

Modèles hiérarchiques bayésiens 2

Contents

Données	1
1. Modèle bayésien de la probabilité de reproduction selon l'âge et du ros	3

Données



Figure 1: Svalbard reindeer

Les données proviennent de mon post-doctorat. Nous analyserons la probabilité d'un renne d'avoir un bébé durant l'été. Dans ce système, un des plus importants facteurs environnementaux est la présence d'épisodes de pluie-sur-neige. Ceux-ci surviennent quand des précipitations surviennent au cours de l'hivers. Celles-ci gèlent et forment ensuite d'épaisse couche de glace bloquant l'accès aux ressources alimentaires.

Cependant, avec le réchauffement continue de l'arctique, certain chercheur croient que le ros n'aura plus d'effet. Quand il y a des épisodes de pluies durables suivi d'une période chaude, la pluie cause un dégagement des ressources alimentaires et a le temps de ruisseler avant de geler. Nous tenterons d'explorer ces changements dans l'effet du ros.

```
library(dplyr)
library(readr)
library(ggplot2)
library(tidyr)
```

```
library(cowplot)
library(lubridate)

dat <- read_csv("../donnees/SvalbardDat.csv")
dat <- dat %>% mutate(age=year-yrbirth) %>% filter(age>1)

ros <- read_csv("../donnees/ROS.csv")

dat <- dat %>% left_join(ros) %>%
  mutate(rosNorm = scale(log(ros)),
         ages=scale(age),
         age2=ages^2,
         obsid=1:n(),
         period=cut(year,breaks = seq(1994,2020,by=5)))
```

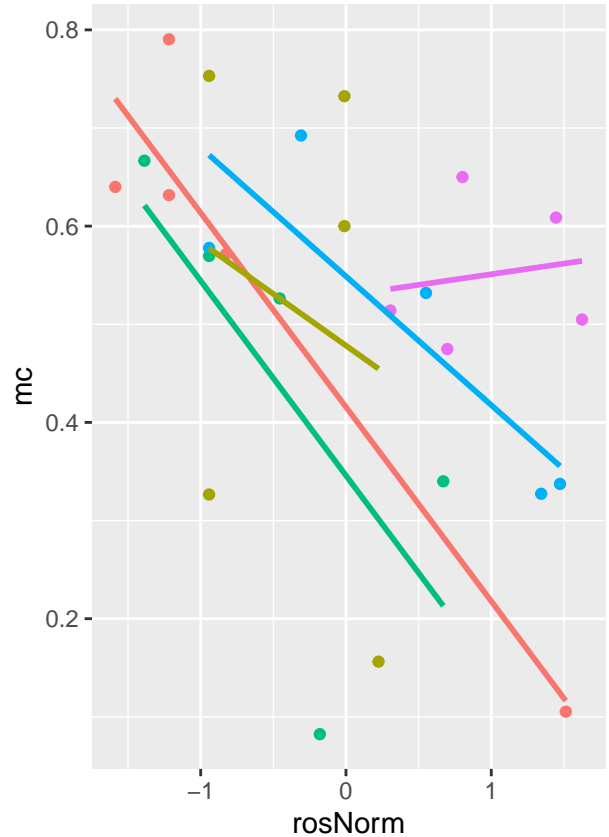
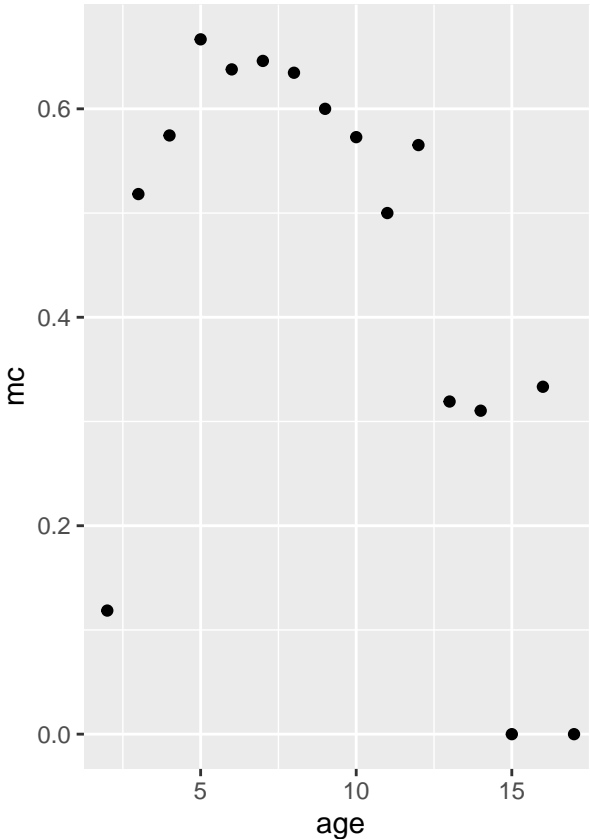
Nous transformons d'abord les prédicteurs:

- *rosNorm* est le logarithme de *ros*, normalisé pour avoir une moyenne de 0 et un écart-type de 1.
- *ages* est la normalisation de *age* pour avoir une moyenne de 0 et un écart-type de 1
- Nous créons une variable *age2* pour l'effet quadratique de l'âge.
- et finalement une variable *période* qui sépare l'étude en 5 périodes

```
g1 <- dat %>% group_by(age) %>% summarise(mc=mean(calf,na.rm=T),n()) %>%
  ggplot(aes(age,mc))+geom_point()

g2 <- dat %>% group_by(period,year,rosNorm) %>% summarise(mc=mean(calf,na.rm=T),n()) %>%
  ggplot(aes(rosNorm,mc,color=period))+geom_point()+geom_smooth(method = lm,se=F)+guides(color='none')

plot_grid(g1,g2)
```



1. Modèle bayésien de la probabilité de reproduction selon l'âge et du ros

Bien entendu, il faudra contrôler pour l'âge des individus. Le modèle sera donc un modèle binomial avec comme effet fixe, l'âge et son carré, et le ros. Les effets aléatoires consisteront de l'année, la période et une pente du ros variant selon la période.

Notes:

- La formule du modèle dans `brm` suit la même syntaxe que `lmer` pour la spécification des effets fixes et aléatoires.
- Bien qu'il serait possible d'ajouter l'interaction `age:ros`, l'année, l'ID, la densité et plusieurs autres variables contrôle. Nous les omettons ici afin de réduire le temps de calcul des modèles.

```
brm(calf ~ ages+age2+rosNorm +(rosNorm|period),family=bernoulli("logit"),
    prior = my_prior,iter = 4000,thin=2,
    data = dat,chains = 2)
```

A Choisissez des distributions *a priori* pour les paramètres du modèle décrit ci-dessus. Voici un exemple de code où il ne manque que la spécification des distributions. Les quatre premières lignes définissent les distributions *a priori* pour l'ordonnée à l'origine et les coefficients des trois effets fixes, les trois suivantes définissent les distributions pour les écarts-types des effets aléatoires (`class = "sd"`), tandis que la dernière réfère à l'écart-type des observations individuelles (`class = "sigma"`).

```
library(brms)
my_prior <- c(set_prior("", class = "Intercept"),
              set_prior("", class = "b", coef = "ages"),
              set_prior("", class = "b", coef = "age2"),
              set_prior("", class = "b", coef = "rosNorm"),
```

```
set_prior("", class = "sd", coef = "Intercept", group = "id"),
set_prior("", class = "sd", coef = "Intercept", group = "period"),
set_prior("", class = "sd", coef = "rosNorm", group = "period"))
```

Il est recommandé de choisir des distributions normales dans tous les cas. Pour “sd”, ces distributions seront interprétées comme des demi-normales car il est sous-entendu que ces paramètres sont ≥ 0 . Pour choisir la moyenne et l’écart-type de chaque distribution normale, considérez l’interprétation de chaque paramètre et en particulier les échelles des prédicteurs `ros`, `ages` et `age2`. Dans `brms`, la famille utilisée sera `family=bernoulli("logit")`.

