Flowering Models Fitted

Loïc Pages

2025-06-20

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())
```

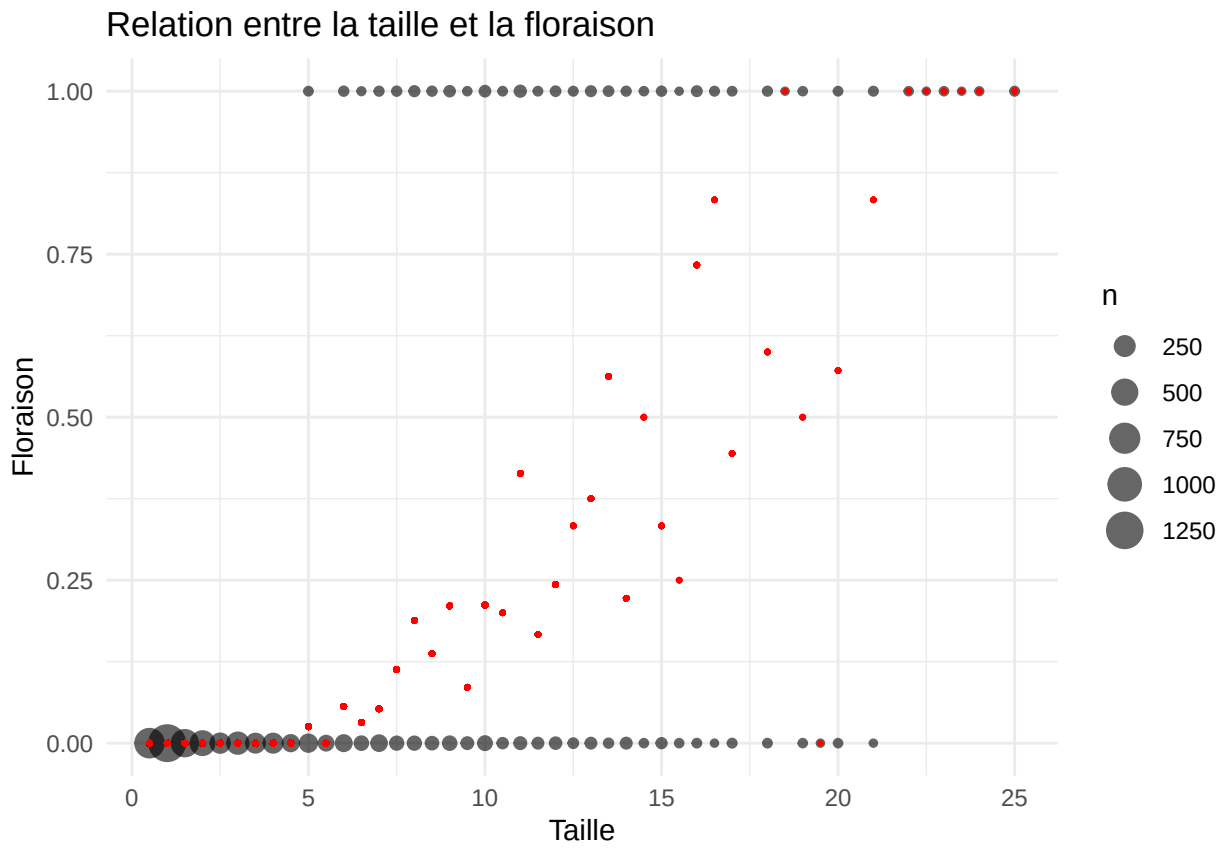
BIC

```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]]
}
```

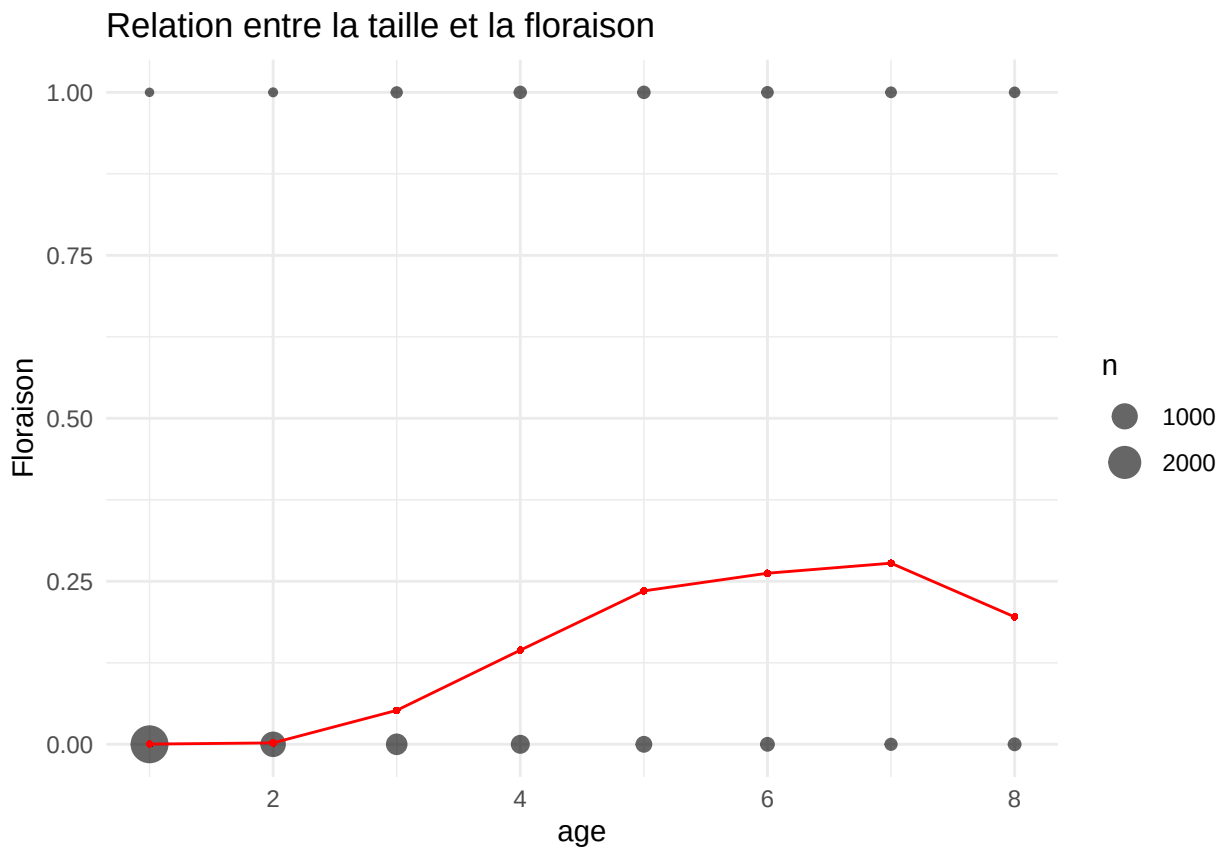
Flowering probability

