Fecundity Models Fitted

Loïc Pages

2025-06-18

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 4.5.35) 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.
```

```
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(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$SizeOMars) & !is.na(IPM_data$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,
  SizeOMars = 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]]
}
```

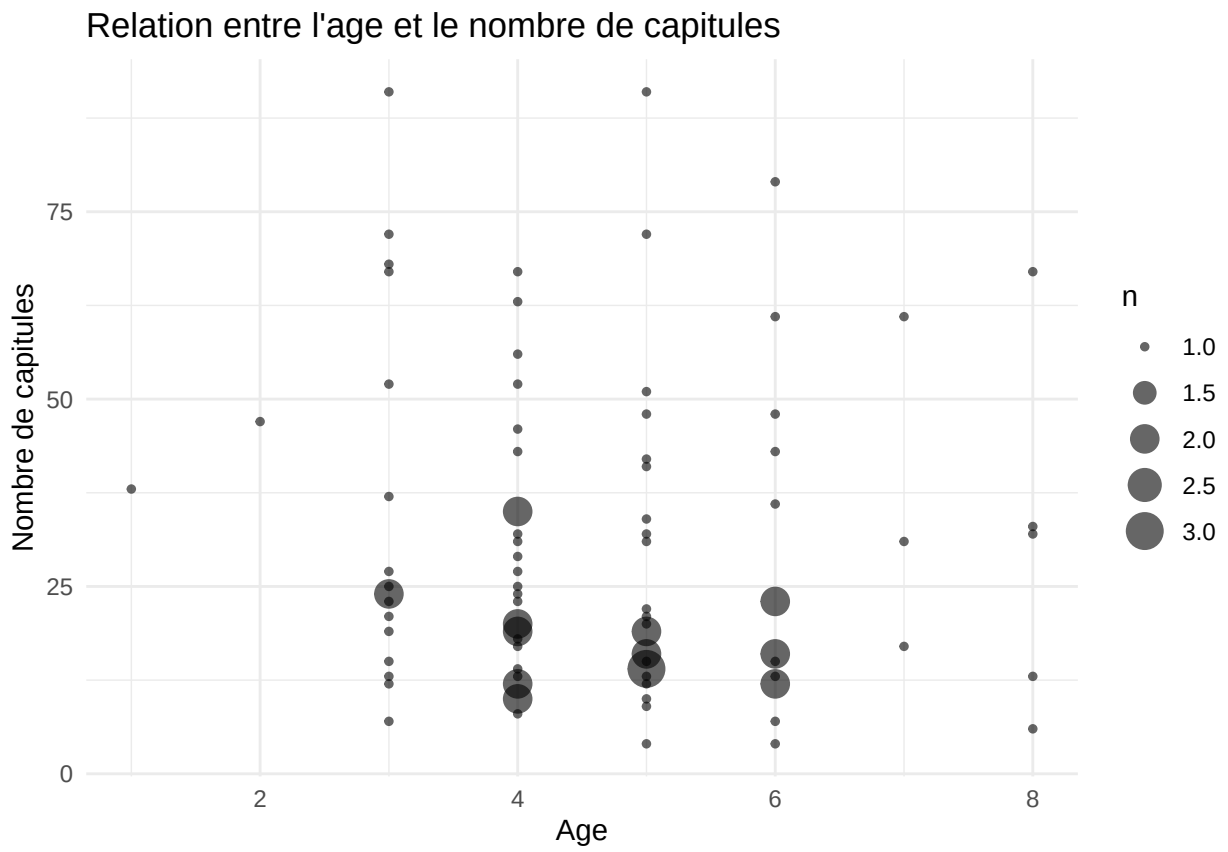
Nombre de capitules

```r
cptldata <- centauree_data[centauree_data$Flowering!=0,]
```

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

capidata%>%
```

