Models Fitted

Loïc Pages

2025-02-11

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 4.5.30) 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.
```

```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 error
```

```r
library(splines)
library(patchwork)

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

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

```r
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])
```

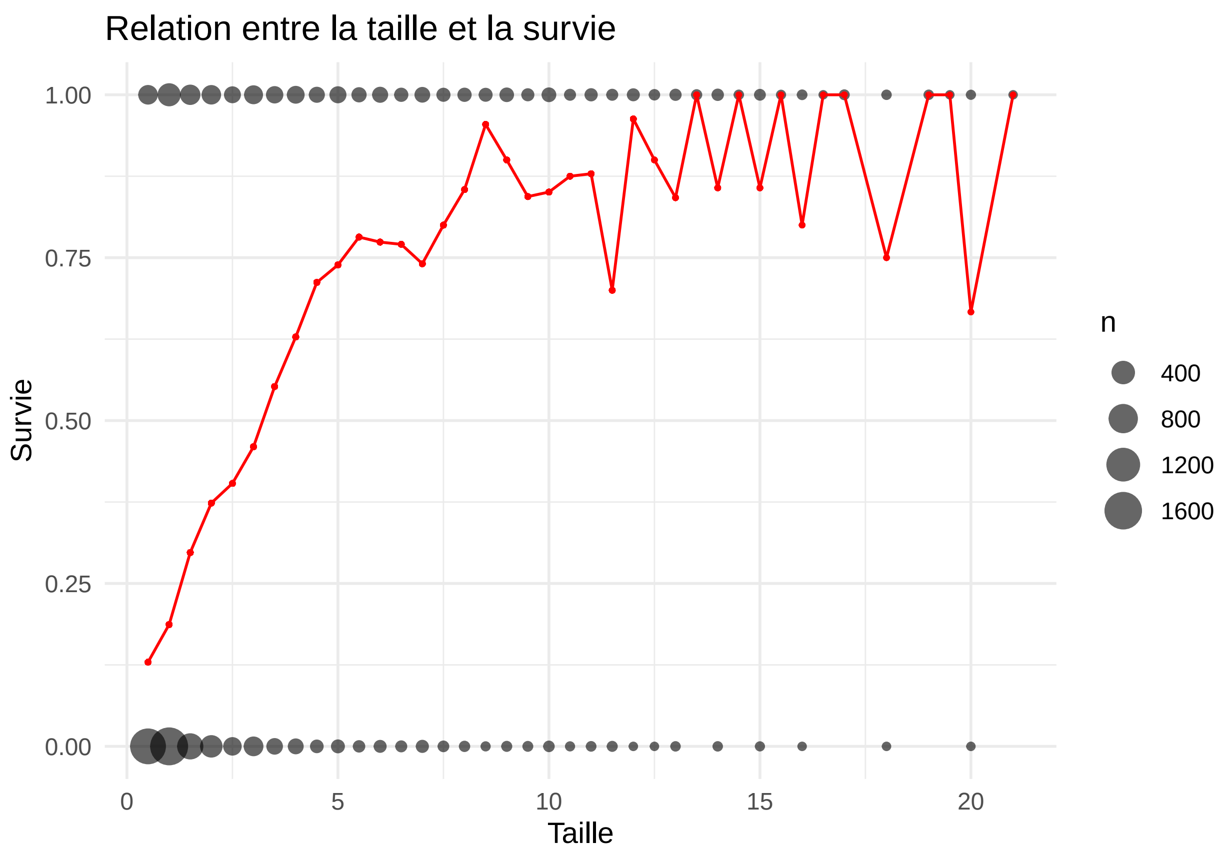
```
## [1] 93
```

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

spaMM.options(separation_max=70)
```

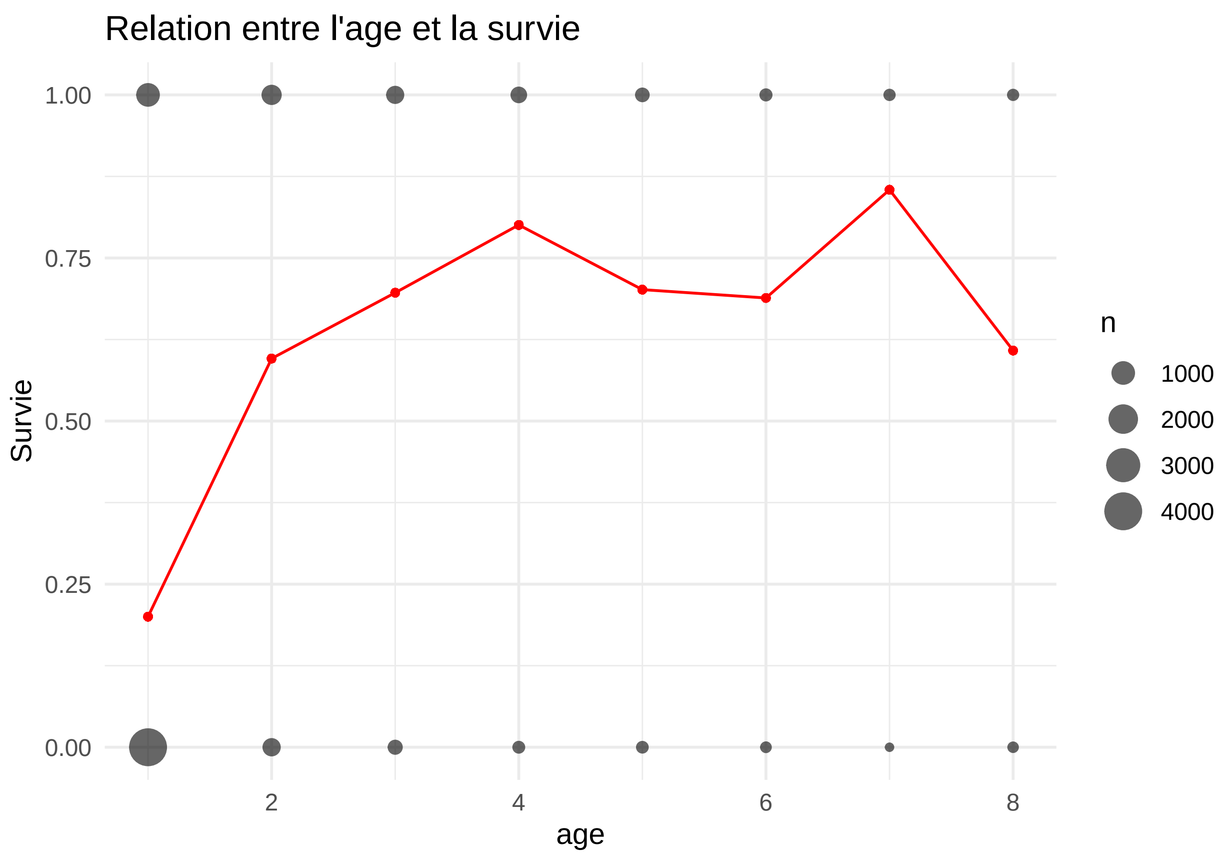
```r
annees <- 1995:2022
populations <- c("Po","Au","Pe","E1","E2","Cr")
taille_range <- seq(0.5, 20, by = 0.5)
age_range <- 1:8

fake_data <- expand.grid(
  year = annees,
  Pop = populations,
  Size0Mars = taille_range,
  age0 = age_range
)

