Introduction à la modélisation statistique bayésienne

Un cours en R et Stan avec brms

Ladislas Nalborczyk (CNRS, LPL, Aix-Marseille Univ)

Planning

Cours n°01 : Introduction à l'inférence bayésienne
Cours n°02 : Modèle Beta-Binomial
Cours n°03 : Introduction à brms, modèle de régression linéaire
Cours n°04 : Modèle de régression linéaire (suite)
Cours n°05 : Markov Chain Monte Carlo
Cours n°06 : Modèle linéaire généralisé
Cours n°07 : Comparaison de modèles
Cours n°08 : Modèles multi-niveaux (généralisés)
Cours n°09 : Examen final

Introduction

$$y_i \sim \mathrm{Normal}(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta_1 \times x_{1i} + \beta_2 \times x_{2i}$$

Le modèle linéaire Gaussien qu'on a vu aux Cours n°03 et n°04 est caractérisé par un ensemble de postulats, entre autres choses :

- Les résidus sont distribués selon une loi Normale.
- La variance de cette distribution Normale est constante (postulat d'homogénéité de la variance).
- Les prédicteurs agissent sur la moyenne de cette distribution.
- La moyenne suit un modèle linéaire ou multi-linéaire.

Introduction

Cette modélisation (le choix d'une distribution Normale) induit plusieurs contraintes, par exemple :

- Les données observées sont définies sur un espace continu.
- Cet espace n'est pas borné.

Comment modéliser certaines données qui ne respectent pas ces contraintes ? Par exemple, la proportion de bonnes réponses à un test (bornée entre 0 et 1), un temps de réponse (restreint à des valeurs positives et souvent distribué de manière approximativement log-normale), un nombre d'avalanches...

Introduction

Nous avons déjà rencontré un modèle différent : le modèle Beta-Binomial (cf. Cours n°02).

$$
\begin{aligned}
y_i &\sim \mathrm{Binomial}(n, p) \\
p &\sim \mathrm{Beta}(a, b)
\end{aligned}
$$

- Les données observées sont binaires (e.g., 0 vs. 1) ou le résultat d'une somme d'observations binaires (e.g., une proportion).
- La probabilité de succès (obtenir 1) a priori se caractérise par une distribution Beta.
- La probabilité de succès (obtenir 1) ne dépend d'aucun prédicteur.

Introduction

Cette modélisation induit deux contraintes :

- Les données observées sont définies sur un espace discret.
- Cet espace est borné.

Comment pourrait-on ajouter des prédicteurs à ce modèle ?

Modèle linéaire généralisé

$$
\begin{aligned}
y_i &\sim \mathrm{Binomial}(n, p_i) \\
f(p_i) &= \alpha + \beta \times x_i
\end{aligned}
$$

Objectifs :

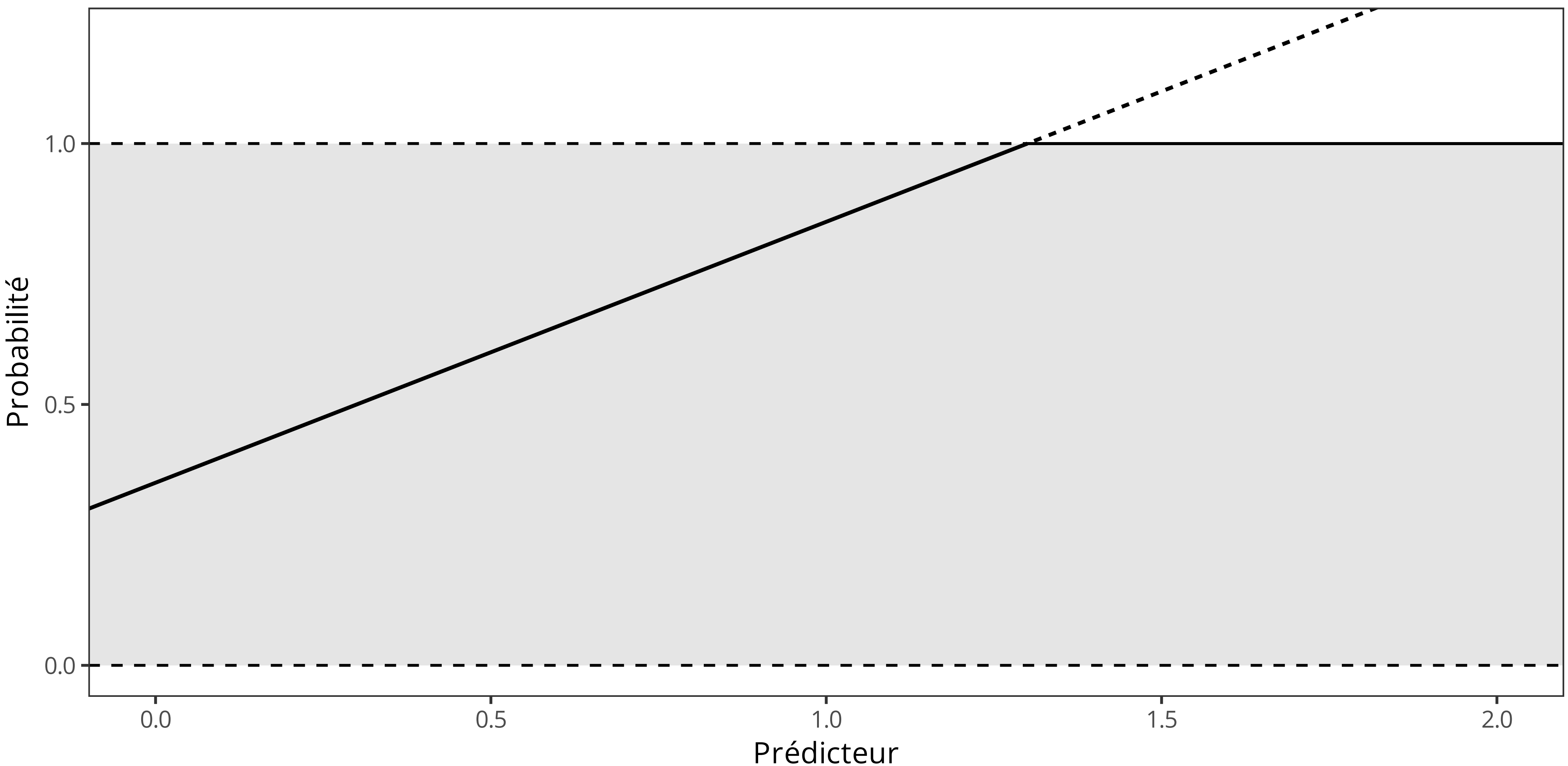
- Rendre compte de données discrètes (e.g., échec/succès) générées par un processus unique.
- Introduire des prédicteurs dans le modèle.

Deux changements par rapport au modèle Gaussien :

- L'utilisation d'une distribution de probabilité Binomiale.
- Le modèle linéaire ne sert plus à décrire directement un des paramètres de la distribution, mais une fonction de ce paramètre (on peut aussi considérer que le modèle Gaussien était formulé avec une fonction identité).

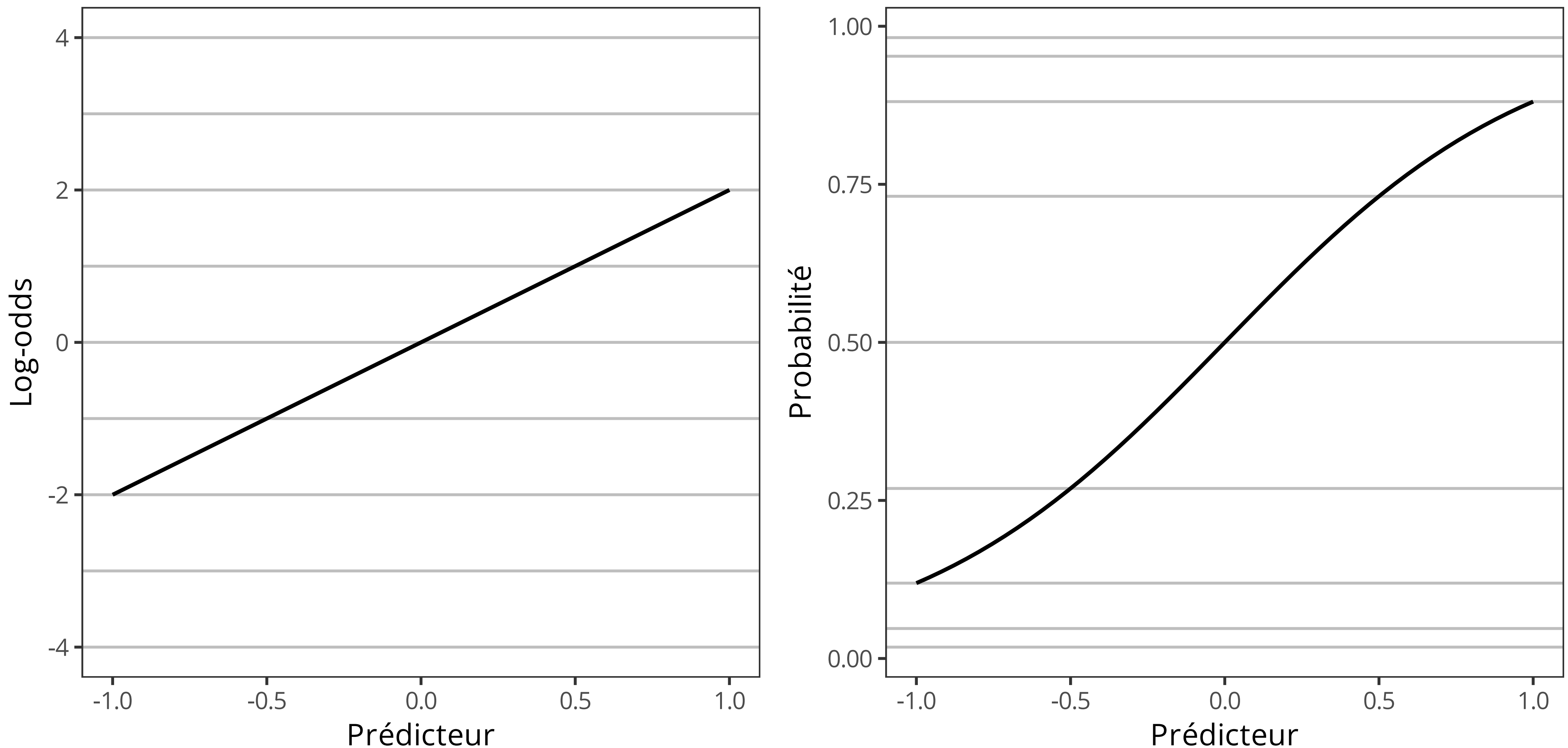
Fonction de lien

Les fonctions de lien ont pour tâche de mettre en correspondance l'espace d'un modèle linéaire (non borné) avec l'espace d'un paramètre potentiellement borné comme une probabilité, définie sur l'intervalle $[0, 1]$.



Fonction de lien

Les fonctions de lien ont pour tâche de mettre en correspondance l'espace d'un modèle linéaire (non borné) avec l'espace d'un paramètre potentiellement borné comme une probabilité, définie sur l'intervalle $[0, 1]$.



Régression logistique

La fonction logit du GLM binomial (on parle de “log-odds”) :

$$\text{logit}(p_i) = \log\left(\frac{p_i}{1 - p_i}\right)$$

La cote d’un évènement (“odds” en anglais) est le ratio entre la probabilité que l’évènement se produise et la probabilité qu’il ne se produise pas. Le logarithme de cette cote est prédit par un modèle linéaire.