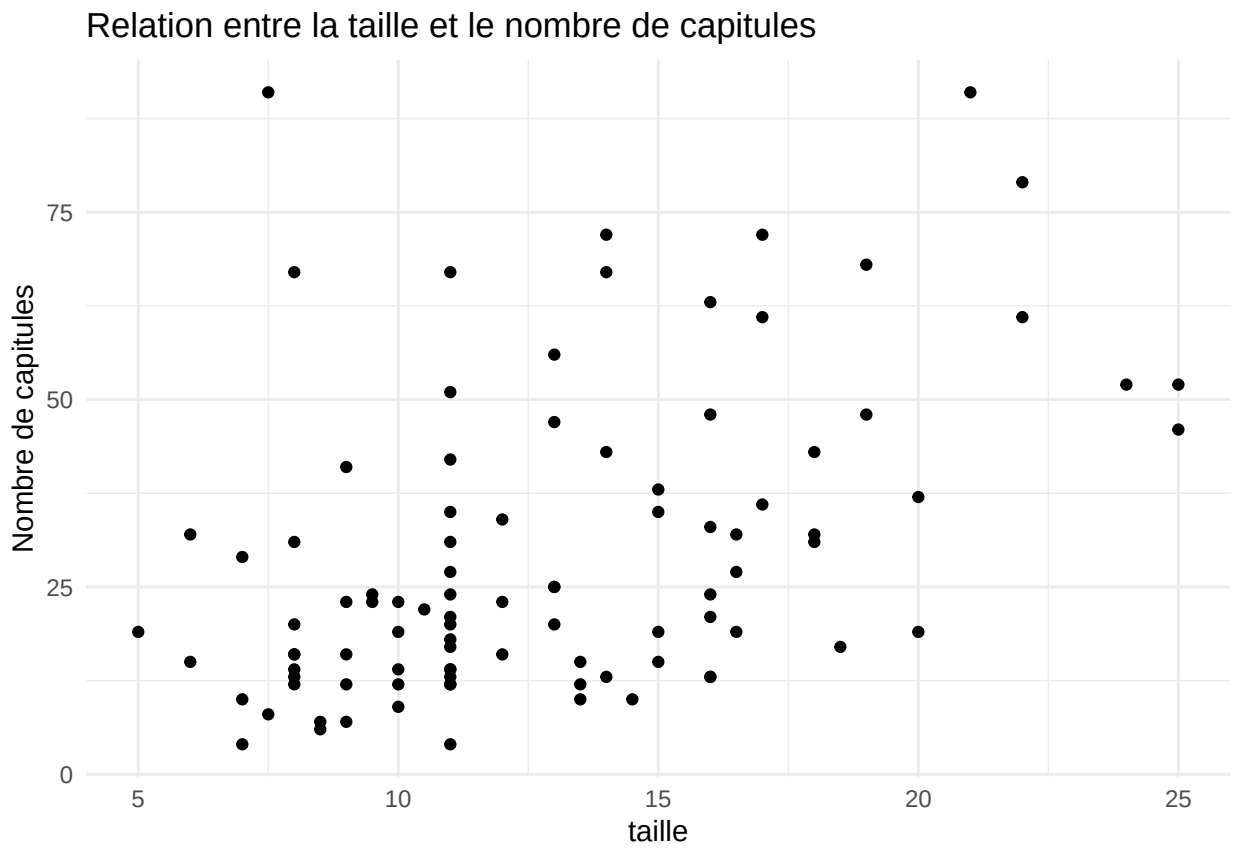
```
  ggplot(aes(x = Age, y = meancptl)) +
  geom_count(aes(y=Capitule), alpha=0.6) +
  labs(title = "Relation entre l'age et le nombre de capitules",
       x = "Age",
       y = "Nombre de capitules") +
  theme_minimal()
```

```
## Warning: Removed 126 rows containing non-finite outside the scale range
## (`stat_sum()`).
```



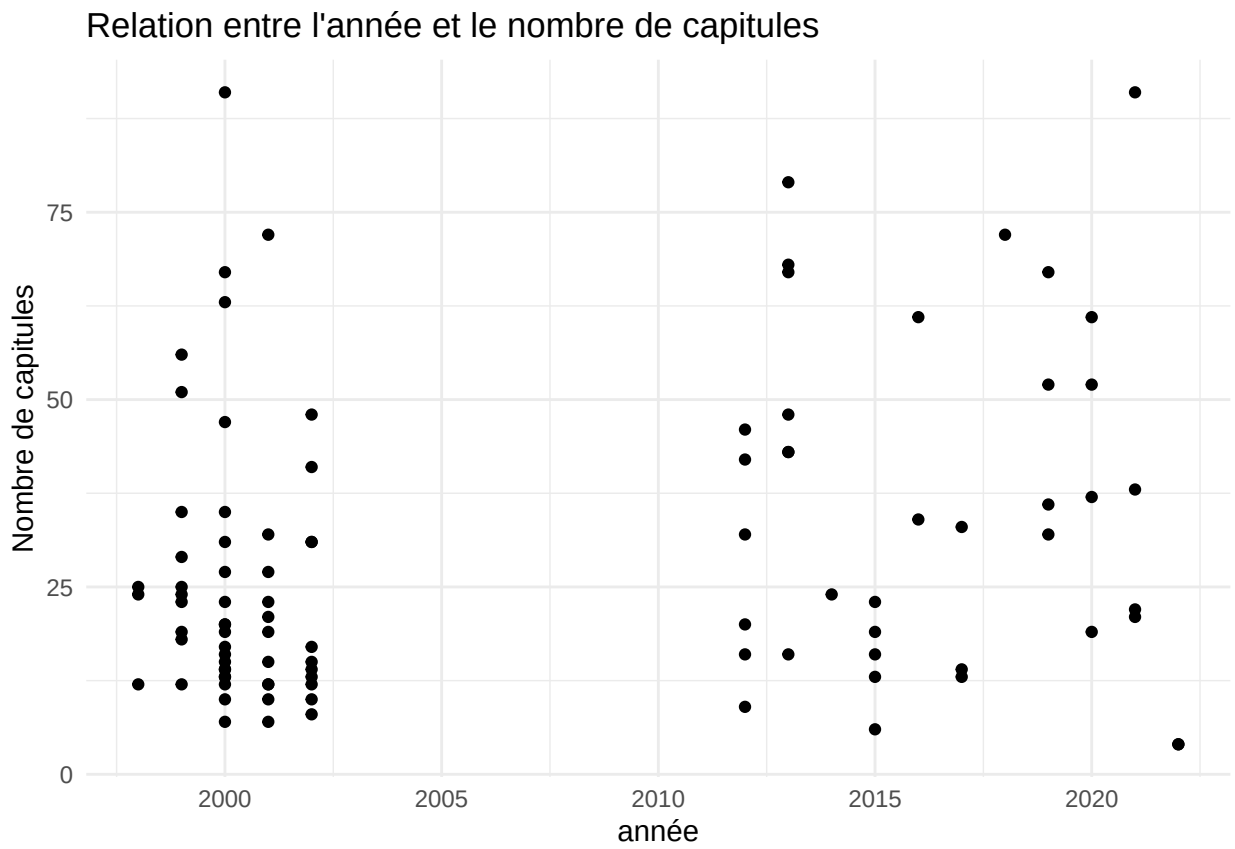

```
# Nombre de capitule / taille
cptldata %>%

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

