

Impact des mammites sur la reproduction et la dynamique des infections intra-mammaires lors de la lactation suivante

Table of contents

1	Données	3
1.1	Chargement des données contrôle laitier Idele	3
1.2	Construction d'un jeu de données lactations	3
1.3	Chargement des données mammites cliniques	5
1.4	Données d'insémination artificielle	7
1.5	Fusion des données d'insémination artificielle avec données lactation	7
1.6	Agrégation des données contrôle laitier par date de contrôle	8
1.7	Agrégation des données à l'échelle élevage-année	8
1.8	Observations conservées pour l'analyse	11
2	Covariables	12
2.1	Parité	12
2.2	Production laitière en début de lactation	12
2.3	Définition des statuts vis-à-vis des mammites	13
3	Jeux de données utilisés pour les prédictions	14
3.1	Catégories mammites	15
3.2	Parité	15
3.3	Maximum de production laitière avant 90 jours et parité	16
4	Modèle mort de la vache	16
5	Modèles mises à la reproduction	20
5.1	Variable d'intérêt	20
5.2	Modèle de référence	20
5.3	Production laitière et parité	21

5.4 Statuts mammites	24
6 Probabilité de revêlage pour les vaches mises à la reproduction	30
7 Probabilité de réussite de la première IA	40
8 Nombre d'inséminations artificielles dans une lactation	46
9 Dynamique des infections intra-mammaire lors de la période tarie suivant la mise à la reproduction	50
9.1 Probabilité d'une CCS élevée au dernier contrôle avant tarissement	50
9.2 Probabilité de guérison	54
9.3 Probabilité de nouvelle infection	57

```
library(tidyverse)
library(lme4)
library(marginaleffects)

## fonction pour passer des coefficients d'une
# régression logistique à une probabilité
invlogit <- function(x) exp(x) / (1 + exp(x))
```

Ce fichier reprend le code qui était auparavant dans le fichier “longevity__mastitis.qmd” pour plus de lisibilité.

Les objectifs sont de modéliser l’impact de différents statuts vis-à-vis des mammites sur la probabilité de réforme. La probabilité de réforme est représentée en différentes composantes :

- Mort de la vache
- Mise à la reproduction : présence d’une première insémination artificielle
- Succès de la mise à la reproduction : re-vêlage
- Succès de la première insémination artificielle
- Nombre d’inséminations artificielles réalisées
- Mammites : nouvelles infections et guérisons pendant la période tarie

L’unité statistique est la lactation.

Les étapes du fichier :

- chargement des données contrôle laitier Idele
- construction d’un jeu de données lactations à partir des données contrôle laitier
- chargement des données mammites cliniques
- chargement des données d’insémination artificielle
- les différents types d’analyses listés sont réalisés tour à tour

Le bloc suivant définit les paramètres de la modélisation.

```
## nombre de jours en dessous duquel on considère qu'on est en début de lactation
## en moyenne avant première IA
early_dim_cutoff <- 90
## SCC cut-off
## seuil de cellules au-delà duquel on considère
## qu'il y a mammite clinique
scc_high_cutoff <- 200
```

1 Données

1.1 Chargement des données contrôle laitier Idele

Un jeu de données construit à partir des données fournies par l’Idele est chargé. Les lignes pour lesquelles soit la production laitière soit les cellules sont manquantes sont supprimées.

```
rec <- read_csv2("generated_datasets/idele_rec_final.csv") |>
  mutate(
    herd_id = as.character(herd_id),
    ctrl_date = as.Date(ctrl_date),
    anim_id = as.character(anim_id),
    parity = as.integer(parity),
    fat = as.integer(fat),
    prot = as.integer(prot),
    scc = as.integer(scc),
    out_cause = as.character(out_cause),
    calv_date = as.Date(calv_date),
    dim = as.integer(dim)
  ) |>
  filter(!is.na(milk) & !is.na(scc))
```

1.2 Construction d'un jeu de données lactations

Le jeu de données lactation est construit à partir des données contrôle laitier. On y rassemble des données qui concernent également la lactation précédente et la lactation suivante.

```

lac <- rec |>
  group_by(anim_id, parity) |>
  summarise(
    herd_id = herd_id[1],
    calv_date = unique(calv_date),
    out_date_last = max(out_date),
    out_cause_last = unique(out_cause[out_date == out_date_last]),
    n_ctrl = length(unique(ctrl_date)),
    n_ctrl_early = length(unique(ctrl_date[dim <= early_dim_cutoff])),
    n_scc_high_early = length(scc[scc > scc_high_cutoff & dim <= early_dim_cutoff]),
    milk_early_max = max(milk[dim <= early_dim_cutoff]),
    dim_first = min(dim),
    scc_first = scc[dim == dim_first],
    dim_last = max(dim),
    milk_last = milk[dim == dim_last],
    scc_last = scc[dim == dim_last]
  ) |>
  ungroup() |>
  group_by(anim_id) |>
  mutate(
    calv_date_prev = lag(calv_date),
    milk_last_prev_lac = lag(milk_last),
    dim_last_prev_lac = lag(dim_last),
    scc_last_prev_lac = lag(scc_last),
    calv_date_nxt = lead(calv_date),
    scc_first_nxt_lac = lead(scc_first),
    infct_end_lac = case_when(
      is.na(scc_last) | is.na(scc_first_nxt_lac) ~ NA,
      !is.na(scc_last) & !is.na(scc_first_nxt_lac) & scc_last < scc_high_cutoff ~ "0",
      !is.na(scc_last) & !is.na(scc_first_nxt_lac) & scc_last >= scc_high_cutoff ~ "1"
    ),
    infct_ctrl1_nxt_lac = case_when(
      is.na(scc_last) | is.na(scc_first_nxt_lac) ~ NA,
      !is.na(scc_last) & !is.na(scc_first_nxt_lac) & scc_first_nxt_lac < scc_high_cutoff ~ "0",
      !is.na(scc_last) & !is.na(scc_first_nxt_lac) & scc_first_nxt_lac >= scc_high_cutoff ~ "1"
    ),
    nxt_cure = case_when(
      infct_end_lac == 0 ~ NA,
      infct_end_lac == 1 & infct_ctrl1_nxt_lac == 1 ~ 0,
      infct_end_lac == 1 & infct_ctrl1_nxt_lac == 0 ~ 1
    ),
    nxt_ninf = case_when(

```

```

infct_end_lac == 1 ~ NA,
infct_end_lac == 0 & infct_ctrl1_nxt_lac == 0 ~ 0,
infct_end_lac == 0 & infct_ctrl1_nxt_lac == 1 ~ 1
)
) |>
ungroup()

```

Warning: There were 21319 warnings in `summarise()`.
 The first warning was:
 i In argument: `milk_early_max = max(milk[dim <= early_dim_cutoff])`.
 i In group 1: `anim_id = "AT445953874"` and `parity = 1`.
 Caused by warning in `max()`:
 ! aucun argument pour max ; -Inf est renvoyé
 i Run `dplyr::last_dplyr_warnings()` to see the 21318 remaining warnings.

`summarise()` has grouped output by 'anim_id'. You can override using the
 `.groups` argument.

Les causes de sorties possibles sont les suivantes :

- B : boucherie
- C : autoconsommation
- E : élevage
- H : prêt ou pension
- M : mort

Le tableau suivant présente les effectifs pour les différentes causes de sortie.

```
table(lac$out_cause_last)
```

	B	C	E	H	M
	137267	402	27910	816	21057

1.3 Chargement des données mammites cliniques

Le fichier de données est chargé. On ne garde que les dates de mammites cliniques. Il existe également des données sur la sévérité des mammites qui ne sont pas conservées à ce stade.

```
path_to_file <- "data/mammities.txt"
mamt <- read_csv2(path_to_file) |>
  rename(anim_id = ANIM,
         cm_date = DAMACL) |>
  select(anim_id, cm_date)
```

Les données mammities cliniques sont fusionnées avec les données lactation. Pour chaque lactation, on extrait les première et dernière dates de mammite clinique. On considère qu'une lactation commence 30 jours avant vêlage. Cette hypothèse pourra être revue.

```
lac_cm <- left_join(
  lac |>
  select(anim_id, calv_date, calv_date_nxt) |>
  mutate(
    dim_minus30 = calv_date - 30,
    dim_early = calv_date + early_dim_cutoff),
  mamt) |>
  filter(!is.na(cm_date)) |>
  group_by(anim_id, calv_date) |>
  summarise(
    cm_first = min(cm_date[cm_date >= calv_date & (cm_date < calv_date_nxt | is.na(calv_date_nxt))],
    cm_last = max(cm_date[cm_date >= calv_date & (cm_date < calv_date_nxt | is.na(calv_date_nxt))],
    n_cm_early = length(unique(cm_date[cm_date >= dim_minus30 & cm_date < dim_early])),
    n_cm_late = length(unique(cm_date[cm_date >= dim_early & (cm_date < calv_date_nxt | is.na(calv_date_nxt))])
  ) |>
  ungroup()
```

Joining with `by = join_by(anim_id)`

```
Warning in left_join(mutate(select(lac, anim_id, calv_date, calv_date_nxt), : Detected an unresolvable join
i Row 24 of `x` matches multiple rows in `y`.
i Row 1 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship = "many-to-many"` to silence this warning.
```

```
Warning: There were 85866 warnings in `summarise()`.
The first warning was:
i In argument: `cm_first = min(...)`.