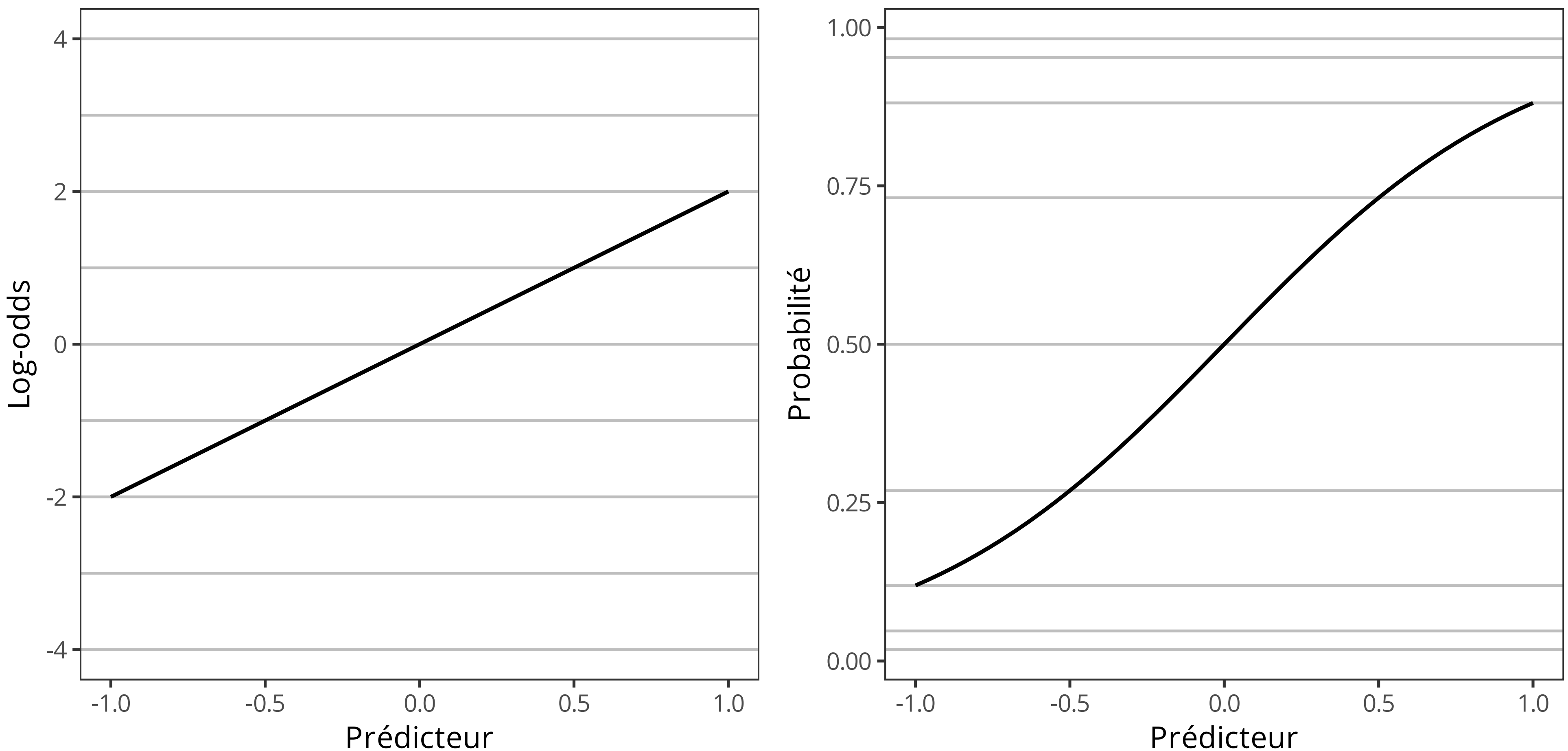
$$\log\left(\frac{p_i}{1 - p_i}\right) = \alpha + \beta \times x_i$$

Pour retrouver la probabilité d’un évènement, il faut utiliser la fonction de **lien inverse**, la fonction **logistique** (ou logit-inverse) :

$$p_i = \frac{\exp(\alpha + \beta \times x_i)}{1 + \exp(\alpha + \beta \times x_i)}$$

Complications induites par la fonction de lien

Ces fonctions de lien posent des problèmes d'interprétation : Un changement d'une unité sur un prédicteur n'a plus un effet constant sur la probabilité mais la modifie plus ou moins en fonction de son éloignement à l'origine. Quand $x = 0$, une augmentation d'une demi-unité (i.e., $\Delta x = 0.5$) se traduit par une augmentation de la probabilité de 0.25. Puis, chaque augmentation d'une demi-unité se traduit par une augmentation de p de plus en plus petite...



Complications induites par la fonction de lien

Deuxième complication : cette fonction de lien “force” chaque prédicteur à interagir avec lui même et à interagir avec TOUS les autres prédicteurs, même si les interactions ne sont pas explicites...

Dans un modèle Gaussien, le taux de changement de y en fonction de x est donné par $\partial(\alpha + \beta x) \, / \, \partial x = \beta$ et ne dépend pas de x (i.e., β est constant).

Dans un GLM binomial (avec une fonction de lien logit), la probabilité d’un évènement est donnée par la fonction logistique :

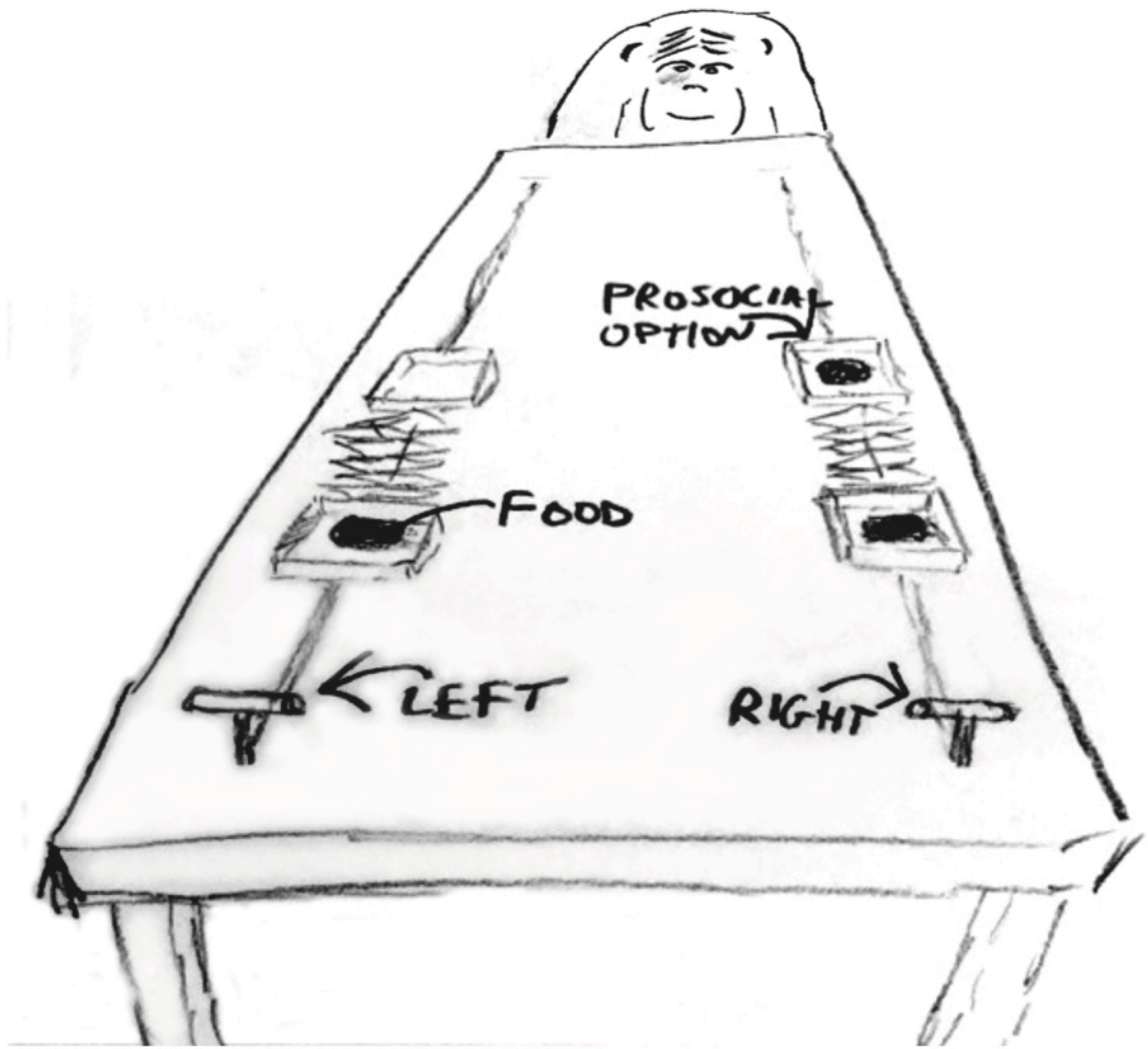
$$p_i = \frac{\exp(\alpha + \beta \times x_i)}{1 + \exp(\alpha + \beta \times x_i)}$$

Et le taux de changement de p en fonction du prédicteur x est donné par :

$$\frac{\partial p}{\partial x} = \frac{\beta}{2(1 + \cosh(\alpha + \beta \times x))}$$

On voit que la variation sur p due au prédicteur x est fonction du prédicteur x, et dépend également de la valeur de α... !

Exemple de régression logistique : La prosocialité chez le chimpanzé



Régression logistique

```r
library(tidyverse)
library(imsb)

df1 <- open_data(chimpanzees)
str(df1)
```

```
'data.frame':   504 obs. of  8 variables:
 $ actor       : int  1 1 1 1 1 1 1 1 1 1 ...
 $ recipient   : int  NA NA NA NA NA NA NA NA NA NA ...
 $ condition   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ block       : int  1 1 1 1 1 1 2 2 2 2 ...
 $ trial       : int  2 4 6 8 10 12 14 16 18 20 ...
 $ prosoc_left : int  0 0 1 0 1 1 1 1 0 0 ...
 $ chose_prosoc: int  1 0 0 1 1 1 0 0 1 1 ...
 $ pulled_left : int  0 1 0 0 1 1 0 0 0 0 ...
```

- **pulled_left** : 1 lorsque le chimpanzé pousse le levier gauche, 0 sinon.
- **prosoc_left** : 1 lorsque le levier gauche est associé à l'option prosociale, 0 sinon.
- **condition** : 1 lorsqu'un partenaire est présent, 0 sinon.

Régression logistique

Le problème

On cherche à savoir si la présence d'un singe partenaire incite le chimpanzé à appuyer sur le levier prosocial, c'est à dire l'option qui donne de la nourriture aux deux individus. Autrement dit, est-ce qu'il existe une interaction entre l'effet de la latéralité et l'effet de la présence d'un autre chimpanzé sur la probabilité d'actionner le levier gauche.

Les variables

- Observations (`pulled_left`) : Ce sont des variables de Bernoulli. Elles prennent comme valeur 0/1.
- Prédicteur (`prosoc_left`) : Est-ce que les deux plats sont sur la gauche ou sur la droite ?
- Prédicteur (`condition`) : Est-ce qu'un partenaire est présent ?

Régression logistique

$$
\begin{aligned}
L_i &\sim \mathrm{Binomial}(1, p_i) \\
\text{(équivalent à)} \quad L_i &\sim \mathrm{Bernoulli}(p_i) \\
\mathrm{logit}(p_i) &= \alpha \\
\alpha &\sim \mathrm{Normal}(0, \omega)
\end{aligned}
$$

Modèle mathématique sans prédicteur. Comment choisir une valeur pour ω... ?