```
## Warning: Removed 126 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




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

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

```
## Warning: Removed 126 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
ACptlglm1 <- fitme(log(Capitule) ~ 1 + SizeOMars + (Age|year),
                   data=cptldata)

ACptlglm2 <- fitme(log(Capitule) ~ 1 + SizeOMars + (Age|year) + (1|Pop),
                   data=cptldata)

ACptlglm3 <- fitme(log(Capitule) ~ 1 + poly(SizeOMars,2) + (Age|year),
                   data=cptldata)

ACptlglm4 <- fitme(log(Capitule) ~ 1 + SizeOMars +Age +(Age|year),
                   data=cptldata)

ACptlglm5 <- fitme(log(Capitule) ~ 1 + SizeOMars + (1|year),
                   data=cptldata)
```

```
BCptlglm1 <- fitme(log(Capitule) ~ 1 + SizeOMars + (Age|year),
                   data=cptldata)

BCptlglm2 <- fitme(log(Capitule) ~ 1 +SizeOMars + (Age|year) + (1|Pop),
                   data=cptldata)

BCptlglm3 <- fitme(log(Capitule) ~ 1 + SizeOMars + (1|year),
                   data=cptldata)

BCptlglm4 <- fitme(log(Capitule) ~ 1 + SizeOMars,
```

```
                 data=cptldata)

BCptlglm5 <- fitme(log(Capitule) ~ 1 + SizeOMars + (1|year) + (1|Pop),
                 data=cptldata)
```

```
summary(ACptlglm1)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars + (Age | year)
## ML: Estimation of ranCoefs and phi by ML.
##     Estimation of fixed effects by ML.
## Estimation of phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)  2.32734   0.2001  11.633
## SizeOMars    0.06752   0.0136   4.963
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term   Var. Corr.
##   year (Intercept) 0.2127
##   year         Age 0.0238    -1
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 0.282245
##  ------------- Likelihood values  -------------
##                          logLik
## logL       (p_v(h)): -82.96256
```

```
summary(ACptlglm2)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars + (Age | year) + (1 | Pop)
## ML: Estimation of lambda, ranCoefs and phi by ML.
##     Estimation of fixed effects by ML.
## Estimation of phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)  2.30931  0.20576  11.223
## SizeOMars    0.06989  0.01349   5.181
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var. Corr.
##   year (Intercept)  0.2146
##   year         Age 0.02458    -1
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    Pop  :  0.018
##             --- Coefficients for log(lambda):
##  Group        Term Estimate Cond.SE
##    Pop (Intercept)   -4.018  0.8667
```

```
## # of obs: 95; # of groups: year, 16; Pop, 6
##  -------------- Residual variance  ------------
## phi estimate was 0.265943
##  ------------- Likelihood values  -------------
##                          logLik
## logL       (p_v(h)): -82.50057
```

```
summary(ACptlglm3)
```

```
## formula: log(Capitule) ~ 1 + poly(SizeOMars, 2) + (Age | year)
## ML: Estimation of ranCoefs and phi by ML.
##     Estimation of fixed effects by ML.
## Estimation of phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)           3.1950  0.07642 41.8074
## poly(SizeOMars, 2)1   2.9216  0.58956  4.9555
## poly(SizeOMars, 2)2   0.1495  0.57139  0.2616
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var. Corr.
##   year (Intercept)  0.2052
##   year         Age 0.02343    -1
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 0.28185
##  ------------- Likelihood values  -------------
##                          logLik
## logL       (p_v(h)): -82.92893
```

```
summary(ACptlglm4)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars + Age + (Age | year)
## ML: Estimation of ranCoefs and phi by ML.
##     Estimation of fixed effects by ML.
## Estimation of phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)  2.38266  0.32556   7.319
## SizeOMars    0.06715  0.01373   4.892
## Age         -0.01310  0.06124  -0.214
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term   Var. Corr.
##   year (Intercept) 0.1994
##   year         Age 0.0229    -1
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 0.282392
```

```
##  ------------- Likelihood values  -------------
##                           logLik
## logL       (p_v(h)): -82.94076
```

```
summary(ACptlglm5)
```

```
## formula: log(Capitule) ~ 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)  2.28701  0.21297  10.739
## SizeOMars    0.06886  0.01446   4.763
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.07845
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 0.312307
##  ------------- Likelihood values  -------------
##                           logLik
## logL       (p_v(h)): -86.00139
```

```
summary(BCptlglm1)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars + (Age | year)
## ML: Estimation of ranCoefs and phi by ML.
##     Estimation of fixed effects by ML.
## Estimation of phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)  2.32734   0.2001  11.633
## SizeOMars    0.06752   0.0136   4.963
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term   Var. Corr.
##   year (Intercept) 0.2127
##   year         Age 0.0238    -1
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 0.282245
##  ------------- Likelihood values  -------------
##                           logLik
## logL       (p_v(h)): -82.96256
```

```
summary(BCptlglm2)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars + (Age | year) + (1 | Pop)
## ML: Estimation of lambda, ranCoefs and phi by ML.
##     Estimation of fixed effects by ML.
## Estimation of phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)  2.30931  0.20576  11.223
## SizeOMars    0.06989  0.01349   5.181
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var. Corr.
##   year (Intercept)  0.2146
##   year         Age 0.02458    -1
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    Pop  :  0.018
##              --- Coefficients for log(lambda):
##  Group        Term Estimate Cond.SE
##    Pop (Intercept)   -4.018  0.8667
## # of obs: 95; # of groups: year, 16; Pop, 6
##  -------------- Residual variance  ------------
## phi estimate was 0.265943
##  ------------- Likelihood values  -------------
##                          logLik
## logL       (p_v(h)): -82.50057
```

```
summary(BCptlglm3)
```

```
## formula: log(Capitule) ~ 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)  2.28701  0.21297  10.739
## SizeOMars    0.06886  0.01446   4.763
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.07845
## # of obs: 95; # of groups: year, 16
##  -------------- Residual variance  ------------
## phi estimate was 0.312307
##  ------------- Likelihood values  -------------
##                          logLik
## logL       (p_v(h)): -86.00139
```

```
summary(BCptlglm4)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars
```

```
## 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)  2.20224  0.18997  11.592
## SizeOMars    0.07455  0.01398   5.331
##  -------------- Residual variance  ------------
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)   -1.003   0.1451
## Estimate of phi=residual var:  0.3669
##  ------------- Likelihood values  -------------
##                         logLik
## logL               : -87.17642
```

```
summary(BCptlglm5)
```

```
## formula: log(Capitule) ~ 1 + SizeOMars + (1 | year) + (1 | Pop)
## 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)  2.24781  0.22148  10.149
## SizeOMars    0.07159  0.01439   4.974
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.09097
##    Pop   :  0.01864
## # of obs: 95; # of groups: year, 16; Pop, 6
##  -------------- Residual variance  ------------
## phi estimate was 0.29351
##  ------------- Likelihood values  -------------
##                         logLik
## logL       (p_v(h)): -85.62765
```