```r
centauree_data %>%
  group_by(SizeOMars) %>%
  mutate(floweringProba = sum(Flowering, na.rm = TRUE) / n()) %>%
  ggplot(aes(x = SizeOMars, y = Flowering)) +
  geom_count(alpha = 0.6) +
  geom_point(aes(y = floweringProba), color = "red", size = 0.5) +
  labs(title = "Relation entre la taille et la floraison",
    x = "Taille",
```

```
    y = "Floraison") +
  ylim(0, 1) +
  theme_minimal()
```




```
centauree_data %>%
  group_by(Age) %>%
  mutate(floweringProba = sum(Flowering, na.rm = TRUE) / n()) %>%
  ggplot(aes(x = Age, y = Flowering)) +
  geom_count(alpha = 0.6) +
  geom_point(aes(y = floweringProba), color = "red", size = 0.5) +
  geom_line(aes(y = floweringProba), color = "red") +
  labs(title = "Relation entre la taille et la floraison",
    x = "age",
    y = "Floraison") +
  ylim(0, 1) +
  theme_minimal()
```



AIC

```
AFlowglm1 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,2) + (Age|Pop),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

AFlowglm2 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,4) + (Age|Pop),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

AFlowglm3 <- fitme(Flowering ~ 1+ poly(SizeOMars,4) + poly(Age,2) + (Age|Pop),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

AFlowglm4 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,2) + (Age|Pop)+ (1|year),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

AFlowglm5 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,3) + (Age|Pop),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")
```

BIC

```
BFlowglm1 <- fitme(Flowering ~ 1 + poly(SizeOMars,3) + poly(Age,2) + (Age|Pop),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

BFlowglm2 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,2) + (Age|Pop)+ (1|year),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

BFlowglm3 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,2) + (Age|Pop)+ (SizeOMars|year),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")
```

```
## Warning in (function (processed, init = list(), fixed = list(), lower = list(),
## : Numerical issue detected; see div_info(<fit object>) for more information.
```

```
BFlowglm4 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,2) + (Age|Pop)+ (Age|year),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")

BFlowglm5 <- fitme(Flowering ~ 1+ poly(SizeOMars,3) + poly(Age,2) + (SizeOMars + Age|Pop),
                   family=binomial,
                   data=centauree_data,
                   method="PQL/L")
```

```
## Warning in (function (processed, init = list(), fixed = list(), lower = list(),
## : Numerical issue detected; see div_info(<fit object>) for more information.
```

```
summary(AFlowglm1)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (Age | Pop)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                      Estimate Cond. SE t-value
## (Intercept)           -10.21   0.9991 -10.222
## poly(SizeOMars, 3)1   226.93  34.0557   6.663
## poly(SizeOMars, 3)2   -76.57  17.1002  -4.478
## poly(SizeOMars, 3)3    39.27  11.3977   3.445
## poly(Age, 2)1         133.48  22.3423   5.974
## poly(Age, 2)2         -54.57   8.8053  -6.197
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.252
```

```
##    Pop         Age 0.09228 -0.9823
## # of obs: 5320; # of groups: Pop, 6
##  ------------- Likelihood values  -------------
##                          logLik
##        h-likelihood: -414.4578
## logL       (p_v(h)): -411.1534
```