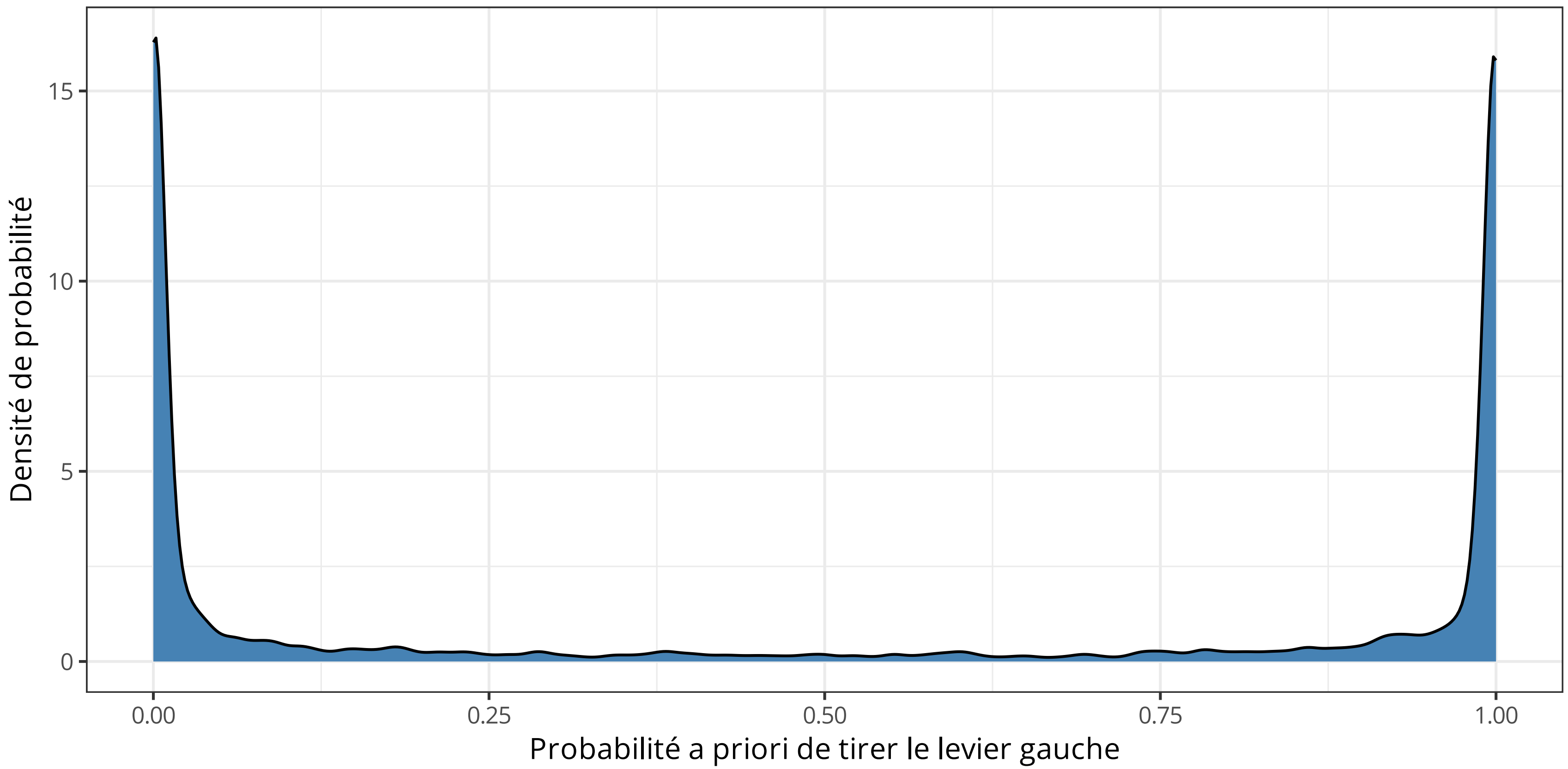
Prior predictive check

On écrit le modèle précédent avec `brms` et on échantillonne à partir du prior afin de vérifier que les prédictions du modèle (sur la base du prior seul) correspondent à nos attentes.

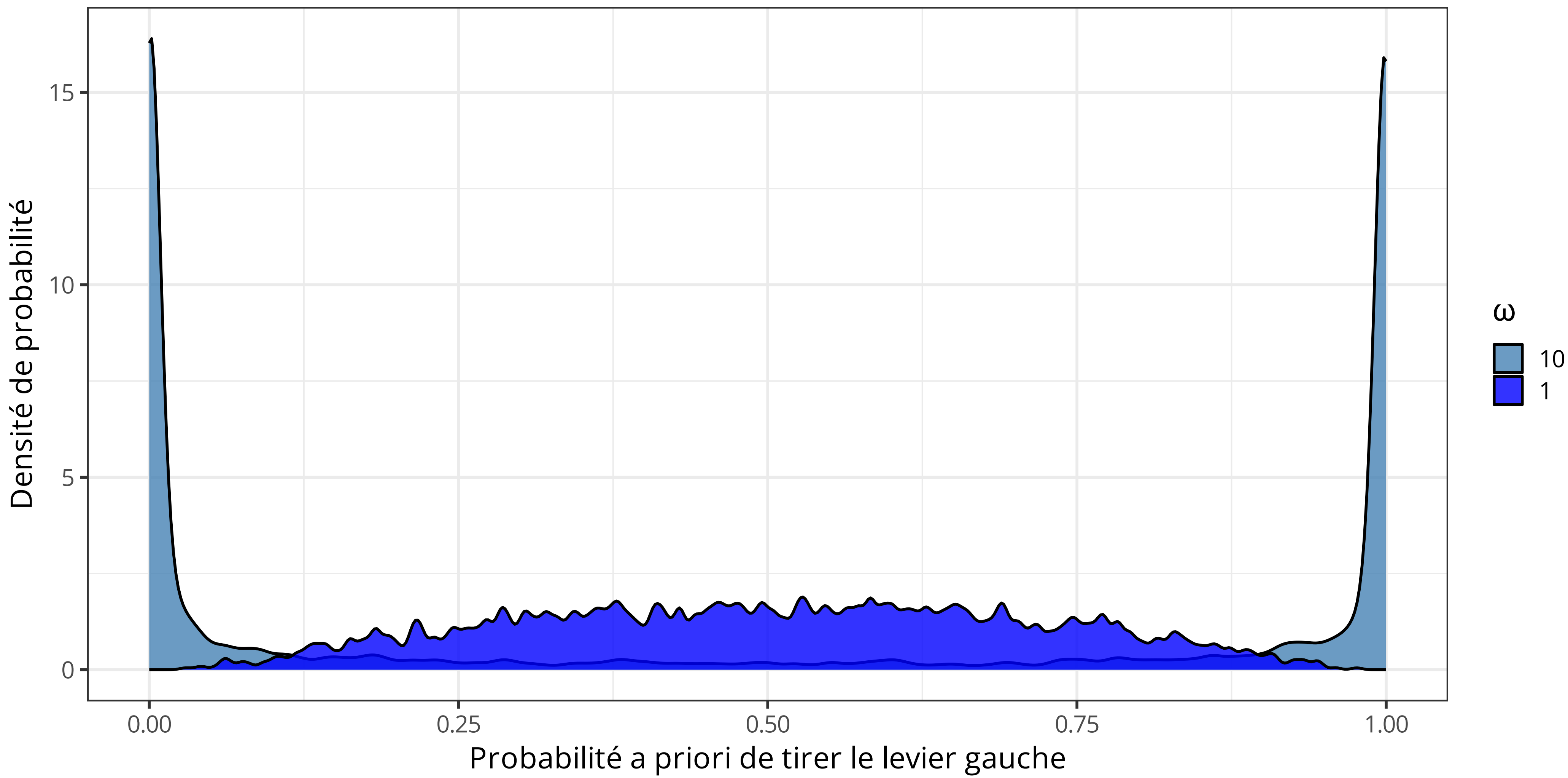
```r
library(brms)

mod1.1 <- brm(
  # "trials" permet de définir le nombre d'essais (i.e., n)
  formula = pulled_left | trials(1) ~ 1,
  family = binomial(),
  prior = prior(normal(0, 10), class = Intercept),
  data = df1,
  chains = 4, cores = 4,
  # on veut récupérer les échantillons issus du prior
  sample_prior = "yes"
  )
```

Prior predictive check

```
# récupère les échantillons (sur la base) du prior
prior_draws(x = mod1.1) %>%
  # applique la fonction de lien inverse
  mutate(p = brms::inv_logit_scaled(Intercept) ) %>%
  ggplot(aes(x = p) ) +
  geom_density(fill = "steelblue", adjust = 0.1) +
  labs(x = "Probabilité a priori de tirer le levier gauche", y = "Densité de probabilité")
```



Prior predictive check



Régression logistique

L'intercept s'interprète dans l'espace des log-odds… pour l'interpréter comme une probabilité, il faut appliquer la fonction de lien inverse. On peut utiliser la fonction `brms::inv_logit_scaled()` ou la fonction `plogis()`.

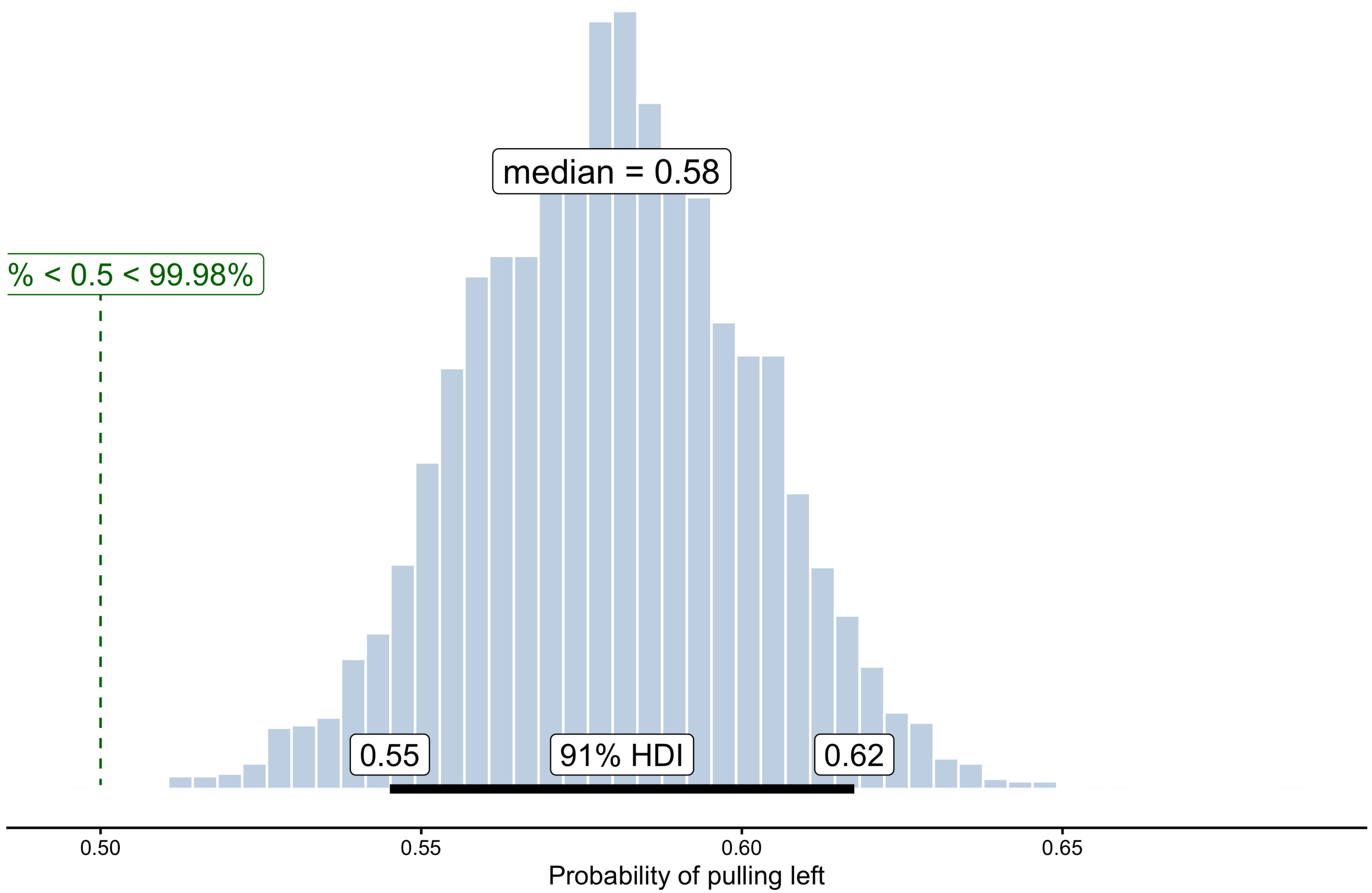
```
fixed_effects <- fixef(mod1.2) # effets fixes (i.e., que l'intercept ici)
plogis(fixed_effects) # fonction de lien inverse
```

```
           Estimate Est.Error      Q2.5     Q97.5
Intercept 0.5796159 0.5223755 0.5368561 0.6214347
```

En moyenne (sans considérer les prédicteurs), il semblerait que les chimpanzés aient légèrement plus tendance à appuyer sur le levier gauche que sur le levier droit…