```

``summarise()`` has grouped output by 'anim_id'. You can override using the ``.groups`` argument.

```
lac_cm$cm_first[is.infinite(lac_cm$cm_first)] <- NA
lac_cm$cm_last[is.infinite(lac_cm$cm_last)] <- NA
```

Fusion avec le jeu de données lac initial.

Joining with ``by = join_by(anim_id, calv_date)``

1.4 Données d'insémination artificielle

Les données d'insémination sont chargées. On ne garde que les identifiants animaux et les dates d'insémination.

```
path_to_file <- "data/extra_femia_20240415.csv"
ai <- read_csv(path_to_file) |>
  select(anim, DAINAR) |>
  rename(anim_id = anim,
         ai_date = DAINAR) |>
  mutate(ai_date = as.Date(ai_date, "%d/%m/%Y"))
```

1.5 Fusion des données d'insémination artificielle avec données lactation

Les données lactations et inséminations artificielles sont fusionnées. On s'assure que chaque date d'IA survient entre un vêlage et le suivant ou entre un vêlage et une absence de vêlage suivant.

```
nlac <- left_join(lac |>
  select(anim_id, calv_date, calv_date_nxt),
  ai) |>
  filter(ai_date > calv_date & (ai_date < calv_date_nxt | is.na(calv_date_nxt))) |>
  group_by(anim_id, calv_date) |>
  summarise(n_ai = length(unique(ai_date)),
           ai_first = min(ai_date),
           ai_last = max(ai_date)) |>
  ungroup()
```

Joining with ``by = join_by(anim_id)``

```
Warning in left_join(select(lac, anim_id, calv_date, calv_date_nxt), ai): Detected an unexpected
i Row 2 of `x` matches multiple rows in `y`.
i Row 1 of `y` matches multiple rows in `x`.
i If a many-to-many relationship is expected, set `relationship =
  "many-to-many"` to silence this warning.
```

```
`summarise()` has grouped output by 'anim_id'. You can override using the
`.groups` argument.
```

```
lac <- left_join(lac, nlac) |>
  mutate(ai1_dim = as.integer(ai_first - calv_date),
         ai_last_dim = as.integer(ai_last - calv_date))
```

```
Joining with `by = join_by(anim_id, calv_date)`
```

1.6 Agrégation des données contrôle laitier par date de contrôle

Les données contrôle laitier sont agrégées par date de contrôle-élevage. Il y a déjà eu une sélection en amont sur le nombre de vaches par contrôle et le nombre de contrôles par élevage par an. Ce jeu de données n'est pas vraiment utilisé pour le moment.

```
td <- rec |>
  group_by(herd_id, ctrl_date) |>
  summarise(
    n_cows = length(unique(anim_id)),
    milk_mean = round(mean(milk), 1),
    scc_geo_mean = round(exp(mean(log(scc))), 1),
    scc_high = round(length(scc[scc > scc_high_cutoff]) / length(scc), 3),
    bmscc = round(sum(scc * milk) / sum(milk), 1)) |>
  ungroup()
```

```
`summarise()` has grouped output by 'herd_id'. You can override using the
`.groups` argument.
```

1.7 Agrégation des données à l'échelle élevage-année

Les données contrôle laitier et inséminations artificielles sont agrégées à l'échelle élevage-année. L'objectif est de sélectionner les élevages dans lesquels une proportion suffisante des vaches sont inséminées.


```
rec_hdyr <- rec |>
  mutate(year = as.integer(format(ctrl_date, "%Y"))) |>
  group_by(herd_id, year) |>
  summarise(n_ctrl = length(unique(ctrl_date)),
            n_cows_rec = length(unique(anim_id)),
            p_primi_rec = round(length(parity[parity == 1]) / length(parity), 3)) |>
  ungroup()
```

`summarise()` has grouped output by 'herd_id'. You can override using the
`.groups` argument.

A partir des données lactation, les premières IA de chaque élevage sont comptées.

```
ai_hdyr <- lac |>
  filter(!is.na(n_ai)) |>
  mutate(year = as.integer(format(ai_first, "%Y"))) |>
  group_by(herd_id, year) |>
  summarise(n_cows_ai = length(n_ai[!is.na(n_ai)])) |>
  ungroup()
```

`summarise()` has grouped output by 'herd_id'. You can override using the
`.groups` argument.

Fusion des 2 jeux de données. Seuls les élevages qui ont à la fois des données contrôles laitier et des données d'IA sont conservés.

```
hdyr <- inner_join(rec_hdyr, ai_hdyr) |>
  mutate(p_ai = round(n_cows_ai / n_cows_rec, 3))
```

Joining with `by = join_by(herd_id, year)`

On regarde également, sur les vaches vélées, celles qui ont une date d'IA ultérieure. Les différents jeux de données sont fusionnés. On examine les proportions de vaches inséminées parmi les vaches présentes et parmi les vaches qui vèlent. Ces 2 variables sont fortement corrélées. Pour sélectionner les élevages à garder par la suite, je choisis de combiner la proportion de vaches inséminées parmi les vaches contrôlées parce que cette variables est plus facile à calculer et à comprendre. Une sélection est également appliquée sur la proportion de primipares contrôlées.

```

lac <- lac |>
  mutate(year = as.integer(format(calv_date, "%Y")))

lac_hrd_yr <- lac |>
  group_by(herd_id, year) |>
  summarise(
    n_calvings = length(anim_id),
    n_calvings_p1 = length(parity[parity == 1]),
    p_calvings_p1 = round(n_calvings_p1 / n_calvings, 3),
    n_ai1 = length(n_ai[!is.na(n_ai)]),
    p_ai_clv = round(n_ai1 / n_calvings, 3)
  )

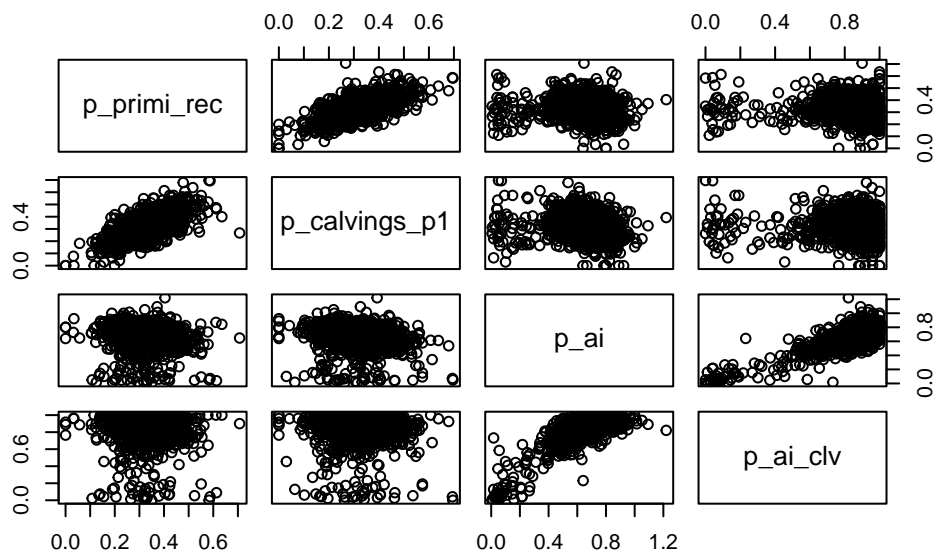
```

`summarise()` has grouped output by 'herd_id'. You can override using the `.groups` argument.

```
hdyr <- left_join(hdyr, lac_hrd_yr)
```

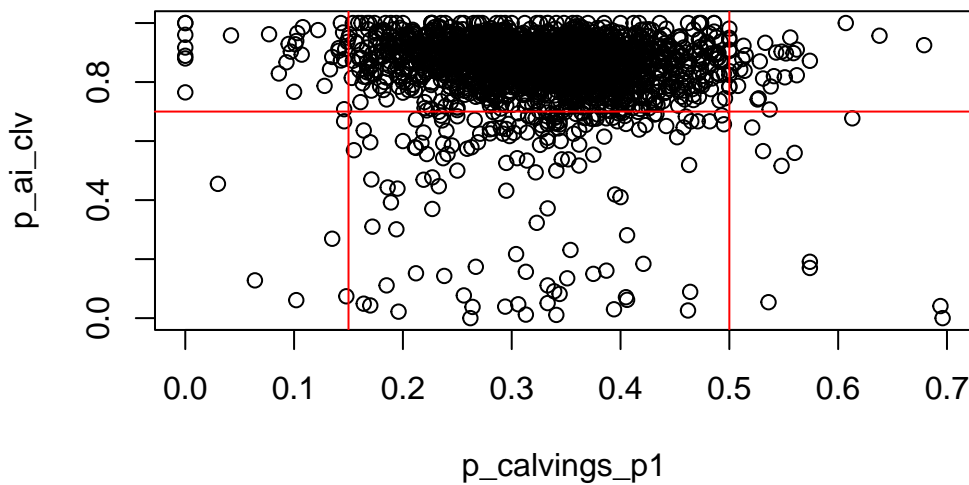
Joining with `by = join_by(herd_id, year)`

```
pairs(hdyr[, c("p_primi_rec", "p_calvings_p1", "p_ai", "p_ai_clv")])
```