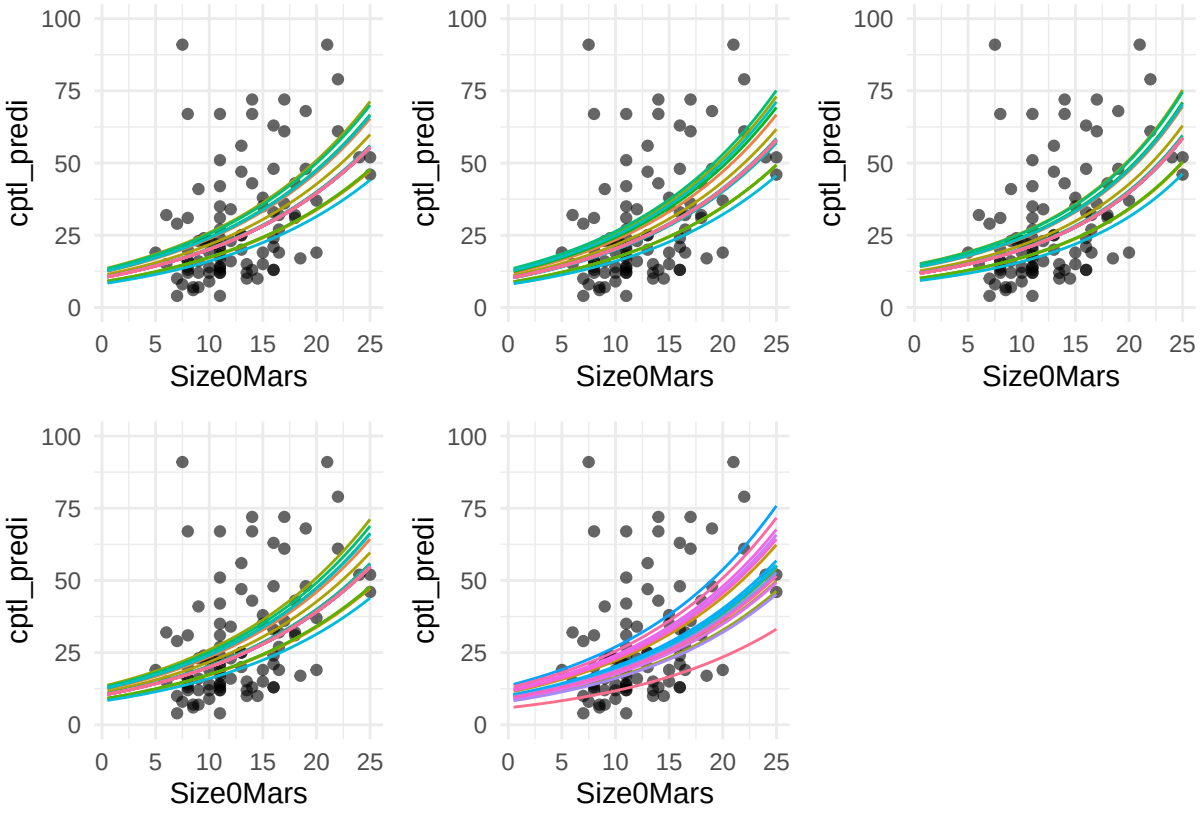
```
ACptlpredict1 <- predict(ACptlglm1, newdata = fake_data)[,1]
ACptlpredict2 <- predict(ACptlglm2, newdata = fake_data)[,1]
ACptlpredict3 <- predict(ACptlglm3, newdata = fake_data)[,1]
ACptlpredict4 <- predict(ACptlglm4, newdata = fake_data)[,1]
ACptlpredict5 <- predict(ACptlglm5, newdata = fake_data)[,1]

BCptlpredict1 <- predict(BCptlglm1, newdata = fake_data)[,1]
BCptlpredict2 <- predict(BCptlglm2, newdata = fake_data)[,1]
BCptlpredict3 <- predict(BCptlglm3, newdata = fake_data)[,1]
BCptlpredict4 <- predict(BCptlglm4, newdata = fake_data)[,1]
BCptlpredict5 <- predict(BCptlglm5, newdata = fake_data)[,1]
```