Régression logistique

```r
post <- as_draws_df(x = mod1.2) # récupère les échantillons du posterior
intercept_samples <- plogis(post$b_Intercept) # échantillons pour l'intercept

posterior_plot(samples = intercept_samples, compval = 0.5) + labs(x = "Probability of pulling left")
```



Régression logistique

Et si on ajoutait des prédicteurs... comment choisir une valeur pour ω ?

$$
\begin{aligned}
L_i &\sim \mathrm{Binomial}(1, p_i) \\
\mathrm{logit}(p_i) &= \alpha + \beta_P P_i + \beta_C C_i + \beta_{PC} P_i C_i \\
\alpha &\sim \mathrm{Normal}(0, 1) \\
\beta_P, \beta_C, \beta_{PC} &\sim \mathrm{Normal}(0, \omega)
\end{aligned}
$$

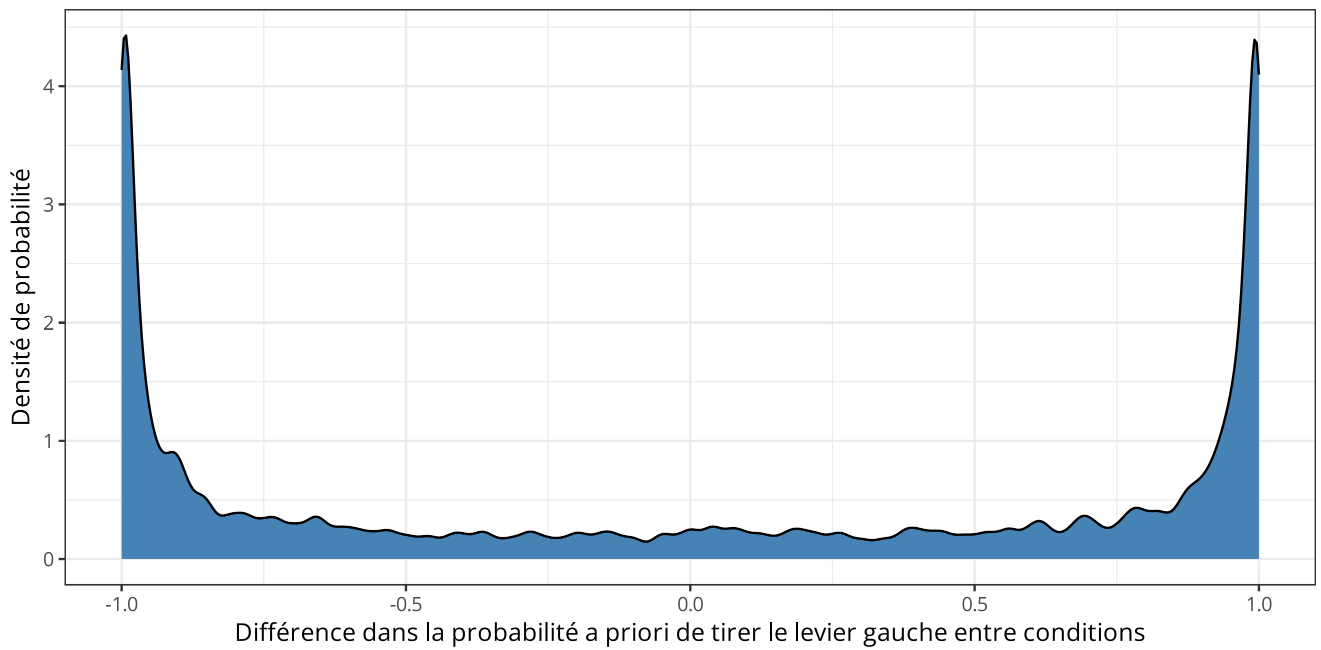
- L_i indique si le singe a poussé le levier gauche (`pulled_left`).
- P_i indique si le coté gauche correspond au coté prosocial.
- C_i indique la présence d'un partenaire.

Régression logistique

```r
# recoding predictors
df1 <- df1 %>%
  mutate(
    prosoc_left = ifelse(prosoc_left == 1, 0.5, -0.5),
    condition = ifelse(condition == 1, 0.5, -0.5)
    )

priors <- c(
  prior(normal(0, 1), class = Intercept),
  prior(normal(0, 10), class = b)
  )

mod2.1 <- brm(
  formula = pulled_left | trials(1) ~ 1 + prosoc_left * condition,
  family = binomial,
  prior = priors,
  data = df1,
  chains = 4, cores = 4,
  sample_prior = "yes"
  )
```

Prior predictive check

```r
prior_draws(x = mod2.1) %>% # échantillons du prior
  mutate(
    condition1 = plogis(Intercept - 0.5 * b), # p dans condition 1
    condition2 = plogis(Intercept + 0.5 * b) # p dans condition 0
    ) %>%
  ggplot(aes(x = condition2 - condition1) ) + # on plot la différence
  geom_density(fill = "steelblue", adjust = 0.1) +
  labs(
    x = "Différence dans la probabilité a priori de tirer le levier gauche entre conditions",
    y = "Densité de probabilité"
    )
```

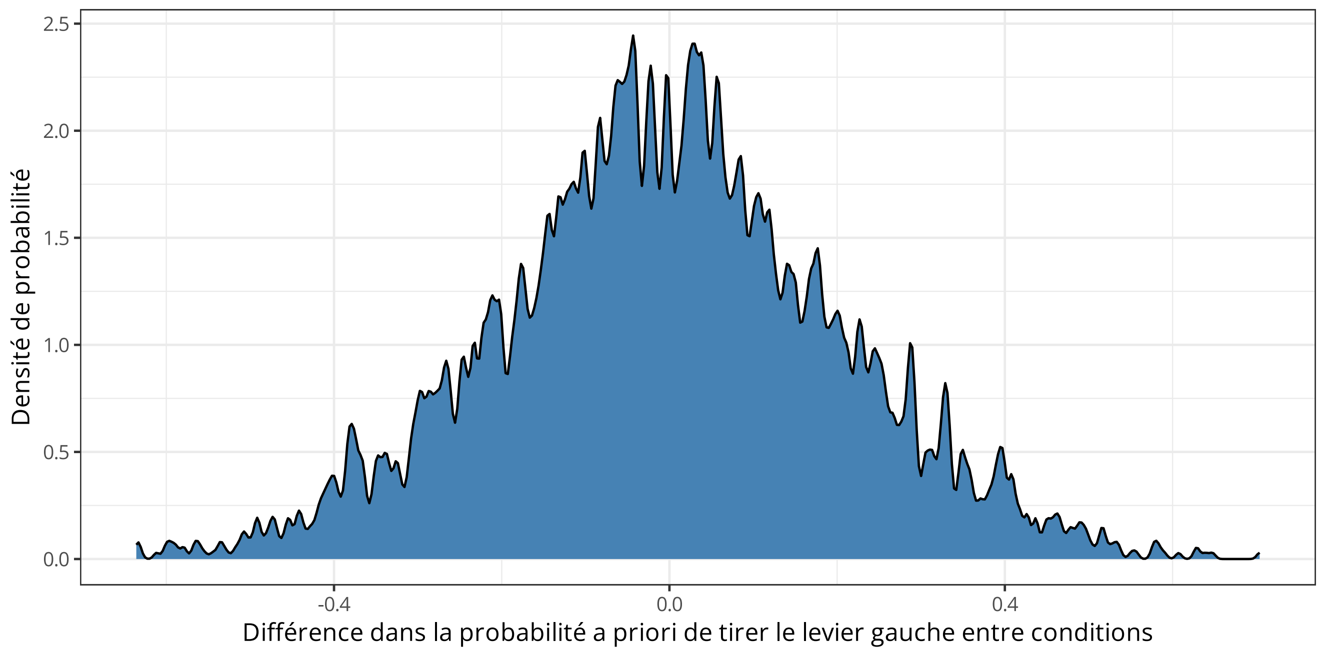


Régression logistique

```r
priors <- c(
  prior(normal(0, 1), class = Intercept),
  prior(normal(0, 1), class = b)
  )

mod2.2 <- brm(
  formula = pulled_left | trials(1) ~ 1 + prosoc_left * condition,
  family = binomial,
  prior = priors,
  data = df1,
  chains = 4, cores = 4,
  sample_prior = "yes"
  )
```

Prior predictive check

```r
prior_draws(mod2.2) %>% # échantillons du prior
  mutate(
    condition1 = plogis(Intercept - 0.5 * b), # p dans condition 1
    condition2 = plogis(Intercept + 0.5 * b) # p dans condition 0
    ) %>%
  ggplot(aes(x = condition2 - condition1) ) +
  geom_density(fill = "steelblue", adjust = 0.1) +
  labs(
    x = "Différence dans la probabilité a priori de tirer le levier gauche entre conditions",
    y = "Densité de probabilité"
    )
```



Régression logistique

```r
summary(mod2.2)
```

```
 Family: binomial 
  Links: mu = logit 
Formula: pulled_left | trials(1) ~ 1 + prosoc_left * condition 
   Data: df1 (Number of observations: 504) 
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
                      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS
Intercept                 0.33      0.09     0.16     0.50 1.00     4504
prosoc_left               0.54      0.18     0.20     0.89 1.00     4313
condition                -0.19      0.18    -0.55     0.16 1.00     3745
prosoc_left:condition     0.16      0.35    -0.52     0.85 1.00     4887
                      Tail_ESS
Intercept                 3099
prosoc_left               3216
condition                 2663
prosoc_left:condition     3395

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).
```

Effet relatif vs. Effet absolu

Le modèle linéaire ne prédit pas directement la probabilité mais le log-odds de la probabilité :

$$\text{logit}(p_i) = \log\left(\frac{p_i}{1 - p_i}\right) = \alpha + \beta \times x_i$$

On peut distinguer et interpréter deux types d'effets.

Effet relatif : L'effet relatif porte sur le logarithme du rapport des probabilités. Il indique une proportion de changement induit par le prédicteur sur **les chances** de succès (ou plutôt, sur la cote). Cela ne nous dit rien de la probabilité de l'évènement, dans l'absolu.

Effet absolu : Effet qui porte directement sur la probabilité d'un évènement. Il dépend de tous les paramètres du modèle et nous donne l'impact effectif d'un changement d'une unité d'un prédicteur (dans l'espace des probabilités).

Effet relatif

Il s'agit d'une **proportion** de changement induit par le prédicteur sur le rapport des chances ou "cote" (odds). Illustration avec un modèle sans interaction.

$$
\begin{aligned}
\log\left(\frac{p_i}{1 - p_i}\right) &= \alpha + \beta x_i \\
\frac{p_i}{1 - p_i} &= \exp(\alpha + \beta x_i)
\end{aligned}
$$

La cote proportionnelle q d'un évènement est le nombre par lequel la cote est multipliée lorsque x_i augmente d'une unité.

$$
q = \frac{\exp(\alpha + \beta(x_i + 1))}{\exp(\alpha + \beta x_i)} = \frac{\exp(\alpha)\exp(\beta x_i)\exp(\beta)}{\exp(\alpha)\exp(\beta x_i)} = \exp(\beta)
$$

Lorsque $q = 2$ (par exemple), une augmentation de x_i d'une unité génère un doublement de la cote.

Interprétation de l'effet relatif

L'effet relatif d'un paramètre **dépend également des autres paramètres**. Dans le modèle précédent, le prédicteur `prosoc_left` augmente le logarithme de la cote d'environ 0.54, ce qui se traduit par une augmentation de la cote de $\exp(0.54) \approx 1.72$ soit une augmentation d'environ 72% de la cote.

Supposons que l'intercept $\alpha = 4$.

- La probabilité de pousser le levier sans autre considération est de $\text{logit}^{-1}(4) \approx 0.98$.
- En considérant l'effet de `prosoc_left`, on obtient $\text{logit}^{-1}(4 + 0.54) \approx 0.99$.