La figure suivante est utilisée pour identifier les valeurs de paramètres à utiliser pour la sélection. En abscisse, on a la proportion de vêlages de génisses et en ordonnées le nombre de premières inséminations ramené au nombre de vêlages. Les lignes rouges matérialisent les valeurs utilisées pour sélectionner les élevages-années retenus.

```
plot(p_ai_clv ~ p_calvings_p1, data = hdyr)
abline(v = c(.15, .5), col = "red")
abline(h = .7, col = "red")
```



```
hdyr_sel <- hdyr |>
  filter(p_ai_clv > .7 & p_calvings_p1 > .15 & p_calvings_p1 < .5) |>
  select(herd_id, year)

lac <- left_join(hdyr_sel, lac)
```

Joining with `by = join_by(herd_id, year)`

1.8 Observations conservées pour l'analyse

On ne conserve que les lactations avec au moins 1 contrôle laitier avant 90 jours de lactation.

```
lac <- lac |>
  filter(n_ctrl_early > 0)
```

2 Covariables

2.1 Parité

Une nouvelle variable parité appelée `parKat` est créée. Les parités supérieures à 3 sont mises dans une catégorie 4. La parité 2 est mise en catégorie de référence pour faciliter la lecture des résultats.

```
lac <- lac |>
  mutate(parKat = relevel(as.factor(ifelse(parity > 3, 4, parity)), ref = "2"))
```

2.2 Production laitière en début de lactation

Le maximum de production laitière sur les contrôles laitiers de 90 premiers jours de lactation sont standardisés : pour chaque valeur, on retranche la moyenne et on divise par l'écart-type de manière à avoir une moyenne de 0 et un écart-type de 1 sur la nouvelle variable centrée-réduite.

Une variable différente est créée pour les primipares et les multipares, de manière à pouvoir prendre en compte un effet différent de la production laitière sur les variables d'intérêt pour ces catégories de vaches.

```
milk_early_max_mean <- with(lac[lac$n_ctrl_early > 0,],
  mean(milk_early_max))

milk_early_max_sd <- with(lac[lac$n_ctrl_early > 0,],
  sd(milk_early_max))

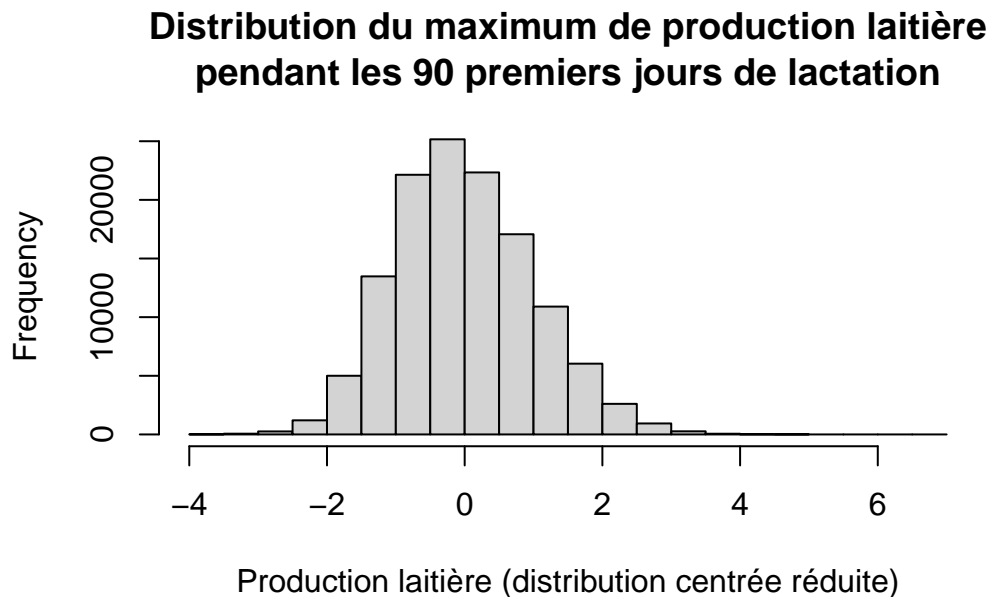
lac <- lac |>
  filter(n_ctrl_early > 0) |>
  mutate(parKat = relevel(as.factor(ifelse(parity > 3, 4, parity)), ref = "2"),
    milk_early_max_scaled = (milk_early_max - milk_early_max_mean) /
      milk_early_max_sd,
    milk_early_max_scaled_p1 = ifelse(parity == 1, milk_early_max_scaled, 0),
    milk_early_max_scaled_p2 = ifelse(parity == 2, milk_early_max_scaled, 0),
    milk_early_max_scaled_p3 = ifelse(parity == 3, milk_early_max_scaled, 0),
    milk_early_max_scaled_p4 = ifelse(parity > 3, milk_early_max_scaled, 0),
```

```

milk_early_max_scaled_p1plus = ifelse(parity > 1, milk_early_max_scaled, 0)
)

hist(lac$milk_early_max_scaled,
     main = "Distribution du maximum de production laitière\ndurant les 90 premiers jours d",
     xlab = "Production laitière (distribution centrée réduite)")

```



Les moyennes et écart-types de la variable originale sont 32.3 et 8 respectivement.

2.3 Définition des statuts vis-à-vis des mammites

Ici on considère :

- pour les primipares, les données cellules et mammites cliniques du début de lactation, c'est à dire jusqu'à 90 jours post-vêlage
- pour les multipares, les données cellules du dernier contrôle de la lactation précédente les données cellules et mammites cliniques du début de lactation.

Une variable `dp_status` (dp pour *dry period*) est créée. Les statuts sont :

- **primi** : primipares, pas de période tarie
- **dpLow** : au-dessous de seuil de CCS choisi pour définir l'infection au dernier contrôle avant tarissement

- **dpHigh** : au-dessus de seuil de CCS choisi pour définir l'infection au dernier contrôle avant tarissement

Les primipares sont prises comme référence.

Une variable **mast_cat** est créée. Cette variable définit un statut vis-à-vis des mammites en début de lactation. L'objectif est d'avoir des catégories pour vache saine, mammite à réservoir mammaire et mammite à réservoir environnemental. Les vache saines sont prises comme référence.

```
lac <- lac |>
  mutate(
    dp_status = case_when(
      parity == 1 ~ "primi",
      scc_last_prev_lac < scc_high_cutoff ~ "dpLow",
      scc_last_prev_lac >= scc_high_cutoff ~ "dpHigh"
    ),
    dp_status = relevel(factor(dp_status), ref = "primi"),
    mast_cat = case_when(
      n_scc_high_early == 0 & n_cm_early == 0 ~ "healthy",
      n_scc_high_early == 1 | (n_scc_high_early == 0 & n_cm_early > 0) ~ "env",
      n_scc_high_early > 1 ~ "cont"),
    mast_cat = relevel(factor(mast_cat), ref = "healthy"),
    mast_cat1 = relevel(factor(paste(dp_status, mast_cat, sep = "_")), ref = "primi_healthy"),
    mast_cat2 = relevel(factor(paste(ifelse(dp_status == "primi", "primi", "multi"), mast_cat, sep = "_")), ref = "primi_healthy")
  )
```

Certaines vaches multipares n'ont pas de CCS pour leur dernière lactation. Elles sont identifiées et exclues de l'analyse.

```
lac <- lac |>
  mutate(dp_mis = ifelse(parity > 1 & is.na(scc_last_prev_lac), 1, 0))
```

3 Jeux de données utilisés pour les prédictions

L'interprétation des résultats des modèles se fait en prédisant les probabilités des différents événements à partir des sorties de modèles. A chaque fois, on fixe les valeurs de toutes les variables à un niveau fixe, à l'exception de celle qui nous intéresse qu'on fait varier.

Les raisons de ces choix s'expliquent plus bas, dans les étapes de construction des modèles. Le code est placé ici parce qu'il est mobilisé de nombreuses fois par la suite.

Les prédictions seront réalisées avec la fonction **prediction** du package **marginaleffects**.

