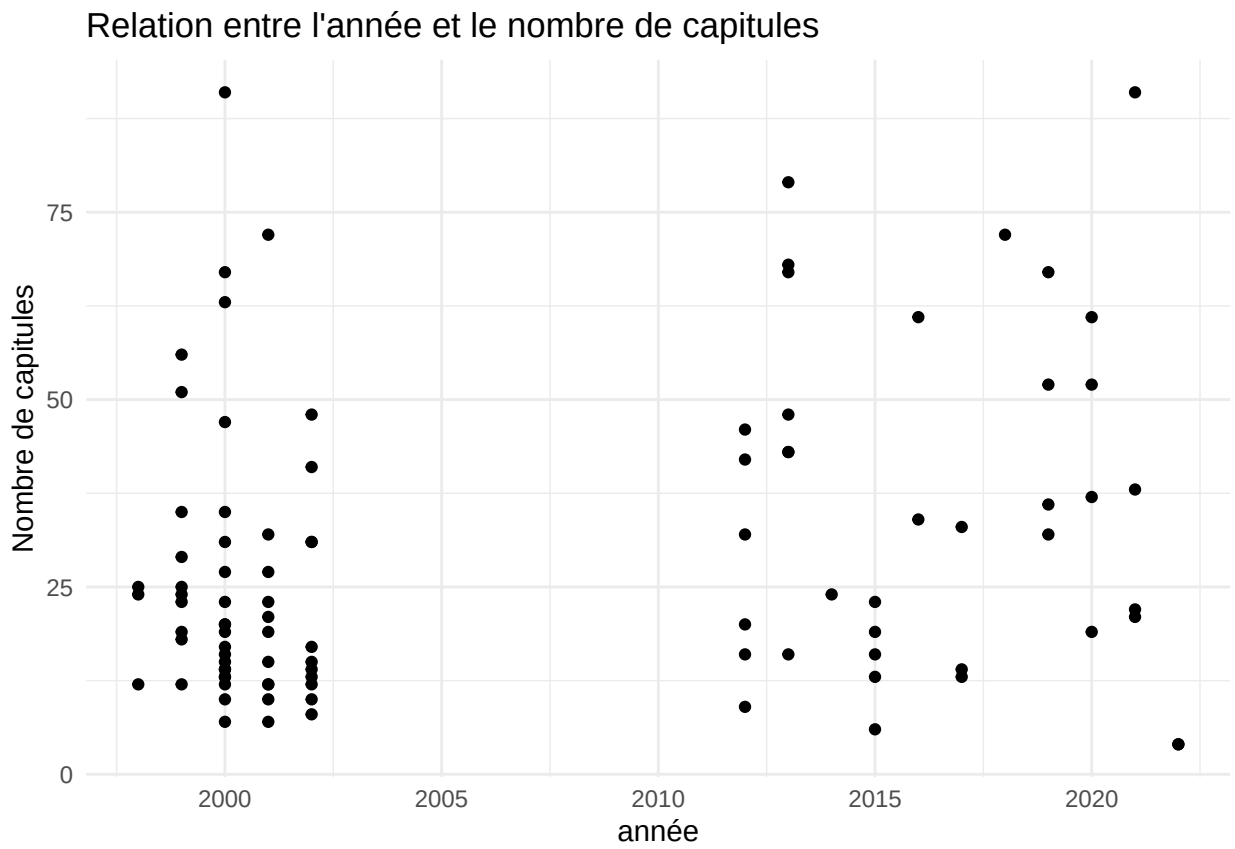
```
  ggplot(aes(x = age0, y = meancptl)) +
  geom_point(color = "red", size = 2) +
  geom_count(aes(y=Cptl0), alpha=0.6) +
  labs(title = "Relation entre l'age et le nombre de capitules",
       x = "Age",
       y = "Nombre de capitules") +
  theme_minimal()
```




```
# Nombre de capitule / taille
cptldata %>%

  ggplot(aes(x=SizeOMars,y=Cptl0))+
  geom_point() +
  labs(title = "Relation entre la taille et le nombre de capitules",
       x = "taille",
       y = "Nombre de capitules") +
  theme_minimal()
```




```
# Nombre de capitule / année
cptldata %>%

  ggplot(aes(x=year,y=Cptl0))+
  geom_point() +
  labs(title = "Relation entre l'année et le nombre de capitules",
       x = "année",
       y = "Nombre de capitules") +
  theme_minimal()
```




```
Cptlglm1 <- fitme(Cptl0 ~ 1 + SizeOMars,
                  data=cptldata)

Cptlglm2 <- fitme(Cptl0 ~ 1 + poly(SizeOMars,2),
                  data=cptldata)

Cptlglm3 <- fitme(Cptl0 ~ 1 + poly(SizeOMars,3),
                  data=cptldata)

Cptlglm4 <- fitme(Cptl0 ~ 1 + SizeOMars +(1|year),
                  data=cptldata)

Cptlglm5 <- fitme(Cptl0 ~ 1 + SizeOMars + age0,
                  data=cptldata)
```

```
n <- length(cptldata$Nrw)
extractAIC(Cptlglm1) ; extractBIC(Cptlglm1, n)
```

```
##      edf      AIC
##   2.0000 825.2362

## [1] 830.344
```

```
extractAIC(Cptlglm2) ; extractBIC(Cptlglm2, n)
```

```
##      edf      AIC
##   3.0000 825.6888
```

```
## [1] 833.3505
```

```
extractAIC(Cptlglm3) ; extractBIC(Cptlglm3, n)
```

```
##      edf      AIC
##   4.0000 826.4869
```

```
## [1] 836.7024
```

```
extractAIC(Cptlglm4) ; extractBIC(Cptlglm4, n)
```

```
##      edf      AIC
##   2.0000 826.8208
```

```
## [1] 831.9286
```

```
extractAIC(Cptlglm5) ; extractBIC(Cptlglm5, n)
```

```
##      edf      AIC
##   3.0000 827.2109
```

```
## [1] 834.8726
```

```
summary(Cptlglm1)
```

```
## formula: Cptl0 ~ 1 + Size0Mars
## ML: Estimation of phi by ML.
##     Estimation of fixed effects by ML.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)    2.951   5.6590  0.5214
## Size0Mars      2.074   0.4166  4.9795
##  -------------- Residual variance  ------------
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)    5.786   0.1451
## Estimate of phi=residual var:  325.6
##  ------------- Likelihood values  -------------
##                         logLik
## logL               : -409.6181
```

```
summary(Cptlglm2)
```

```
## formula: Cptl0 ~ 1 + poly(SizeOMars, 2)
## ML: Estimation of phi by ML.
##     Estimation of fixed effects by ML.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##                      Estimate Cond. SE t-value
## (Intercept)             29.58    1.836  16.108
## poly(SizeOMars, 2)1     89.85   17.898   5.020
## poly(SizeOMars, 2)2     22.36   17.898   1.249
##  -------------- Residual variance  ------------
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)    5.769   0.1451
## Estimate of phi=residual var:  320.3
##  ------------- Likelihood values  -------------
##                          logLik
## logL               : -408.8444
```

```
summary(Cptlglm3)
```

```