```
summary(AFlowglm2)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 4) + (Age | Pop)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)           -9.359    1.128 -8.2973
## poly(SizeOMars, 3)1  230.238   34.257  6.7209
## poly(SizeOMars, 3)2  -77.695   17.149 -4.5305
## poly(SizeOMars, 3)3   40.294   11.520  3.4978
## poly(Age, 4)1         86.897   38.122  2.2794
## poly(Age, 4)2        -21.244   23.053 -0.9215
## poly(Age, 4)3        -21.022   13.643 -1.5408
## poly(Age, 4)4         11.235    6.698  1.6774
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term   Var.   Corr.
##    Pop (Intercept)  2.449
##    Pop         Age 0.1002 -0.9829
## # of obs: 5320; # of groups: Pop, 6
##  ------------- Likelihood values  -------------
##                          logLik
##        h-likelihood: -412.9489
## logL       (p_v(h)): -409.9265
```

```
summary(AFlowglm3)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 4) + poly(Age, 2) + (Age | Pop)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)          -10.592    1.571 -6.7425
## poly(SizeOMars, 4)1  243.382   62.370  3.9022
## poly(SizeOMars, 4)2  -88.378   40.618 -2.1758
## poly(SizeOMars, 4)3   45.243   21.934  2.0627
## poly(SizeOMars, 4)4   -3.993   12.121 -0.3294
## poly(Age, 2)1        133.221   22.299  5.9744
## poly(Age, 2)2        -54.444    8.799 -6.1877
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
```

```
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.236
##    Pop         Age 0.09138 -0.9824
## # of obs: 5320; # of groups: Pop, 6
##  ------------- Likelihood values  -------------
##                          logLik
##        h-likelihood: -414.4553
## logL       (p_v(h)): -411.1082
```

```
summary(AFlowglm4)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (Age | Pop) +
##     (1 | year)
## Estimation of lambda and ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)           -10.21   0.9991 -10.222
## poly(SizeOMars, 3)1   226.93  34.0557   6.663
## poly(SizeOMars, 3)2   -76.57  17.1002  -4.478
## poly(SizeOMars, 3)3    39.27  11.3977   3.445
## poly(Age, 2)1         133.48  22.3423   5.974
## poly(Age, 2)2         -54.57   8.8053  -6.197
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.252
##    Pop         Age 0.09228 -0.9823
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  9.294e-07
##             --- Coefficients for log(lambda):
##  Group        Term Estimate Cond.SE
##   year (Intercept)   -13.89   129.8
## # of obs: 5320; # of groups: Pop, 6; year, 28
##  ------------- Likelihood values  -------------
##                          logLik
##        h-likelihood: -245.7454
## logL       (p_v(h)): -411.1534
```

```
summary(AFlowglm5)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 3) + (Age | Pop)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                      Estimate Cond. SE   t-value
## (Intercept)         -10.20937    1.136 -8.984556
## poly(SizeOMars, 3)1 226.93131   34.065  6.661806
## poly(SizeOMars, 3)2 -76.56728   17.101 -4.477471
```

```
## poly(SizeOMars, 3)3  39.26638   11.399  3.444796
## poly(Age, 3)1       133.31406   34.641  3.848485
## poly(Age, 3)2       -54.46331   18.315 -2.973718
## poly(Age, 3)3        -0.05541    8.685 -0.006379
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.252
##    Pop         Age 0.09228 -0.9823
## # of obs: 5320; # of groups: Pop, 6
##  ------------- Likelihood values  -------------
##                          logLik
##        h-likelihood: -414.4577
## logL       (p_v(h)): -411.1534
```

```
summary(BFlowglm1)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (Age | Pop)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)           -10.21   0.9991 -10.222
## poly(SizeOMars, 3)1   226.93  34.0557   6.663
## poly(SizeOMars, 3)2   -76.57  17.1002  -4.478
## poly(SizeOMars, 3)3    39.27  11.3977   3.445
## poly(Age, 2)1         133.48  22.3423   5.974
## poly(Age, 2)2         -54.57   8.8053  -6.197
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.252
##    Pop         Age 0.09228 -0.9823
## # of obs: 5320; # of groups: Pop, 6
##  ------------- Likelihood values  -------------
##                          logLik
##        h-likelihood: -414.4578
## logL       (p_v(h)): -411.1534
```

```
summary(BFlowglm2)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (Age | Pop) +
##     (1 | year)
## Estimation of lambda and ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)           -10.21   0.9991 -10.222
## poly(SizeOMars, 3)1   226.93  34.0557   6.663
```

```
## poly(SizeOMars, 3)2   -76.57  17.1002  -4.478
## poly(SizeOMars, 3)3    39.27  11.3977   3.445
## poly(Age, 2)1         133.48  22.3423   5.974
## poly(Age, 2)2         -54.57   8.8053  -6.197
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.252
##    Pop         Age 0.09228 -0.9823
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  9.294e-07
##             --- Coefficients for log(lambda):
##  Group        Term Estimate Cond.SE
##   year (Intercept)   -13.89   129.8
## # of obs: 5320; # of groups: Pop, 6; year, 28
##  ------------- Likelihood values  -------------
##                         logLik
##        h-likelihood: -245.7454
## logL       (p_v(h)): -411.1534
```