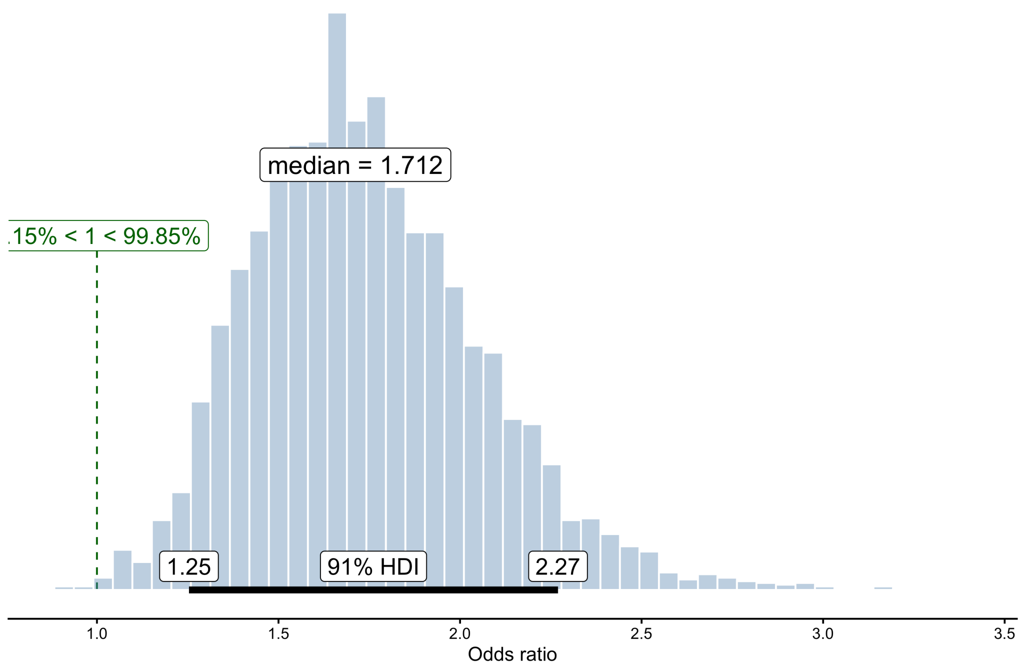
Une augmentation de 72% sur le log-odds se traduit par une augmentation de seulement 1% sur la probabilité effective... Les effets relatifs peuvent conduire à de mauvaises interprétations lorsqu'on ne considère pas l'échelle de la variable mesurée.

Interprétation de l'effet relatif

```r
fixef(mod2.2) # récupère les estimations des effets dits "fixes"
```

```
                        Estimate Est.Error       Q2.5     Q97.5
Intercept              0.3270807 0.0893073  0.1573382 0.5020609
prosoc_left            0.5403104 0.1795264  0.1953843 0.8932221
condition             -0.1904050 0.1808399 -0.5517242 0.1649813
prosoc_left:condition  0.1645906 0.3517610 -0.5164286 0.8530099
```

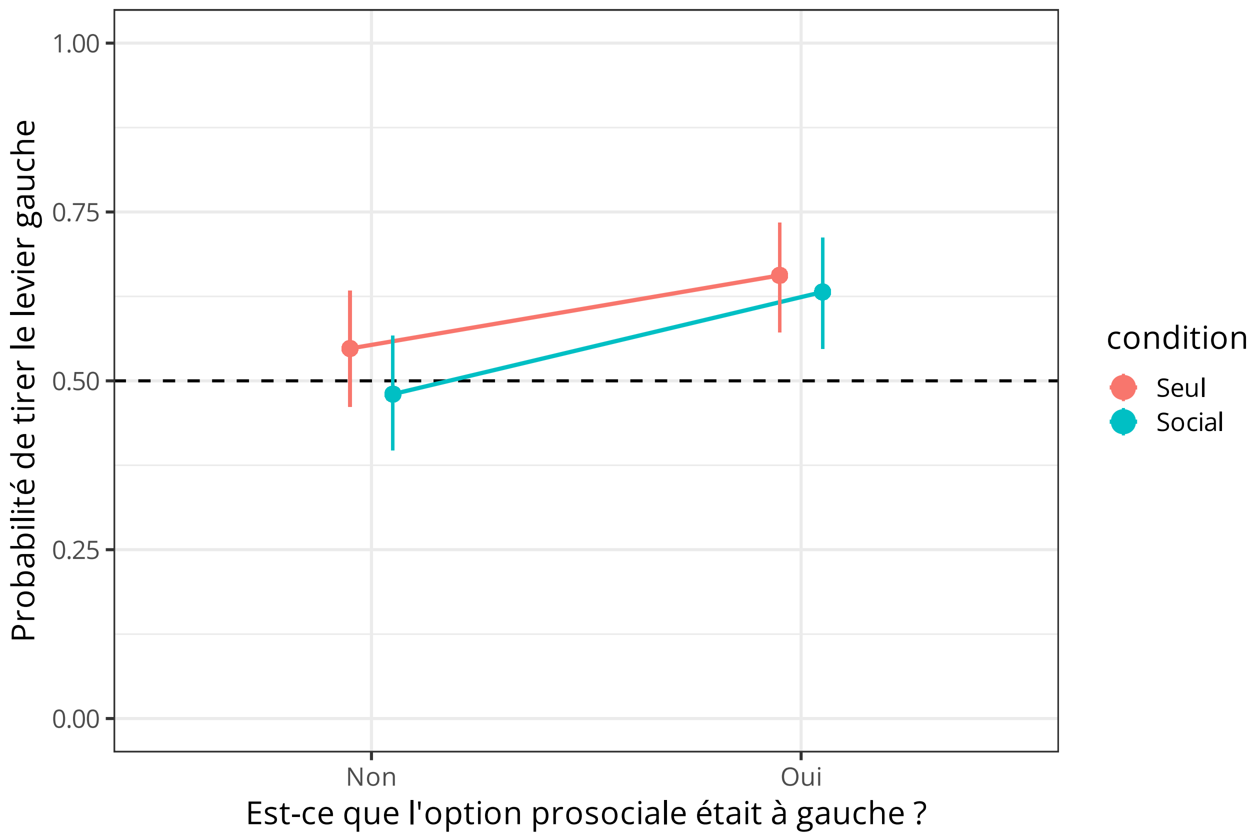
```r
post <- as_draws_df(x = mod2.2) # échantillons du posterior
posterior_plot(samples = exp(post$b_prosoc_left), compval = 1) + labs(x = "Odds ratio")
```



Effet absolu

L'effet absolu dépend de tous les paramètres du modèle et nous donne l'impact effectif d'un changement d'une unité d'un prédicteur (dans l'espace des probabilités).

```
model_predictions <- fitted(mod2.2) %>% # prédiction pour p (i.e., la probabilité)
  data.frame() %>%
  bind_cols(df1) %>%
  mutate(condition = factor(condition), prosoc_left = factor(prosoc_left) )
```



Régression binomiale agrégée

Ces données représentent le nombre de candidatures à l'université de Berkeley par sexe et par département. Chaque candidature est acceptée ou rejetée et les résultats sont agrégés par département et par sexe.

```
(df2 <- open_data(admission) )
```

```
   dept gender admit reject applications
1     A   Male   512    313          825
2     A Female    89     19          108
3     B   Male   353    207          560
4     B Female    17      8           25
5     C   Male   120    205          325
6     C Female   202    391          593
7     D   Male   138    279          417
8     D Female   131    244          375
9     E   Male    53    138          191
10    E Female    94    299          393
11    F   Male    22    351          373
12    F Female    24    317          341
```

Existe-t-il un biais de recrutement lié au sexe ?

Régression binomiale agrégée

On va construire un modèle de la décision d'admission en prenant comme prédicteur le sexe du candidat.

$$
\begin{aligned}
\color{orangered}{\text{admit}_i} &\color{orangered}{\sim \text{Binomial}(n_i, p_i)} \\
\text{logit}(p_i) &= \alpha + \beta_m \times m_i \\
\color{steelblue}{\alpha} &\color{steelblue}{\sim \text{Normal}(0, 1)} \\
\color{steelblue}{\beta_m} &\color{steelblue}{\sim \text{Normal}(0, 1)}
\end{aligned}
$$

Les variables :

- admit_i : Le nombre de candidatures acceptées (`admit`).
- n_i : Le nombre total de candidatures (`applications`).
- m_i : Le sexe du candidat (`1 = Male`).

Régression binomiale agrégée

```r
priors <- c(prior(normal(0, 1), class = Intercept) )

mod3 <- brm(
  formula = admit | trials(applications) ~ 1,
  family = binomial(link = "logit"),
  prior = priors,
  data = df2,
  chains = 4, cores = 4,
  sample_prior = "yes"
  )
```

Régression binomiale agrégée

```r
priors <- c(
  prior(normal(0, 1), class = Intercept),
  prior(normal(0, 1), class = b)
  )

# dummy-coding
df2$male <- ifelse(df2$gender == "Male", 1, 0)

mod4 <- brm(
  formula = admit | trials(applications) ~ 1 + male,
  family = binomial(link = "logit"),
  prior = priors,
  data = df2,
  chains = 4, cores = 4,
  sample_prior = "yes"
  )
```

Régression binomiale agrégée

```r
summary(mod4)
```

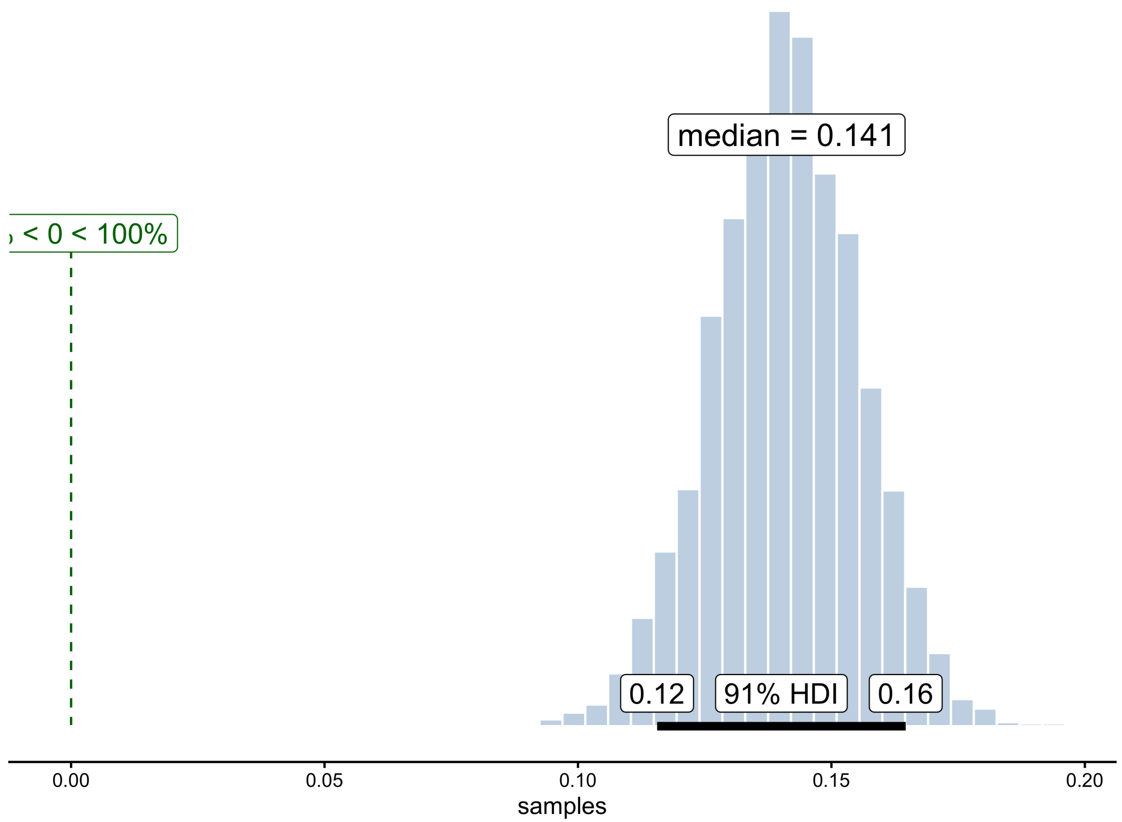
```
 Family: binomial 
  Links: mu = logit 
Formula: admit | trials(applications) ~ 1 + male 
   Data: df2 (Number of observations: 12) 
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
          Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept    -0.83      0.05    -0.93    -0.73 1.00     2151     2348
male          0.61      0.06     0.48     0.73 1.00     2528     2313

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).
```

Être un homme semble être un avantage... ! Le rapport des cotes est de $\exp(0.61) \approx 1.84$.