Quant aux écarts-types des effets aléatoires (“sd”), leur distribution *a priori* peut avoir la même largeur que celle du coefficient “b” correspondant.

```
library(brms)
```

```
## Warning: package 'brms' was built under R version 4.2.2
```

```
## Loading required package: Rcpp
```

```
## Warning: package 'Rcpp' was built under R version 4.2.2
```

```
## Loading 'brms' package (version 2.18.0). Useful instructions
```

```
## can be found by typing help('brms'). A more detailed introduction
```

```
## to the package is available through vignette('brms_overview').
```

```
##
```

```
## Attaching package: 'brms'
```

```
## The following object is masked from 'package:stats':
```

```
##
```

```
##      ar
```

```
my_prior <- c(set_prior("normal(0,1.5)", class = "Intercept"),
  set_prior("normal(.5,.25)", class = "b", coef = "ages"),
  set_prior("normal(-.5,.25)", class = "b", coef = "age2"),
  set_prior("normal(0,.5)", class = "b", coef = "rosNorm"),
  # set_prior("normal(0,.5)", class = "sd", coef = "Intercept", group = "id"),
  set_prior("normal(0,.5)", class = "sd", coef = "Intercept", group = "period"),
  set_prior("normal(0,.5)", class = "sd", coef = "rosNorm", group = "period"))
```

l’intercept sera centré autour de 0 avec une `sd=1.5`. Ceci résulte en un prior assez peu informatif sur l’échelle des probabilités. Les prior pour l’âge sont choisis pour aller d’environ 0 jusqu’à une valeur raisonnable. celle-ci est positive pour `ages` et négative pour `age2`. Ceci résulte en une forme de U inversé pour l’effet de l’âge. Le prior pour `ros` est `normal(0,.5)`, la moyenne est peu informative (centré sur 0), la `sd=0.5` signifie que la pente pourrait aller de -1 à 1 ($2*sd$). Dans un contexte de régression binomial avec des variables explicatives normalisées, une pente de 1 est considéré comme un effet fort.

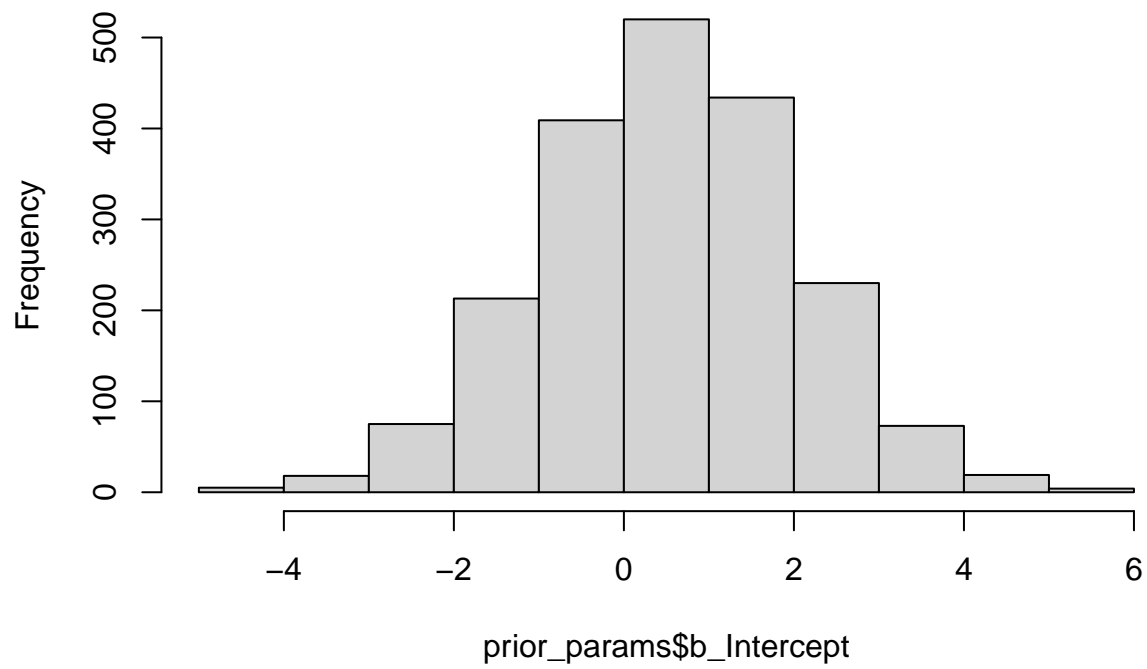
B Tirez maintenant un échantillon de la distribution conjointe *a priori* des paramètres avec `brm`. Je suggère de spécifier `chains = 1`, `iter = 1500`, `warmup = 1000` pour produire une seule chaîne de Markov avec 1000 itérations de rodage et 500 itérations d’échantillonnage. Visualisez ensuite la distribution de `calf` prédite pour chaque itération des paramètres *a priori*.

```
res_prior <- brm(calf ~ ages+age2+rosNorm +(rosNorm|period),family=bernoulli("logit"),
  prior = my_prior,sample_prior = "only",
  data = dat,chains = 1, iter = 3000, warmup = 1000)
```

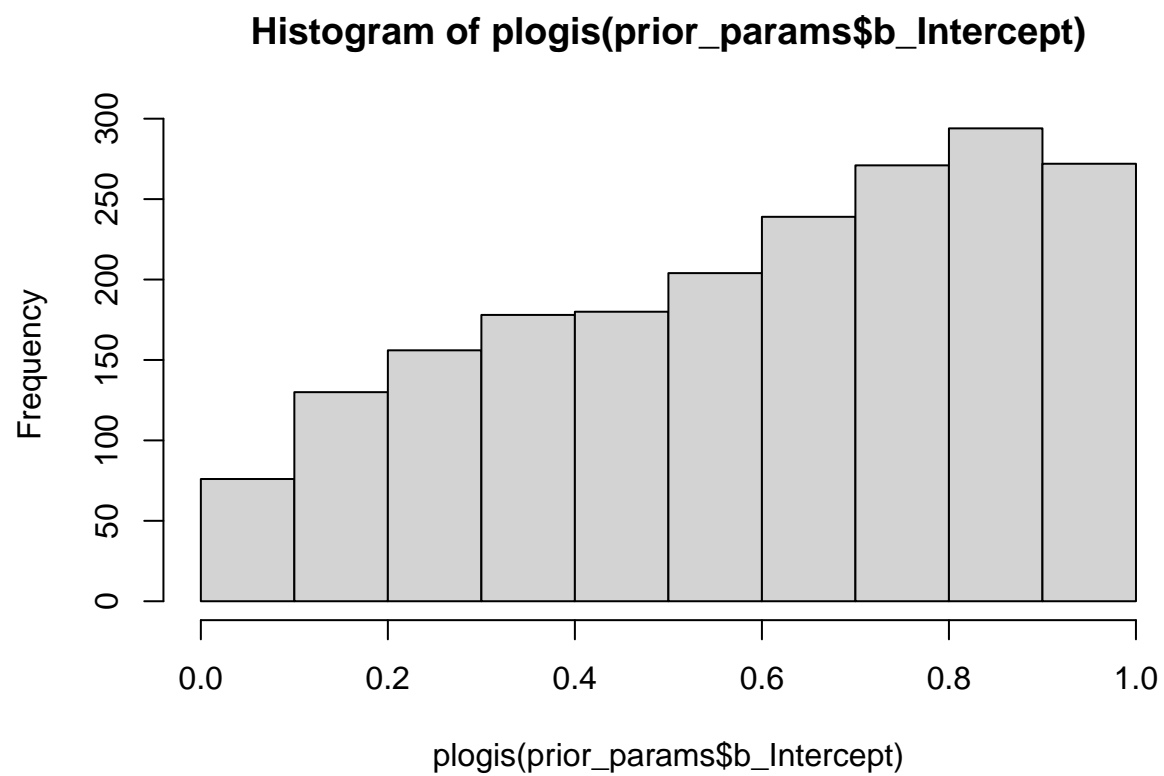
```
prior_params <- as_draws_df(res_prior) %>% mutate(id=1:n())
```

```
hist(prior_params$b_Intercept) # prior sur l'échelle de la fct lien
```

Histogram of prior_params\$b_Intercept

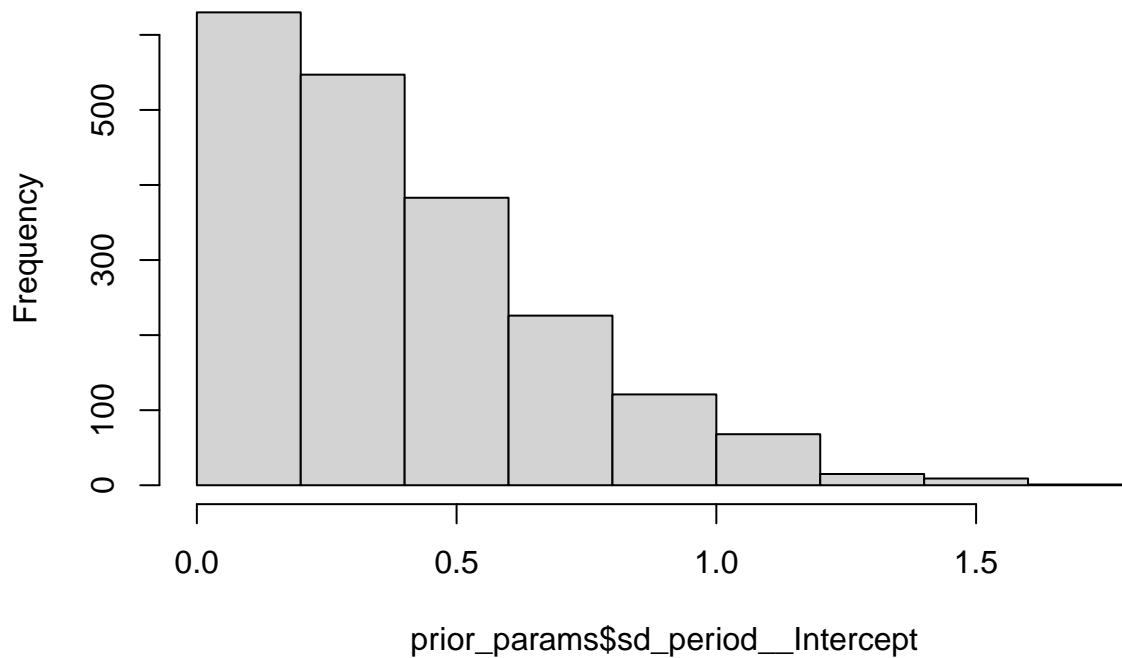


```
hist(plogis(prior_params$b_Intercept)) # prior sur l'échel de la réponse
```