## formula: Cptl0 ~ 1 + poly(SizeOMars, 3)
## ML: Estimation of phi by ML.
##     Estimation of fixed effects by ML.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##                      Estimate Cond. SE t-value
## (Intercept)             29.58    1.825  16.210
## poly(SizeOMars, 3)1     89.85   17.785   5.052
## poly(SizeOMars, 3)2     22.36   17.785   1.257
## poly(SizeOMars, 3)3    -19.56   17.785  -1.100
##  -------------- Residual variance  ------------
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)    5.757   0.1451
## Estimate of phi=residual var:  316.3
##  ------------- Likelihood values  -------------
##                          logLik
## logL               : -408.2434
```

```
summary(Cptlglm4)
```

```
## formula: Cptl0 ~ 1 + SizeOMars + (1 | year)
## Estimation of fixed effects by ML.
## Estimation of lambda and phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)    4.198   5.9466  0.7059
## SizeOMars      2.010   0.4253  4.7263
```

```
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##             --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  19.26
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 308.566
##  ------------- Likelihood values  -------------
##                          logLik
## logL       (p_v(h)): -409.4104
```

```
summary(Cptlglm5)
```

```
## formula: Cptl0 ~ 1 + Size0Mars + age0
## ML: Estimation of phi by ML.
##     Estimation of fixed effects by ML.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)   1.7588   9.3904  0.1873
## Size0Mars     2.0883   0.4257  4.9059
## age0          0.2176   1.3685  0.1590
##  -------------- Residual variance  ------------
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)    5.785   0.1451
## Estimate of phi=residual var:  325.5
##  ------------- Likelihood values  -------------
##                          logLik
## logL               : -409.6055
```

```
Cptlpredict1 <- predict(Cptlglm1, newdata = fake_data)[,1]
Cptlpredict2 <- predict(Cptlglm2, newdata = fake_data)[,1]
Cptlpredict3 <- predict(Cptlglm3, newdata = fake_data)[,1]
Cptlpredict4 <- predict(Cptlglm4, newdata = fake_data)[,1]
Cptlpredict5 <- predict(Cptlglm5, newdata = fake_data)[,1]
```

```
plot_capitule <- function(data = fake_data, prediction, var, c1, valc1 = 1, c2, valc2 = "Au", fact) {
  data %>%
    mutate(cptl_predi = prediction) %>%
    filter(!!sym(c1) == valc1, !!sym(c2) == valc2) %>%
    ggplot(aes(x = .data[[var]], y = cptl_predi)) +
    geom_point(data=cptldata, aes(y = Cptl0), alpha=0.6)+
    geom_line(aes(color = as.factor(.data[[fact]]))) +
    theme_minimal() +
    ylim(0,50)
}

plot_capitule2 <- function(data = fake_data, prediction, var, c1, valc1 = 1, c2, valc2 = "Au", fact) {
  data %>%
    mutate(cptl_predi = prediction) %>%
    filter(!!sym(c1) == valc1, !!sym(c2) == valc2) %>%
```

```
    ggplot(aes(x = .data[[var]], y = cptl_predi)) +
    geom_line(aes(color = as.factor(.data[[fact]]))) +
    theme_minimal() +
    ylim(0,50)
}
```

Nombre de capitules en fonction de la taille

En fixant la population : voir l'effet année

var <- "Size0Mars"; c1 <- "age0"; valc1 <- 1; c2 <- "Pop"; valc2 <- "Au"; fact <- "year"

Nombre de capitules en fonction de l'age

En fixant la population : voir l'effet année

var <- "age0"; c1 <- "Size0Mars"; valc1 <- 1; c2 <- "Pop"; valc2 <- "Au"; fact <- "year"