Régression binomiale agrégée

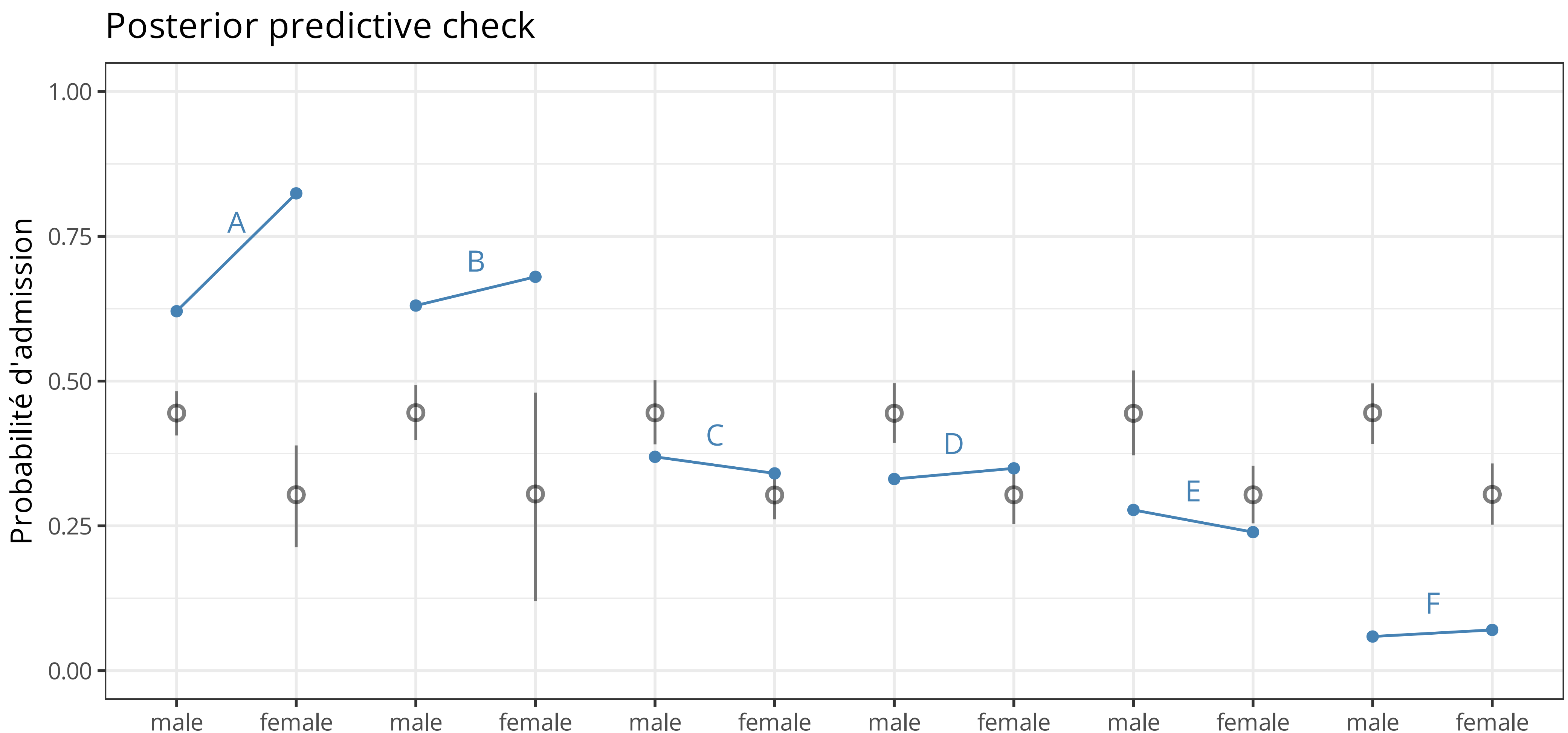
Calculons la différence de probabilité d'admission entre hommes et femmes.

```r
post <- as_draws_df(x = mod4)
p.admit.male <- plogis(post$b_Intercept + post$b_male)
p.admit.female <- plogis(post$b_Intercept)
diff.admit <- p.admit.male - p.admit.female
posterior_plot(samples = diff.admit, compval = 0)
```



Visualiser les prédictions du modèle

On examine les prédictions du modèle (en noir) par département.



Régression binomiale agrégée

Les prédictions du modèle sont très mauvaises… Il n’y a que deux départements pour lesquels les femmes ont de moins bonnes prédictions que les hommes (C et E) alors que le modèle prédit une probabilité d’admission plus basse pour tous les départements…

Le problème est double :

- Les hommes et les femmes ne postulent pas aux mêmes départements.
- Les départements n’ont pas tous les mêmes effectifs.

C’est le “paradoxe” de Simpson… remarques :

- La distribution postérieure seule n’aurait pas permis de détecter ce problème.
- C’est l’étude des prédictions du modèle qui nous a permis de mettre le doigt sur le problème…

Régression binomiale agrégée

On construit donc un modèle de la décision d'admission en fonction du genre, au sein de chaque département.

$$
\begin{aligned}
\text{admit}_i &\sim \text{Binomial}(n_i, p_i) \\
\text{logit}(p_i) &= \alpha_{\text{dept}[i]} + \beta_m \times m_i \\
\alpha_{\text{dept}[i]} &\sim \text{Normal}(0, 1) \\
\beta_m &\sim \text{Normal}(0, 1)
\end{aligned}
$$

Régression binomiale aggrégée

```
# modèle sans prédicteur
mod5 <- brm(
  admit | trials(applications) ~ 0 + dept,
  family = binomial(link = "logit"),
  prior = prior(normal(0, 1), class = b),
  chains = 4, cores = 4,
  data = df2
  )

# modèle avec prédicteur
mod6 <- brm(
  admit | trials(applications) ~ 0 + dept + male,
  family = binomial(link = "logit"),
  prior = prior(normal(0, 1), class = b),
  chains = 4, cores = 4,
  data = df2
  )
```

Régression binomiale aggrégée

```r
summary(mod6)
```

```
 Family: binomial
  Links: mu = logit
Formula: admit | trials(applications) ~ 0 + dept + male
   Data: df2 (Number of observations: 12)
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
deptA     0.68      0.10     0.49     0.88 1.00     1941     2886
deptB     0.64      0.12     0.42     0.87 1.00     2220     2495
deptC    -0.58      0.08    -0.73    -0.43 1.00     3584     2644
deptD    -0.61      0.09    -0.78    -0.44 1.00     2789     2746
deptE    -1.05      0.10    -1.25    -0.87 1.00     3926     2596
deptF    -2.58      0.15    -2.89    -2.28 1.00     4564     3140
male     -0.10      0.08    -0.26     0.05 1.00     1570     2462

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).
```

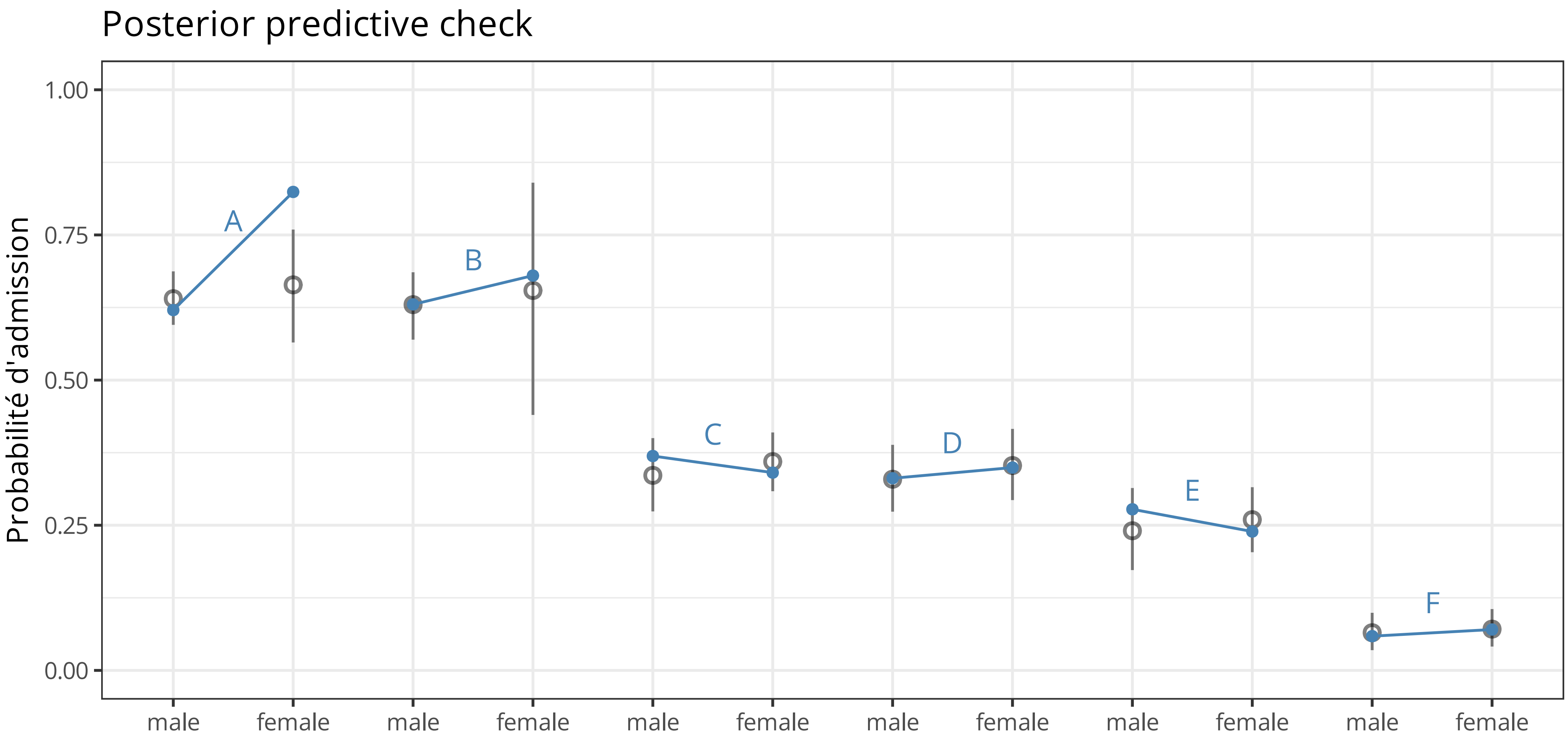
Régression binomiale agrégée

```r
fixef(mod6)
```

```
        Estimate  Est.Error       Q2.5       Q97.5
deptA  0.6828594 0.09837012  0.4918927  0.87713765
deptB  0.6372442 0.11646729  0.4151340  0.86960104
deptC -0.5766685 0.07656305 -0.7275404 -0.43223696
deptD -0.6083267 0.08668628 -0.7804546 -0.43544673
deptE -1.0479897 0.09731482 -1.2479170 -0.86515973
deptF -2.5751091 0.15490640 -2.8940589 -2.28317230
male  -0.1045555 0.08095063 -0.2646223  0.04587855
```

Maintenant, la prédiction pour β_m va dans l'autre sens… La rapport des cotes (odds ratio) est de $\exp(-0.1) = 0.9$, la cote (odds) des hommes est estimée à 90% de la cote des femmes.

Régression binomiale agrégée



Conclusions

Les hommes et les femmes ne postulent pas aux mêmes départements et les départements varient par leur probabilité d'admission. En l'occurrence, les femmes ont plus postulé aux départements E et F (avec une probabilité d'admission plus faible) et ont moins postulé aux départements A ou B, avec une probabilité d'admission plus haute.

Pour évaluer l'effet du sexe sur la probabilité d'admission, il faut donc se poser la question suivante : "Quelle est la différence de probabilité d'admission entre hommes et femmes **au sein de chaque département** ?" (plutôt que de manière générale).

Retenir que le modèle de régression peut être généralisé à différents modèles de génération des données (i.e., différentes distributions de probabilité, comme la distribution Normale, Binomiale, Poisson, etc) et que l'espace des paramètres peut être "connecté" à l'espace des prédicteurs (variables mesurées) grâce à des fonctions de lien (e.g., la fonction logarithme, exponentielle, logit, etc).

Retenir la distinction entre **effet relatif** (e.g., un changement de cote) et **effet absolu** (e.g., une différence de probabilité).