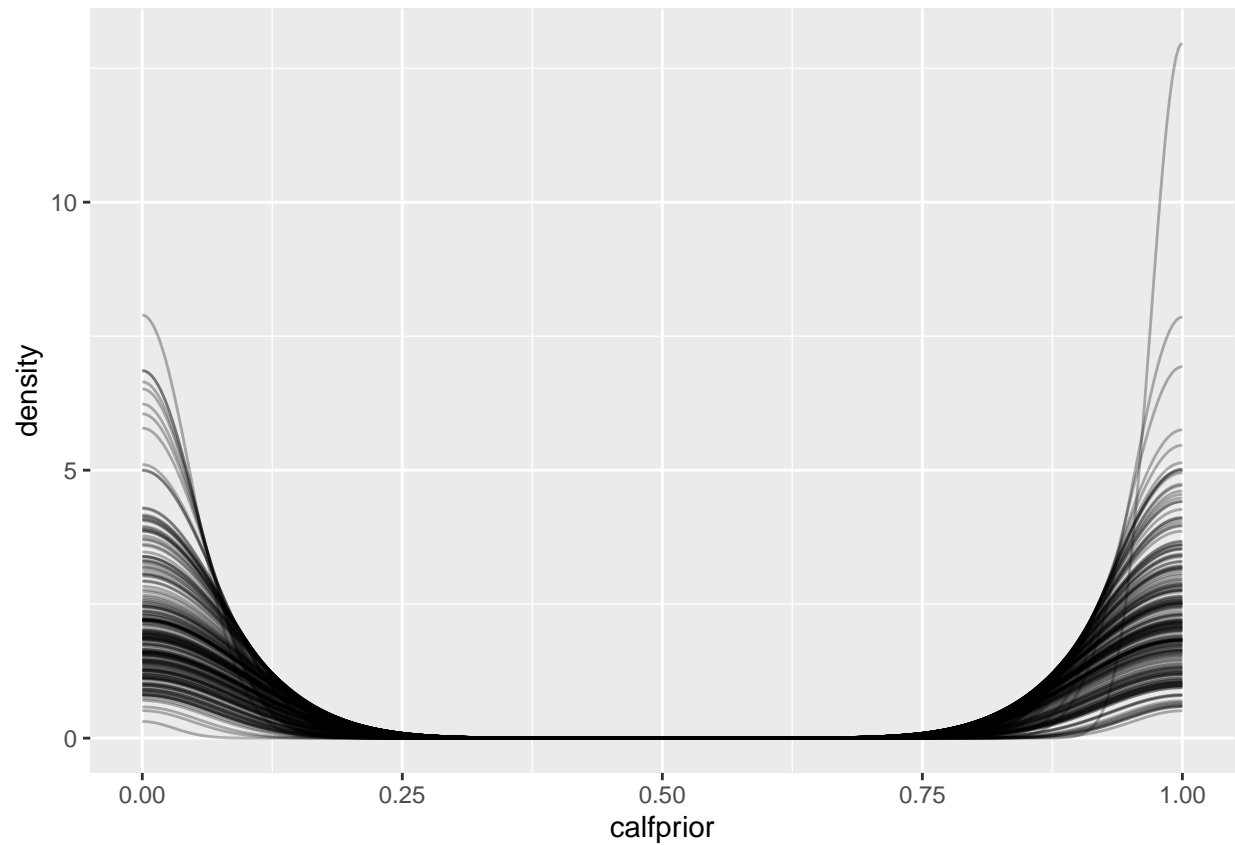
```
hist(prior_params$sd_period__Intercept)
```

Histogram of prior_params\$sd_period__Intercept



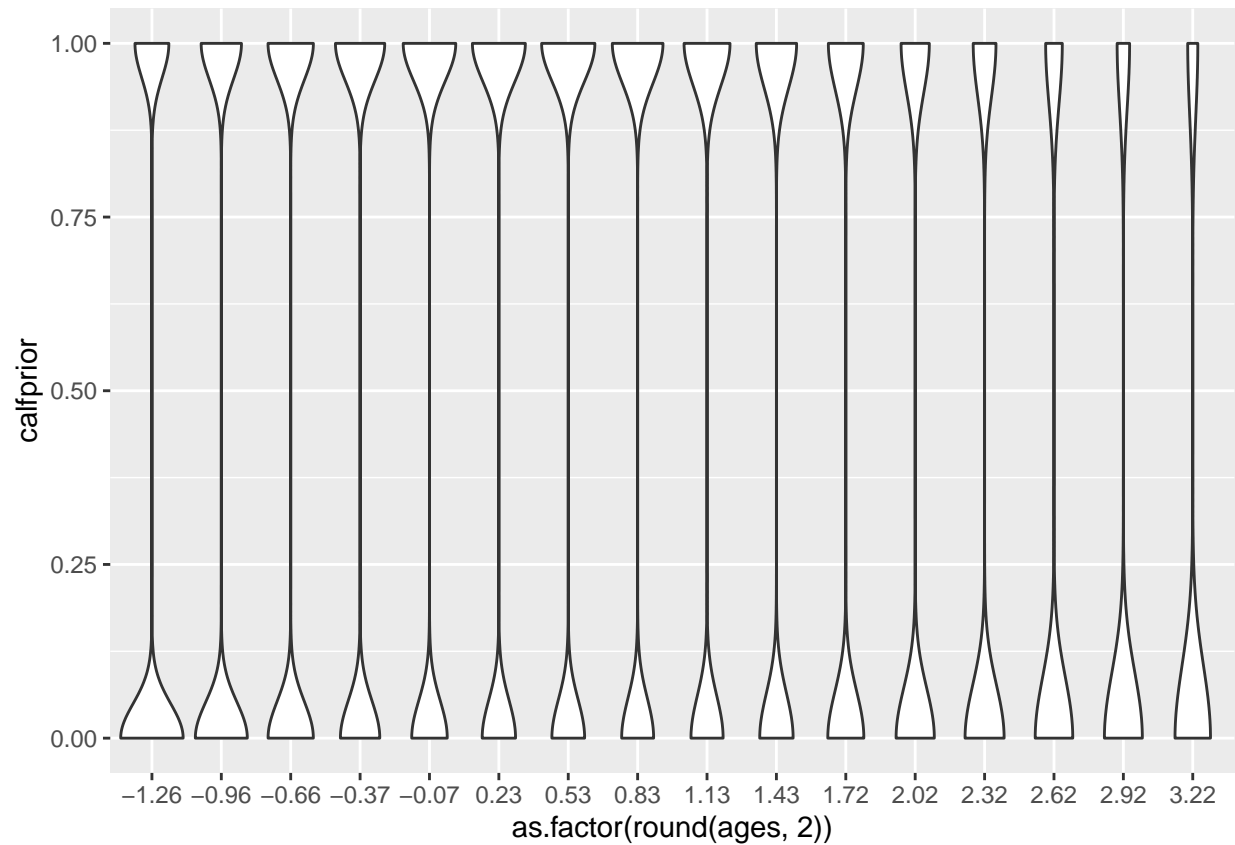
```
# situation de valeur réponse basé
# sur les parametre simulé à partir des prior
prior_pred <- posterior_predict(res_prior)
prior_df <- data.frame(prior_pred)[1:200,] # en garder juste 200 pour garder léger
prior_df$sim_id <- 1:nrow(prior_df)
prior_df <- pivot_longer(prior_df, cols = -sim_id,
names_to = "obsid", values_to = "calfprior") %>%
  mutate(obsid=as.numeric(substr(obsid,2,9))) #pivoter pour la manipulation
# et extraite l'id de l'observation (pour un join futur)

# regardon d'abord la distribution de valeurs prédite
# ceci est peu informatif puisqu'il ne s'agit que de 0 et de 1
ggplot(prior_df, aes(x = calfprior)) +
stat_density(aes(group = sim_id), position = "identity", geom = "line", alpha = 0.3)
```

