Plantules size Models Fitted

Loïc Pages

2025-06-18

Introduction

```r
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 4.5.35) 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.
```

```r
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
```

```r
library(splines)
library(foreach)
```

```
##
## Attaching package: 'foreach'
##
## The following objects are masked from 'package:purrr':
##
##     accumulate, when
```

```r
library(doParallel)
```

```
## Loading required package: iterators
## Loading required package: parallel
```

```r
library(patchwork)

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

IPM_data <- read.csv("newdata.csv")

centauree_data <- IPM_data[!is.na(IPM_data$Size0Mars) & !is.na(IPM_data$Age),]
centauree_data$Age[centauree_data$Age > 8] <- 8

spaMM.options(separation_max=70)
```

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

fake_data <- expand.grid(
  year = annees,
  Pop = populations,
  Size0Mars = taille_range,
  Age = age_range
)

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

BIC

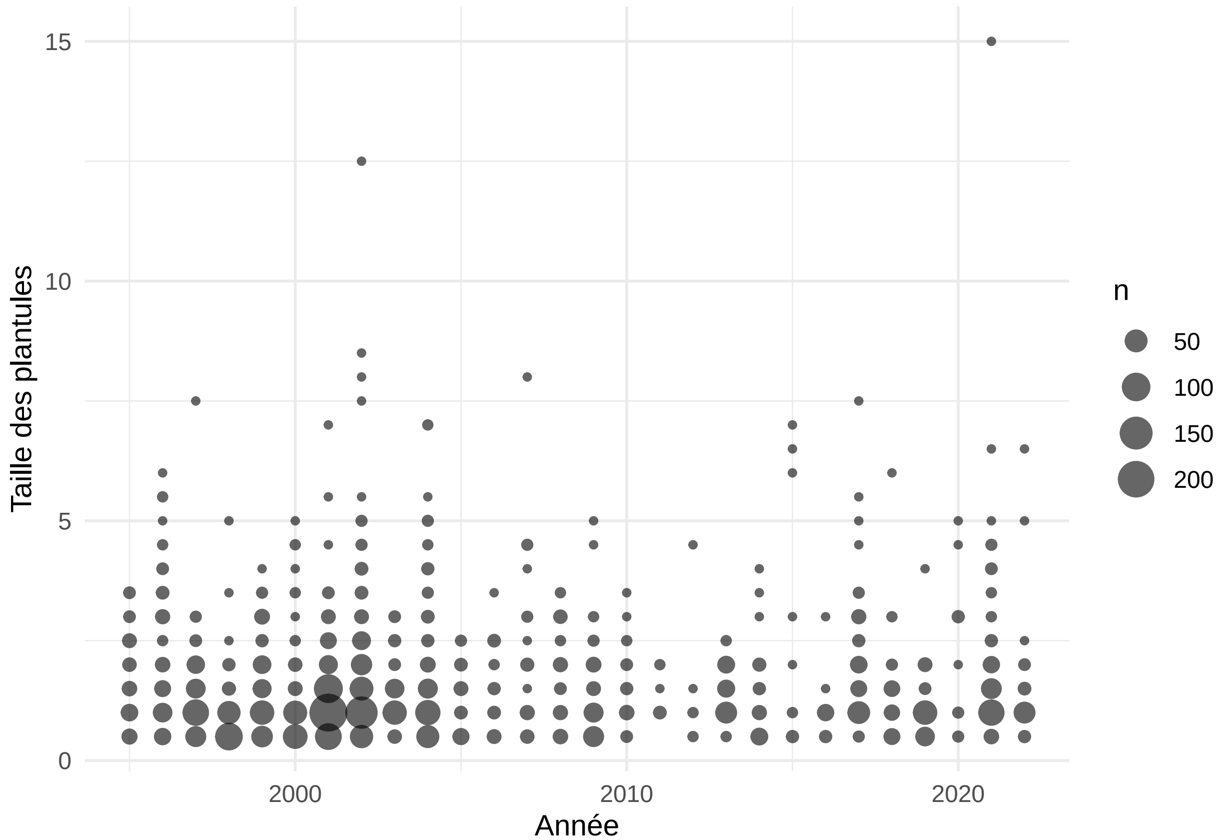
```r
# N the number of subjects
# ntot the total number of observations
extractBIC <- function(fit, ntot, N){
  extractAIC(fit)[[2]] +(log(ntot)-2)*DoF(fit)[[3]] + log(N)*DoF(fit)[[1]]
}
```