Travaux pratiques - Absentéisme expérimental

Travailler avec des sujets humains implique un minimum de coopération réciproque. Mais ce n'est pas toujours le cas. Une partie non-négligeable des étudiants qui s'inscrivent pour passer des expériences de Psychologie ne se présentent pas le jour prévu... On a voulu estimer la **probabilité de présence d'un étudiant inscrit** en fonction de l'envoi (ou non) d'un mail de rappel (cet exemple est présenté en détails dans deux articles de blog, accessibles ici, et ici).

```r
df3 <- open_data(absence)
df3 %>% sample_frac %>% head(10)
```

```
         day inscription reminder absence presence total
1  Wednesday       panel      yes       0       14    14
2     Friday      doodle       no       7       11    18
3     Monday       panel      yes       6       12    18
4  Wednesday      doodle      yes       0        4     4
5     Monday      doodle       no       5        4     9
6     Friday       panel      yes       0       10    10
7    Tuesday      doodle       no       4       10    14
8     Monday      doodle      yes       2        6     8
9    Tuesday       panel      yes       0        9     9
10 Wednesday      doodle       no       6       11    17
```

Travaux pratiques

- **Quelle est la probabilité qu'un participant, qui s'est inscrit de son propre chef, vienne effectivement passer l'expérience ?**
- Quel est l'effet du rappel ?
- Quel est l'effet du mode d'inscription ?
- Quel est l'effet conjoint de ces deux prédicteurs ?

Travaux pratiques

Écrire le modèle qui prédit la présence d'un participant sans prédicteur.

$$
\begin{aligned}
\color{orangered}{y_{i}} \ &\color{orangered}{\sim \mathrm{Binomial}(n_{i}, p_{i})} \\
\mathrm{logit}(p_{i}) \ &= \alpha \\
\color{steelblue}{\alpha} \ &\color{steelblue}{\sim \mathrm{Normal}(0, 1)}
\end{aligned}
$$

Travaux pratiques

```r
mod7 <- brm(
    presence | trials(total) ~ 1,
    family = binomial(link = "logit"),
    prior = prior(normal(0, 1), class = Intercept),
    data = df3,
    chains = 4, cores = 4
    )
```

```r
fixef(mod7) # effet relatif (log de la cote)
```

```
          Estimate Est.Error      Q2.5    Q97.5
Intercept  1.14834 0.1952533 0.7625212 1.533169
```

```r
fixef(mod7) %>% plogis # effet absolu (probabilité de présence)
```

```
           Estimate Est.Error      Q2.5     Q97.5
Intercept 0.7592075 0.5486588 0.6819009 0.8224695
```

Travaux pratiques

- Quelle est la probabilité qu'un participant, qui s'est inscrit de son propre chef, vienne effectivement passer l'expérience ?
- **Quel est l'effet du rappel ?**
- Quel est l'effet du mode d'inscription ?
- Quel est l'effet conjoint de ces deux prédicteurs ?

Travaux pratiques

On commence par recoder en dummy variables `reminder` et `inscription`.

```
df3 <-
  df3 %>%
  mutate(
    reminder = ifelse(reminder == "no", 0, 1),
    inscription = ifelse(inscription == "panel", 0, 1)
    )

head(df3, n = 10)
```

```
        day inscription reminder absence presence total
1    Friday           1        0       7       11    18
2    Friday           1        1       0        2     2
3    Friday           0        1       0       10    10
4    Monday           1        0       5        4     9
5    Monday           1        1       2        6     8
6    Monday           0        1       6       12    18
7  Thursday           1        0       3       11    14
8   Tuesday           1        0       4       10    14
9   Tuesday           1        1       1        7     8
10  Tuesday           0        1       0        9     9
```

Travaux pratiques

Écrire le modèle qui prédit la présence en fonction du rappel.

$$
\begin{aligned}
y_i &\sim \mathrm{Binomial}(n_i, p_i) \\
\mathrm{logit}(p_i) &= \alpha + \beta \times \mathrm{reminder}_i \\
\alpha &\sim \mathrm{Normal}(0, 1) \\
\beta &\sim \mathrm{Normal}(0, 1)
\end{aligned}
$$

Travaux pratiques

Écrire le modèle qui prédit la présence en fonction du rappel.

```r
priors <- c(
  prior(normal(0, 1), class = Intercept),
  prior(normal(0, 1), class = b)
  )

mod8 <- brm(
    presence | trials(total) ~ 1 + reminder,
    family = binomial(link = "logit"),
    prior = priors,
    data = df3,
    chains = 4, cores = 4
    )
```

Travaux pratiques

Quel est l'effet **relatif** du mail de rappel ?

```
exp(fixef(mod8)[2]) # rapport des cotes sans vs. avec mail de rappel
```

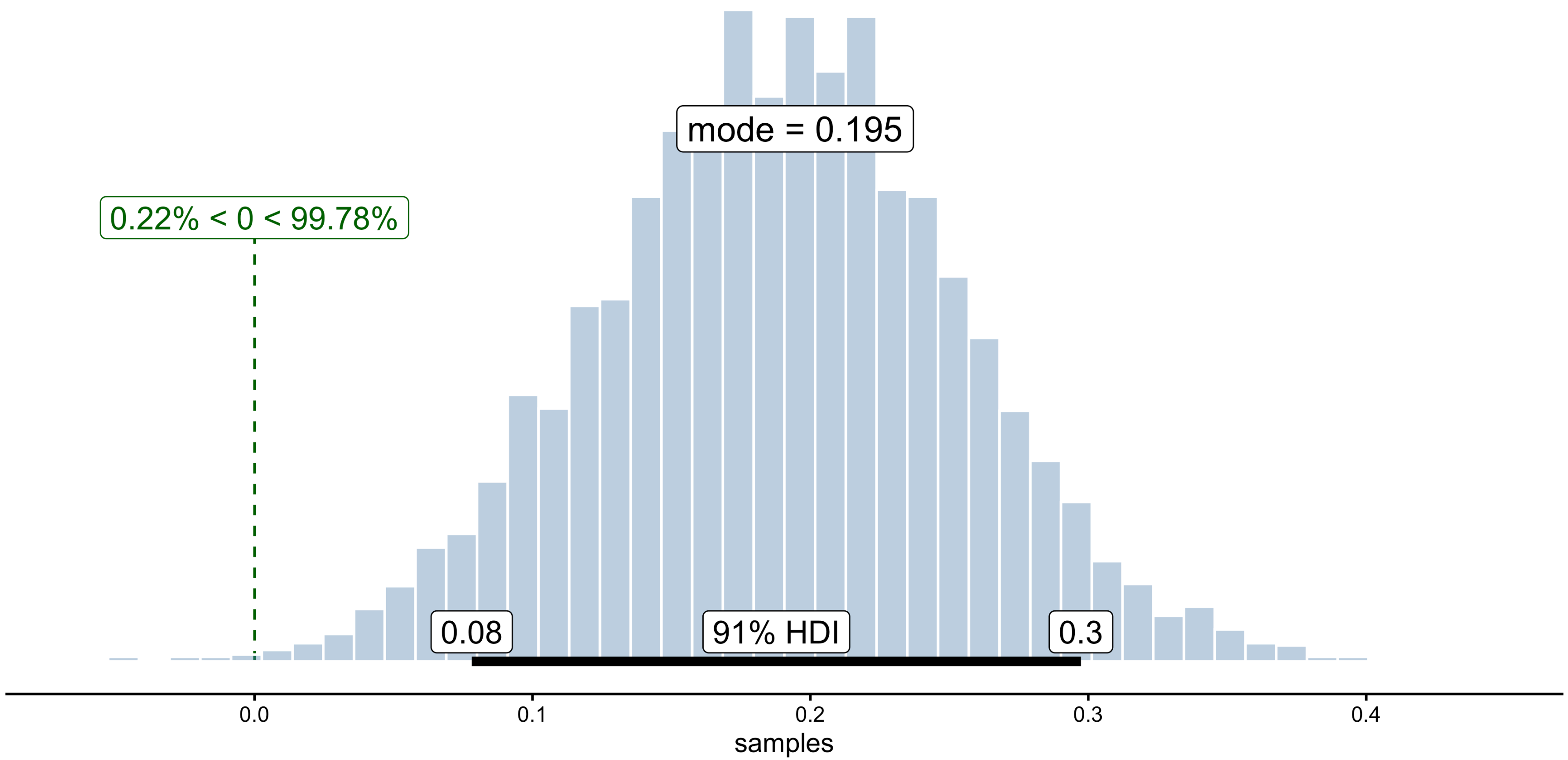
```
[1] 2.969455
```

Envoyer un mail de rappel augmente la cote (le rapport des chances) par environ 3.

Travaux pratiques

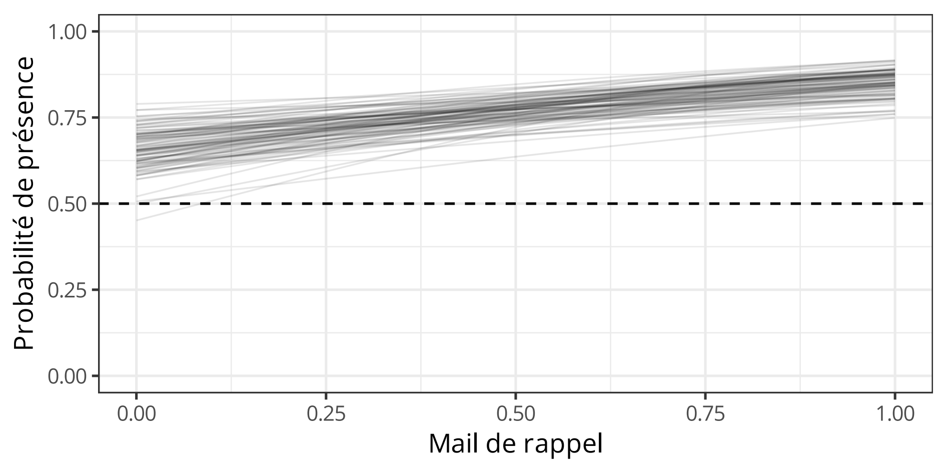
Quel est l'effet **absolu** du mail de rappel ?

```
post <- as_draws_df(x = mod8) # récupères les échantillons du posterior
p.no <- plogis(post$b_Intercept) # probabilité de présence sans mail de rappel
p.yes <- plogis(post$b_Intercept + post$b_reminder) # probabilité de présence avec mail de rappel
posterior_plot(samples = p.yes - p.no, compval = 0, usemode = TRUE)
```



Travaux pratiques

```r
library(tidybayes)
library(modelr)

df3 %>%
  group_by(total) %>%
  data_grid(reminder = seq_range(reminder, n = 1e2) ) %>%
  add_fitted_draws(mod8, newdata = ., n = 100, scale = "linear") %>%
  mutate(estimate = plogis(.value) ) %>%
  group_by(reminder, .draw) %>%
  summarise(estimate = mean(estimate) ) %>%
  ggplot(aes(x = reminder, y = estimate, group = .draw) ) +
  geom_hline(yintercept = 0.5, lty = 2) +
  geom_line(aes(y = estimate, group = .draw), size = 0.5, alpha = 0.1) +
  ylim(0, 1) +
  labs(x = "Mail de rappel", y = "Probabilité de présence")
```



Travaux pratiques

- Quelle est la probabilité qu'un participant, qui s'est inscrit de son propre chef, vienne effectivement passer l'expérience ?
- Quel est l'effet du rappel ?
- **Quel est l'effet du mode d'inscription ?**
- Quel est l'effet conjoint de ces deux prédicteurs ?

Travaux pratiques

Écrire le modèle qui prédit la présence en fonction du mode d'inscription.