3.1 Catégories mammites

Les différents catégories de mammites selon la parité sont incluses. On fixe la production laitière à la moyenne de la population (32.3). La parité est fixée à 2 pour les multipares.

```
## catégories de statuts mammites pour les primipares
mast_lev_p1 <- lac |>
  filter(n_ctrl_early > 0 & dp_mis == 0 & parKat == 1) |>
  distinct(mast_cat1) |>
  mutate(mast_cat1 = as.character(mast_cat1)) |>
  pull(mast_cat1)
## catégories de statuts mammites pour les multipares
mast_lev_p2 <- lac |>
  filter(n_ctrl_early > 0 & dp_mis == 0 & parKat != 1) |>
  distinct(mast_cat1) |>
  mutate(mast_cat1 = as.character(mast_cat1)) |>
  pull(mast_cat1)

## jeu de données
mast_pred <- tibble(
  parKat = factor(c(rep(1, length(mast_lev_p1)),
                    rep(2, length(mast_lev_p2)))),
  mast_cat1 = factor(c(mast_lev_p1, mast_lev_p2)),
  milk_early_max_scaled_p1 = 0,
  milk_early_max_scaled_p2 = 0,
  milk_early_max_scaled_p3 = 0,
  milk_early_max_scaled_p4 = 0
)

level_order <- c("primi_healthy", "primi_env", "primi_cont",
                 "dpLow_healthy", "dpLow_env", "dpLow_cont",
                 "dpHigh_healthy", "dpHigh_env", "dpHigh_cont")
```

3.2 Parité

Pour les parités, on utilise les catégories incluses dans les modèles : 1, 2, 3 ou supérieure à 3.

```
parit_pred <- expand_grid(
  parKat = factor(1:4),
  milk_early_max = 0
```

```

) |>
  mutate(
    milk_early_max_scaled_p1 = 0,
    milk_early_max_scaled_p2 = 0,
    milk_early_max_scaled_p3 = 0,
    milk_early_max_scaled_p4 = 0,
    mast_cat1 = factor(ifelse(parKat == 1, "primi_healthy", "dpLow_healthy"))
  )

```

3.3 Maximum de production laitière avant 90 jours et parité

On utilise les mêmes catégories de parités que ci-dessus. Les productions laitières sont variées de 10 à 50kg et standardisées pour les faire correspondre aux variables production laitière du modèle. Les catégories de référence pour les mammites sont primipare saine et multipare saine.

```

milk_pred <- expand_grid(
  parKat = factor(1:4),
  milk_early_max = seq(10, 50, by = .1)
) |>
  mutate(
    milk_early_max_scaled = (milk_early_max - milk_early_max_mean) /
      milk_early_max_sd,
    milk_early_max_scaled_p1 = ifelse(parKat == 1, milk_early_max_scaled, 0),
    milk_early_max_scaled_p2 = ifelse(parKat == 2, milk_early_max_scaled, 0),
    milk_early_max_scaled_p3 = ifelse(parKat == 3, milk_early_max_scaled, 0),
    milk_early_max_scaled_p4 = ifelse(parKat == 4, milk_early_max_scaled, 0),
    mast_cat1 = factor(ifelse(parKat == 1, "primi_healthy", "dpLow_healthy"))
  )

```

4 Modèle mort de la vache

L'évènement mort est défini d'après la dernière cause de sortie enregistrée, s'il n'y a pas de date de vêlage suivant. Le premier modèle contient uniquement un intercept. La probabilité de mort dans une lactation est d'environ 4%.

```

lac <- lac |>
  mutate(death = ifelse(out_cause_last == "M" & is.na(calv_date_nxt), 1, 0))

```



```
death_null <- glmer(death ~ 1 + (1 | herd_id),
  data = lac,
  subset = n_ctrl_early > 0 & dp_mis == 0,
  family = binomial(link = "logit"))

summary(death_null)
```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]

Family: binomial (logit)

Formula: death ~ 1 + (1 | herd_id)

Data: lac

Subset: n_ctrl_early > 0 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
41520.4	41539.7	-20758.2	41516.4	115195

Scaled residuals:

Min	1Q	Median	3Q	Max
-0.5198	-0.2451	-0.1963	-0.1572	9.4466

Random effects:

Groups	Name	Variance	Std.Dev.
--------	------	----------	----------

herd_id	(Intercept)	0.4785	0.6917
---------	-------------	--------	--------

Number of obs: 115197, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-3.16303	0.05075	-62.32	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

On teste un modèle complet avec toutes les variables incluses dans les modèles suivants.

```
death_full <- glmer(death ~ parkat +
  milk_early_max_scaled_p1 +
  milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 +
  milk_early_max_scaled_p4 +
  mast_cat1 +
  (1 | herd_id),
```

```
data = lac,
subset = n_ctrl_early > 0 & dp_mis == 0,
family = binomial(link = "logit"))
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(death_full)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula:
death ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
      milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +
      (1 | herd_id)
Data: lac
Subset: n_ctrl_early > 0 & dp_mis == 0
```

AIC	BIC	logLik	deviance	df.resid
41263.9	41418.3	-20615.9	41231.9	115181

Scaled residuals:

Min	1Q	Median	3Q	Max
-0.6712	-0.2438	-0.1963	-0.1544	10.6435

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.4954	0.7038

Number of obs: 115197, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-3.31213	0.06072	-54.552	< 2e-16 ***
parKat1	-0.12749	0.05399	-2.361	0.018209 *
parKat3	0.14456	0.05102	2.833	0.004605 **
parKat4	0.50105	0.04465	11.221	< 2e-16 ***
milk_early_max_scaled_p1	-0.08883	0.03911	-2.271	0.023135 *
milk_early_max_scaled_p2	0.12270	0.03279	3.742	0.000183 ***
milk_early_max_scaled_p3	0.11190	0.03468	3.226	0.001254 **
milk_early_max_scaled_p4	-0.04217	0.02717	-1.552	0.120621
mast_cat1dpHigh_cont	-0.26308	0.06812	-3.862	0.000112 ***

```

mast_cat1dpHigh_env      0.05093      0.05955      0.855 0.392434
mast_cat1dpHigh_healthy  0.02599      0.04385      0.593 0.553390
mast_cat1dpLow_cont      -0.19344      0.08136     -2.378 0.017421 *
mast_cat1dpLow_env       0.04100      0.05824      0.704 0.481457
mast_cat1primi_cont      -0.16256      0.09718     -1.673 0.094387 .
mast_cat1primi_env       0.05277      0.06683      0.790 0.429697
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or
`vcov(x)` if you need it

fit warnings:

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

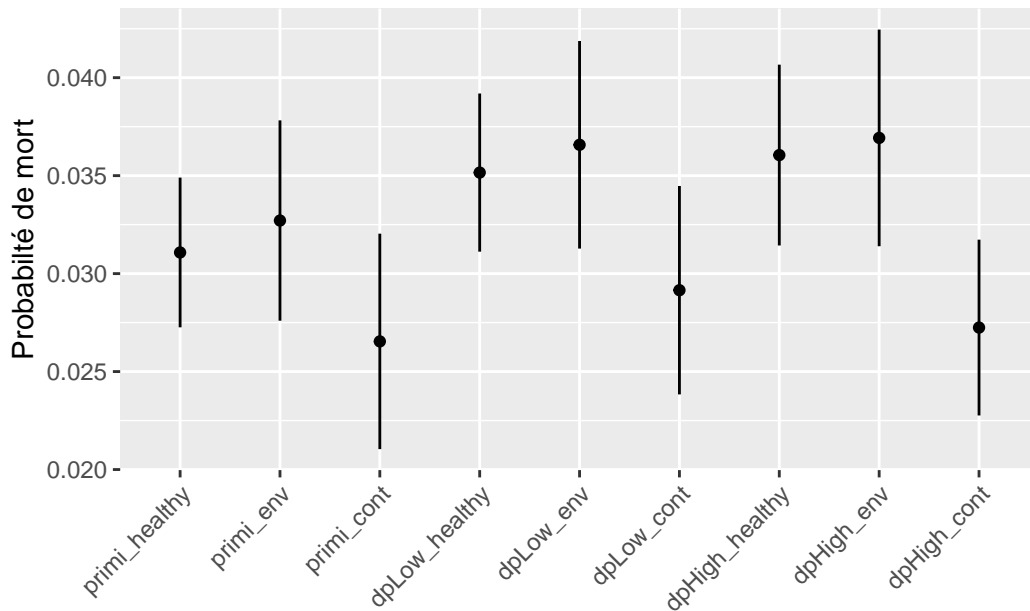
Les résultats de ce modèle peuvent sembler surprenant. Les vaches avec une mammite chronique semblent moins susceptibles de mourir, comme si ce type d'infection prévenait la mortalité. La raison de cette association est probablement à rechercher dans la définition des mammites contagieuses (chroniques). Pour avoir eu plus d'une CCS supérieure à 200 000 pendant les 90 premiers jours de lactation, il faut avoir eu au moins 2 contrôles. Il est probable que beaucoup de mortalités surviennent avant qu'un 2ème contrôle ait été effectué.