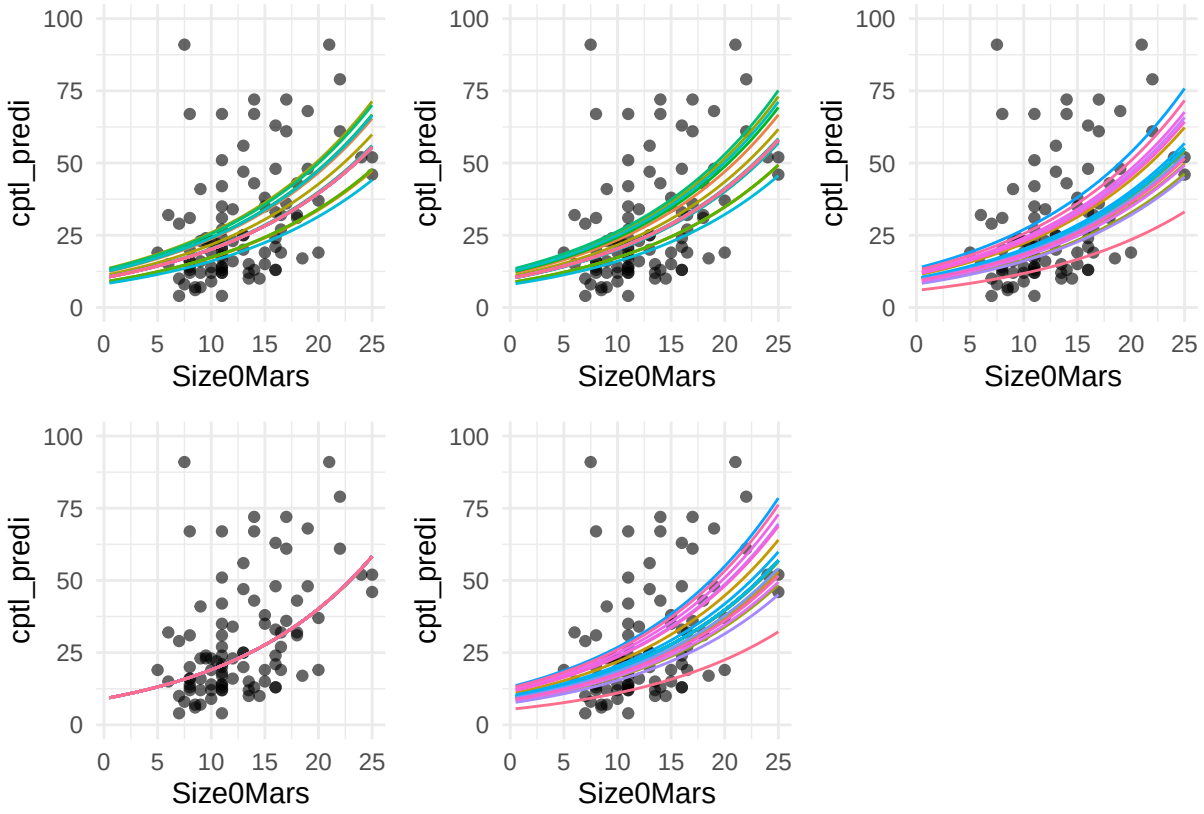
```
plot_capitule <- function(data = fake_data, prediction, var, fact) {
  data %>%
    mutate(cptl_predi = exp(prediction)) %>%
    group_by(!!sym(var),!!sym(fact)) %>%
    summarise(cptl_predi = mean(cptl_predi)) %>%
    ggplot(aes(x = .data[[var]], y = cptl_predi)) +
    geom_point(data = cptldata, aes(y = Capitule), alpha=0.6)+
    geom_line(aes(color = as.factor(.data[[fact]])),show.legend = FALSE) +
    theme_minimal() +
    ylim(0,100)
}

plot_capitule2 <- function(data = fake_data, prediction, var, fact) {
  data %>%
    mutate(cptl_predi = exp(prediction)) %>%
    group_by(!!sym(var),!!sym(fact)) %>%
    summarise(cptl_predi = mean(cptl_predi)) %>%
    ggplot(aes(x = .data[[var]], y = cptl_predi)) +
    geom_point(data = cptldata, aes(y = Capitule), alpha=0.6)+
    geom_line(aes(color = as.factor(.data[[fact]])),show.legend = FALSE) +
    theme_minimal() +
    scale_color_viridis_d(option = "plasma")+
    ylim(0,100)
}
```