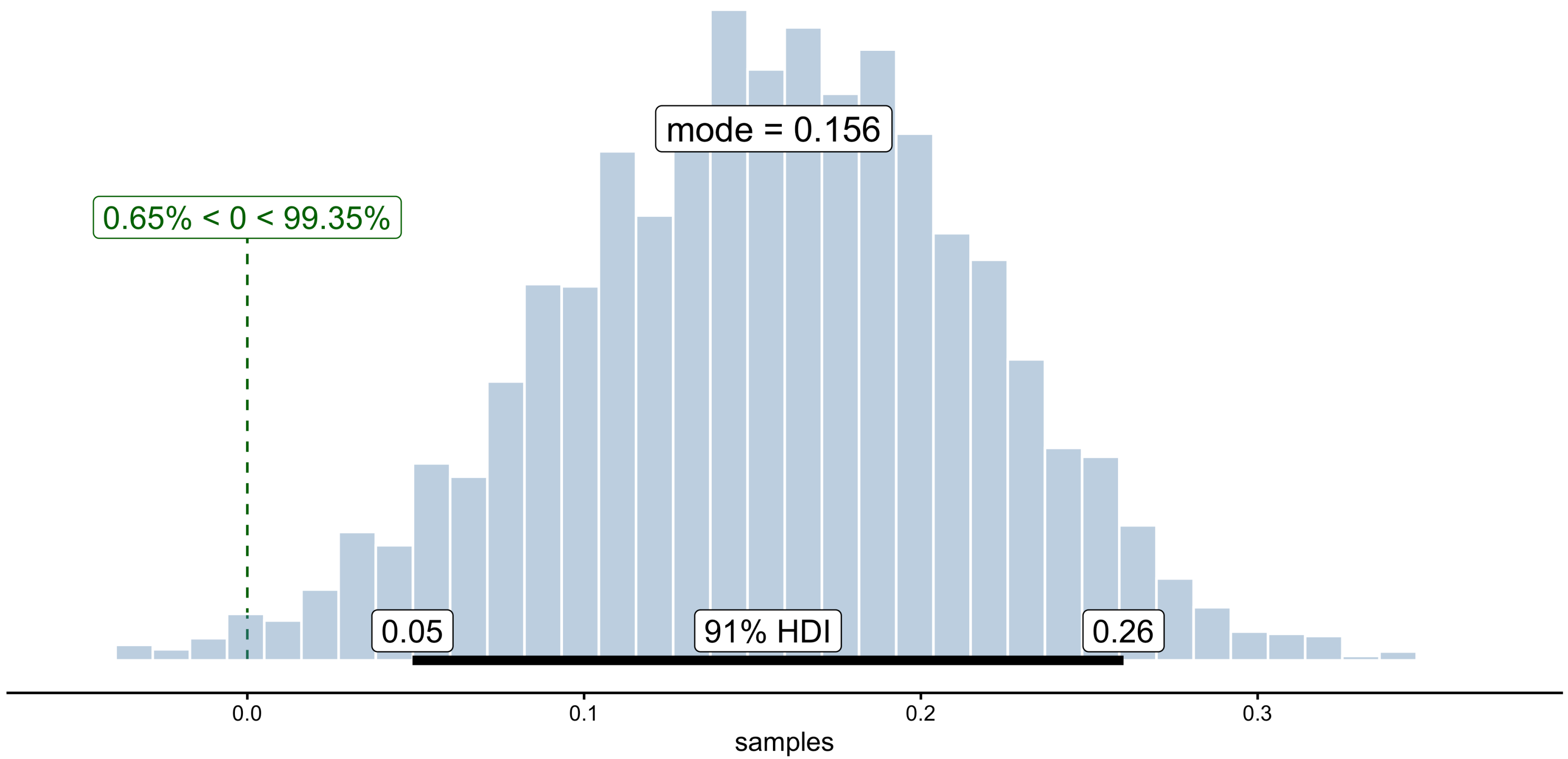
$$
\begin{aligned}
y_i &\sim \mathrm{Binomial}(n_i, p_i) \\
\mathrm{logit}(p_i) &= \alpha + \beta \times \mathrm{inscription}_i \\
\alpha &\sim \mathrm{Normal}(0, 1) \\
\beta &\sim \mathrm{Normal}(0, 1)
\end{aligned}
$$

Travaux pratiques

```r
priors <- c(
  prior(normal(0, 1), class = Intercept),
  prior(normal(0, 1), class = b)
  )

mod9 <- brm(
    presence | trials(total) ~ 1 + inscription,
    family = binomial(link = "logit"),
    prior = priors,
    data = df3,
    chains = 4, cores = 4
    )
```

Travaux pratiques

```r
post <- as_draws_df(x = mod9)
p.panel <- plogis(post$b_Intercept) # probabilité moyenne de présence - panel
p.doodle <- plogis(post$b_Intercept + post$b_inscription) # probabilité moyenne de présence - doodle
posterior_plot(samples = p.panel - p.doodle, compval = 0, usemode = TRUE)
```



La probabilité de présence est augmentée d'environ 0.16 lorsque l'on s'inscrit sur un panel comparativement à une inscription sur un Doodle (effet légèrement plus faible que pour le rappel).

Travaux pratiques

- Quelle est la probabilité qu'un participant, qui s'est inscrit de son propre chef, vienne effectivement passer l'expérience ?
- Quel est l'effet du rappel ?
- Quel est l'effet du mode d'inscription ?
- **Quel est l'effet conjoint de ces deux prédicteurs ?**

Travaux pratiques

Écrire le modèle complet.

$$
\begin{aligned}
y_i &\sim \mathrm{Binomial}(n_i, p_i) \\
\mathrm{logit}(p_i) &= \alpha + \beta_1 \times \mathrm{reminder}_i + \beta_2 \times \mathrm{inscription}_i \\
\alpha &\sim \mathrm{Normal}(0, 1) \\
\beta_1, \beta_2 &\sim \mathrm{Normal}(0, 1)
\end{aligned}
$$

Travaux pratiques

```r
priors <- c(
  prior(normal(0, 1), class = Intercept),
  prior(normal(0, 1), class = b)
  )

mod10 <- brm(
    presence | trials(total) ~ 1 + reminder + inscription,
    family = binomial(link = "logit"),
    prior = priors,
    data = df3,
    chains = 4, cores = 4
    )
```

Travaux pratiques

```r
summary(mod10)
```

```
 Family: binomial 
  Links: mu = logit 
Formula: presence | trials(total) ~ 1 + reminder + inscription 
   Data: df3 (Number of observations: 13) 
  Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
         total post-warmup draws = 4000

Regression Coefficients:
            Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
Intercept       1.01      0.60    -0.16     2.20 1.00     1911     2416
reminder        0.93      0.50    -0.01     1.93 1.00     2017     2273
inscription    -0.34      0.57    -1.44     0.77 1.00     2090     2512

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).
```

Travaux pratiques

Le mail de rappel semble avoir moins d'effet dans le modèle complet que dans le modèle simple… Pourquoi ?

```r
fixef(mod8) %>% exp() # calcul du "odds ratio" (i.e., exp(beta))
```

```
          Estimate Est.Error     Q2.5    Q97.5
Intercept 1.975957  1.280300 1.219978 3.285039
reminder  2.969455  1.471665 1.403602 6.522752
```

```r
fixef(mod9) %>% exp()
```

```
             Estimate Est.Error      Q2.5      Q97.5
Intercept   6.3712460  1.466393 3.0870809 13.6501657
inscription 0.3764592  1.539638 0.1597557  0.8535957
```

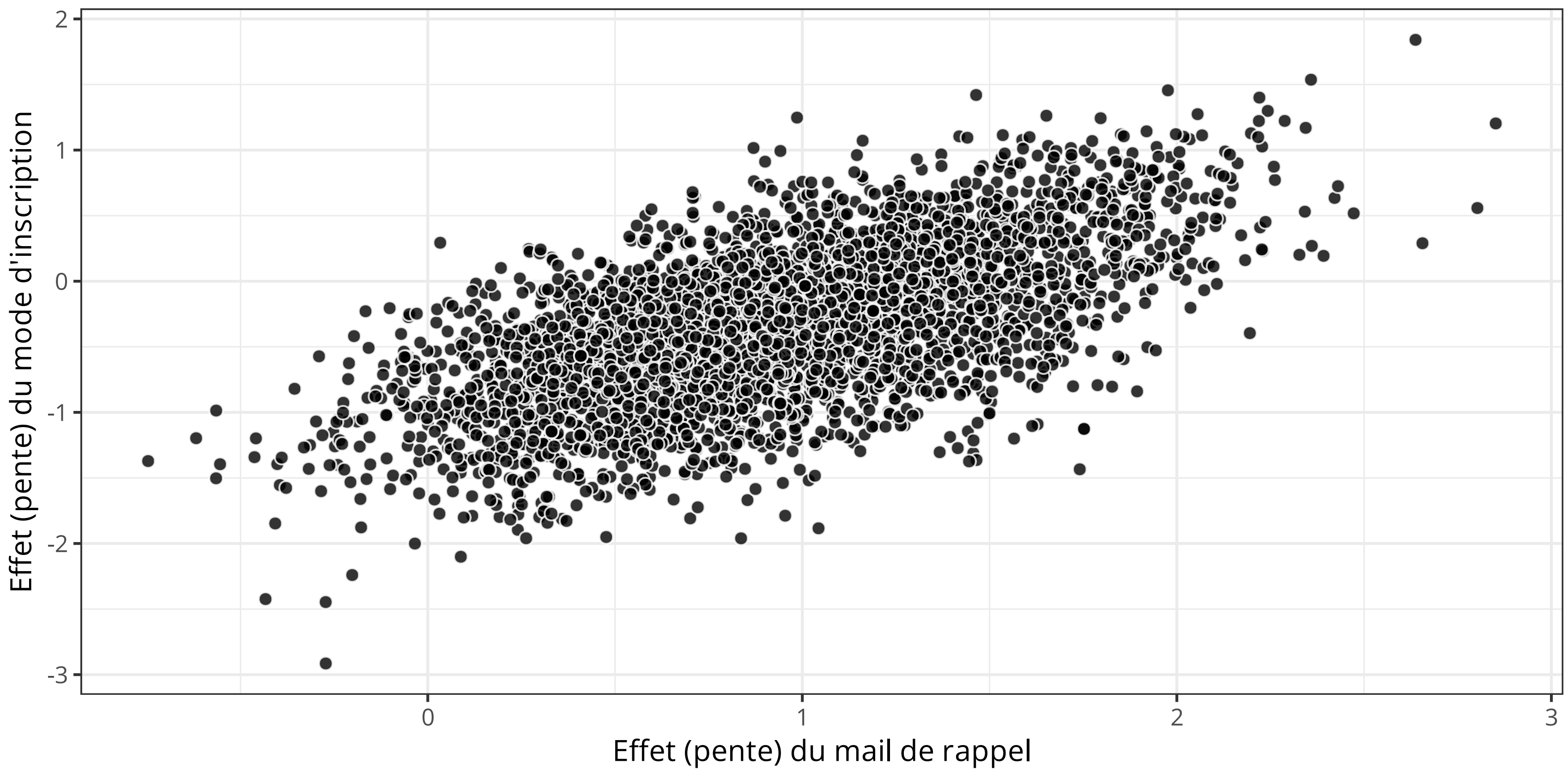
```r
fixef(mod10) %>% exp()
```

```
             Estimate Est.Error      Q2.5    Q97.5
Intercept   2.7486413  1.818883 0.8528041 9.012623
reminder    2.5389822  1.650270 0.9903892 6.871404
inscription 0.7089088  1.766321 0.2373591 2.165044
```

Travaux pratiques

On a déjà rencontré ce cas de figure (cf. Cours n°04). Lorsque deux prédicteurs contiennent une part d'information commune, l'estimation des pentes est corrélée...

```
as_draws_df(x = mod10) %>%
    ggplot(aes(b_reminder, b_inscription) ) +
    geom_point(size = 3, pch = 21, alpha = 0.8, color = "white", fill = "black") +
    labs(x = "Effet (pente) du mail de rappel", y = "Effet (pente) du mode d'inscription")
```



Travaux pratiques

En effet, les données ont été collectées par deux expérimentateurs. L'un d'entre eux a recruté tous ses participants via Doodle, et n'envoyait pas souvent de mail de rappel. Le deuxième expérimentateur a recruté tous ses participants via un panneau physique présent dans le laboratoire et envoyait systématiquement un mail de rappel. Autrement dit, ces deux variables sont presque parfaitement confondues.

```r
open_data(absence) %>%
  group_by(inscription, reminder) %>%
  summarise(n = sum(total) ) %>%
  spread(key = reminder, value = n) %>%
  data.frame()
```

```
  inscription no yes
1      doodle 72  22
2       panel NA  51
```