```

prd <- marginaleseffects::predictions(model = death_full,
                                     newdata = mast_pred,
                                     re.form = NA,
                                     type = "response")

ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité de mort") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))

```



5 Modèles mises à la reproduction

5.1 Variable d'intérêt

On utilise des modèles logistiques mixtes, avec la fonction `glmer` du package `lme4`. Dans tous les modèles on met un effet aléatoire élevage qui représente la probabilité pour une vache d'être mise à la reproduction dans son élevage.

```
lac <- lac |>
  mutate(
    repro = ifelse(!is.na(n_ai) | !is.na(calv_date_nxt), 1, 0))
```

5.2 Modèle de référence

Modèle avec intercept uniquement et effet aléatoire troupeau.

```
repro_null <- glmer(repro ~ 1 + (1 | herd_id),
  data = lac,
  subset = n_ctrl_early > 0 & dp_mis == 0,
  family = binomial(link = "logit"))
```

```
summary(repro_null)
```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]

Family: binomial (logit)

Formula: repro ~ 1 + (1 | herd_id)

Data: lac

Subset: n_ctrl_early > 0 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
92394.7	92414.2	-46195.3	92390.7	125738

Scaled residuals:

Min	1Q	Median	3Q	Max
-4.6767	0.2902	0.3600	0.4038	0.5573

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.1733	0.4163

Number of obs: 125740, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	2.0484	0.0305	67.16	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

On obtient une probabilité de mise à la reproduction de 0.886

5.3 Production laitière et parité

```
repro_parity <- glmer(repro ~ parKat + (1 | herd_id),  
                      data = lac,  
                      subset = n_ctrl_early > 0 & dp_mis == 0,  
                      family = binomial(link = "logit"))  
  
summary(repro_parity)
```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: repro ~ parKat + (1 | herd_id)
Data: lac
Subset: n_ctrl_early > 0 & dp_mis == 0

```

AIC	BIC	logLik	deviance	df.resid
87796.5	87845.2	-43893.2	87786.5	125735

```

Scaled residuals:
    Min       1Q   Median       3Q      Max
-7.5725  0.2335  0.3042  0.3923  0.9354

```

```

Random effects:
 Groups Name      Variance Std.Dev.
herd_id (Intercept) 0.272    0.5216
Number of obs: 125740, groups: herd_id, 231

```

```

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.44944    0.04136   59.22  <2e-16 ***
parKat1      0.30147    0.02755   10.94  <2e-16 ***
parKat3     -0.42885    0.02827  -15.17  <2e-16 ***
parKat4     -1.18731    0.02463  -48.20  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Correlation of Fixed Effects:
      (Intr) parKt1 parKt3
parKat1 -0.342
parKat3 -0.336  0.500
parKat4 -0.402  0.573  0.567

```

Ce modèle plus complet inclut les catégories de parité et les productions laitières de début de lactation standardisées pour les parités 1, 2, 3 et supérieure à 3. L'hypothèse est qu'il y a un effet additionnel de la parité qui va au-delà de la production laitière.

```

repro_parity_milk <- glmer(repro ~ parKat +
                           milk_early_max_scaled_p1 +
                           milk_early_max_scaled_p2 +
                           milk_early_max_scaled_p3 +

```

```

        milk_early_max_scaled_p4 +
        (1 | herd_id),
data = lac,
subset = n_ctrl_early > 0 & dp_mis == 0,
family = binomial(link = "logit"))

summary(repro_parity_milk)

```

Generalized linear mixed model fit by maximum likelihood (Laplace

Approximation) [glmerMod]

Family: binomial (logit)

Formula:

```

repro ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
        milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + (1 | herd_id)

```

Data: lac

Subset: n_ctrl_early > 0 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
83756.0	83843.7	-41869.0	83738.0	125731

Scaled residuals:

Min	1Q	Median	3Q	Max
-45.656	0.181	0.283	0.401	2.220

Random effects:

Groups	Name	Variance	Std.Dev.
--------	------	----------	----------

herd_id	(Intercept)	0.5394	0.7344
---------	-------------	--------	--------

Number of obs: 125740, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	2.52711	0.05422	46.61	<2e-16 ***
parKat1	1.57060	0.04400	35.69	<2e-16 ***
parKat3	-0.63210	0.02958	-21.37	<2e-16 ***
parKat4	-1.43760	0.02598	-55.33	<2e-16 ***
milk_early_max_scaled_p1	1.42183	0.03178	44.74	<2e-16 ***
milk_early_max_scaled_p2	0.69348	0.02328	29.79	<2e-16 ***
milk_early_max_scaled_p3	0.57282	0.02273	25.20	<2e-16 ***
milk_early_max_scaled_p4	0.56250	0.01645	34.20	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

	(Intr)	parKt1	parKt3	parKt4	m____1	m____2	m____3
parKat1	-0.159						
parKat3	-0.262	0.298					
parKat4	-0.315	0.328	0.553				
mlk_rly___1	0.018	0.764	-0.024	-0.044			
mlk_rly___2	0.061	0.002	-0.099	-0.132	0.080		
mlk_rly___3	0.021	0.047	-0.229	-0.053	0.075	0.087	
mlk_rly___4	0.036	0.059	-0.035	-0.239	0.095	0.109	0.111

5.4 Statuts mammites

Modèle incluant les variables précédentes.

```
repro_full <- glmer(repro ~ parKat +
                    milk_early_max_scaled_p1 +
                    milk_early_max_scaled_p2 +
                    milk_early_max_scaled_p3 +
                    milk_early_max_scaled_p4 +
                    dp_status + mast_cat +
                    (1 | herd_id),
                    data = lac,
                    subset = n_ctrl_early > 0 & dp_mis == 0,
                    family = binomial(link = "logit"))
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(repro_full)
```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]

Family: binomial (logit)

Formula:

```
repro ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
        milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + dp_status +
        mast_cat + (1 | herd_id)
```

Data: lac

Subset: n_ctrl_early > 0 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
-----	-----	--------	----------	----------

81668.2 81785.1 -40822.1 81644.2 125728

Scaled residuals:

Min	1Q	Median	3Q	Max
-32.809	0.170	0.268	0.389	2.160

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.5393	0.7344

Number of obs: 125740, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	2.85404	0.05512	51.78	<2e-16 ***
parKat1	1.46477	0.04464	32.81	<2e-16 ***
parKat3	-0.53954	0.03002	-17.97	<2e-16 ***
parKat4	-1.20873	0.02698	-44.81	<2e-16 ***
milk_early_max_scaled_p1	1.39284	0.03199	43.54	<2e-16 ***
milk_early_max_scaled_p2	0.67594	0.02346	28.81	<2e-16 ***
milk_early_max_scaled_p3	0.55542	0.02294	24.21	<2e-16 ***
milk_early_max_scaled_p4	0.52468	0.01665	31.51	<2e-16 ***
dp_statusdpHigh	-0.37402	0.02170	-17.24	<2e-16 ***
mast_catcont	-0.89945	0.02480	-36.27	<2e-16 ***
mast_catenv	-0.60924	0.02252	-27.05	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

	(Intr)	parKt1	parKt3	parKt4	m____1	m____2	m____3	m____4	dp_stH
parKat1	-0.173								
parKat3	-0.247	0.286							
parKat4	-0.269	0.290	0.551						
mlk_rly__1	0.018	0.749	-0.022	-0.038					
mlk_rly__2	0.057	0.005	-0.095	-0.125	0.081				
mlk_rly__3	0.022	0.045	-0.230	-0.049	0.076	0.088			
mlk_rly__4	0.031	0.059	-0.036	-0.238	0.096	0.109	0.112		
dp_sttsdpHg	-0.116	0.145	-0.062	-0.200	-0.023	0.004	-0.015	0.004	
mast_catcnt	-0.078	-0.026	-0.030	-0.062	0.007	-0.004	0.006	0.023	-0.135
mast_catenv	-0.098	-0.020	-0.022	-0.044	0.025	0.008	0.014	0.024	-0.075

mst_ctc

parKat1

parKat3

parKat4

```

mlk_rly___1
mlk_rly___2
mlk_rly___3
mlk_rly___4
dp_sttsdpHg
mast_catcnt
mast_catenv 0.305
fit warnings:
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```

```

repro_full1 <- glmer(repro ~ parKat +
                     milk_early_max_scaled_p1 +
                     milk_early_max_scaled_p2 +
                     milk_early_max_scaled_p3 +
                     milk_early_max_scaled_p4 +
                     mast_cat1 +
                     (1 | herd_id),
                     data = lac,
                     subset = n_ctrl_early > 0 & dp_mis == 0,
                     family = binomial(link = "logit"))

```

```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```

```

summary(repro_full1)

```

```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula:
repro ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +
(1 | herd_id)
Data: lac
Subset: n_ctrl_early > 0 & dp_mis == 0

      AIC      BIC   logLik deviance df.resid
81645.1 81801.0 -40806.6 81613.1   125724

Scaled residuals:
      Min       1Q   Median       3Q      Max
-35.752   0.172   0.268   0.386   2.104