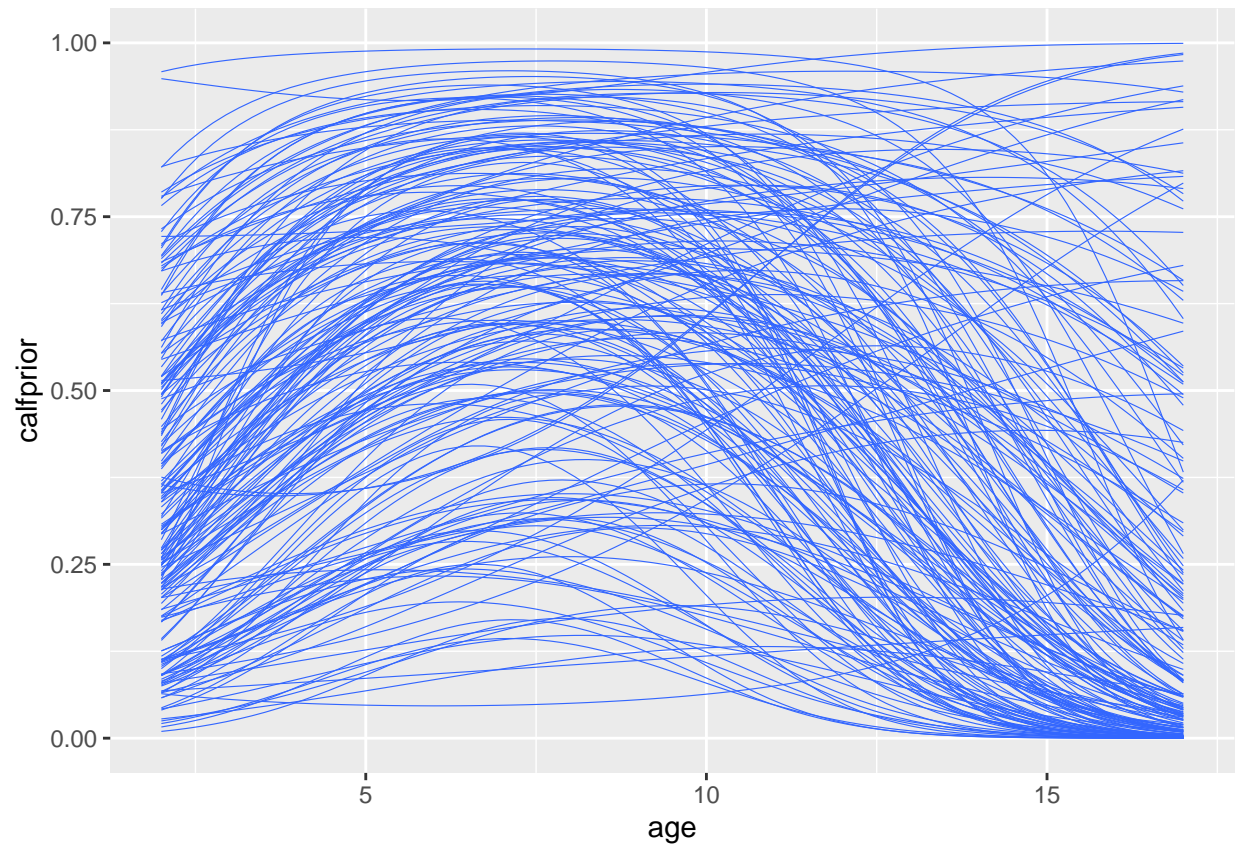


```
# joignons plutot les prediction a prior au donnée pour voir
# comment cele-ci change en fct de différente var explicative
# premierement l'age
# les prior semblent résulter en prédiction plus extrême que prévues
# dans ce cas, ceci est du à l'effet cumulatif de l'intercept et de l'effets aléatoire
```

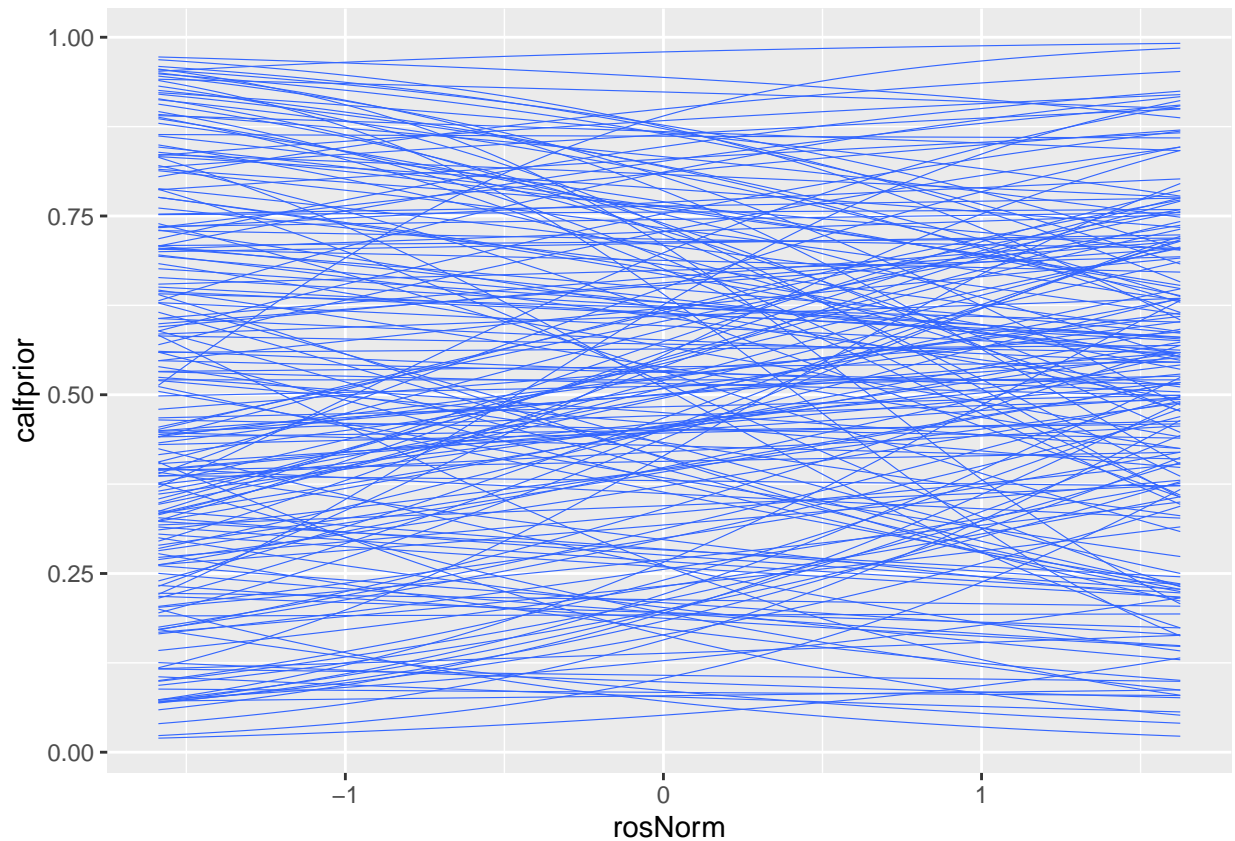
```
prior_df <- prior_df %>% left_join(dat)
prior_df %>% ggplot(aes(x=as.factor(round(ages,2)),y=calfprior))+geom_violin()
```