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

Survival probability

```r
survdata <- centauree_data[centauree_data$Flowering0!=1,]
survdata <- survdata[!is.na(survdata$Size1Mars),]


survdata %>%
  group_by(Size0Mars) %>%
  mutate(survivalProba = sum(SurvieMars, na.rm = TRUE) / n()) %>%
  ggplot(aes(x = Size0Mars, y = SurvieMars)) +
  geom_count(alpha = 0.6) +  # Points dimensionnés selon la fréquence
  geom_point(aes(y = survivalProba), color = "red", size = 0.5) +
  geom_line(aes(y = survivalProba), color = "red") +
  labs(title = "Relation entre la taille et la survie",
    x = "Taille",
    y = "Survie") +
  ylim(0, 1) +
  theme_minimal()
```




```r
survdata %>%
  group_by(age0) %>%
  mutate(survivalProba = sum(SurvieMars, na.rm = TRUE) / n()) %>%
  ggplot(aes(x = age0, y = SurvieMars)) +
    geom_count(alpha = 0.6) +  # Points dimensionnés selon la fréquence
    geom_point(aes(x = age0, y = survivalProba), color = "red", size = 1) +
    geom_line(aes(x = age0, y = survivalProba), color = "red") +
    labs(title = "Relation entre l'age et la survie",
    x = "age",
    y = "Survie") +
  ylim(0, 1) +
  theme_minimal()
```




```
Survglm1 <- fitme(SurvieMars ~ 1+ bs(Size0Mars,df=5,degree=3) + bs(age0,degree=3,knots = 6.5)+ (Size0Ma
                 family=binomial,
                 data=survdata,
                 method="PQL/L")

Survglm2 <- fitme(SurvieMars ~ 1 + bs(Size0Mars,df=5,degree=3) + bs(age0,degree=3,knots = 6.5)
+ (Size0Mars + age0|year) + (age0|Pop) ,
                 family=binomial,
                 data=survdata,
                 method="PQL/L")

Survglm3 <- fitme(SurvieMars ~ 1 + bs(Size0Mars,df=5,degree=3) + bs(age0,degree=3,knots = c(1.5,6.5)) +
                 family=binomial,
                 data=survdata,
                 method="PQL/L")
```

```
## Warning in (function (formula, resid.formula = NULL, data, prior.weights, :
## 'c(' detected in formula: did you mean cbind() for binomial response or for
## poly()?
```

```
Survglm4 <- fitme(SurvieMars ~ 1 + bs(Size0Mars,df=5,degree=3) + bs(age0,degree=3,knots = c(1.5,6.5))
+ (Size0Mars + age0|year) + (age0|Pop) ,
                 family=binomial,
                 data=survdata,
                 method="PQL/L")
```

```
## Warning in (function (formula, resid.formula = NULL, data, prior.weights, :
## 'c(' detected in formula: did you mean cbind() for binomial response or for
## poly()?
```

```r
Survglm5 <- fitme(SurvieMars ~ 1 + bs(Size0Mars,df=5,degree=3) + bs(age0,degree=3,knots = 6.5)
+ (Size0Mars|year) + (Size0Mars + age0|Pop) ,
                 family=binomial,
                 data=survdata,
                 method="PQL/L")
```

```r
summary(Survglm1)
```

```
## formula: SurvieMars ~ 1 + bs(Size0Mars, df = 5, degree = 3) + bs(age0,
##     degree = 3, knots = 6.5) + (Size0Mars | year) + (age0 | 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)                           -2.2406   0.2462 -9.1006
## bs(Size0Mars, df = 5, degree = 3)1   0.1103   0.1999  0.5517
## bs(Size0Mars, df = 5, degree = 3)2   0.8657   0.1272  6.8056
## bs(Size0Mars, df = 5, degree = 3)3   4.2121   0.4778  8.8158
## bs(Size0Mars, df = 5, degree = 3)4   3.8150   0.9696  3.9348
## bs(Size0Mars, df = 5, degree = 3)5   3.5933   1.2790  2.8095
## bs(age0, degree = 3, knots = 6.5)1   2.5146   0.2908  8.6480
## bs(age0, degree = 3, knots = 6.5)2  -0.8001   0.5510 -1.4520
## bs(age0, degree = 3, knots = 6.5)3   2.1587   0.6255  3.4513
## bs(age0, degree = 3, knots = 6.5)4   0.1423   0.4088  0.3480
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term     Var.  Corr.
##   year (Intercept)     1.02
##   year   Size0Mars  0.01061 -0.672
##    Pop (Intercept)   0.1431
##    Pop        age0 0.009096  -0.76
## # of obs: 7293; # of groups: year, 27; Pop, 6
##  ------------- Likelihood values  -------------
##                         logLik
##       h-likelihood: -3345.794
## logL       (p_v(h)): -3347.972
```

```r
summary(Survglm2)
```

```
## formula: SurvieMars ~ 1 + bs(Size0Mars, df = 5, degree = 3) + bs(age0,
##     degree = 3, knots = 6.5) + (Size0Mars + age0 | year) + (age0 |
##     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)                         -2.24874   0.2508 -8.9673
## bs(Size0Mars, df = 5, degree = 3)1  0.09729   0.2001  0.4862
## bs(Size0Mars, df = 5, degree = 3)2  0.87965   0.1261  6.9774
## bs(Size0Mars, df = 5, degree = 3)3  4.33060   0.4547  9.5240
## bs(Size0Mars, df = 5, degree = 3)4  3.76265   0.9328  4.0336
## bs(Size0Mars, df = 5, degree = 3)5  3.44384   1.2172  2.8294
## bs(age0, degree = 3, knots = 6.5)1  2.48418   0.3082  8.0608
## bs(age0, degree = 3, knots = 6.5)2 -0.84350   0.5967 -1.4135
## bs(age0, degree = 3, knots = 6.5)3  2.18330   0.7078  3.0848
## bs(age0, degree = 3, knots = 6.5)4  0.06919   0.5440  0.1272
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term      Var.   Corr. Corr..1
##   year (Intercept)     1.304
##   year   Size0Mars 0.0005913      -1
##   year        age0   0.04756  -0.663   0.663
##    Pop (Intercept)    0.1543
##    Pop        age0   0.01036 -0.7826
## # of obs: 7293; # of groups: year, 27; Pop, 6
##  ------------- Likelihood values  -------------
##                        logLik
##       h-likelihood: -3359.315
## logL       (p_v(h)): -3341.086
```

```r
summary(Survglm3)
```

```
## formula: SurvieMars ~ 1 + bs(Size0Mars, df = 5, degree = 3) + bs(age0,
##     degree = 3, knots = c(1.5, 6.5)) + (Size0Mars | year) + (age0 |
##     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)                                 -2.23985   0.2463 -9.09280
## bs(Size0Mars, df = 5, degree = 3)1           0.10932   0.1999  0.54682
## bs(Size0Mars, df = 5, degree = 3)2           0.86720   0.1272  6.81565
## bs(Size0Mars, df = 5, degree = 3)3           4.22462   0.4787  8.82432
## bs(Size0Mars, df = 5, degree = 3)4           3.80246   0.9711  3.91554
## bs(Size0Mars, df = 5, degree = 3)5           3.59587   1.2796  2.81021
## bs(age0, degree = 3, knots = c(1.5, 6.5))1  0.01095   0.3491  0.03138
## bs(age0, degree = 3, knots = c(1.5, 6.5))2  2.61115   0.5908  4.41987
## bs(age0, degree = 3, knots = c(1.5, 6.5))3 -0.95676   0.7692 -1.24392
## bs(age0, degree = 3, knots = c(1.5, 6.5))4  2.37011   0.7184  3.29903
## bs(age0, degree = 3, knots = c(1.5, 6.5))5  0.13872   0.4095  0.33875
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term     Var.   Corr.
##   year (Intercept)    1.021
##   year   Size0Mars  0.01077 -0.6711
##    Pop (Intercept)   0.1436
```

```
##    Pop        age0 0.009156 -0.7602
## # of obs: 7293; # of groups: year, 27; Pop, 6
##  ------------- Likelihood values  -------------
##                         logLik
##        h-likelihood: -3345.466
## logL       (p_v(h)): -3347.771
```

```r
summary(Survglm4)
```

```
## formula: SurvieMars ~ 1 + bs(Size0Mars, df = 5, degree = 3) + bs(age0,
##     degree = 3, knots = c(1.5, 6.5)) + (Size0Mars + age0 | year) +
##     (age0 | 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)                                -2.2482   0.2508 -8.9626
## bs(Size0Mars, df = 5, degree = 3)1          0.0967   0.2001  0.4833
## bs(Size0Mars, df = 5, degree = 3)2          0.8804   0.1261  6.9830
## bs(Size0Mars, df = 5, degree = 3)3          4.3379   0.4554  9.5250
## bs(Size0Mars, df = 5, degree = 3)4          3.7544   0.9336  4.0213
## bs(Size0Mars, df = 5, degree = 3)5          3.4392   1.2172  2.8256
## bs(age0, degree = 3, knots = c(1.5, 6.5))1   0.1132   0.3500  0.3234
## bs(age0, degree = 3, knots = c(1.5, 6.5))2   2.4172   0.5974  4.0460
## bs(age0, degree = 3, knots = c(1.5, 6.5))3  -0.8176   0.8049 -1.0158
## bs(age0, degree = 3, knots = c(1.5, 6.5))4   2.2926   0.7865  2.9149
## bs(age0, degree = 3, knots = c(1.5, 6.5))5   0.0687   0.5444  0.1262
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##         --- Random-coefficients Cov matrices:
##  Group        Term      Var.   Corr. Corr..1
##   year (Intercept)     1.303
##   year   Size0Mars 0.0006054      -1
##   year        age0   0.04765 -0.6614  0.6614
##    Pop (Intercept)    0.1546
##    Pop        age0   0.01038 -0.7827
## # of obs: 7293; # of groups: year, 27; Pop, 6
##  ------------- Likelihood values  -------------
##                         logLik
##        h-likelihood: -3359.220
## logL       (p_v(h)): -3341.027
```

```r
summary(Survglm5)
```

```
## formula: SurvieMars ~ 1 + bs(Size0Mars, df = 5, degree = 3) + bs(age0,
##     degree = 3, knots = 6.5) + (Size0Mars | year) + (Size0Mars +
##     age0 | 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)                        -2.2250   0.2397 -9.2840
## bs(Size0Mars, df = 5, degree = 3)1   0.1091   0.1999  0.5459
## bs(Size0Mars, df = 5, degree = 3)2   0.8570   0.1278  6.7070
## bs(Size0Mars, df = 5, degree = 3)3   4.1960   0.4967  8.4476
## bs(Size0Mars, df = 5, degree = 3)4   3.9689   1.0148  3.9109
## bs(Size0Mars, df = 5, degree = 3)5   3.6233   1.3663  2.6519
## bs(age0, degree = 3, knots = 6.5)1   2.5437   0.2918  8.7182
## bs(age0, degree = 3, knots = 6.5)2  -0.8567   0.5554 -1.5426
## bs(age0, degree = 3, knots = 6.5)3   2.2147   0.6319  3.5047
## bs(age0, degree = 3, knots = 6.5)4   0.1441   0.4172  0.3455
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##          --- Random-coefficients Cov matrices:
##  Group        Term     Var.   Corr. Corr..1
##   year (Intercept)    1.011
##   year   Size0Mars  0.01131 -0.6471
##    Pop (Intercept)   0.1314
##    Pop   Size0Mars 0.001472   0.353
##    Pop        age0 0.009911 -0.9029 -0.3149
## # of obs: 7293; # of groups: year, 27; Pop, 6
##  ------------- Likelihood values  -------------
##                        logLik
##       h-likelihood: -3348.440
## logL       (p_v(h)): -3346.683
```

```r
Survpredict1 <- predict(Survglm1, newdata = fake_data)[,1]
Survpredict2 <- predict(Survglm2, newdata = fake_data)[,1]
Survpredict3 <- predict(Survglm3, newdata = fake_data)[,1]
Survpredict4 <- predict(Survglm4, newdata = fake_data)[,1]
Survpredict5 <- predict(Survglm5, newdata = fake_data)[,1]
```

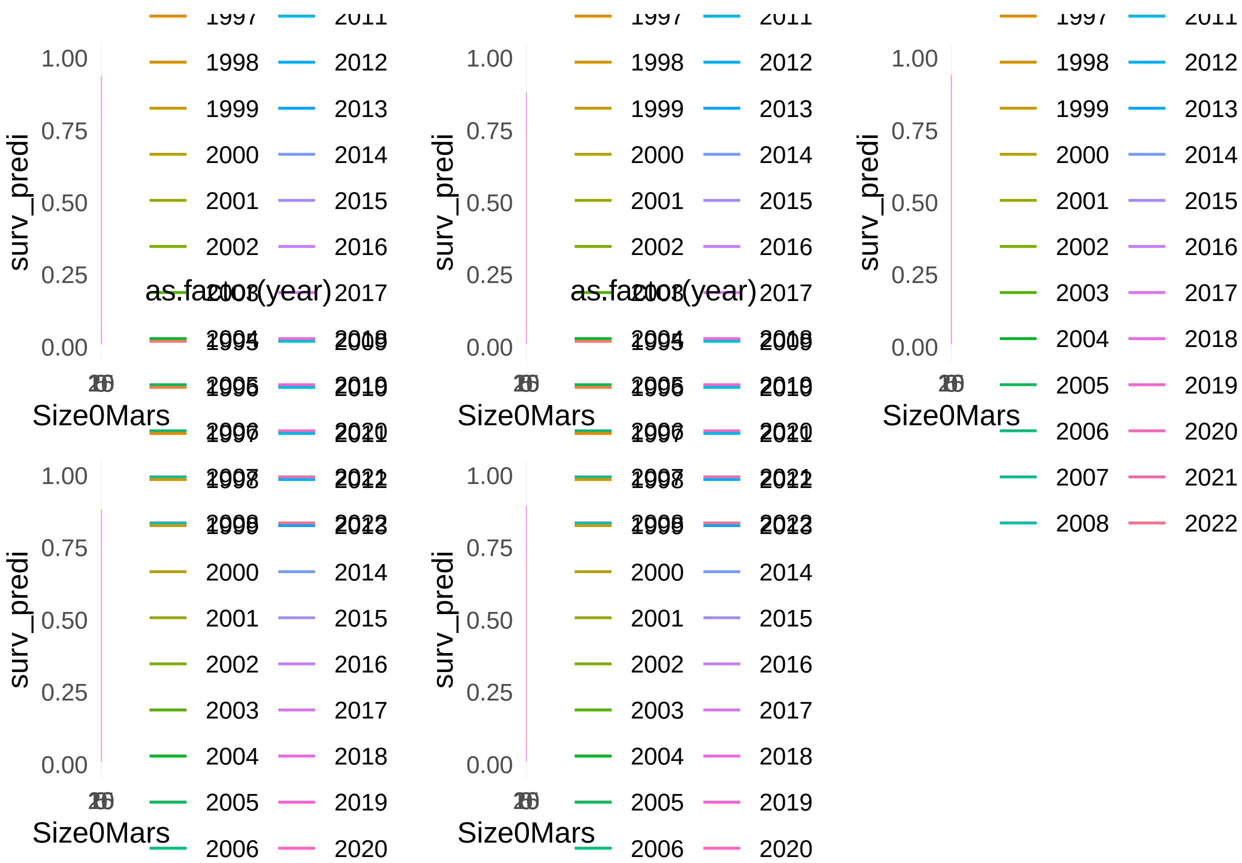
```r
plot_survie <- function(data = fake_data, prediction, var, c1, valc1 = 1, c2, valc2 = "Au", fact) {
  data %>%
    mutate(surv_predi = prediction) %>%
    filter(!!sym(c1) == valc1, !!sym(c2) == valc2) %>%
    ggplot(aes(x = .data[[var]], y = surv_predi)) +
    geom_line(aes(color = as.factor(.data[[fact]]))) +
    theme_minimal() +
    ylim(0, 1)
}
```

Survie en fonction de la taille

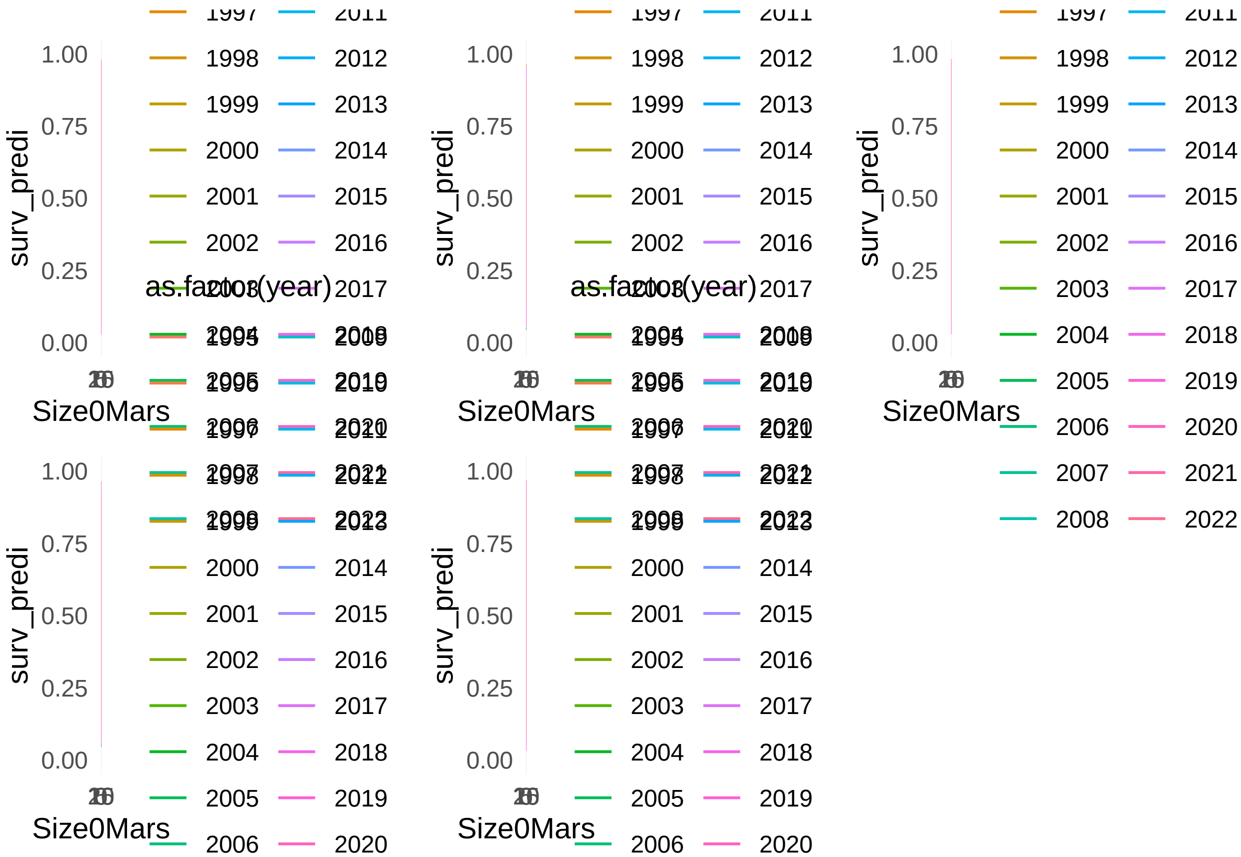
En fixant la population : voir l’effet année