```

Random effects:

```
Groups   Name      Variance Std.Dev.
herd_id (Intercept) 0.5388    0.734
Number of obs: 125740, groups: herd_id, 231
```

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	2.83833	0.05564	51.01	<2e-16 ***
parKat1	1.42973	0.04787	29.87	<2e-16 ***
parKat3	-0.53800	0.03004	-17.91	<2e-16 ***
parKat4	-1.20622	0.02702	-44.64	<2e-16 ***
milk_early_max_scaled_p1	1.39728	0.03199	43.69	<2e-16 ***
milk_early_max_scaled_p2	0.67517	0.02347	28.76	<2e-16 ***
milk_early_max_scaled_p3	0.55404	0.02297	24.12	<2e-16 ***
milk_early_max_scaled_p4	0.52229	0.01668	31.31	<2e-16 ***
mast_cat1dpHigh_cont	-1.32393	0.03429	-38.61	<2e-16 ***
mast_cat1dpHigh_env	-1.00703	0.03459	-29.11	<2e-16 ***
mast_cat1dpHigh_healthy	-0.29752	0.02895	-10.28	<2e-16 ***
mast_cat1dpLow_cont	-0.84608	0.04270	-19.82	<2e-16 ***
mast_cat1dpLow_env	-0.57450	0.03607	-15.93	<2e-16 ***
mast_cat1primi_cont	-0.69112	0.05950	-11.62	<2e-16 ***
mast_cat1primi_env	-0.51394	0.04708	-10.92	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or
`vcov(x)` if you need it

fit warnings:

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

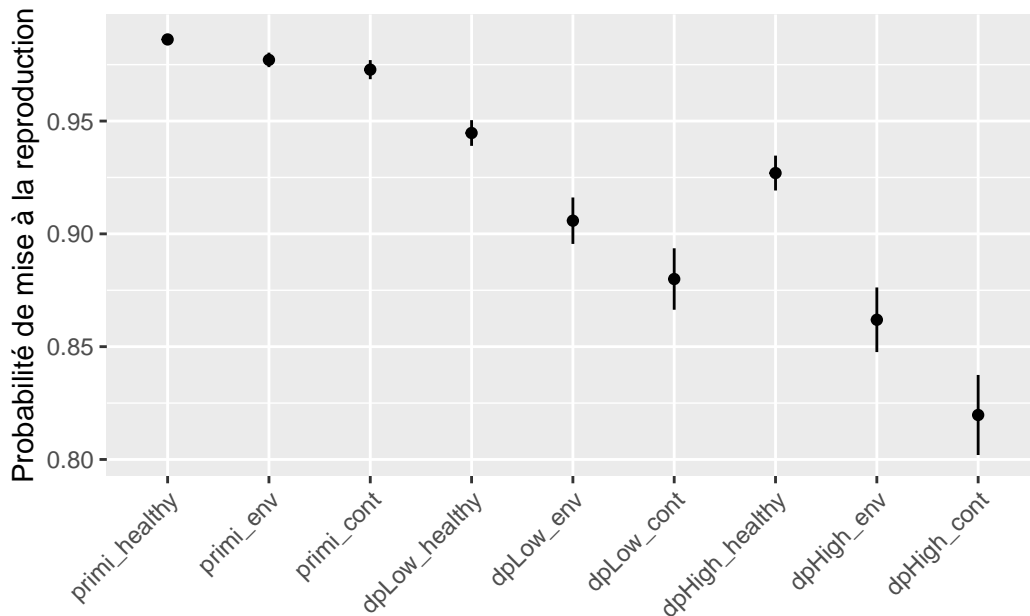
La figure ci-dessous représente les probabilités prédites qu'une vache soit mise à la reproduction en fonction de son statut vis-à-vis des mammites. Une production laitière moyenne a été prise en compte. Cela avantage les primipares dont la production laitière moyenne est inférieure à la la production laitière moyenne du reste des vaches. On a considéré que les multipares étaient de parité 2.

```

prd <- marginales::predictions(model = repro_full1,
                              newdata = mast_pred,
                              re.form = NA, type = "response")

ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité de mise à la reproduction") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))

```



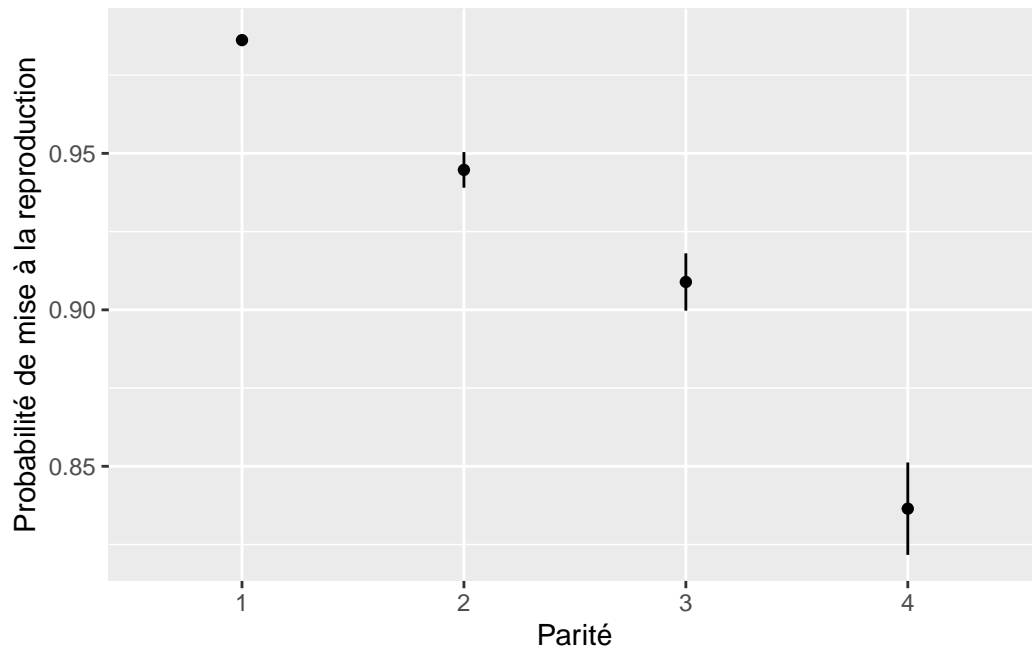
Effet de la parité sur la probabilité d'être mise à la reproduction.

```

prd <- marginales::predictions(model = repro_full1,
                              newdata = parit_pred,
                              re.form = NA, type = "response")

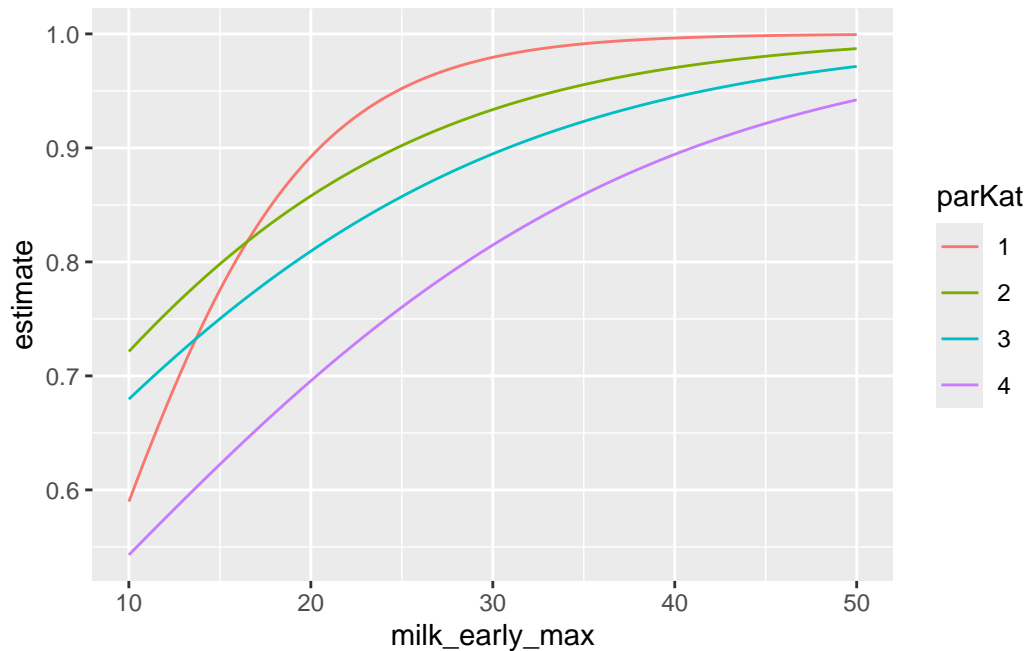
ggplot(prd, aes(x = factor(parKat), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = parKat, y = conf.low, xend = parKat, yend = conf.high)) +
  xlab("Parité") +
  ylab("Probabilité de mise à la reproduction")