```
summary(BFlowglm3)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (Age | Pop) +
##     (SizeOMars | year)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)           -10.31    1.015 -10.159
## poly(SizeOMars, 3)1   229.84   34.922   6.581
## poly(SizeOMars, 3)2   -76.90   17.337  -4.436
## poly(SizeOMars, 3)3    42.04   11.796   3.564
## poly(Age, 2)1         135.06   22.207   6.082
## poly(Age, 2)2         -54.70    8.912  -6.138
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term    Var.   Corr.
##    Pop (Intercept)   2.121
##    Pop         Age 0.08652 -0.9772
##   year (Intercept)   0.289
##   year   SizeOMars 0.00457      -1
## # of obs: 5320; # of groups: Pop, 6; year, 28
##  ------------- Likelihood values  -------------
##                         logLik
##        h-likelihood: -460.4618
## logL       (p_v(h)): -410.1739
## Numerical issue detected; see div_info(<fit object>) for more information.
```

```
summary(BFlowglm4)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (Age | Pop) +
##     (Age | year)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)          -10.18    1.004 -10.137
## poly(SizeOMars, 3)1   226.68   34.059   6.656
## poly(SizeOMars, 3)2   -76.10   17.105  -4.449
## poly(SizeOMars, 3)3    39.07   11.414   3.423
## poly(Age, 2)1         131.79   22.995   5.731
## poly(Age, 2)2         -54.22    8.976  -6.040
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term   Var.   Corr.
##    Pop (Intercept)  2.309
##    Pop         Age 0.0977 -0.9811
##   year (Intercept) 0.1897
##   year         Age 0.0131      -1
## # of obs: 5320; # of groups: Pop, 6; year, 28
##  ------------- Likelihood values  -------------
##                         logLik
##        h-likelihood: -462.8984
## logL       (p_v(h)): -410.6572
```

```
summary(BFlowglm5)
```

```
## formula: Flowering ~ 1 + poly(SizeOMars, 3) + poly(Age, 2) + (SizeOMars +
##     Age | Pop)
## Estimation of ranCoefs by ML (p_v approximation of logL).
## Estimation of fixed effects by h-likelihood approximation.
## family: binomial( link = logit )
##  ------------ Fixed effects (beta) ------------
##                     Estimate Cond. SE t-value
## (Intercept)          -10.28   0.9852 -10.434
## poly(SizeOMars, 3)1   229.28  34.3337   6.678
## poly(SizeOMars, 3)2   -78.27  17.1907  -4.553
## poly(SizeOMars, 3)3    39.35  11.4245   3.444
## poly(Age, 2)1         133.06  22.0510   6.034
## poly(Age, 2)2         -54.61   8.7957  -6.208
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term      Var.   Corr. Corr..1
##    Pop (Intercept)     1.562
##    Pop   SizeOMars 0.0005318  0.8445
##    Pop         Age   0.08771 -0.9946 -0.7841
## # of obs: 5320; # of groups: Pop, 6
```

```
##  ------------- Likelihood values  -------------
##                           logLik
##        h-likelihood: -419.5491
## logL       (p_v(h)): -410.8177
## Numerical issue detected; see div_info(<fit object>) for more information.
```

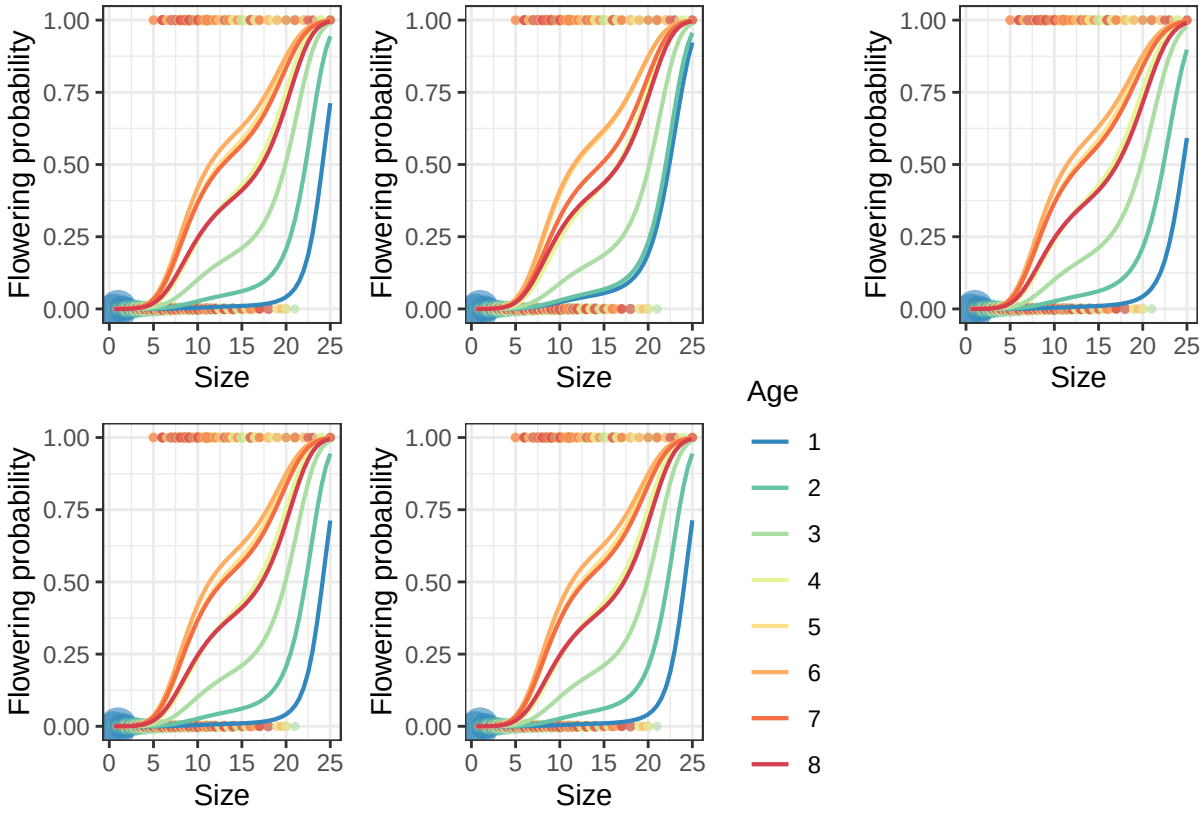
```
AFlowpredict1 <- predict(AFlowglm1, newdata = fake_data)[,1]
AFlowpredict2 <- predict(AFlowglm2, newdata = fake_data)[,1]
AFlowpredict3 <- predict(AFlowglm3, newdata = fake_data)[,1]
AFlowpredict4 <- predict(AFlowglm4, newdata = fake_data)[,1]
AFlowpredict5 <- predict(AFlowglm5, newdata = fake_data)[,1]