Nombre de capitules en fonction de la taille

voir l'effet année

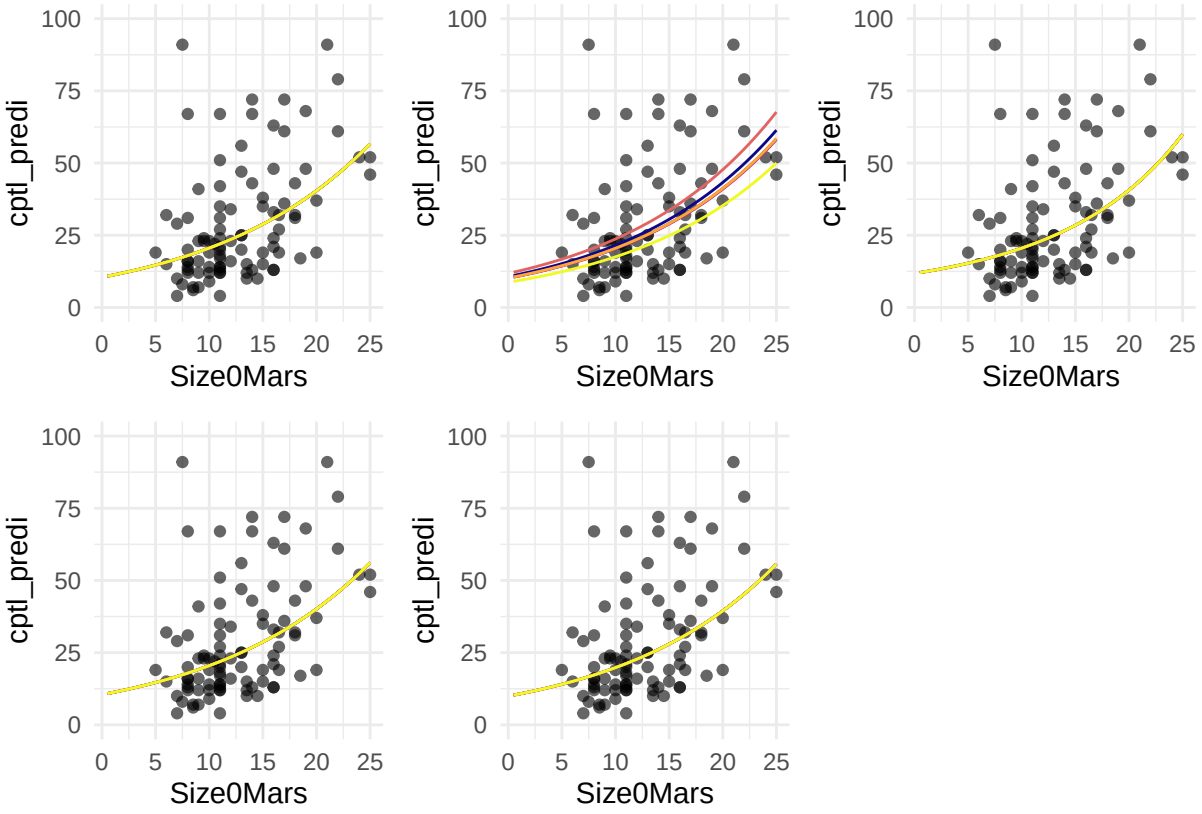
```
var <- "SizeOMars"
fact <- "year"
```