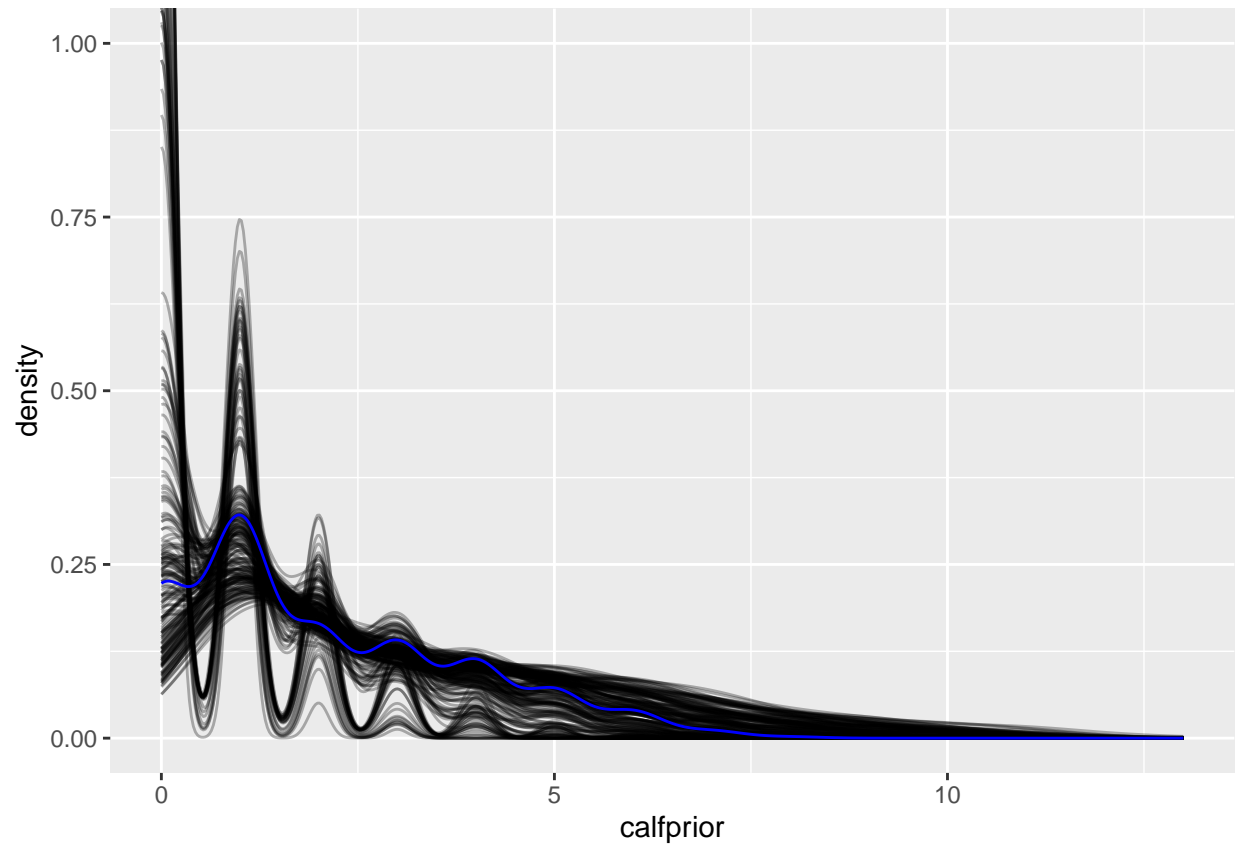
```
# mais si nous regardont les tendance moyenne, elles ont la bonne forme
# la moajorité comment faible, augmente et sont faible pour les vieilles femmes
ggplot(prior_df,aes(x=age,y=calfprior))+
  geom_smooth(aes(group = sim_id),method="glm",formula = y~x+I(x^2),method.args = list(family='binomial
```



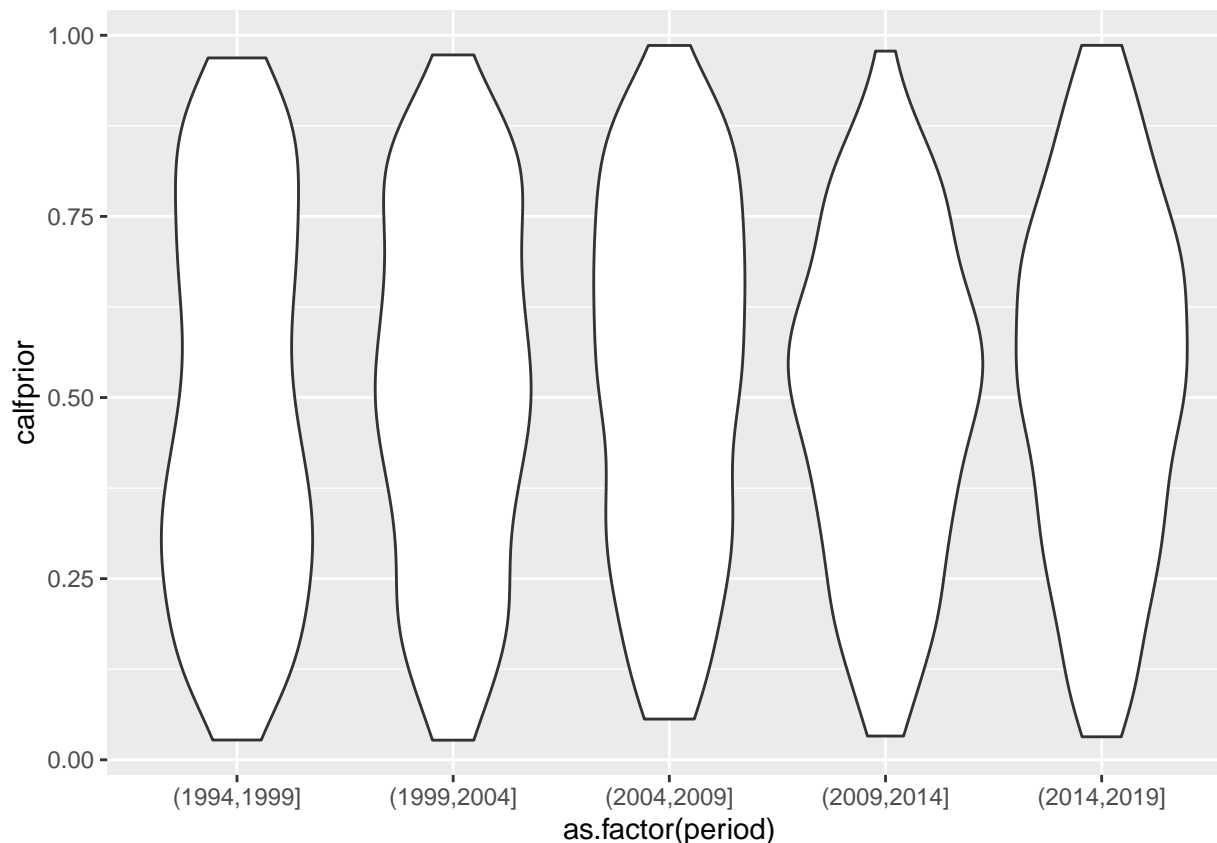
```
#on peut faire le meme exercice pour le ros
ggplot(prior_df,aes(x=rosNorm,y=calfprior))+
  geom_smooth(aes(group = sim_id),method="glm",formula = y~x,method.args = list(family='binomial'),se=F
```



```
#Parfois, il peut être bien de regarder les prédictions de variable dérivée
# et non directement mesurer par le modèle.
# ici on regarde la somme de bébé produit pour chaque femelle
# les valeurs prédites s'alignent bien avec celle observée en bleu
prior_df %>%group_by(sim_id,id) %>% summarise(calfprior=sum(calfprior)) %>%
  ggplot(aes(x=calfprior))+
  stat_density(aes(group = sim_id), position = "identity", geom = "line", alpha = 0.3)+
  stat_density(data=data.frame(calfprior=tapply(dat$scalf,INDEX = dat$id,FUN = sum)),
    aes(x=calfprior),
    position = "identity",
    geom = "line", color="blue")+
  coord_cartesian(ylim=c(0,1))
```



```
prior_df %>%group_by(sim_id,period) %>% summarise(calfprior=mean(calfprior)) %>%  
  ggplot(aes(x=as.factor(period),y=calfprior))+geom_violin()
```