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

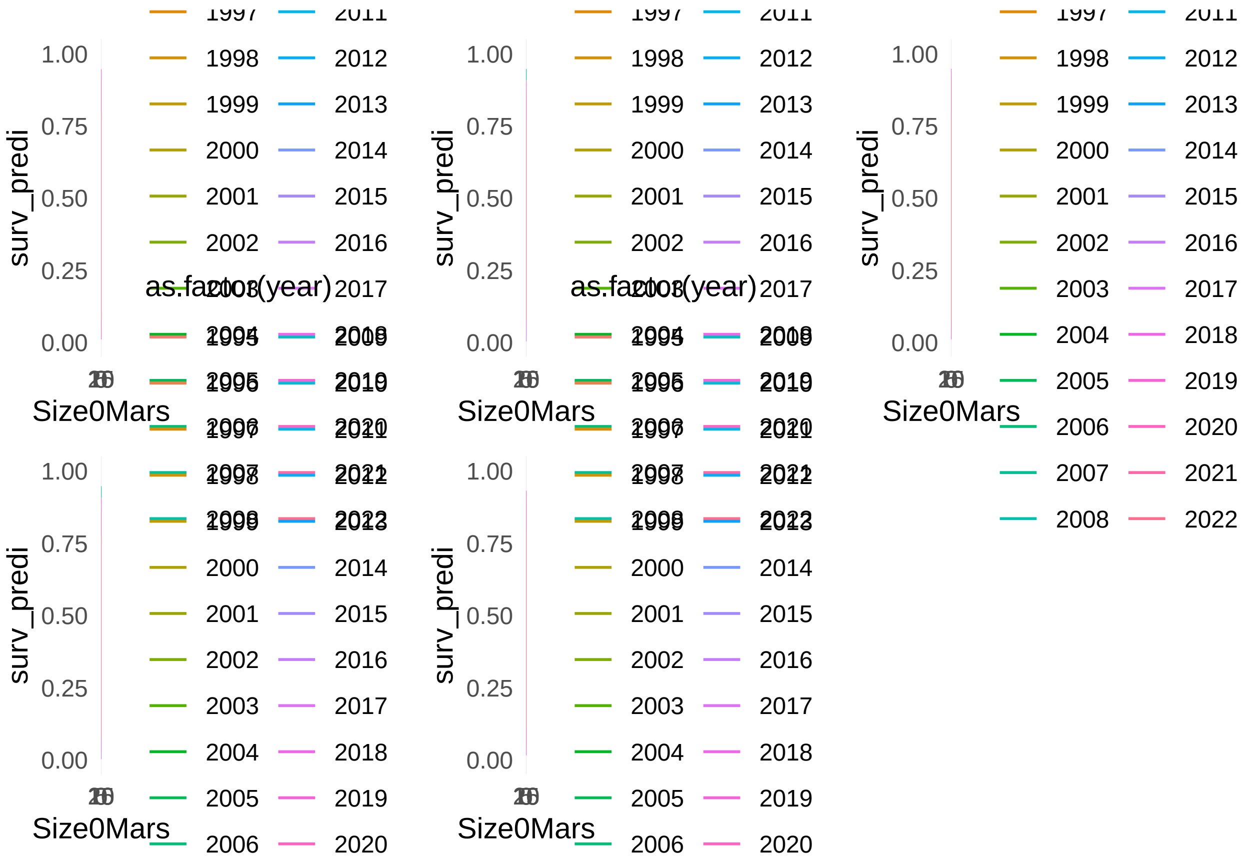
```r
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```




```r
valc1 <- 3
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

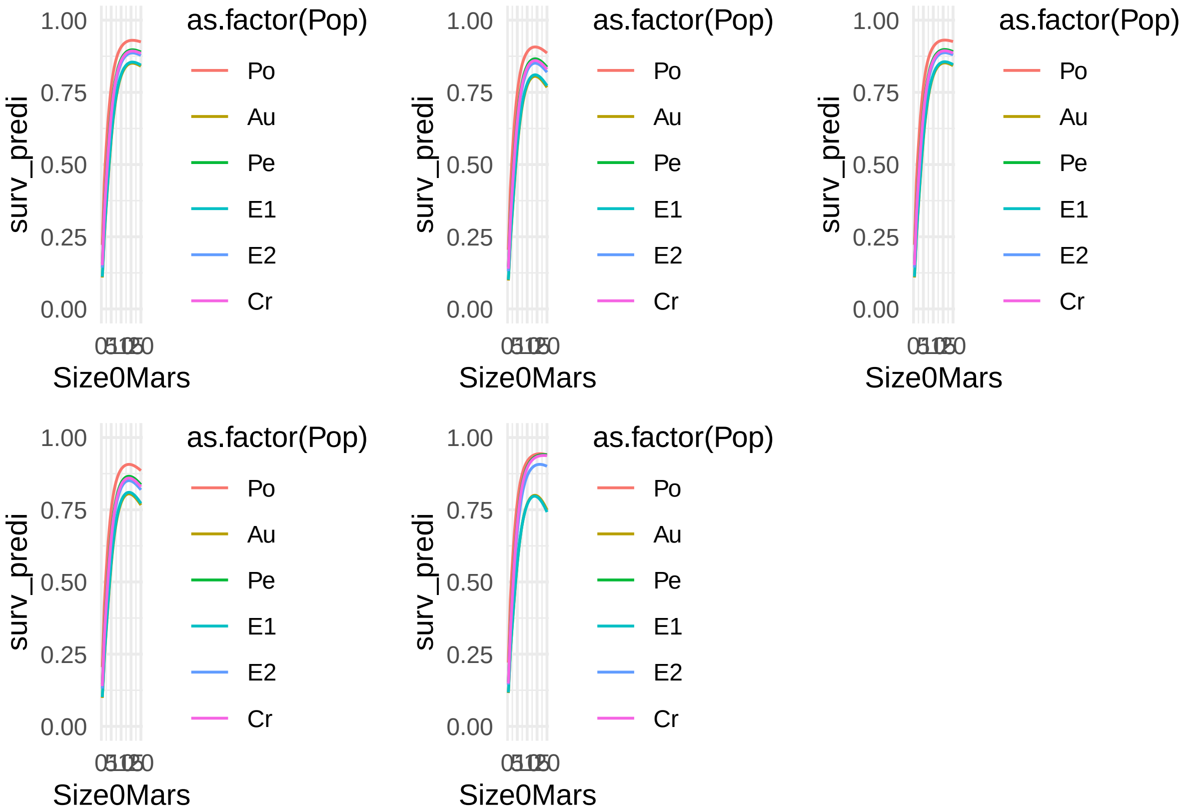



```r
valc1 <- 8
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

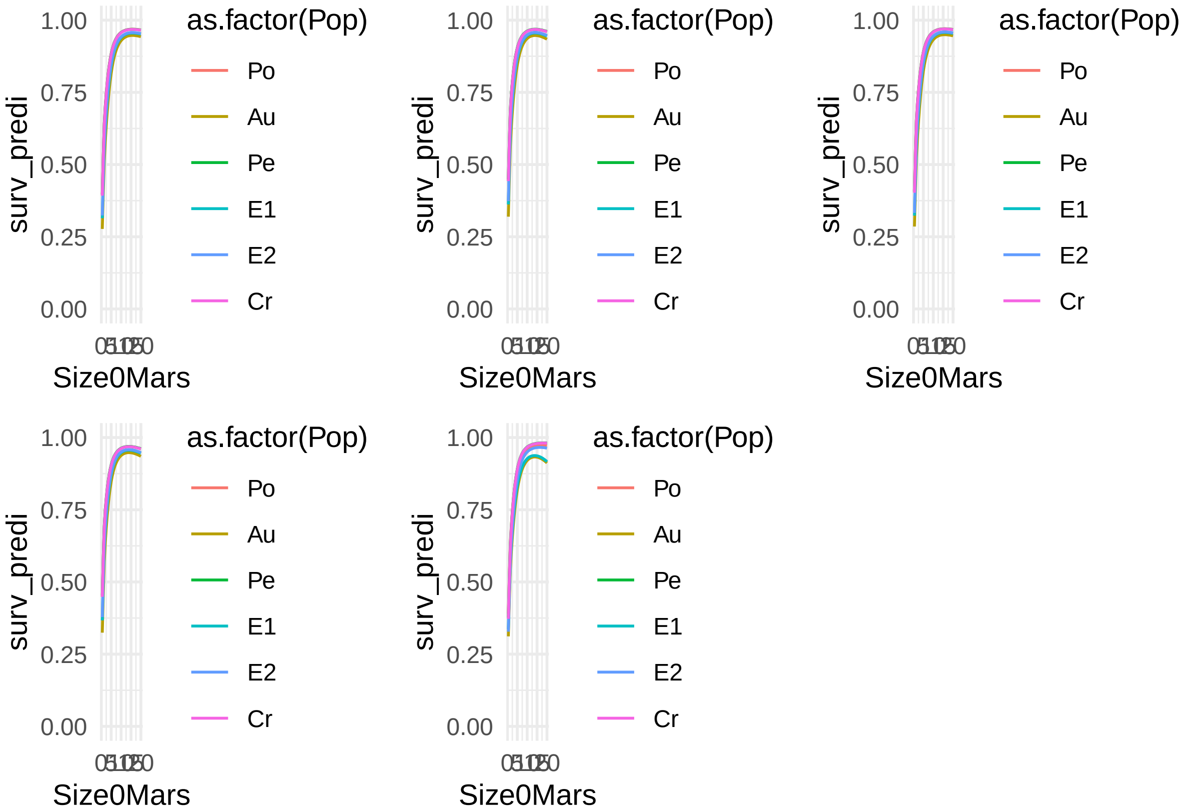


En fixant l'année : voir l'effet population

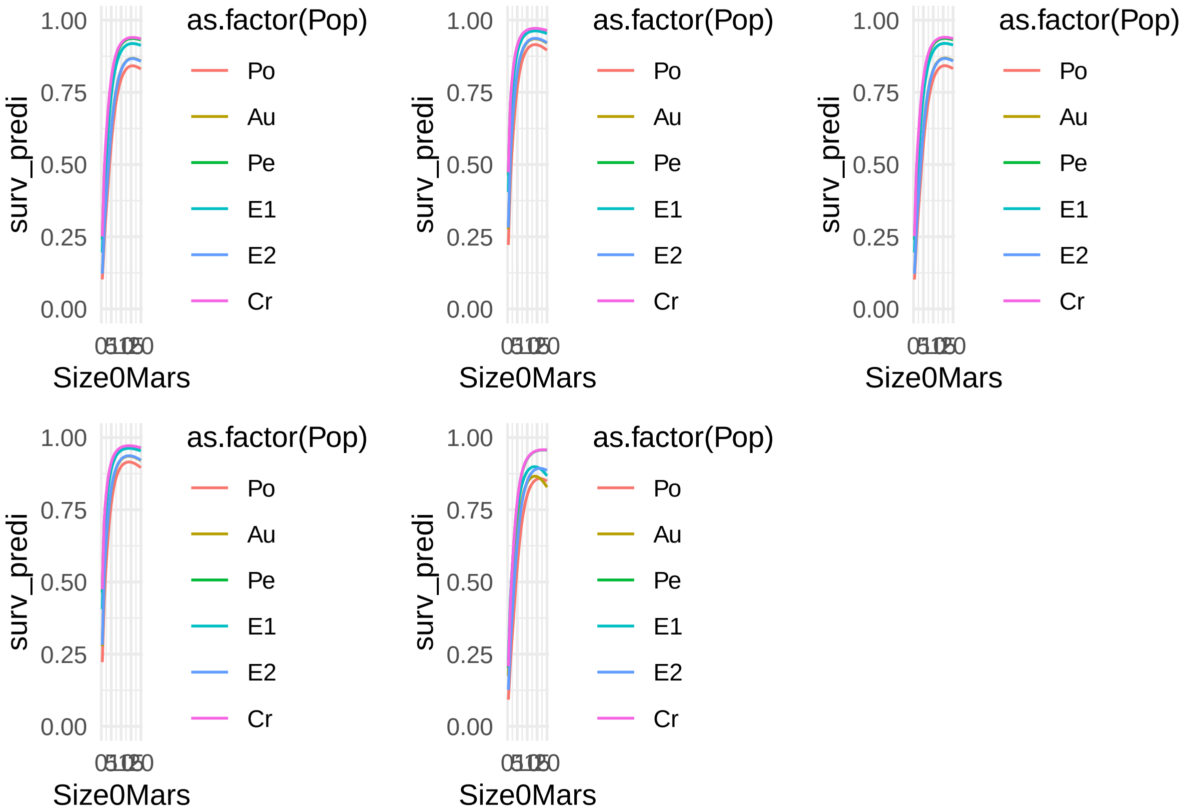
```
var <- "Size0Mars"
c1 <- "age0"
valc1 <- 1
c2 <- "year"
valc2 <- 2000
fact <- "Pop"

wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```




```r
valc1 <- 3
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```



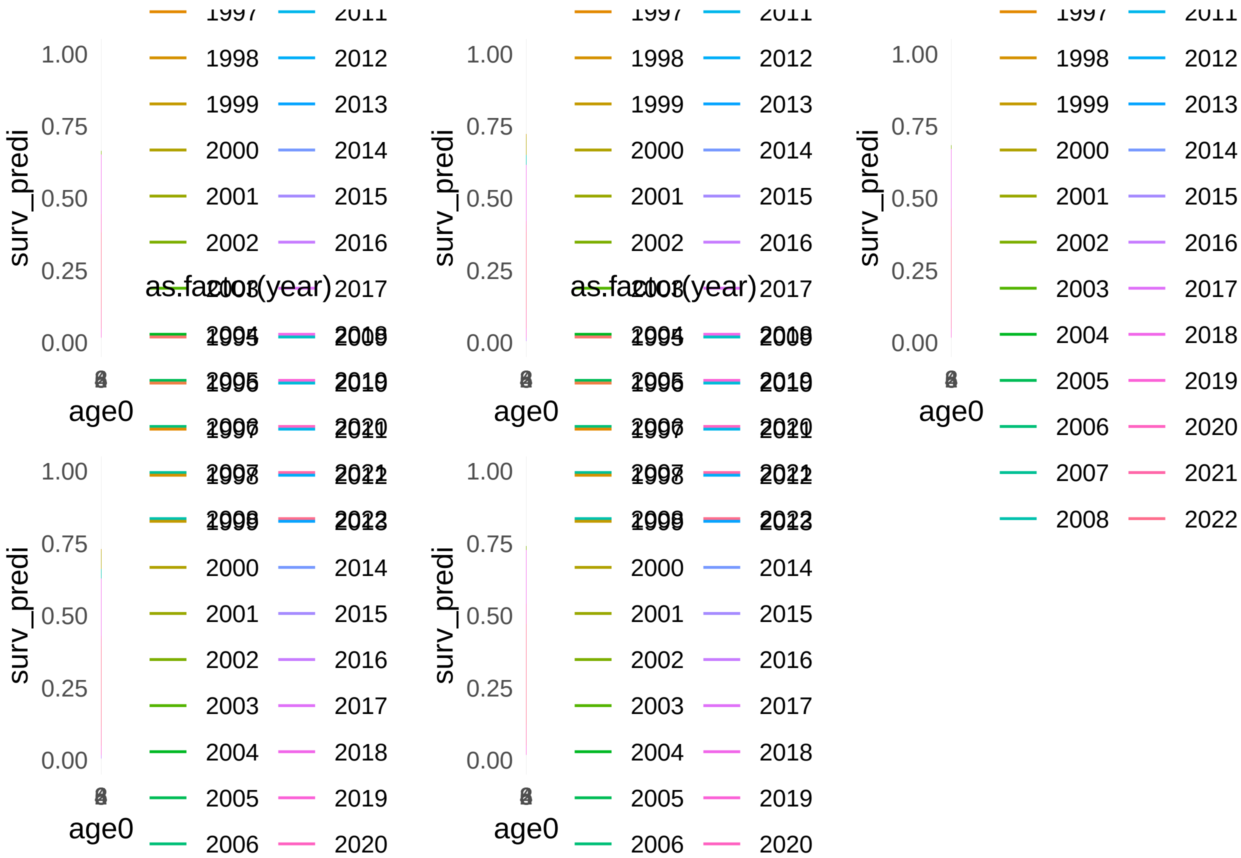

```r
valc1 <- 8
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```



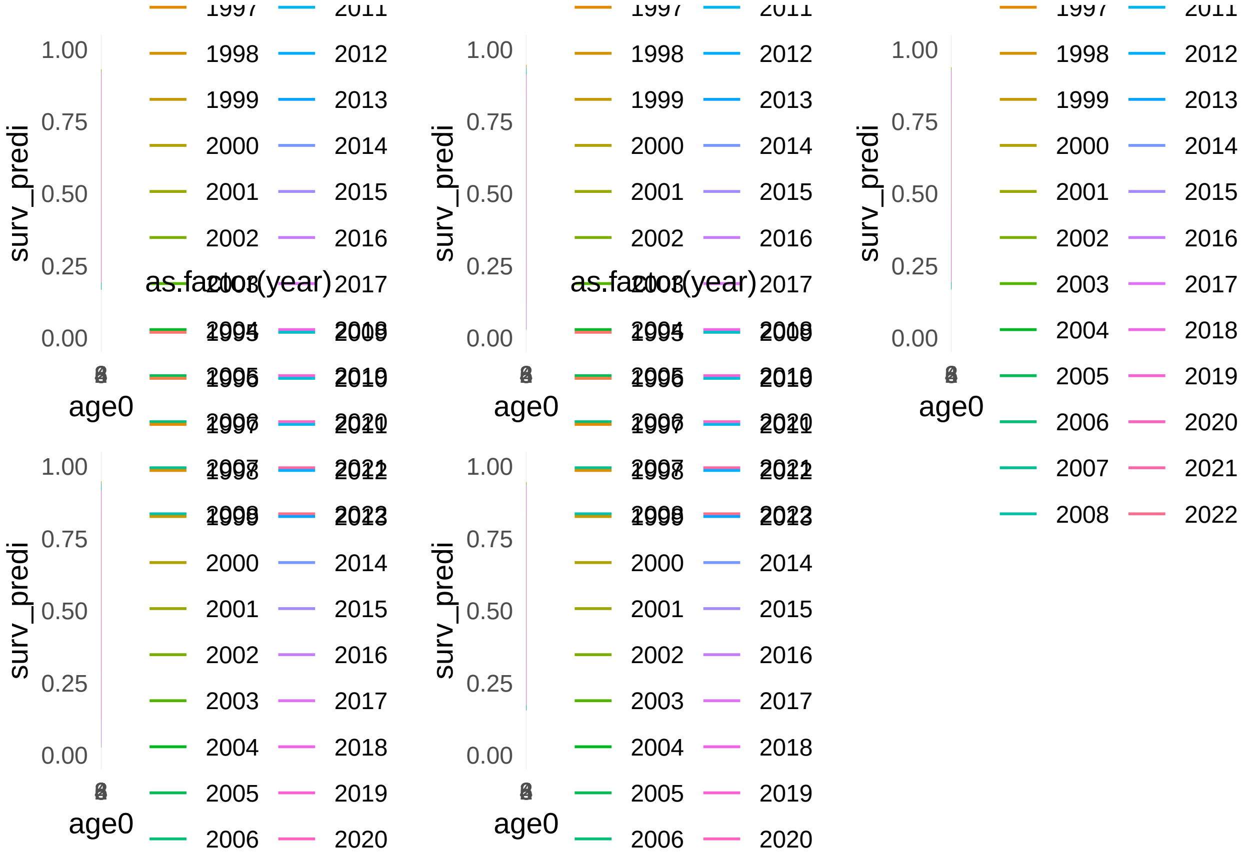
Survie en fonction de l'age

En fixant la population : voir l'effet année

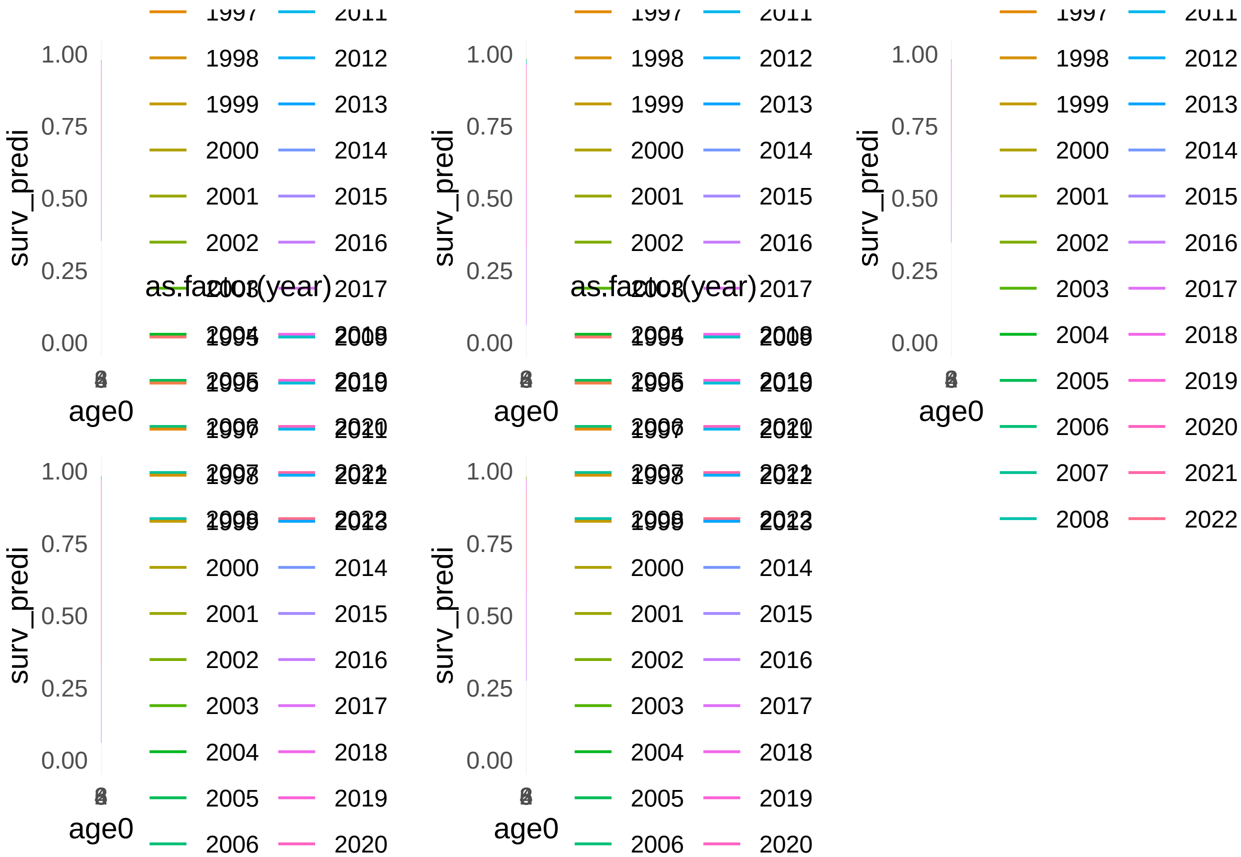
```
var <- "age0"
c1 <- "Size0Mars"
valc1 <- 1
c2 <- "Pop"
valc2 <- "Au"
fact <- "year"

wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```