BFlowpredict1 <- predict(BFlowglm1, newdata = fake_data)[,1]
BFlowpredict2 <- predict(BFlowglm2, newdata = fake_data)[,1]
BFlowpredict3 <- predict(BFlowglm3, newdata = fake_data)[,1]
BFlowpredict4 <- predict(BFlowglm4, newdata = fake_data)[,1]
BFlowpredict5 <- predict(BFlowglm5, newdata = fake_data)[,1]
```

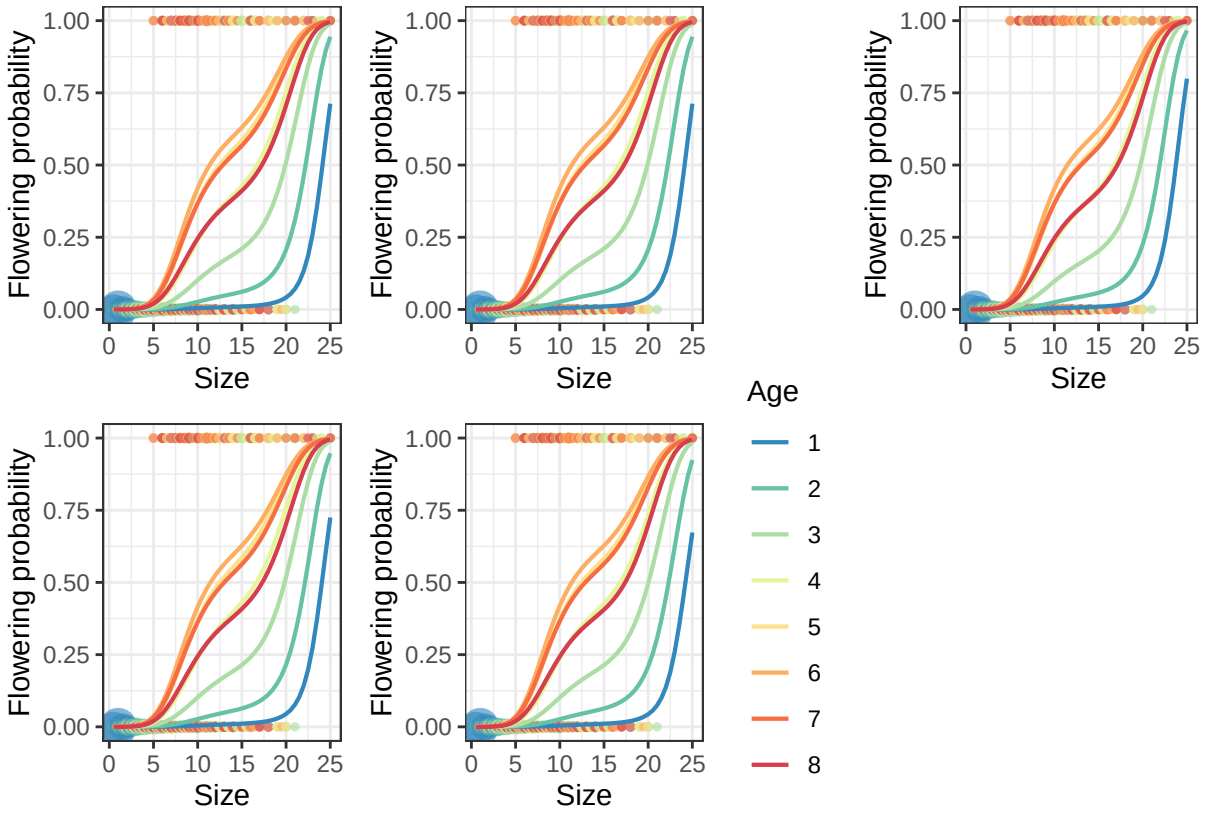
```
plot_Flowering <- function(data = fake_data, prediction) {
  data %>%
  mutate(flow_predi = prediction) %>%
  group_by(SizeOMars, Age) %>%
  summarise(flow_predi = mean(flow_predi),
            .groups = "drop") %>%
  ggplot(aes(x = SizeOMars, y = flow_predi)) +
  geom_count(data=centauree_data,aes(x=SizeOMars,y=Flowering,col=as.factor(Age)),alpha = 0.6,show.legend
  geom_line(aes(color = as.factor(Age)),size=0.75,show.legend = FALSE) +
  theme_bw()+
  ylim(0, 1) +
  labs(x = "Size",
      y = "Flowering probability",
      color = "Age")+
  scale_color_brewer(palette = "Spectral", direction = -1)
}