```



Effets combinés de la parité et de la production laitière.

```
prd <- marginales::predictions(model = repro_full1,  
                               newdata = milk_pred,  
                               re.form = NA, type = "response")  
  
ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +  
  geom_line()
```



6 Probabilité de revêlage pour les vaches mises à la reproduction

Une variable binaire prenant la valeur 1 si la vache revêlage et 0 autrement est construite.

```
lac <- lac |>
mutate(
  repro_revel = case_when(
    repro == 1 & !is.na(calv_date_nxt) ~ 1,
    repro == 1 & is.na(calv_date_nxt) ~ 0,
    repro == 0 ~ NA
  )
)
```

Le modèle suivant estime la probabilité qu'une vache mise à la reproduction revêlage.

```
repro_revel_null <- glmer(repro_revel ~ 1 + (1 | herd_id),
  family = binomial(link = "logit"),
  data = lac,
  subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0)

summary(repro_revel_null)
```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]

Family: binomial (logit)

Formula: repro_revel ~ 1 + (1 | herd_id)

Data: lac

Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
136873.8	136893.1	-68434.9	136869.8	110249

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.9104	-1.2031	0.6016	0.7078	3.8621

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	1.03	1.015

Number of obs: 110251, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.35090	0.06756	5.194	2.06e-07 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Le modèle suivant inclut toutes les variables retenues dans le modèle de mise à la reproduction. Le statut au tarissement ne semble pas avoir une influence importante. Ci-après le même modèle est réestimé en faisant abstraction du statut au tarissement. Les vaches qui ont une des formes de mammites prises en compte ont une probabilité de revêlage différente.

```
repro_mast_cat1 <- glmer(repro_revel ~ parKat +
  milk_early_max_scaled_p1 +
  milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 +
  milk_early_max_scaled_p4 +
  mast_cat1 +
  (1 |herd_id),
  family = binomial(link = "logit"),
  data = lac,
  subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0)
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(repro_mast_cat1)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula:
repro_revel ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +
  (1 | herd_id)
Data: lac
Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0
```

```
      AIC      BIC   logLik deviance df.resid
135243.2 135397.0 -67605.6 135211.2   110235
```

```
Scaled residuals:
```

```
      Min       1Q   Median       3Q      Max
-3.8028 -1.1405  0.5754  0.7129  3.7528
```

```
Random effects:
```

```
Groups Name      Variance Std.Dev.
herd_id (Intercept) 1.107    1.052
Number of obs: 110251, groups: herd_id, 231
```

```
Fixed effects:
```

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.46179	0.07200	6.414	1.42e-10 ***
parKat1	0.30303	0.02385	12.706	< 2e-16 ***
parKat3	-0.12998	0.02236	-5.813	6.13e-09 ***
parKat4	-0.43241	0.02136	-20.240	< 2e-16 ***
milk_early_max_scaled_p1	0.24108	0.01856	12.991	< 2e-16 ***
milk_early_max_scaled_p2	0.08344	0.01557	5.358	8.41e-08 ***
milk_early_max_scaled_p3	0.01082	0.01752	0.618	0.53687
milk_early_max_scaled_p4	0.10643	0.01560	6.821	9.03e-12 ***
mast_cat1dpHigh_cont	-0.39825	0.03405	-11.696	< 2e-16 ***
mast_cat1dpHigh_env	-0.19504	0.03193	-6.108	1.01e-09 ***
mast_cat1dpHigh_healthy	-0.05237	0.02111	-2.481	0.01310 *
mast_cat1dpLow_cont	-0.32456	0.03738	-8.683	< 2e-16 ***
mast_cat1dpLow_env	-0.15079	0.02836	-5.317	1.05e-07 ***
mast_cat1primi_cont	-0.10553	0.04020	-2.625	0.00866 **
mast_cat1primi_env	0.02720	0.03003	0.906	0.36500

```
---
```


Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

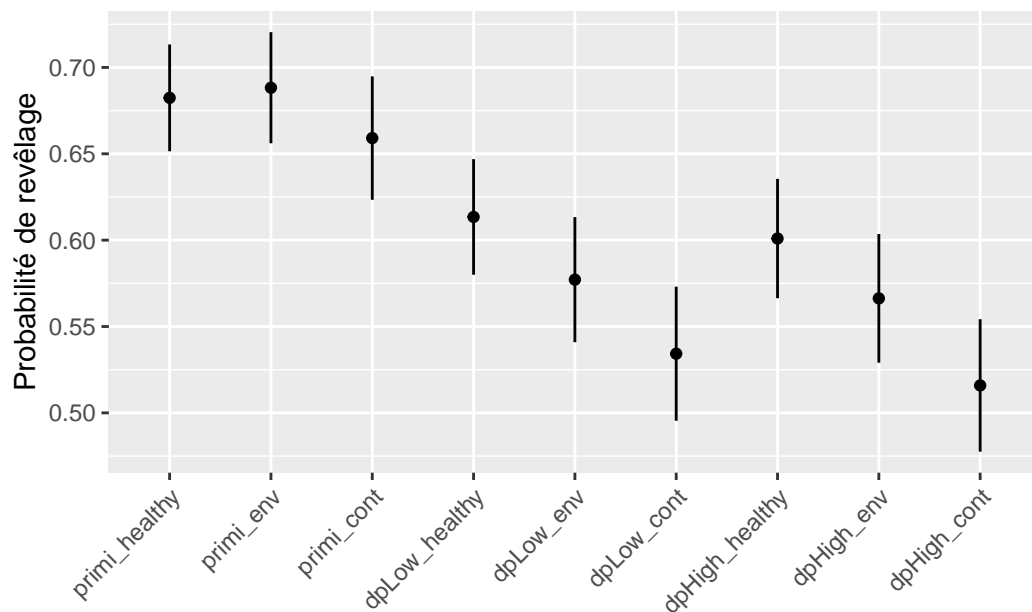
Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or
`vcov(x)` if you need it

fit warnings:

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
prd <- margineffects::predictions(model = repro_mast_cat1,  
                                newdata = mast_pred,  
                                re.form = NA, type = "response")  
  
ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +  
  geom_point() +  
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +  
  xlab("") +  
  ylab("Probabilité de revêlage") +  
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```

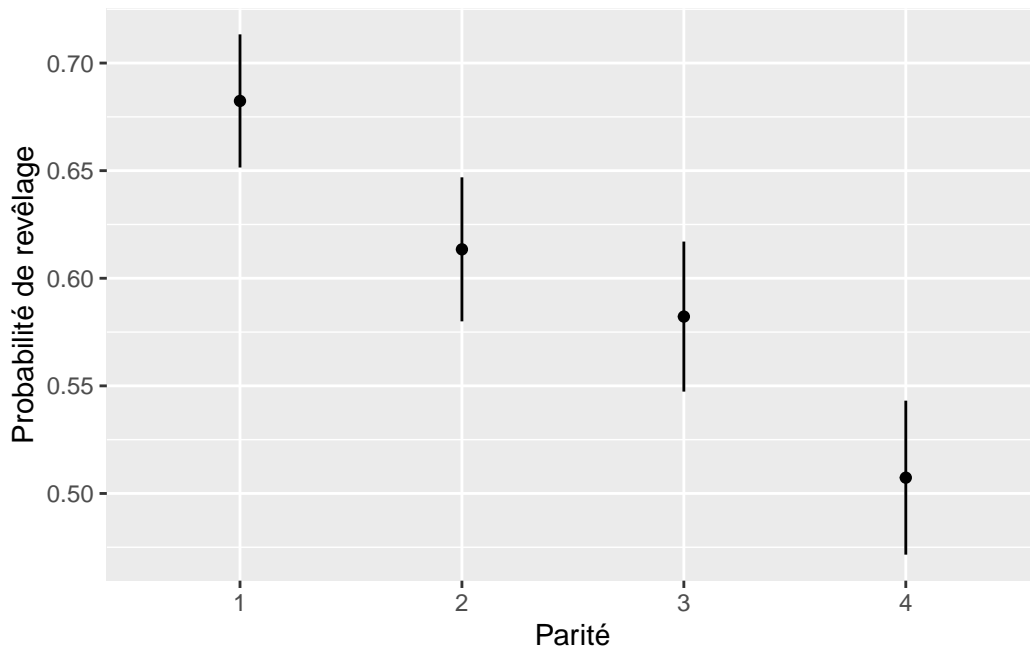


```

prd <- marginales::predictions(model = repro_mast_cat1,
                              newdata = parit_pred,
                              re.form = NA, type = "response")

ggplot(prd, aes(x = factor(parKat), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = parKat, y = conf.low, xend = parKat, yend = conf.high)) +
  xlab("Parité") +
  ylab("Probabilité de revêlage")