```r
valc1 <- 5
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

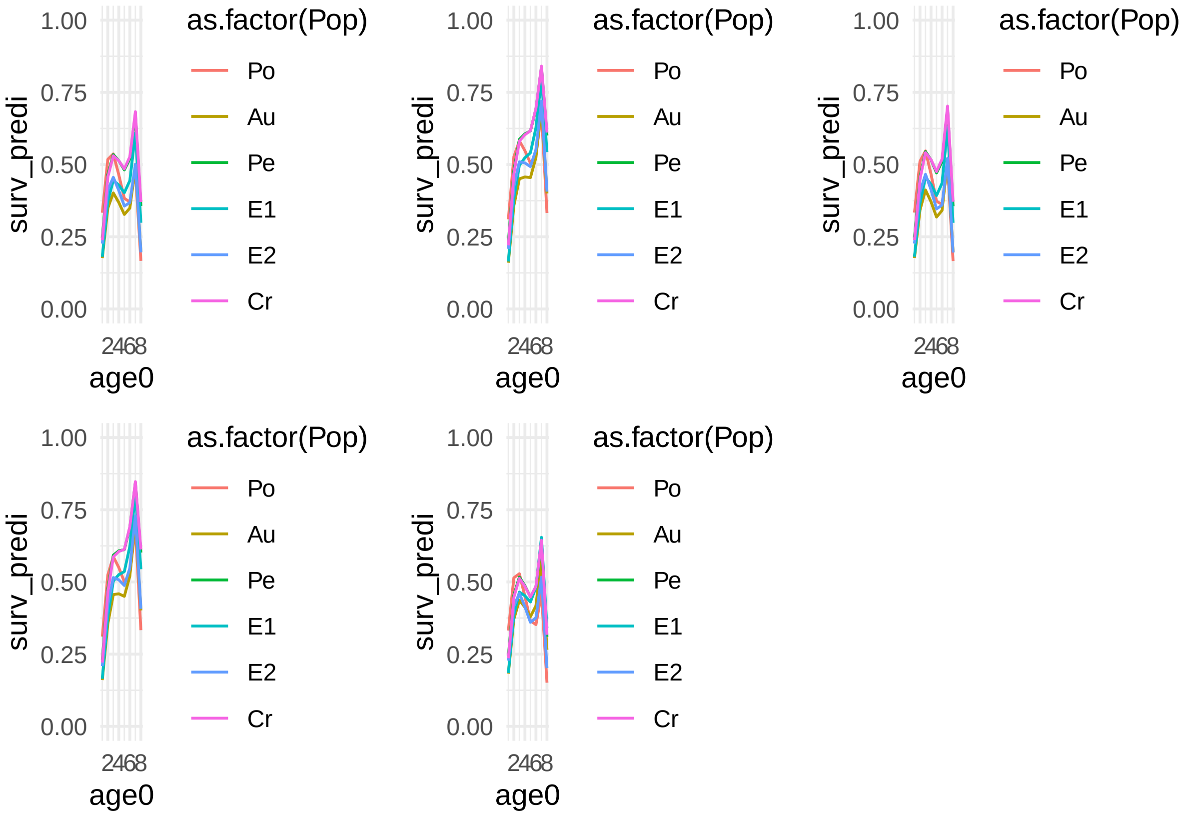



```
valc1 <- 10
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

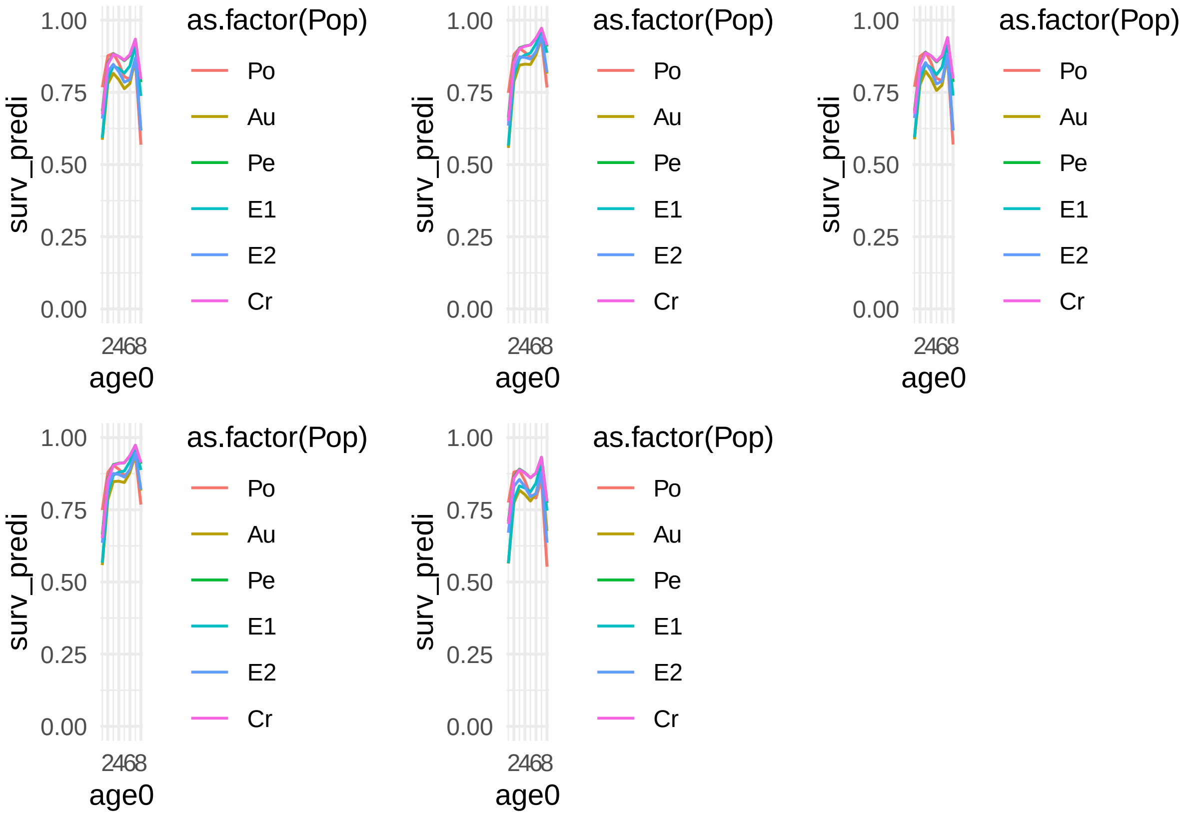


En fixant l'année : voir l'effet population

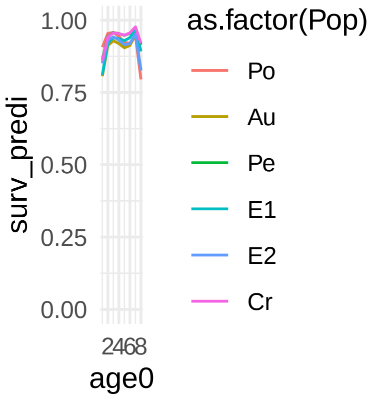
```
var <- "age0"
c1 <- "Size0Mars"
valc1 <- 1
c2 <- "year"
valc2 <- 2000
fact <- "Pop"

wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

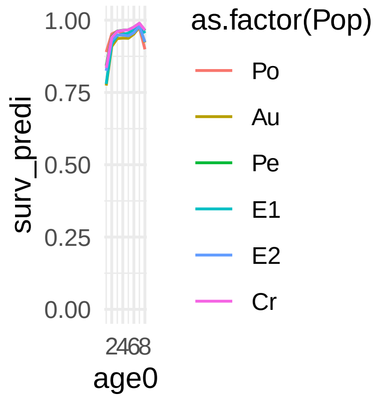


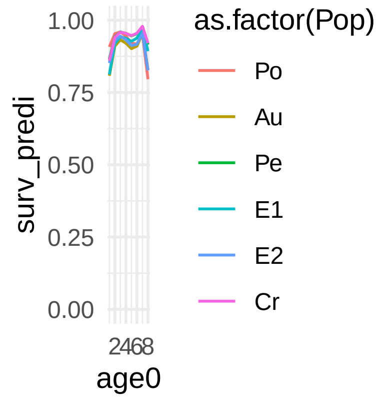

```r
valc1 <- 5
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

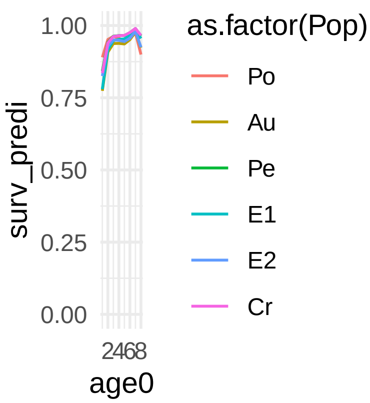


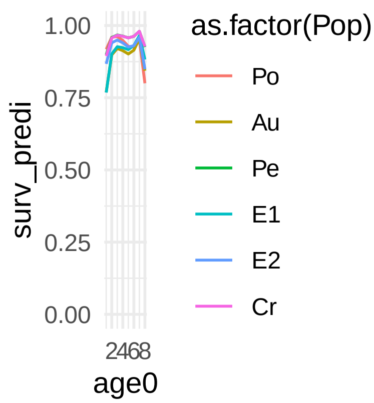

```r
valc1 <- 10
wrap_plots(
plot_survie(prediction = Survpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_survie(prediction = Survpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```







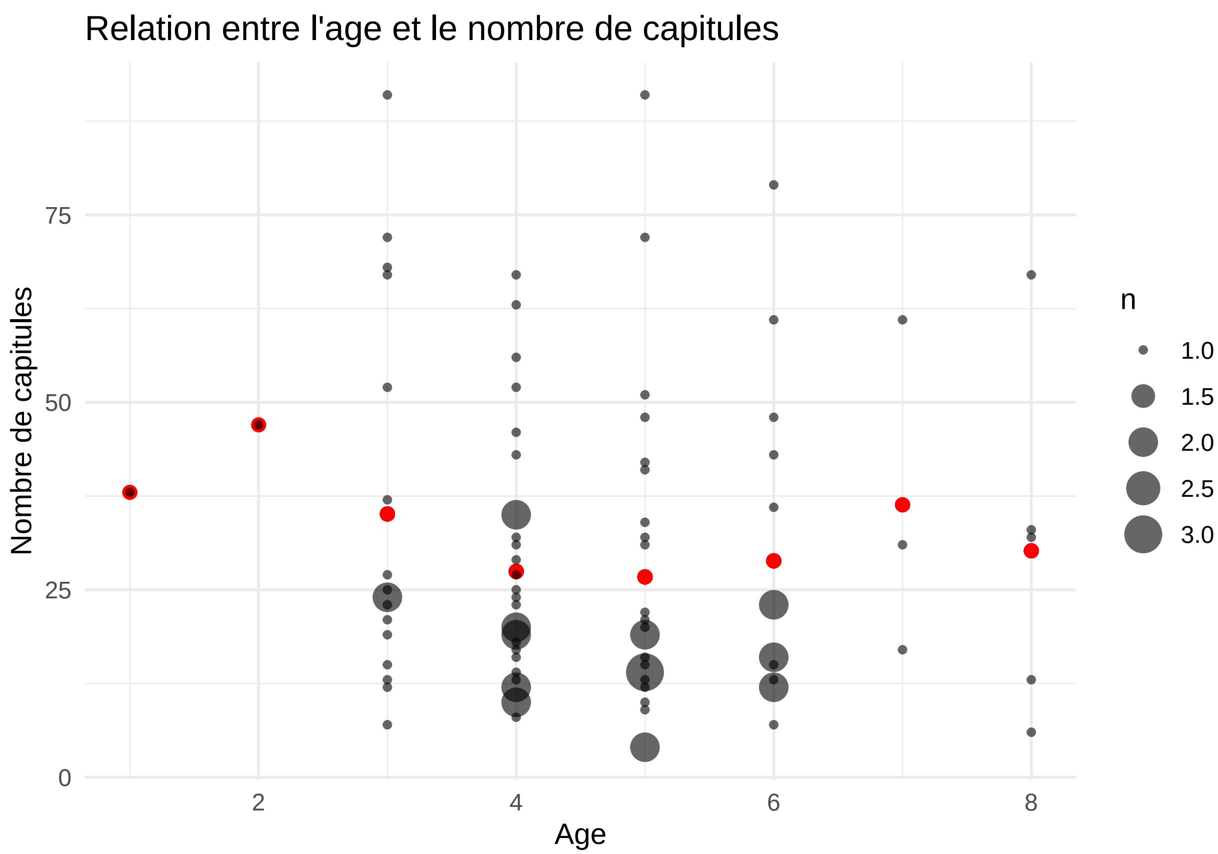




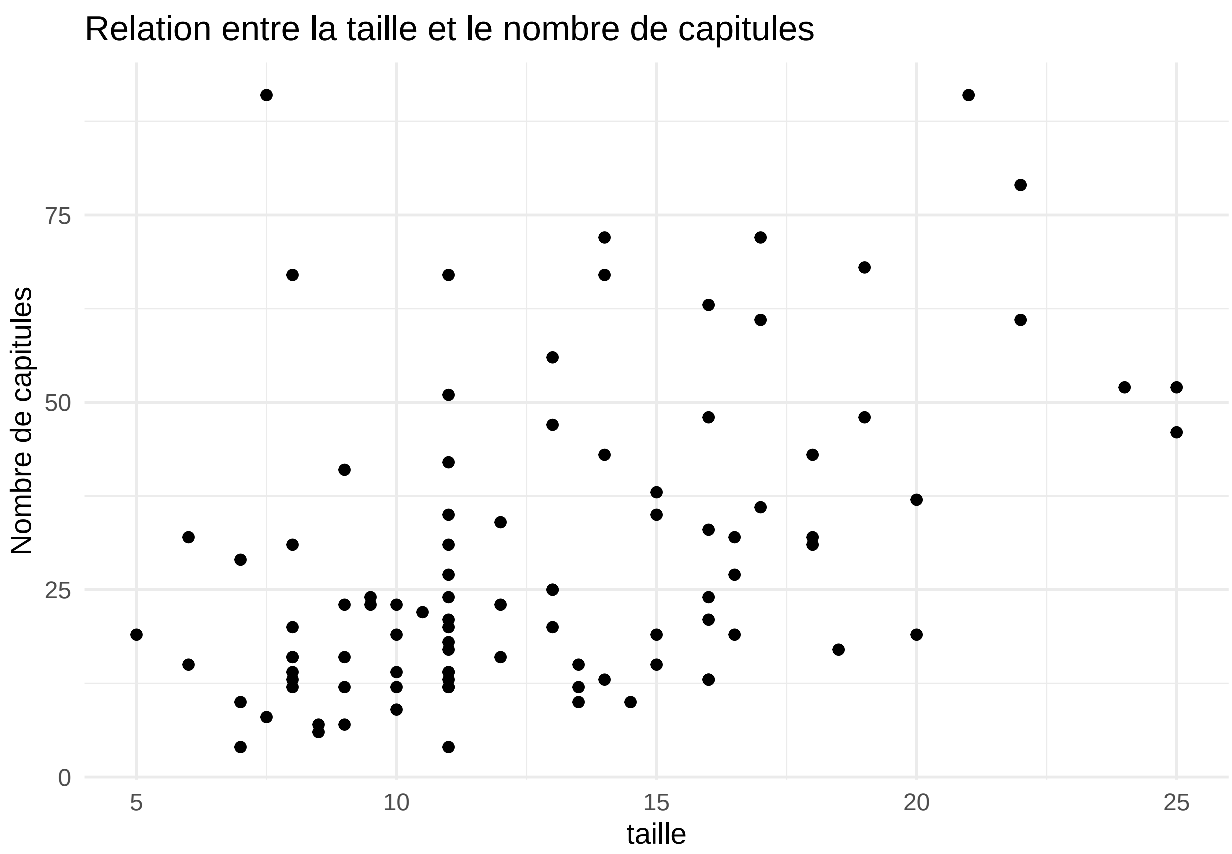
Nombre de capitules

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

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

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 / année
cptldata %>%

  ggplot(aes(x=Size0Mars,y=Cptl0))+
    geom_point() +
labs(title = "Relation entre la taille et le nombre de capitules",
    x = "taille",
    y = "Nombre de capitules") +
    theme_minimal()
```