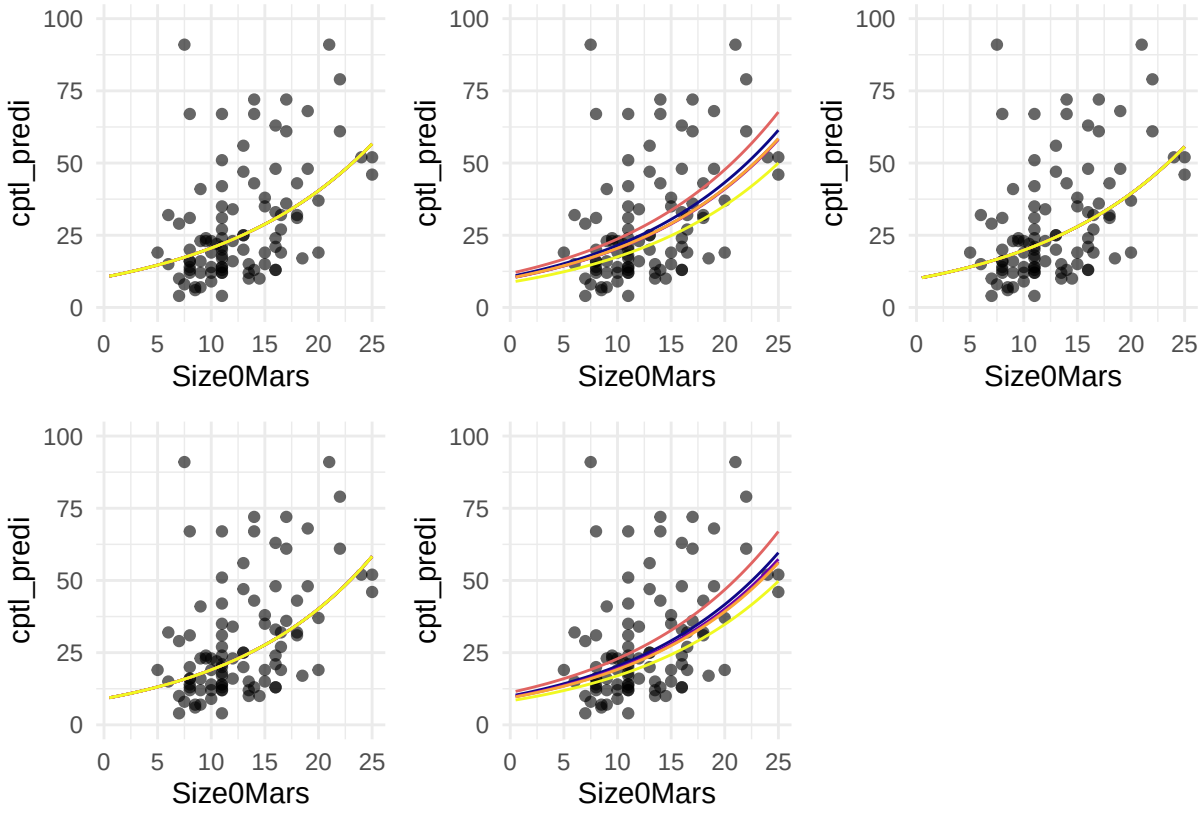

cptl_predi
Size0Mars
0
25
50
75
100
0
5
10
15
20
25



voir l'effet pop

```
var <- "Size0Mars"
fact <- "Pop"
```

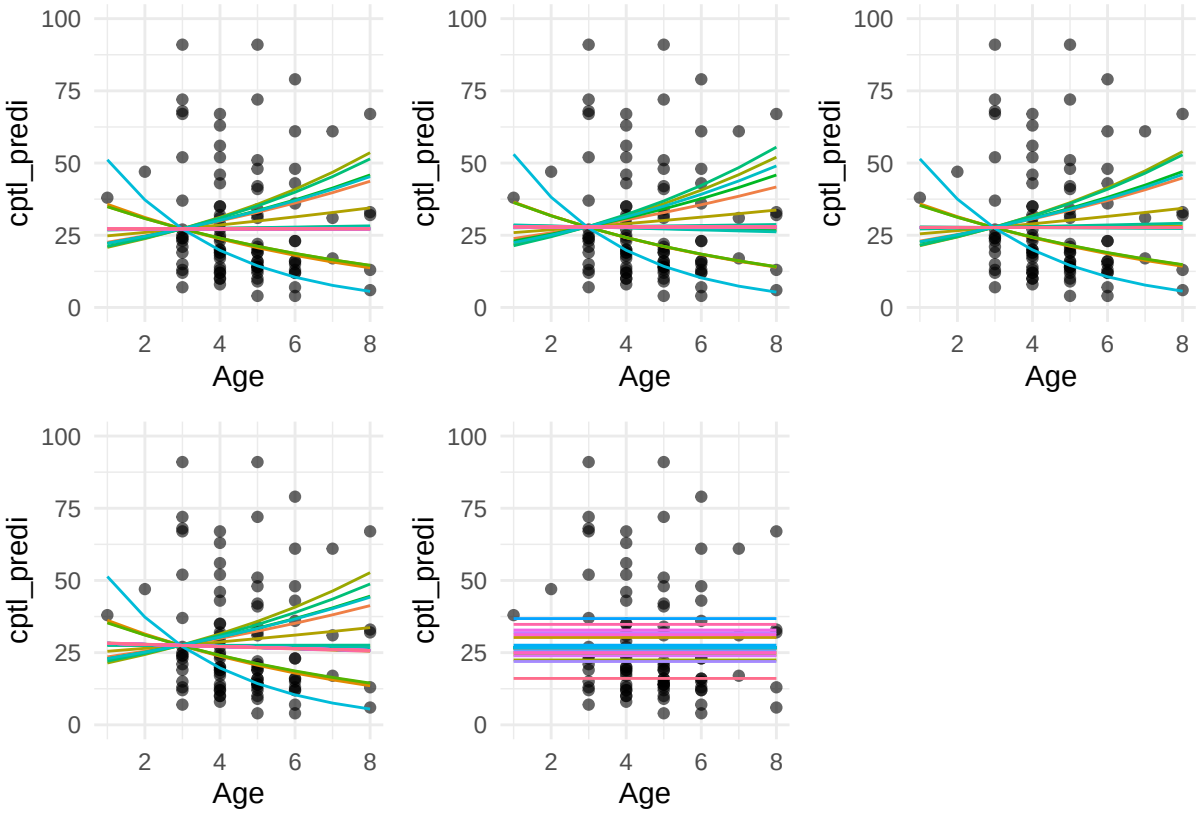

cptl_predi
Size0Mars
0
25
50
75
100
0
5
10
15
20