Taille des plantules

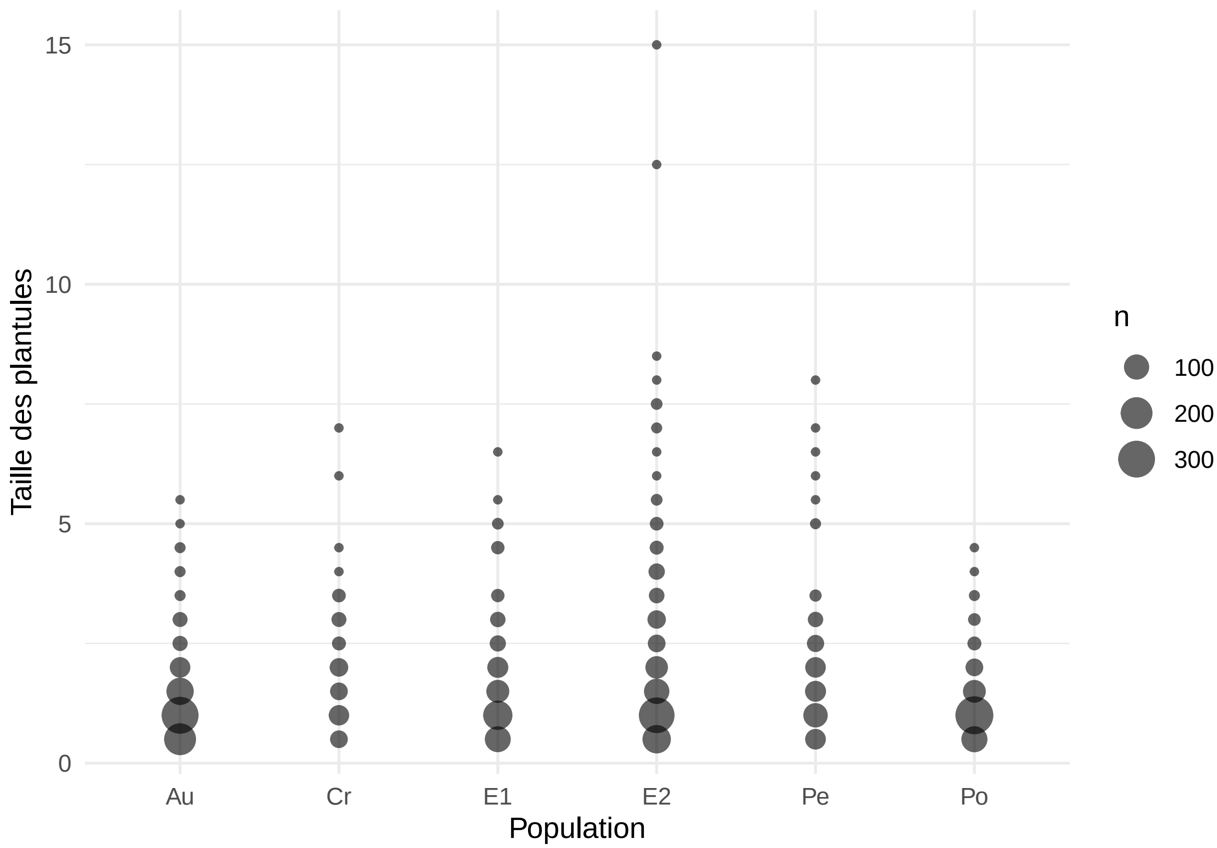
```r
plantule_data <- centauree_data[centauree_data$Age==1,]
```

```r
# Taille des plantules / année

plantule_data %>%
  ggplot(aes(x = year, y = Size0Mars)) +
  geom_count(alpha=0.6) +
  labs(x = "Année",
```

```
      y = "Taille des plantules") +
  theme_minimal()
```




```
# Taille des plantules / population

plantule_data %>%
  ggplot(aes(x = Pop, y = Size0Mars)) +
  geom_count(alpha=0.6) +
  labs(x = "Population",
       y = "Taille des plantules") +
  theme_minimal()
```




```r
Pltglm1 <- fitme(Size0Mars ~ 1 + (1|year) + (1|Pop) + (1|Pop:year),
                 data=plantule_data,
                 family = Gamma(log))

Pltglm2 <- fitme(Size0Mars ~ 1 + (1|Pop) + (1|Pop:year),
                 data=plantule_data,
                 family = Gamma(log))

Pltglm3 <- fitme(Size0Mars ~ 1 + (1|year) + (1|Pop:year),
                 data=plantule_data,
                 family = Gamma(log))

Pltglm4 <- fitme(Size0Mars ~ 1 + (1|Pop:year),
                 data=plantule_data,
                 family = Gamma(log))

Pltglm5 <- fitme(Size0Mars ~ 1 + (1|year) + (1|Pop),
                 data=plantule_data,
                 family = Gamma(log))
```

```r
summary(Pltglm1)
```

```
## formula: Size0Mars ~ 1 + (1 | year) + (1 | Pop) + (1 | Pop:year)
## Estimation of lambda and phi by ML (P_v approximation of logL).
## Estimation of fixed effects by ML (P_v approximation of logL).
```

```
## family: Gamma( link = log )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)   0.2873   0.0769   3.736
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.02495
##    Pop  :  0.02477
##    Pop:year  :  0.07813
##              --- Coefficients for log(lambda):
##     Group        Term Estimate Cond.SE
##      year (Intercept)   -3.691  0.3807
##       Pop (Intercept)   -3.698  0.6437
##  Pop:year (Intercept)   -2.549  0.1537
## # of obs: 2904; # of groups: year, 28; Pop, 6; Pop:year, 142
##  --- Residual variation ( var = phi * mu^2 )  --
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)   -1.356  0.02615
## Estimate of phi:  0.2577
##  ------------- Likelihood values  -------------
##                         logLik
## logL      (P_v(h)): -2715.877
```

```r
summary(Pltglm2)
```

```
## formula: Size0Mars ~ 1 + (1 | Pop) + (1 | Pop:year)
## Estimation of lambda and phi by ML (P_v approximation of logL).
## Estimation of fixed effects by ML (P_v approximation of logL).
## family: Gamma( link = log )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)   0.2788  0.06931   4.023
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    Pop  :  0.02247
##    Pop:year  :  0.1089
##              --- Coefficients for log(lambda):
##     Group        Term Estimate Cond.SE
##       Pop (Intercept)   -3.796  0.6547
##  Pop:year (Intercept)   -2.217  0.1399
## # of obs: 2904; # of groups: Pop, 6; Pop:year, 142
##  --- Residual variation ( var = phi * mu^2 )  --
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)   -1.358  0.02617