```
Cptlglm1 <- fitme(Cptl0 ~ 1 + Size0Mars,
              data=cptldata)

Cptlglm2 <- fitme(Cptl0 ~ 1 + poly(Size0Mars,2),
              data=cptldata)

Cptlglm3 <- fitme(Cptl0 ~ 1 + poly(Size0Mars,3),
              data=cptldata)

Cptlglm4 <- fitme(Cptl0 ~ 1 + Size0Mars +(1|year),
              data=cptldata)

Cptlglm5 <- fitme(Cptl0 ~ 1 + Size0Mars + age0,
              data=cptldata)
```

```
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
```

```r
summary(Cptlglm2)
```

```
## formula: Cptl0 ~ 1 + poly(Size0Mars, 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(Size0Mars, 2)1    89.85   17.898   5.020
## poly(Size0Mars, 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
```

```r
summary(Cptlglm3)
```

```
## formula: Cptl0 ~ 1 + poly(Size0Mars, 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(Size0Mars, 3)1    89.85   17.785   5.052
## poly(Size0Mars, 3)2    22.36   17.785   1.257
## poly(Size0Mars, 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
```

```r
summary(Cptlglm4)
```

```
## formula: Cptl0 ~ 1 + Size0Mars + (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
## Size0Mars      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)   0.9787   9.3256  0.1049
## Size0Mars     2.0965   0.4246  4.9372
## age0          0.3645   1.3705  0.2660
##  -------------- Residual variance  ------------
## Coefficients for log(phi)  ~ 1  :
##             Estimate Cond. SE
## (Intercept)    5.785   0.1451
## Estimate of phi=residual var:  325.4
##  ------------- Likelihood values  -------------
##                         logLik
## logL               : -409.5828
```

```
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]
```

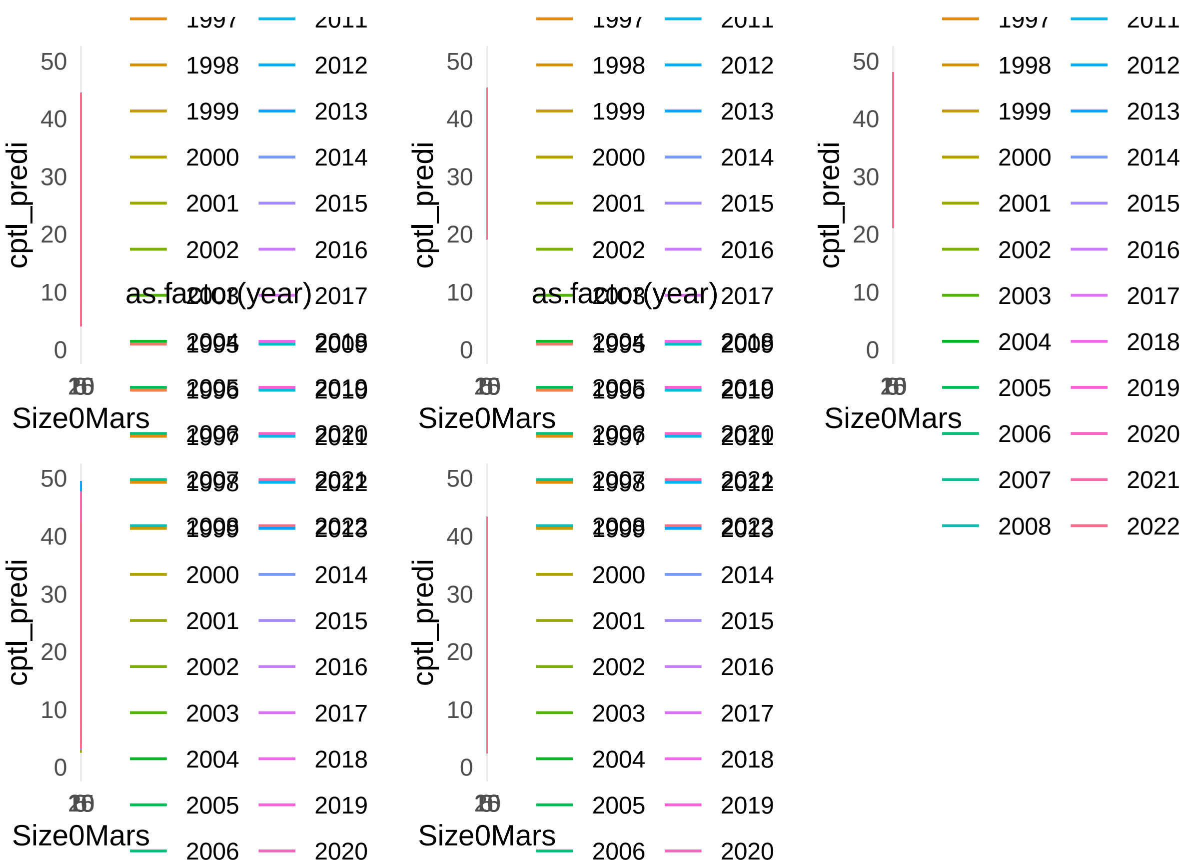
```
plot_capitule <- function(data = fake_data, prediction, var, c1, valc1 = 1, c2, valc2 = "Au", fact) {
  data %>%
    mutate(cptl_predi = prediction) %>%
    filter(!!sym(c1) == valc1, !!sym(c2) == valc2) %>%
    ggplot(aes(x = .data[[var]], y = cptl_predi)) +
    geom_line(aes(color = as.factor(.data[[fact]])))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

```
    ylim(0,50)
  }
```

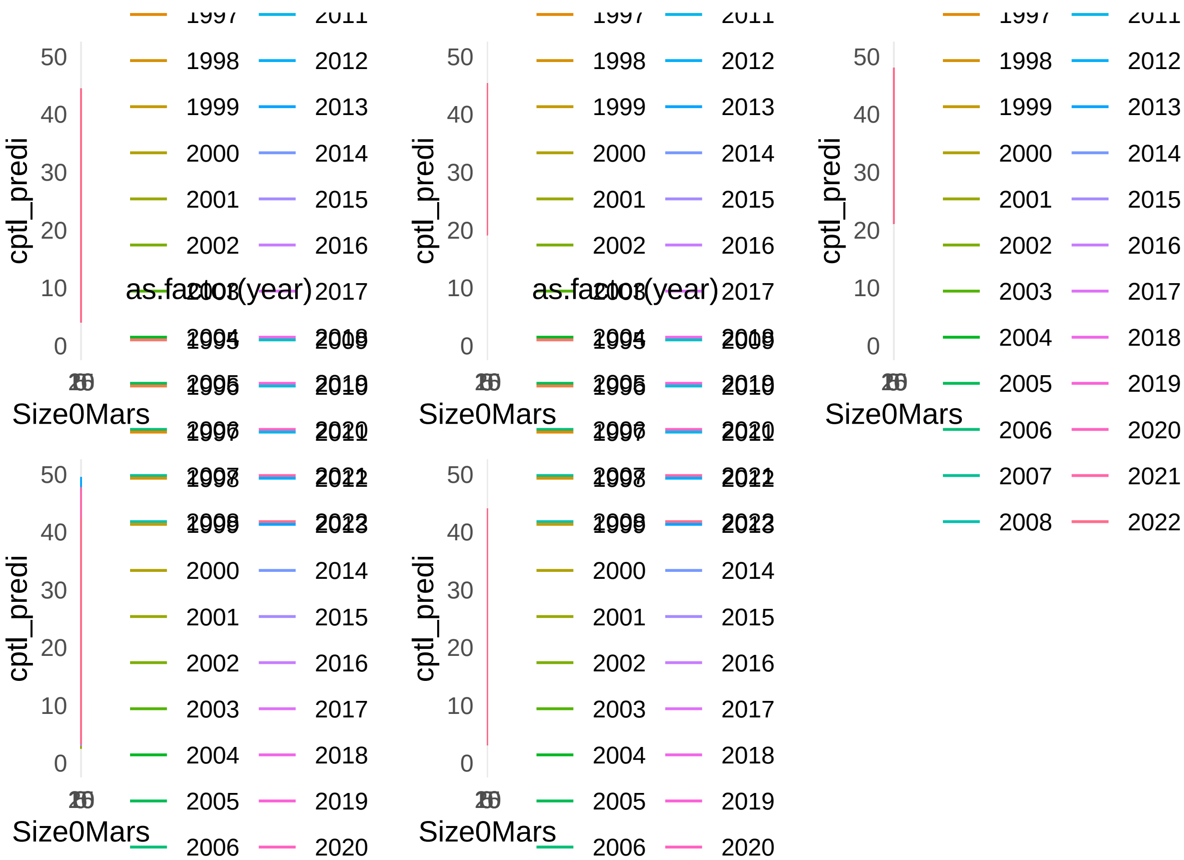
Nombre de capitules en fonction de la taille