plot_Floweringbis <- function(data = fake_data, prediction) {
  data %>%
  mutate(flow_predi = prediction) %>%
  group_by(SizeOMars, Age) %>%
  summarise(flow_predi = mean(flow_predi),
            .groups = "drop") %>%
  ggplot(aes(x = SizeOMars, y = flow_predi)) +
  geom_count(data=centauree_data,aes(x=SizeOMars,y=Flowering,col=as.factor(Age)),alpha = 0.6,show.legend
  geom_line(aes(color = as.factor(Age)),size=0.75) +
  theme_bw()+
  ylim(0, 1) +
  labs(x = "Size",
      y = "Flowering probability",
      color = "Age")+
  scale_color_brewer(palette = "Spectral", direction = -1)
}

plot_Flowering2 <- function(data = fake_data, prediction, var, fact) {
  data %>%
```

```
    mutate(flow_predi = prediction) %>%
    filter(SizeOMars==10) %>%
    group_by(!!sym(var), !!sym(fact)) %>%
     summarise(flow_predi = mean(flow_predi),
            .groups = "drop") %>%
    ggplot(aes(x = .data[[var]], y = flow_predi)) +
    geom_line(aes(color = as.factor(.data[[fact]])),show.legend = FALSE) +
    theme_minimal() +
    scale_color_viridis_d(option = "plasma")+
    ylim(0, 1)
}