## Estimate of phi:  0.2572
##  ------------- Likelihood values  -------------
##                         logLik
## logL      (P_v(h)): -2719.756
```

```r
summary(Pltglm3)
```

```
## formula: Size0Mars ~ 1 + (1 | year) + (1 | Pop:year)
## Estimation of lambda and phi by ML (P_v approximation of logL).
## Estimation of fixed effects by ML (P_v approximation of logL).
## family: Gamma( link = log )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)   0.2646  0.04245   6.232
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.01974
##    Pop:year  :  0.1112
##              --- Coefficients for log(lambda):
##     Group        Term Estimate Cond.SE
##      year (Intercept)   -3.925  0.4267
##  Pop:year (Intercept)   -2.196  0.1431
## # of obs: 2904; # of groups: year, 28; Pop:year, 142
##  --- Residual variation ( var = phi * mu^2 )  --
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)   -1.358  0.02618
## Estimate of phi:  0.2572
##  ------------- Likelihood values  -------------
##                        logLik
## logL      (P_v(h)): -2722.887
```

```r
summary(Pltglm4)
```

```
## formula: Size0Mars ~ 1 + (1 | Pop:year)
## Estimation of lambda and phi by ML (P_v approximation of logL).
## Estimation of fixed effects by ML (P_v approximation of logL).
## family: Gamma( link = log )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)   0.2625  0.03484   7.533
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    Pop:year  :  0.1333
##              --- Coefficients for log(lambda):
##     Group        Term Estimate Cond.SE
##  Pop:year (Intercept)   -2.015  0.1348
## # of obs: 2904; # of groups: Pop:year, 142
##  --- Residual variation ( var = phi * mu^2 )  --
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)   -1.359  0.02618
## Estimate of phi:  0.257
```

```
##  ------------- Likelihood values  -------------
##                         logLik
## logL      (P_v(h)): -2724.726
```

```
summary(Pltglm5)
```

```
## formula: Size0Mars ~ 1 + (1 | year) + (1 | Pop)
## Estimation of lambda and phi by ML (P_v approximation of logL).
## Estimation of fixed effects by ML (P_v approximation of logL).
## family: Gamma( link = log )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)   0.3318   0.0851   3.898
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.0437
##    Pop  :  0.03254
##              --- Coefficients for log(lambda):
##  Group        Term Estimate Cond.SE
##   year (Intercept)    -3.13  0.2916
##    Pop (Intercept)   -3.425  0.5969
## # of obs: 2904; # of groups: year, 28; Pop, 6
##  --- Residual variation ( var = phi * mu^2 )  --
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)   -1.235  0.02576
## Estimate of phi:  0.291
##  ------------- Likelihood values  -------------
##                         logLik
## logL      (P_v(h)): -2828.364
```