En fixant la population : voir l'effet année

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

wrap_plots(
plot_capitule(prediction = Cptlpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```



```r
valc1 <- 3
wrap_plots(
plot_capitule(prediction = Cptlpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

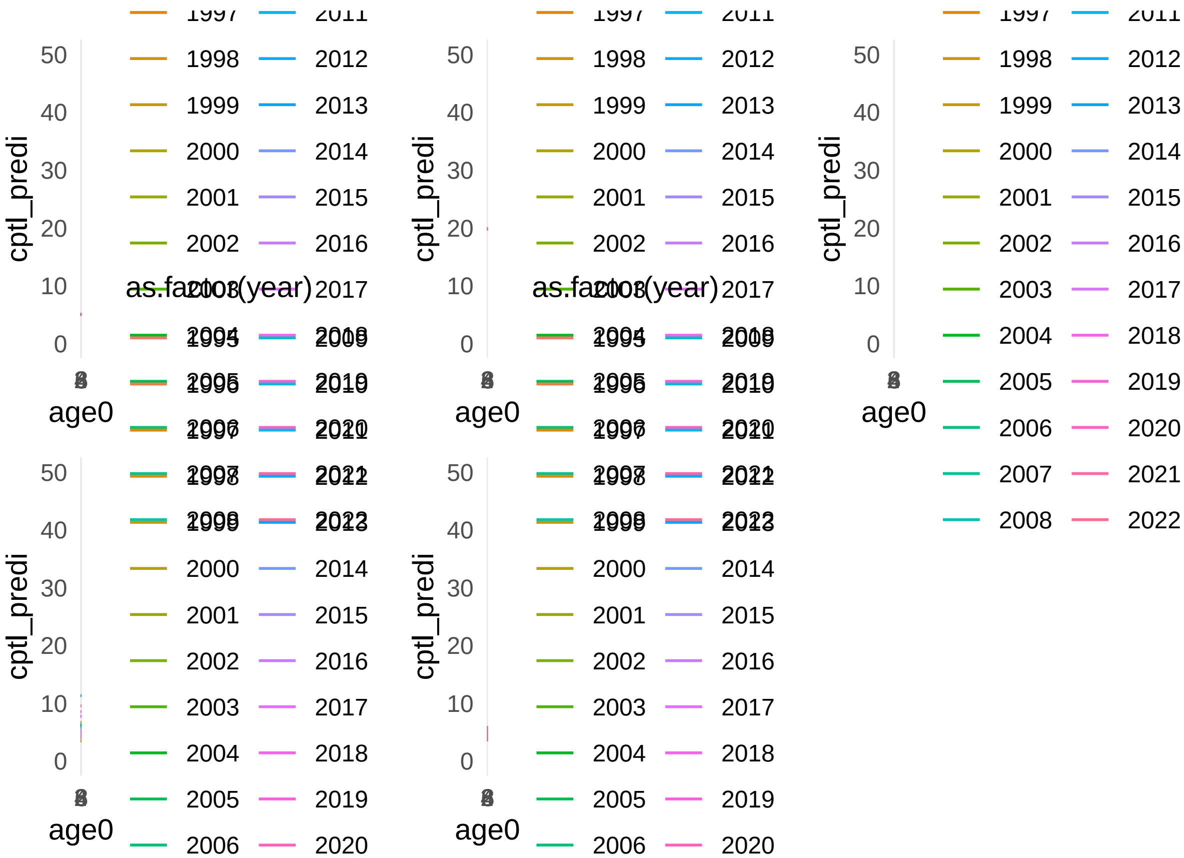


Nombre de capitules en fonction de l'age

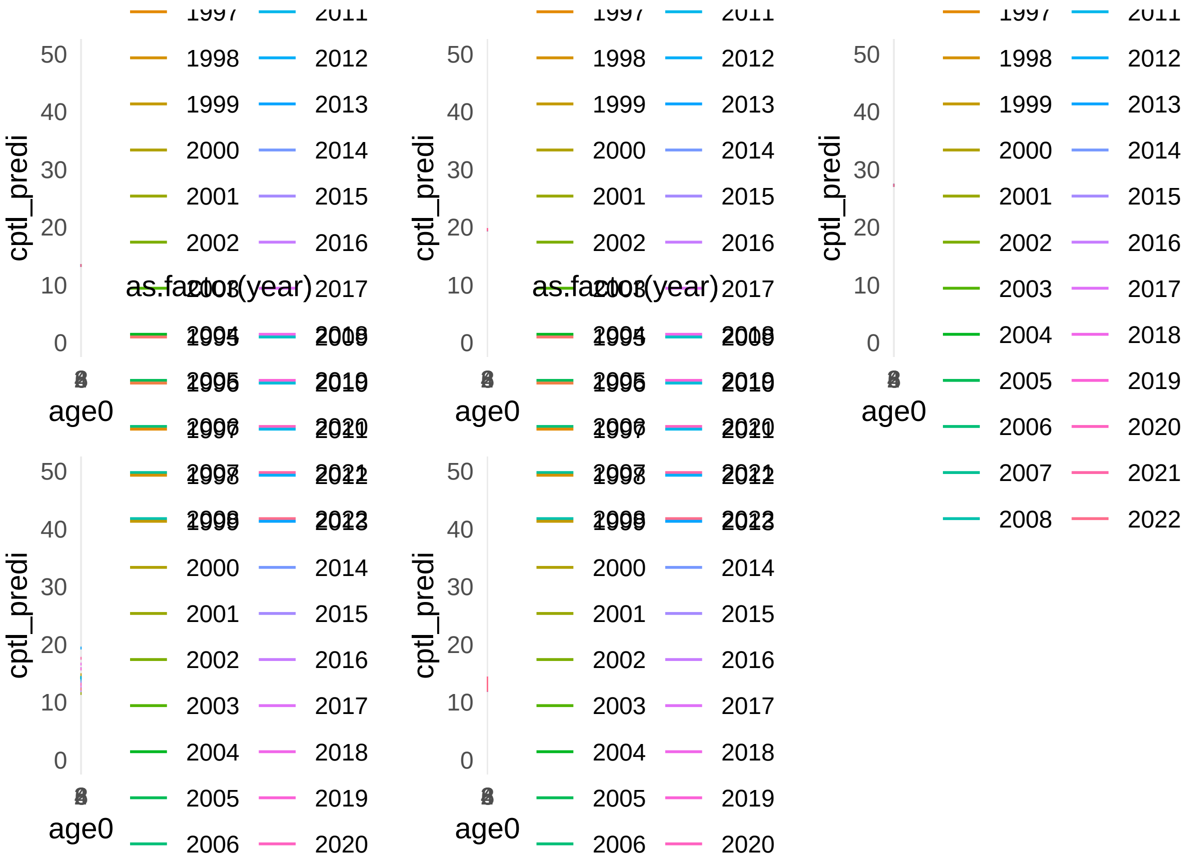
En fixant la population : voir l'effet année

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

wrap_plots(
plot_capitule(prediction = Cptlpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
```

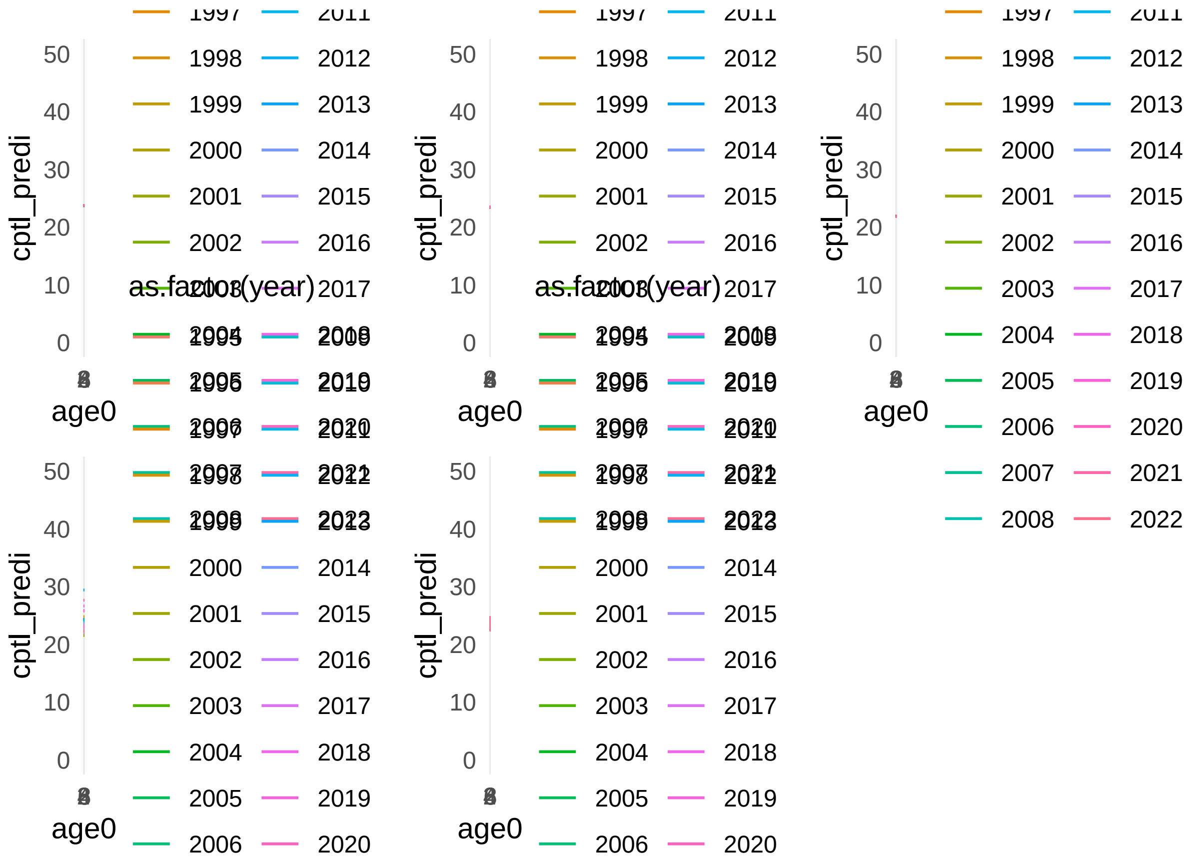
```
plot_capitule(prediction = Cptlpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```




```
valc1 <- 5
wrap_plots(
plot_capitule(prediction = Cptlpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```




```r
valc1 <- 10
wrap_plots(
plot_capitule(prediction = Cptlpredict1, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict2, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict3, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict4, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact),
plot_capitule(prediction = Cptlpredict5, var=var, c1=c1, valc1=valc1, c2=c2, valc2=valc2, fact=fact)
)
```

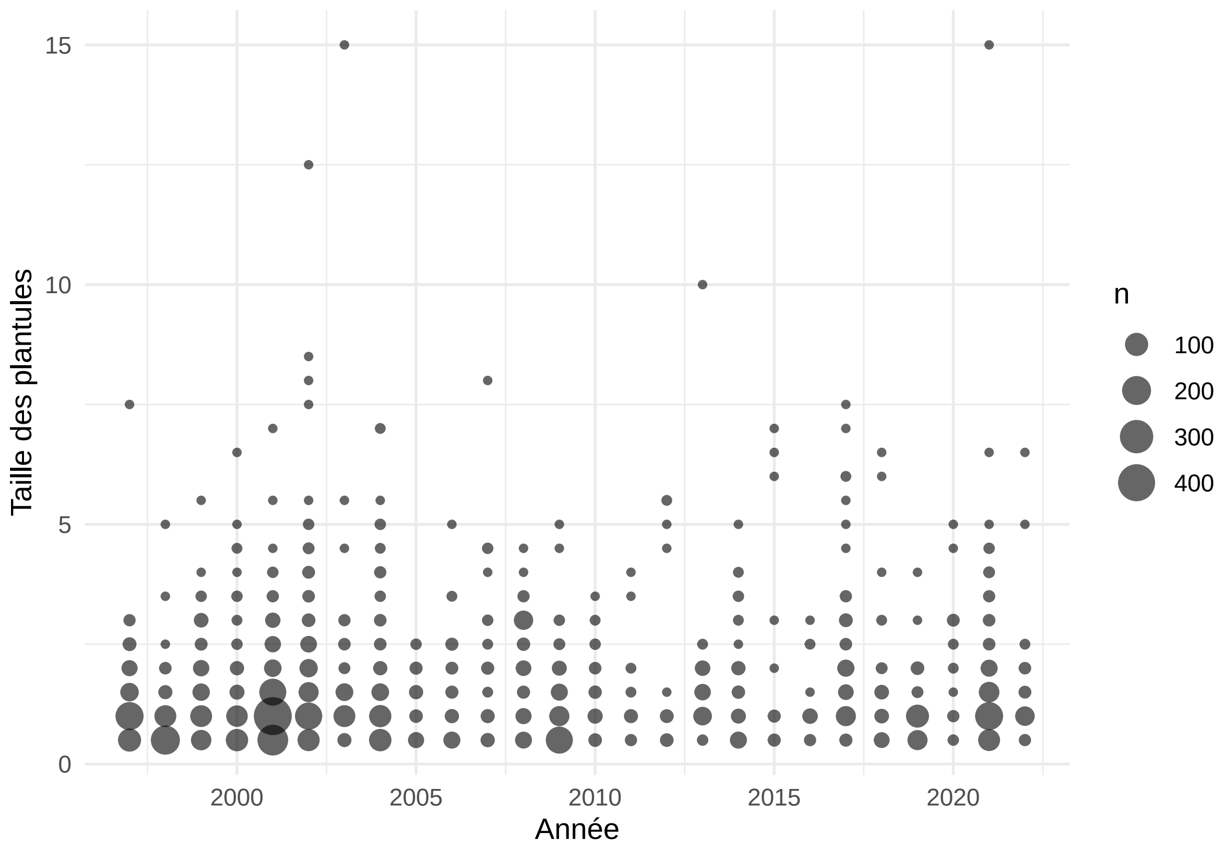


Taille des plantules

```r
plantule_data <- centauree_data_complet[centauree_data_complet$age0==1,]
```