plot_Flowering2bis <- function(data = fake_data, prediction, var, fact) {
  data %>%
    mutate(flow_predi = prediction) %>%
    filter(SizeOMars==10) %>%
    group_by(!!sym(var), !!sym(fact)) %>%
    summarise(flow_predi = mean(flow_predi),
            .groups = "drop") %>%
    ggplot(aes(x = .data[[var]], y = flow_predi)) +
    geom_line(aes(color = as.factor(.data[[fact]])),show.legend = FALSE) +
    theme_minimal() +
    ylim(0, 1)
}
```

Flowering en fonction de la taille

En fixant la population et l'année : voir l'effet age

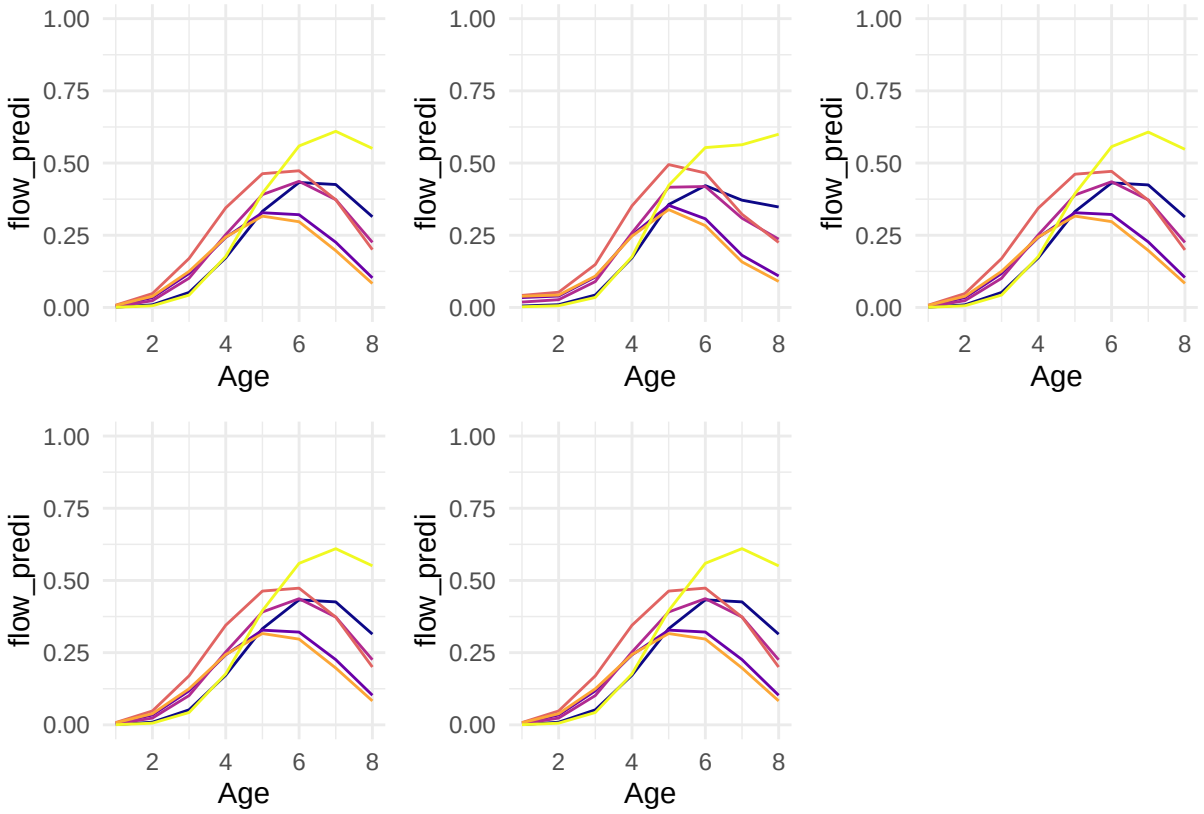


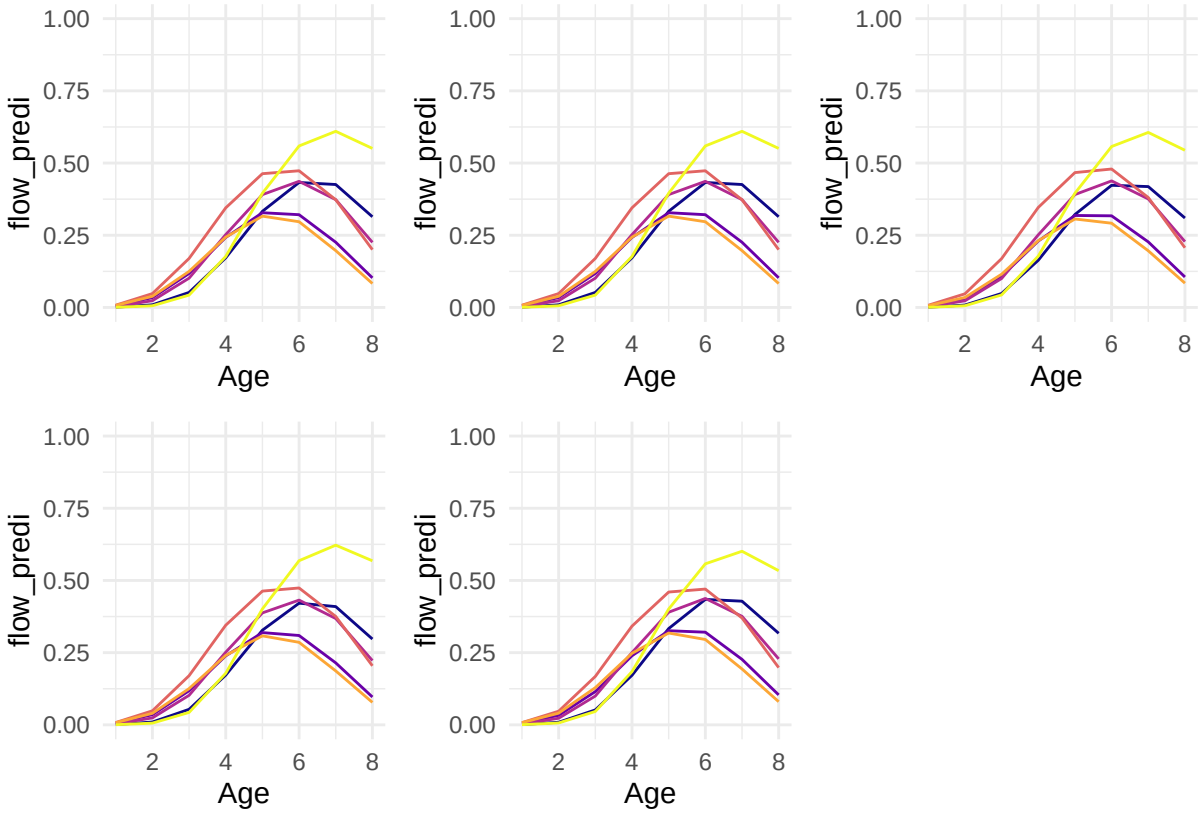


Flowering en fonction de l'age (taille fixé)

En moyennant sur les années : voir l'effet population

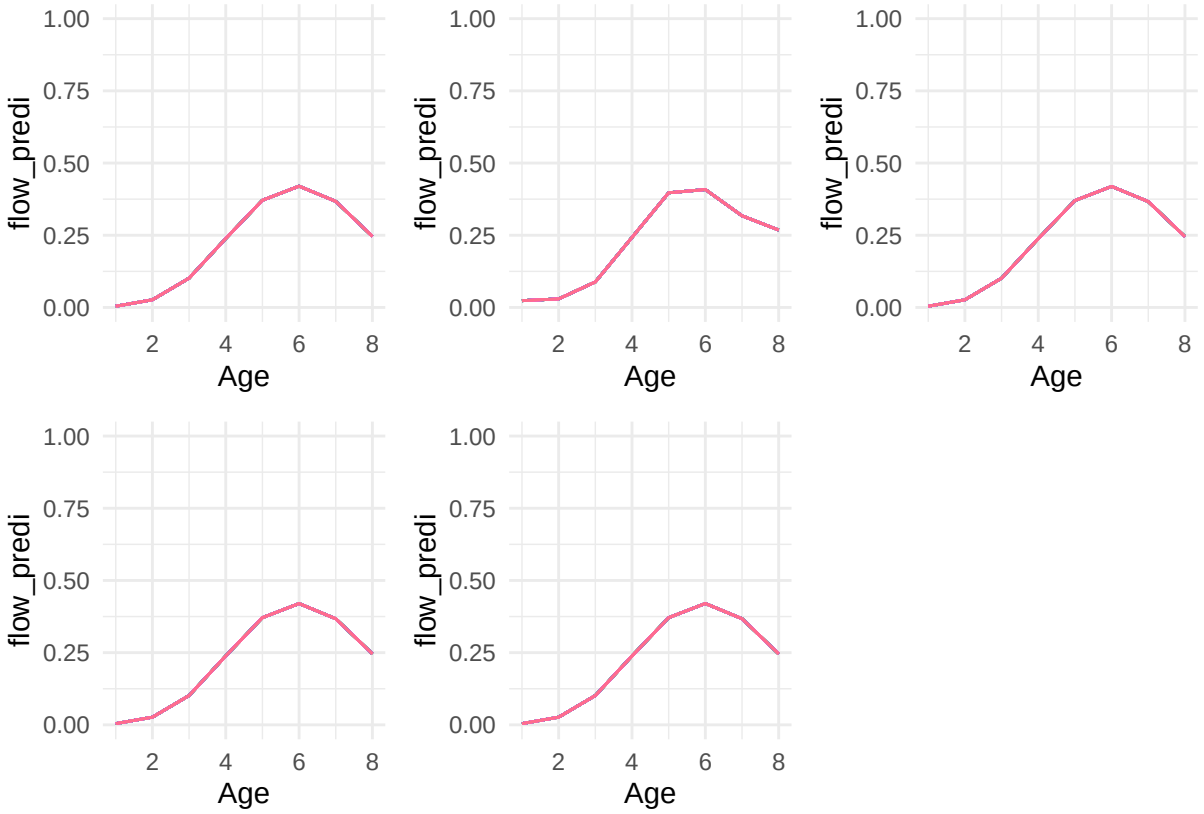
```
var <- "Age"
fact <- "Pop"
```

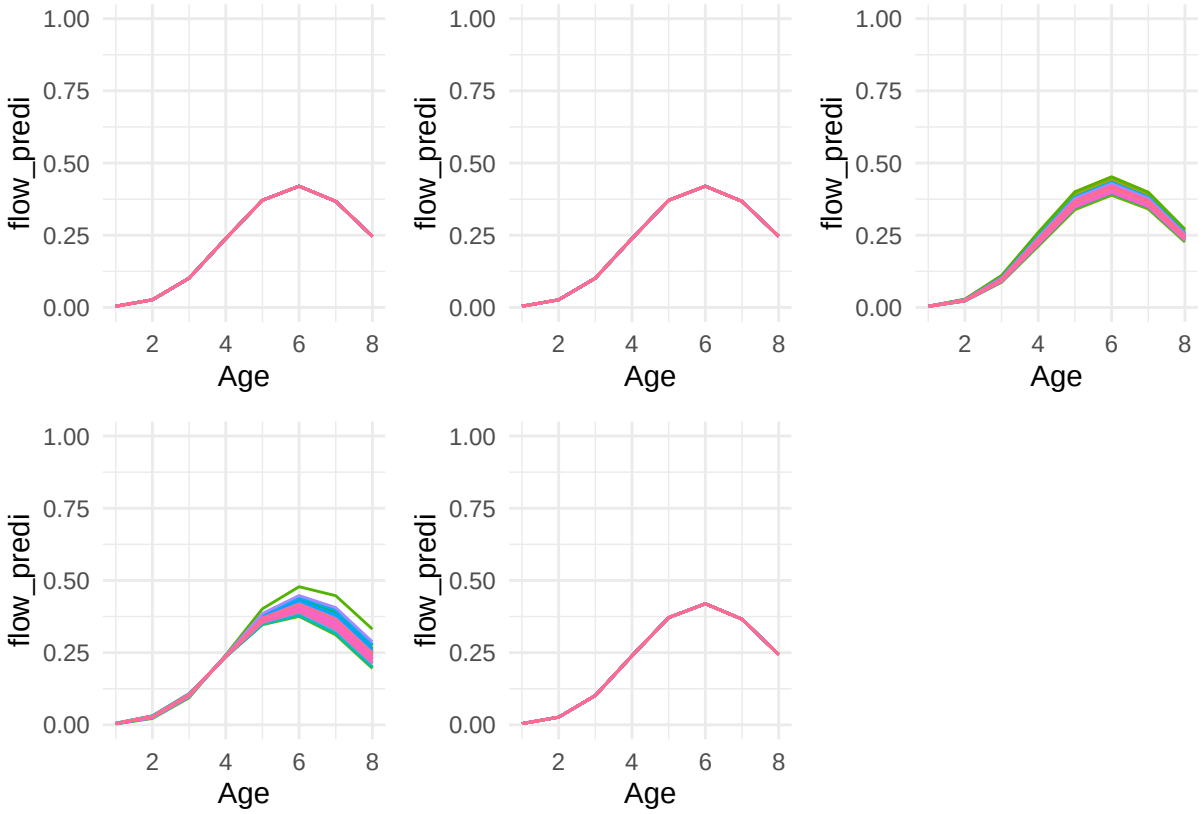

flow_predi
1.00
0.75
0.50
0.25
0.00
2
4
6
8
Age



En moyennant sur les populations : voir l'effet année

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


flow_predi
1.00
0.75
0.50
0.25
0.00
2
4
6
8
Age


flow_predi
1.00
0.75
0.50
0.25
0.00
2
4
6
8
Age