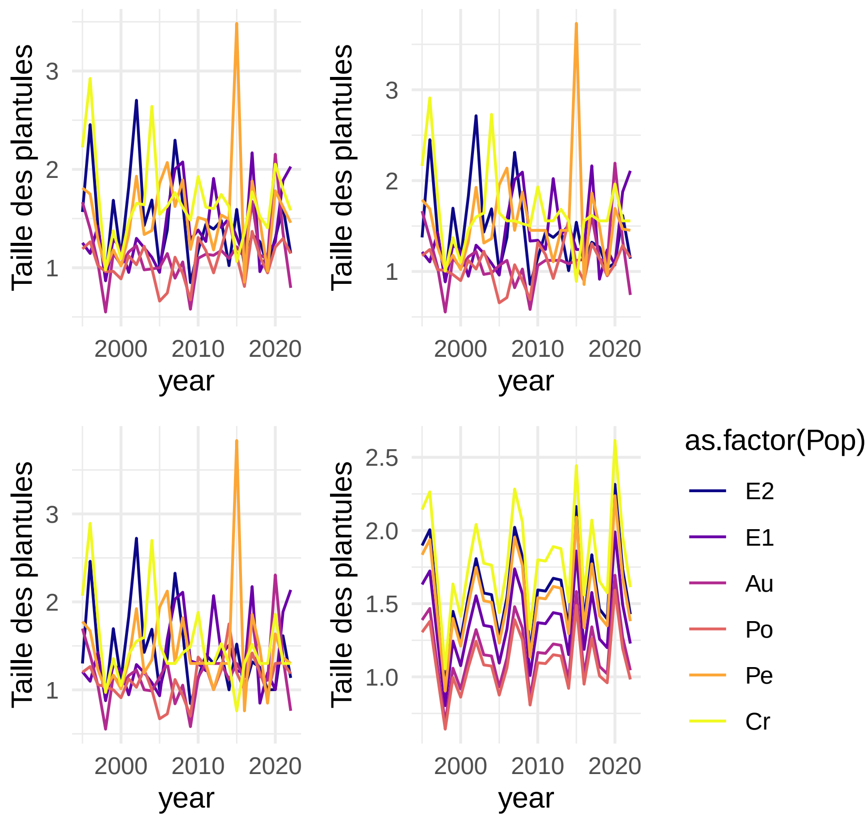
```
Pltpredict1 <- predict(Pltglm1, newdata = fake_data)[,1]
Pltpredict2 <- predict(Pltglm2, newdata = fake_data)[,1]
Pltpredict3 <- predict(Pltglm3, newdata = fake_data)[,1]
Pltpredict4 <- predict(Pltglm4, newdata = fake_data)[,1]
Pltpredict5 <- predict(Pltglm5, newdata = fake_data)[,1]
```

```
plot_plantule <- function(data = fake_data, prediction, var, fact) {
  data %>%
    mutate(plt_predi = prediction) %>%
    ggplot(aes(x = .data[[var]], y = plt_predi)) +
    geom_line(aes(color = as.factor(.data[[fact]])),show.legend = FALSE) +
    labs(y="Taille des plantules")+
    scale_color_viridis_d(option = "plasma")+
    theme_minimal()
}

plot_plantule1 <- function(data = fake_data, prediction, var, fact) {
  data %>%
    mutate(plt_predi = prediction) %>%
    ggplot(aes(x = .data[[var]], y = plt_predi)) +
```

```r
    geom_line(aes(color = as.factor(.data[[fact]]))) +
    labs(y="Taille des plantules")+
    scale_color_viridis_d(option = "plasma")+
    theme_minimal()
}

plot_plantule2 <- function(data = fake_data, prediction) {
  data %>%
    mutate(plt_predi = prediction) %>%
    ggplot(aes(x = plt_predi)) +
    stat_bin(binwidth = 0.25,fill="grey",color="black")+
    labs(x="Taille des plantules")+
    theme_minimal()
}
```

Taille des plantules en fonction de l'année

```r
var <- "year"
fact <- "Pop"
```



Densité de taille de plantule