25

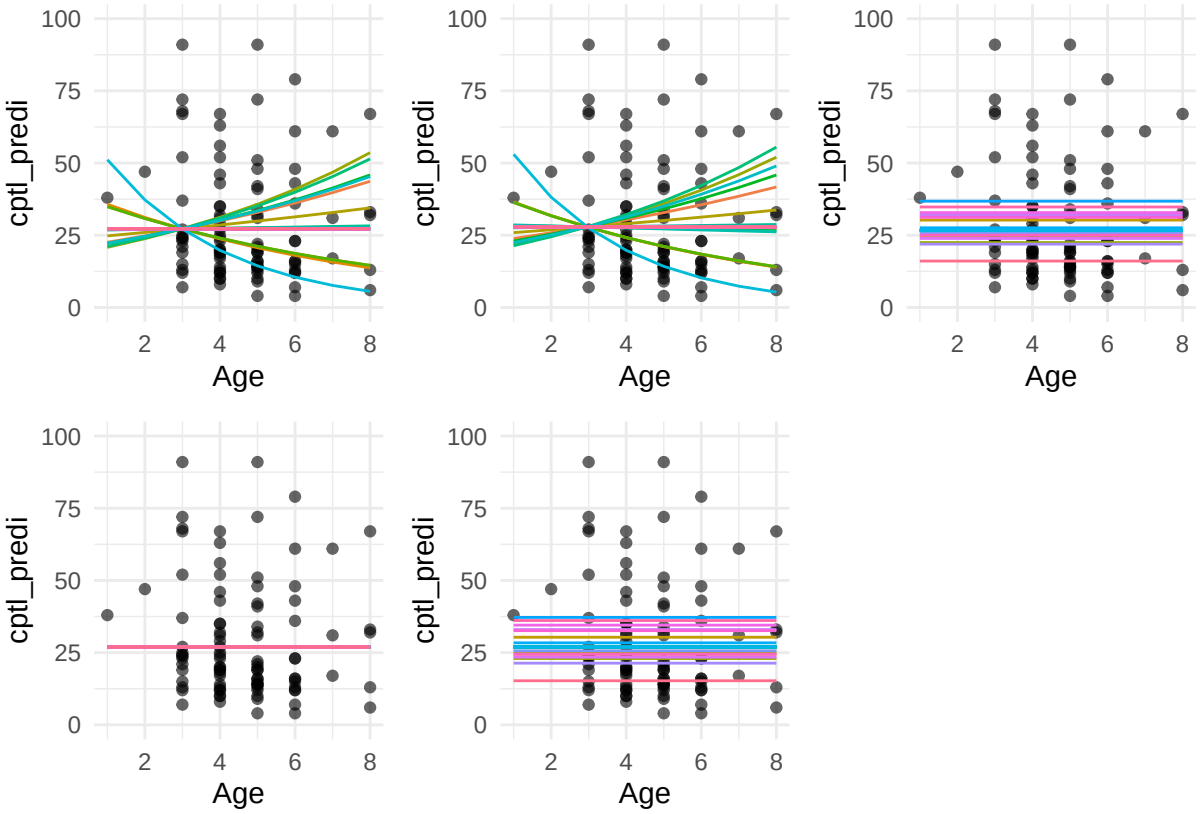


Nombre de capitules en fonction de l'age

En fixant la population : voir l'effet année

```
var <- "Age"
fact <- "year"
```


cptl_predi
Age
100
75
50
25
0
2
4
6
8


cptl_predi
Age
100
75
50
25
0
2
4
6
8