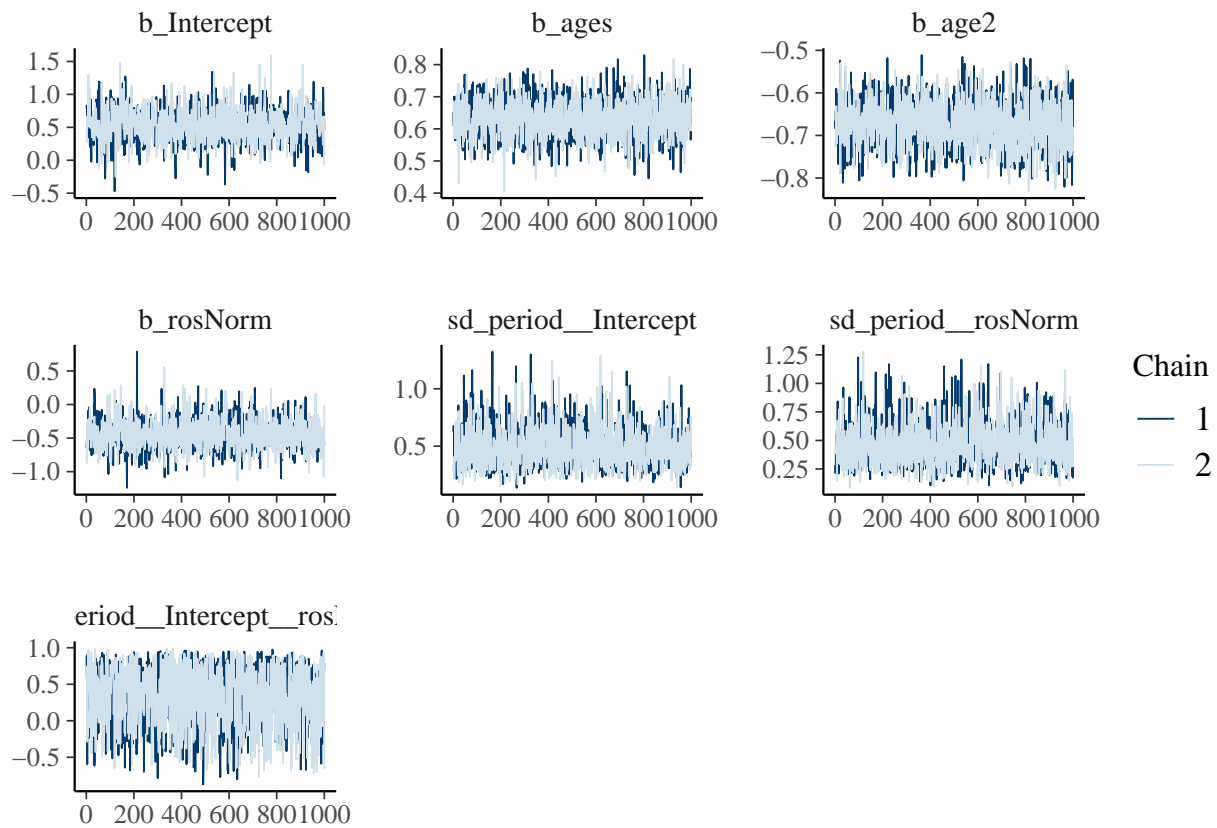
En raison du grand nombre d'effets estimés et du fait que nous n'imposons que des contraintes légères sur chaque distribution *a priori*, on doit s'attendre à des valeurs extrêmes voire impossibles (grandes valeurs positives et négatives); l'important est que la densité soit plus grande dans une plage de valeurs réalistes. Il peut être utile de faire un “zoom” sur une partie du graphique `ggplot` en y ajoutant `coord_cartesian(xlim = c(..., ...), ylim = c(..., ...))` avec des limites en x et y .

C Ajustez maintenant le modèle avec `brm`. Vous pouvez réduire le nombre de chaînes de Markov à 2 pour sauver du temps, mais conservez les valeurs par défaut pour le nombre d'itérations. (Vous pouvez ignorer l'avertissement selon lequel la taille effective de l'échantillon ou ESS est faible.) Comment pouvez-vous évaluer la convergence du modèle?

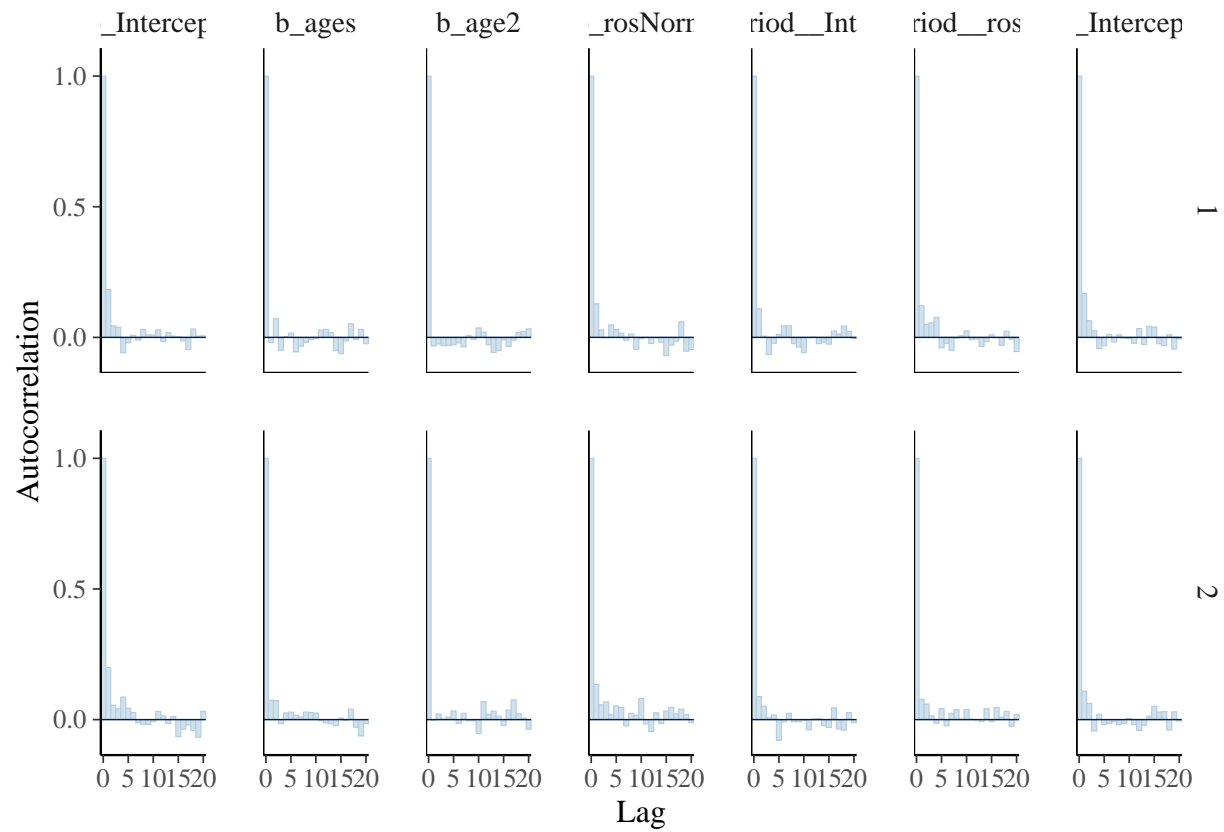
```
res_br <- brm(calf ~ ages+age2+rosNorm +(rosNorm|period),family=bernoulli("logit"),
  prior = my_prior,iter = 4000,thin=2,
  data = dat,chains = 2)
```

```
# as_draw extrait les postérieur (oui prior) du modèle et
# les arrange dans un data.frame
post_params <- as_draws_df(res_br)
```

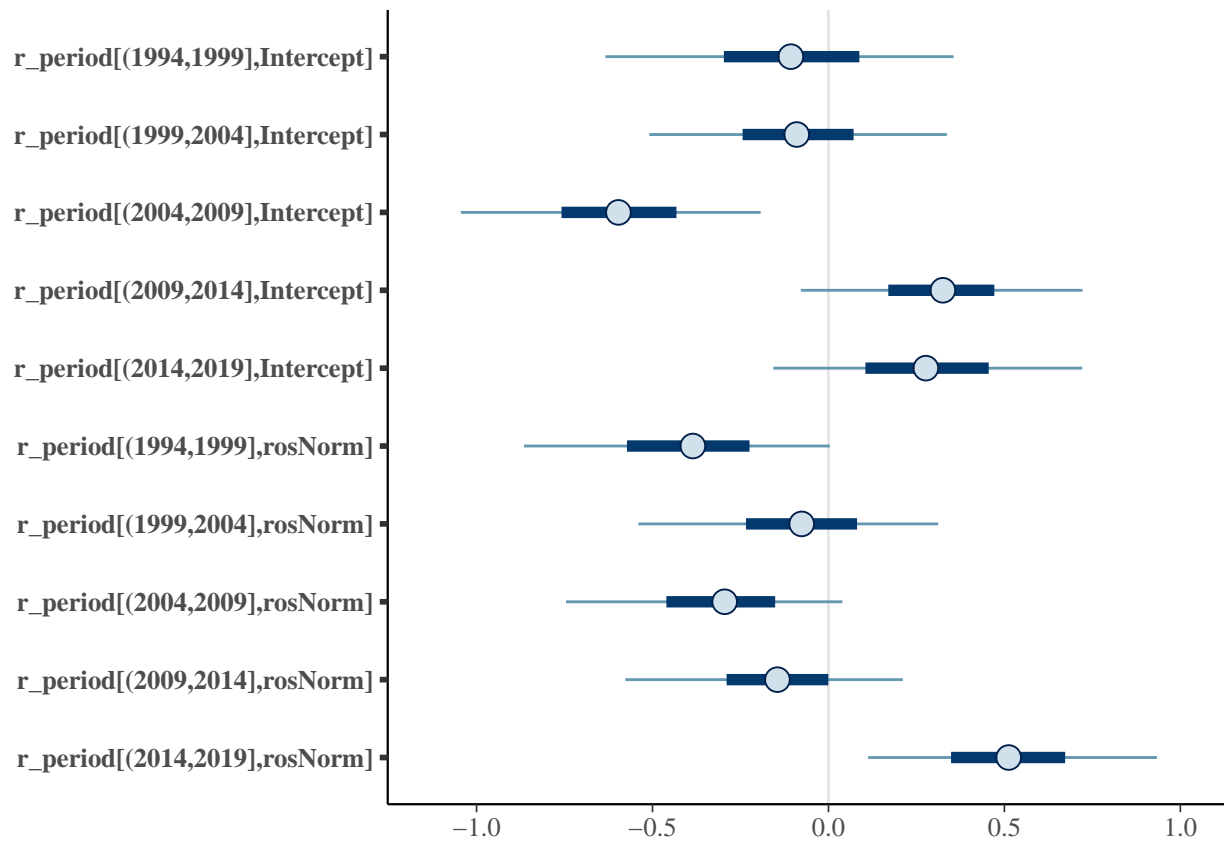
```
mcmc_plot(res_br, type = "trace") # voir les trace plot
```