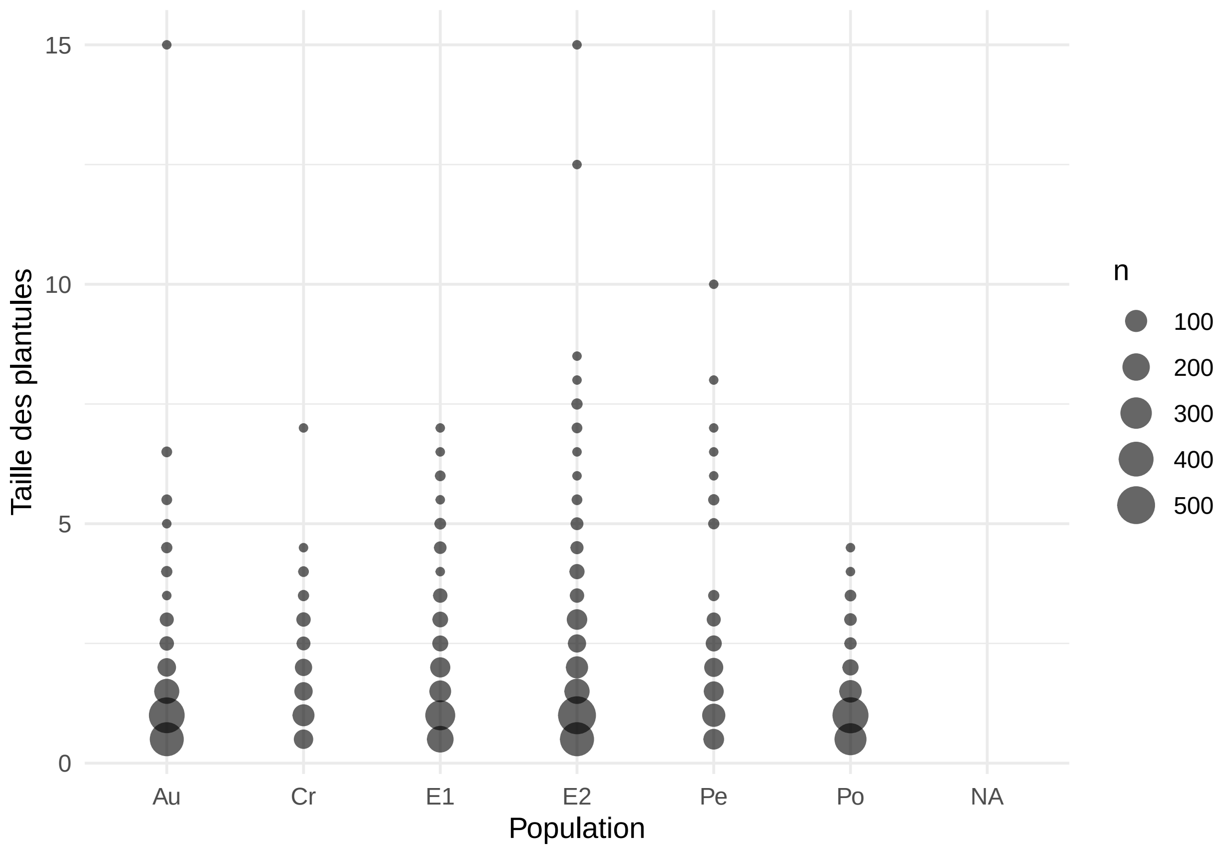
```r
# Taille des plantules / année

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

```
## Warning: Removed 2360 rows containing non-finite outside the scale range
## (‘stat_sum()‘).
```




```r
# 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()
```

```
## Warning: Removed 2360 rows containing non-finite outside the scale range
## (‘stat_sum()‘).
```




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

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

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

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

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

```r
summary(Pltglm1)
```

```
## formula: Size0Mars ~ 1 + (1 | year) + (1 | Pop) + (1 | Pop: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)    1.382   0.1267   10.91
##  --------------- Random effects ---------------
```

```
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.03363
##    Pop  :  0.06829
##    Pop:year  :  0.3535
## # of obs: 4648; # of groups: year, 26; Pop, 6; Pop:year, 136
##  -------------- Residual variance  ------------
## phi estimate was 0.688727
##  ------------- Likelihood values  -------------
##                         logLik
## logL       (p_v(h)): -5891.028
```

```r
summary(Pltglm2)
```

```
## formula: Size0Mars ~ 1 + (1 | Pop) + (1 | Pop: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)    1.374   0.1213   11.32
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    Pop  :  0.06649
##    Pop:year  :  0.3966
## # of obs: 4648; # of groups: Pop, 6; Pop:year, 136
##  -------------- Residual variance  ------------
## phi estimate was 0.688347
##  ------------- Likelihood values  -------------
##                        logLik
## logL       (p_v(h)): -5891.66
```

```r
summary(Pltglm3)
```

```
## formula: Size0Mars ~ 1 + (1 | Pop: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)    1.354  0.06375   21.24
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    Pop:year  :  0.4668
## # of obs: 4648; # of groups: Pop:year, 136
##  -------------- Residual variance  ------------
## phi estimate was 0.688088
##  ------------- Likelihood values  -------------
```

```
##                         logLik
## logL       (p_v(h)): -5895.849
```

```r
summary(Pltglm4)
```

```
## formula: Size0Mars ~ 1 + (1 | year) + (1 | Pop: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)    1.357  0.06779   20.01
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.01713
##    Pop:year  :  0.4463
## # of obs: 4648; # of groups: year, 26; Pop:year, 136
##  -------------- Residual variance  ------------
## phi estimate was 0.688217
##  ------------- Likelihood values  -------------
##                         logLik
## logL       (p_v(h)): -5895.717
```

```r
summary(Pltglm5)
```

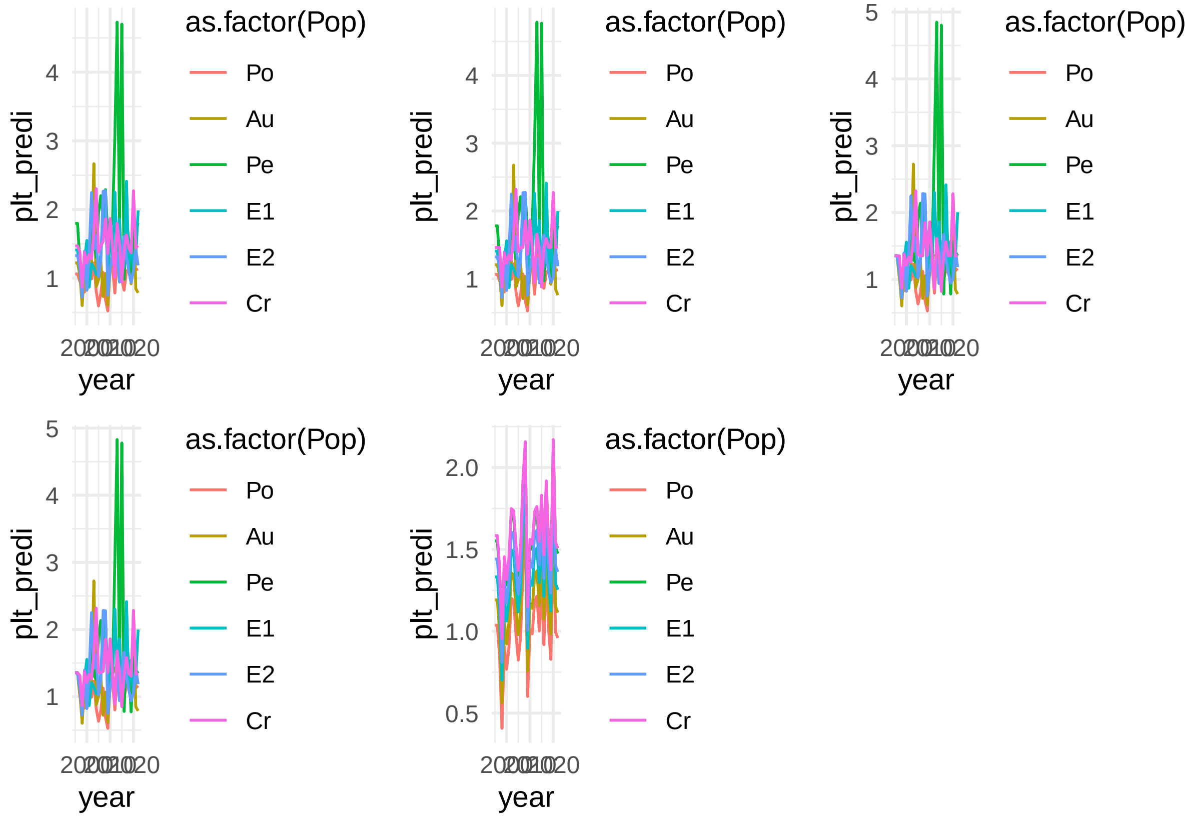
```
## formula: Size0Mars ~ 1 + (1 | year) + (1 | Pop)
## Estimation of fixed effects by ML.
## Estimation of lambda and phi by 'outer' ML, maximizing logL.
## family: gaussian( link = identity )
##  ------------ Fixed effects (beta) ------------
##             Estimate Cond. SE t-value
## (Intercept)    1.357   0.1032   13.14
##  --------------- Random effects ---------------
## Family: gaussian( link = identity )
##            --- Variance parameters ('lambda'):
## lambda = var(u) for u ~ Gaussian;
##    year  :  0.08524
##    Pop  :  0.04151
## # of obs: 4648; # of groups: year, 26; Pop, 6
##  -------------- Residual variance  ------------
## phi estimate was 0.77653
##  ------------- Likelihood values  -------------
##                         logLik
## logL       (p_v(h)): -6050.509
```

```r
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]
```

```r
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]]))) +
    theme_minimal()
}

plot_plantule2 <- function(data = fake_data, prediction, var, fact) {
  data %>%
    mutate(plt_predi = prediction) %>%
    ggplot(aes(x = .data[[var]], y = plt_predi)) +
    geom_point(aes(color = as.factor(.data[[fact]]))) +
    theme_minimal()
}
```

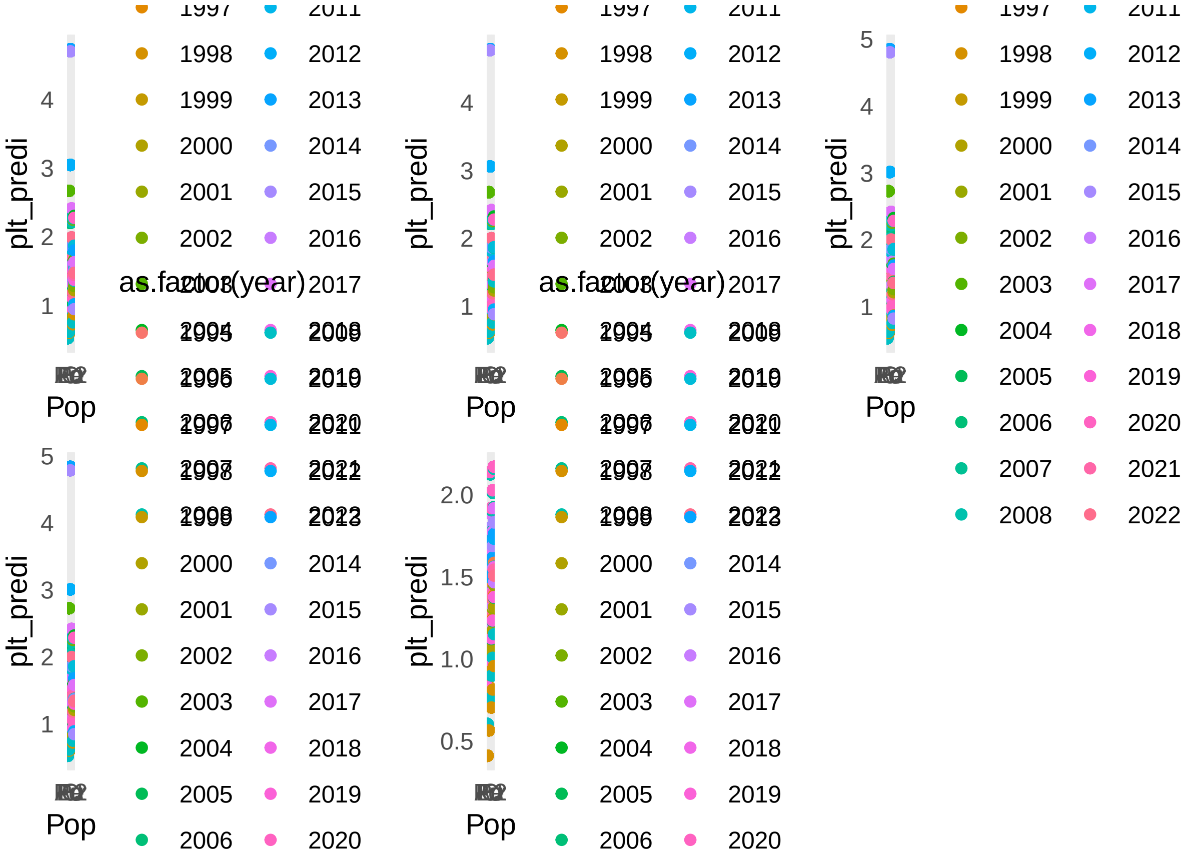
Taille des plantules en fonction de l'année

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

wrap_plots(
plot_plantule(prediction = Pltpredict1, var=var, fact=fact),
plot_plantule(prediction = Pltpredict2, var=var, fact=fact),
plot_plantule(prediction = Pltpredict3, var=var, fact=fact),
plot_plantule(prediction = Pltpredict4, var=var, fact=fact),
plot_plantule(prediction = Pltpredict5, var=var, fact=fact)
)
```



Taille des plantules en fonction de la population

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

wrap_plots(
plot_plantule2(prediction = Pltpredict1, var=var, fact=fact),
plot_plantule2(prediction = Pltpredict2, var=var, fact=fact),
plot_plantule2(prediction = Pltpredict3, var=var, fact=fact),
plot_plantule2(prediction = Pltpredict4, var=var, fact=fact),
plot_plantule2(prediction = Pltpredict5, var=var, fact=fact)
)
```


plt_predi
Pop
as.factor(year)
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022