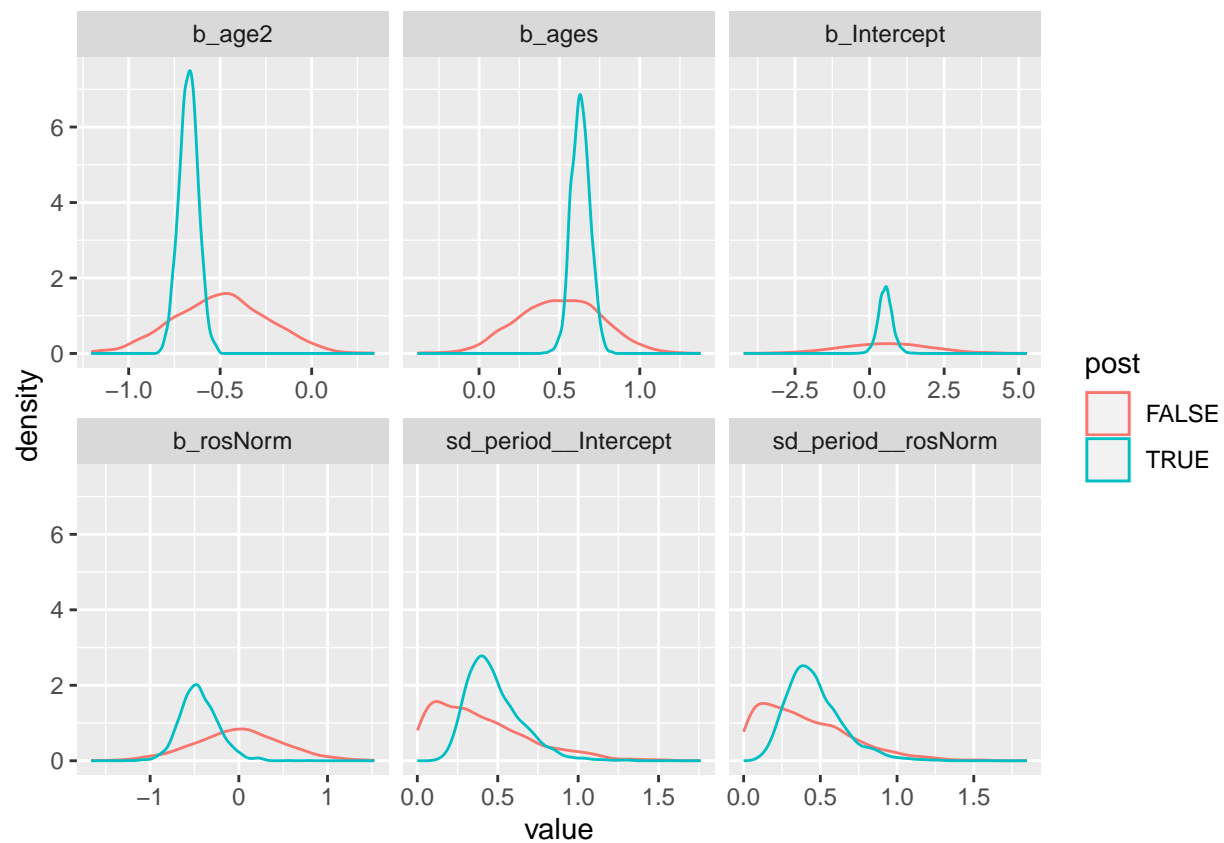
```
mcmc_plot(res_br, type = "acf_bar") # voir l'auto-correlation
```



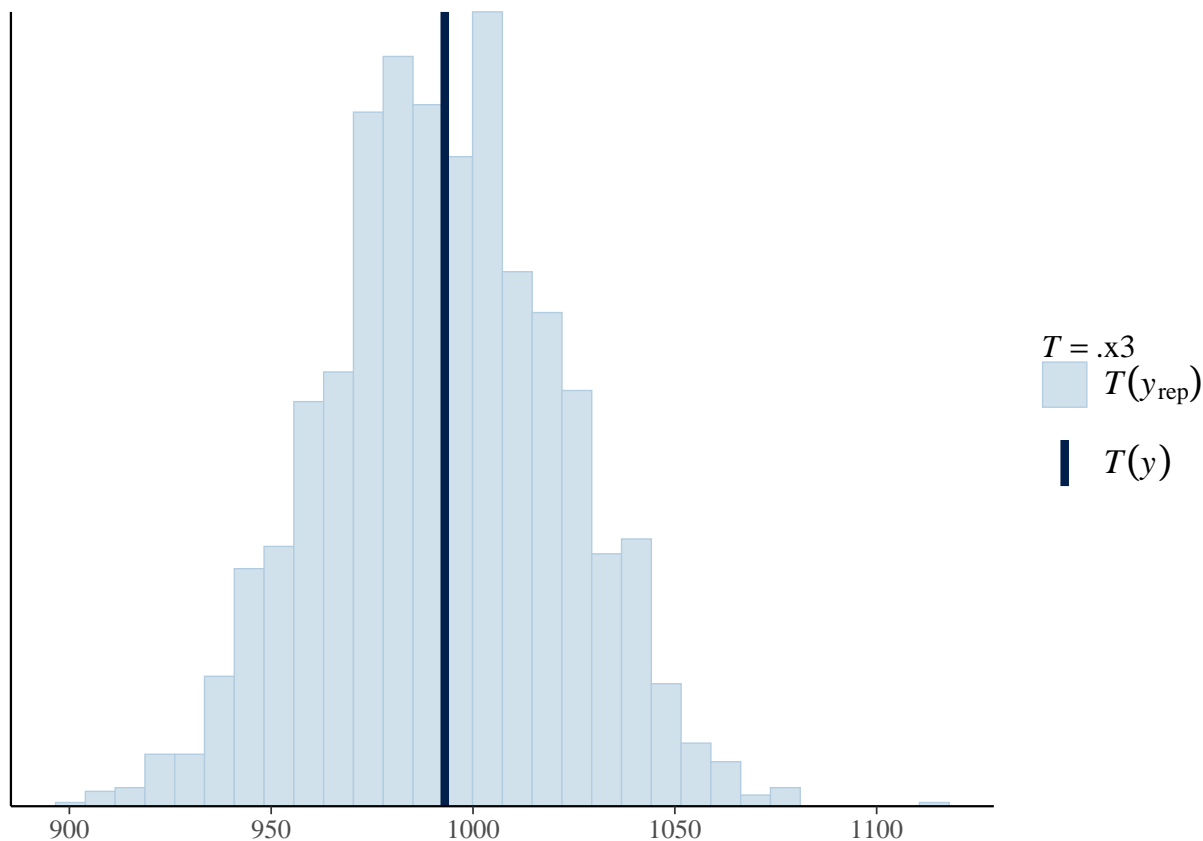
```
# voir les estimés des paramètres et leurs CI (50 et 95%)
mcmc_plot(res_br, variable = 'r_period', type = "intervals")
```



```
# il est très informatif de comparer les postérieurs au prior
rbind(
  post_params[,c( "b_Intercept", "b_ages", "b_age2", "b_rosNorm",
    "sd_period__Intercept" , "sd_period__rosNorm" )] %>%
  mutate(post=T),
  prior_params[,c( "b_Intercept", "b_ages", "b_age2", "b_rosNorm",
    "sd_period__Intercept" , "sd_period__rosNorm" )] %>%
  mutate(post=F)
) %>% pivot_longer(-post) %>%
  ggplot(aes(x=value))+geom_density(aes(color=post))+facet_wrap(~name, scales = "free_x")
```

```
# nous pouvons aussi faire un test de comparaison des prédictions ppsterieur au vrai donnée
pp_check(res_br, type = "stat", stat = function(x) sum(x == 1))
```



D Comparez la magnitude du coefficient de `rosNorm` à celle des effets aléatoires. Qu'est-ce que cette comparaison vous apprend?

```
summary(res_br)
```

```
## Family: bernoulli
## Links: mu = logit
## Formula: calf ~ ages + age2 + rosNorm + (rosNorm | period)
## Data: dat (Number of observations: 1922)
## Draws: 2 chains, each with iter = 4000; warmup = 2000; thin = 2;
## total post-warmup draws = 2000
##
## Group-Level Effects:
## ~period (Number of levels: 5)
##
```

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS
sd(Intercept)	0.48	0.17	0.23	0.87	1.00	1587
sd(rosNorm)	0.46	0.18	0.19	0.89	1.00	1355
cor(Intercept,rosNorm)	0.33	0.39	-0.54	0.91	1.00	1476

```
##
```

	Tail_ESS
sd(Intercept)	1691
sd(rosNorm)	1750
cor(Intercept,rosNorm)	1651

```
##
## Population-Level Effects:
##
```

	Estimate	Est.Error	1-95% CI	u-95% CI	Rhat	Bulk_ESS	Tail_ESS
Intercept	0.53	0.24	0.06	1.03	1.00	1235	1300
ages	0.63	0.06	0.52	0.74	1.00	1682	1482

```
## age2          -0.67      0.05   -0.77   -0.57 1.00      2081      1942
## rosNorm       -0.45      0.22   -0.84    0.01 1.00      1276      1400
##
## Draws were sampled using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

E Vérifier les prédictions a posteriori: Appliquez `predict` au modèle pour obtenir la moyenne, l'écart-type et l'intervalle à 95% pour la prédiction *a posteriori*. Vous pouvez donner un `data.frame` à l'argument `newdata` pour obtenir les prédictions voulues (`expand.grid(rosNorm=seq(-1.6,1.6,l=30),period=unique(dat$period),...)`). Illustrez les prédictions du modèle et leurs intervalles de crédibilité pour les différentes périodes pour un individu de 7 ans.

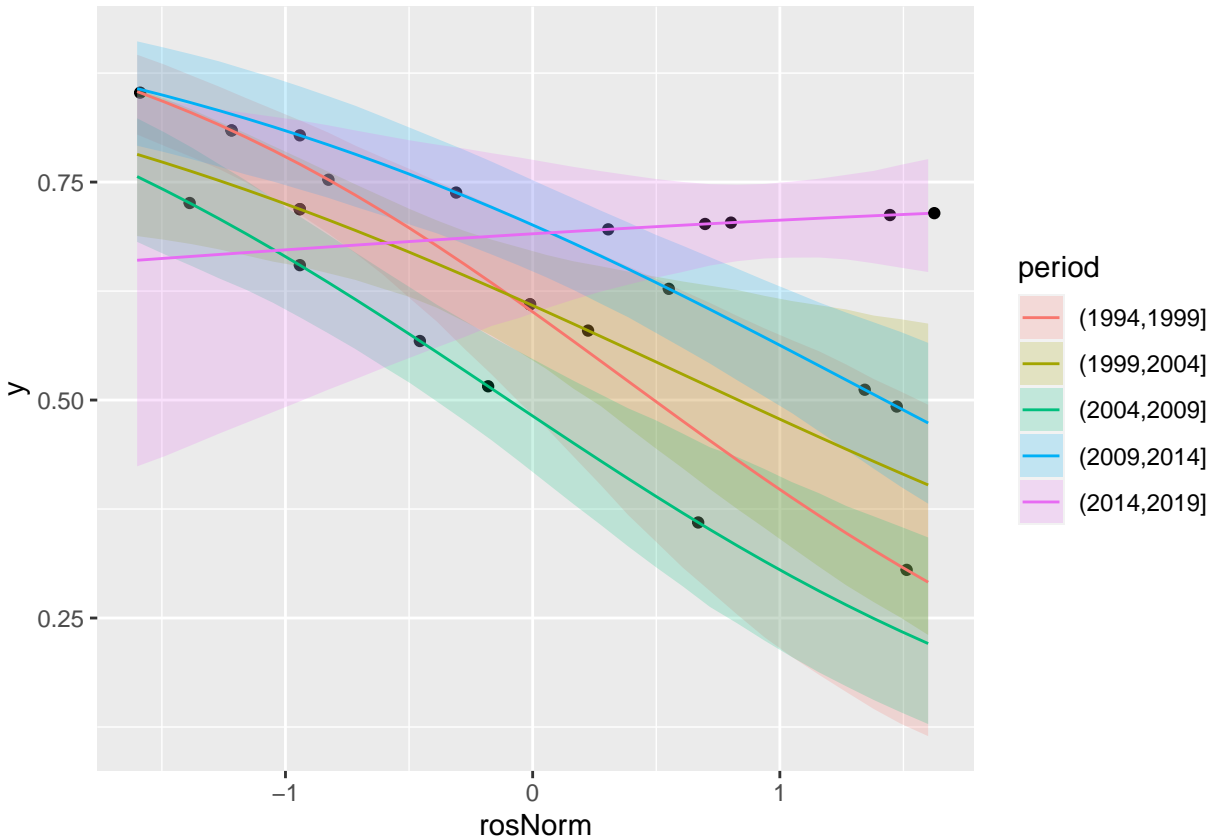
```
# nous faisons la premiere prédiction en changeant l'age pour 0.
# ceci résult en un controle de cette variable inutile dans le graphique
# l'argument re_formula nous permet de contrôler quel effets aléatoires
# sont considérer lors de la prédiction.

post_pred <- posterior_epred(res_br,
                             newdata = mutate(dat,ages=0,age2=0),
                             re_formula = ~(rosNorm|period))

dat$y=apply(post_pred,2,mean)
dat$y.sd=apply(post_pred,2,sd)
dat$ymin=apply(post_pred,2,function(x) quantile(x,0.025))
dat$ymax=apply(post_pred,2,function(x) quantile(x,0.975))
pt <- dat %>% group_by(period,year) %>% summarise_if(is.numeric,mean)

# nous pouvons aussi partir d'un nouveau data.frame ou
# nous générons une séquence de valeurs de ros allant du minimum au maximum.
# si cette sequence contient assez de point, les relier les un au autre
# résultera une courbe de prédiction
newd=expand.grid(ages=0,age2=0,rosNorm=seq(-1.6,1.6,l=30),period=unique(dat$period),id="W16")
post_pred2 <- posterior_epred(res_br,newdata =newd, re_formula = ~(rosNorm|period) )
newd$y=apply(post_pred2,2,mean)
newd$y.sd=apply(post_pred2,2,sd)
newd$ymin=apply(post_pred2,2,function(x) quantile(x,0.025))
newd$ymax=apply(post_pred2,2,function(x) quantile(x,0.975))

ggplot(newd,aes(x=rosNorm,y=y))+
  geom_point(data=pt)+
  geom_ribbon(aes(fill=period,ymin=ymin,ymax=ymax),alpha=0.2)+
  geom_path(aes(color=period))
```



```
# on peut recommencer pour l'effet de l'age. en fixant ros à sa
# moyenne (0 puisque normalisé)
post_pred <- posterior_epred(res_br,
                             newdata = mutate(dat,rosNorm=0),
                             re_formula =NA)

dat$y=apply(post_pred,2,mean)
dat$y.sd=apply(post_pred,2,sd)
dat$ymin=apply(post_pred,2,function(x) quantile(x,0.025))
dat$ymax=apply(post_pred,2,function(x) quantile(x,0.975))
pt <- dat %>% group_by(period,age,year) %>% summarise_if(is.numeric,mean)

newd=dat %>% select(age,ages,age2) %>% unique() %>% mutate(rosNorm=0)
post_pred2 <- posterior_epred(res_br,newdata =newd, re_formula =NA)
newd$y=apply(post_pred2,2,mean)
newd$y.sd=apply(post_pred2,2,sd)
newd$ymin=apply(post_pred2,2,function(x) quantile(x,0.025))
newd$ymax=apply(post_pred2,2,function(x) quantile(x,0.975))

ggplot(newd)+
  stat_summary(data=pt,aes(x=age,y=calf),fun.data = 'mean_cl_boot',geom='pointrange')+
  geom_ribbon(aes(x=age,y=y,ymin=ymin,ymax=ymax),alpha=0.2)+
  geom_line(aes(x=age,y=y,ymin=ymin,ymax=ymax))

## Warning in geom_line(aes(x = age, y = y, ymin = ymin, ymax = ymax)): Ignoring
## unknown aesthetics: ymin and ymax
```