```



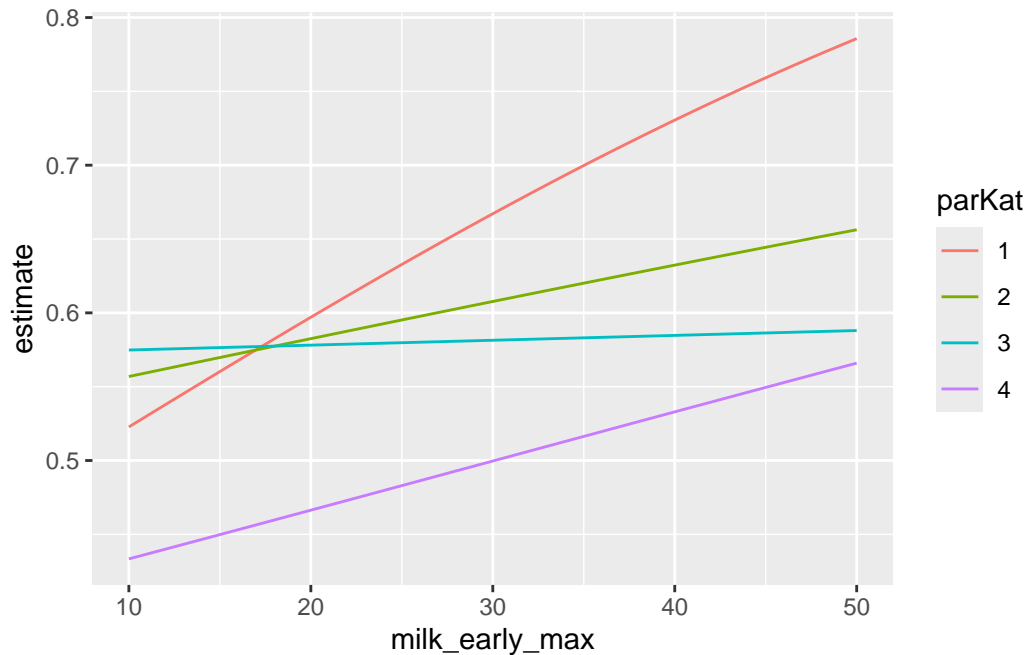
Les effets de la production laitière sont plus difficiles à comprendre.

```

prd <- marginales::predictions(model = repro_mast_cat1,
                              newdata = milk_pred,
                              re.form = NA, type = "response")

ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +
  geom_line()

```



Même modèle que plus haut avec cette fois les différentes catégories de statuts liées au tarissement regroupées en multipares.

```
repro_mast_cat2 <- glmer(repro_revel ~ parKat +
  milk_early_max_scaled_p1 +
  milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 +
  milk_early_max_scaled_p4 +
  mast_cat2 +
  (1 | herd_id),
  family = binomial(link = "logit"),
  data = lac,
  subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0)
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(repro_mast_cat2)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula:
```

```
repro_revel ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat2 +
  (1 | herd_id)
```

Data: lac

Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
135246.8	135371.7	-67610.4	135220.8	110238

Scaled residuals:

Min	1Q	Median	3Q	Max
-3.7901	-1.1412	0.5756	0.7130	3.7708

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	1.106	1.052

Number of obs: 110251, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.45030	0.07186	6.266	3.69e-10 ***
parKat1	0.31357	0.02331	13.451	< 2e-16 ***
parKat3	-0.13403	0.02232	-6.005	1.92e-09 ***
parKat4	-0.44535	0.02094	-21.266	< 2e-16 ***
milk_early_max_scaled_p1	0.24020	0.01855	12.946	< 2e-16 ***
milk_early_max_scaled_p2	0.08405	0.01557	5.397	6.76e-08 ***
milk_early_max_scaled_p3	0.01061	0.01752	0.605	0.54487
milk_early_max_scaled_p4	0.10711	0.01560	6.866	6.61e-12 ***
mast_cat2multi_cont	-0.34700	0.02566	-13.524	< 2e-16 ***
mast_cat2multi_env	-0.15240	0.02174	-7.010	2.38e-12 ***
mast_cat2primi_cont	-0.10455	0.04021	-2.600	0.00932 **
mast_cat2primi_env	0.02776	0.03002	0.925	0.35515

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:

	(Intr)	parKt1	parKt3	parKt4	m____1	m____2	m____3	m____4
parKat1	-0.118							
parKat3	-0.109	0.314						
parKat4	-0.113	0.309	0.407					
mlk_rly__1	0.001	0.548	-0.031	-0.055				
mlk_rly__2	-0.034	0.197	0.083	0.066	0.126			
mlk_rly__3	0.001	0.072	-0.395	-0.053	0.105	0.107		

```

mlk_rly___4    0.001  0.083 -0.034 -0.381  0.113  0.116  0.103
mst_ct2mlt_c -0.039  0.124 -0.041 -0.110  0.007  0.018  0.026  0.052
mst_ct2mlt_n -0.052  0.164 -0.022 -0.065  0.006  0.006  0.003  0.018
mst_ct2prm_c  0.002 -0.183  0.001  0.000  0.024 -0.006 -0.005 -0.004
mst_ct2prm_n -0.001 -0.249 -0.001 -0.004  0.021 -0.001  0.000  0.000
               mst_ct2mlt_c mst_ct2mlt_n mst_ct2prm_c
parKat1
parKat3
parKat4
mlk_rly___1
mlk_rly___2
mlk_rly___3
mlk_rly___4
mst_ct2mlt_c
mst_ct2mlt_n  0.185
mst_ct2prm_c  0.007          0.007
mst_ct2prm_n  0.006          0.007          0.157
fit warnings:
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```

```

## catégories de statuts mammites pour les primipares
mast_lev_p1 <- lac |>
  filter(n_ctrl_early > 0 & dp_mis == 0 & parKat == 1) |>
  distinct(mast_cat2) |>
  mutate(mast_cat2 = as.character(mast_cat2)) |>
  pull(mast_cat2)
## catégories de statuts mammites pour les multipares
mast_lev_p2 <- lac |>
  filter(n_ctrl_early > 0 & dp_mis == 0 & parKat != 1) |>
  distinct(mast_cat2) |>
  mutate(mast_cat2 = as.character(mast_cat2)) |>
  pull(mast_cat2)

## jeu de données
mast_pred1 <- tibble(
  parKat = factor(c(rep(1, length(mast_lev_p1)),
                    rep(2, length(mast_lev_p2)))),
  mast_cat2 = factor(c(mast_lev_p1, mast_lev_p2)),
  milk_early_max_scaled_p1 = 0,
  milk_early_max_scaled_p2 = 0,
  milk_early_max_scaled_p3 = 0,
  milk_early_max_scaled_p4 = 0
)

```

```

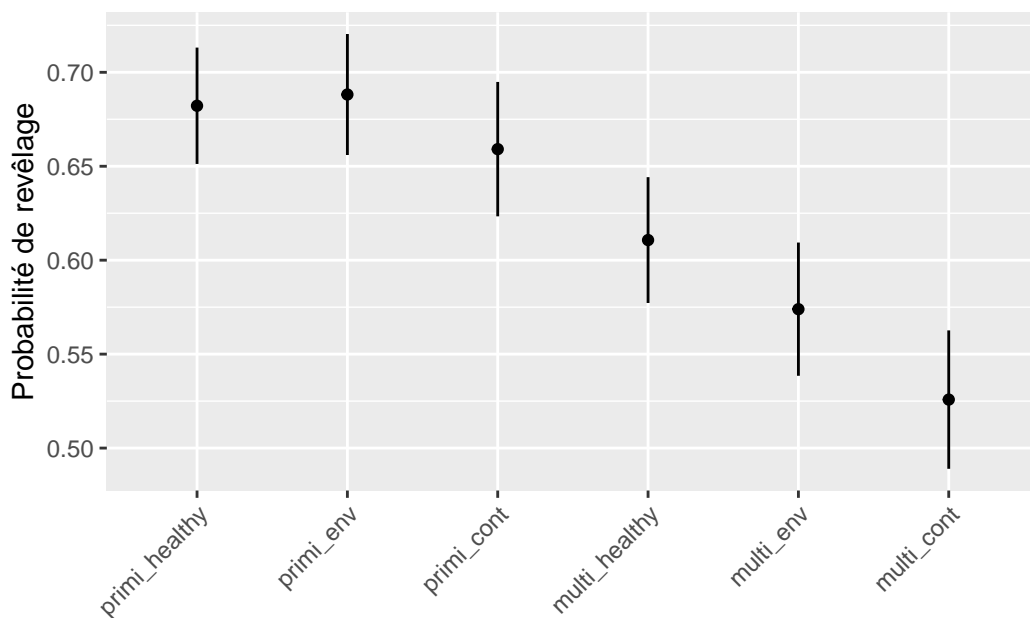
)

level_order1 <- c("primi_healthy", "primi_env", "primi_cont",
                  "multi_healthy", "multi_env", "multi_cont")

prd <- margineffects::predictions(model = repro_mast_cat2,
                                newdata = mast_pred1,
                                re.form = NA, type = "response")

ggplot(prd, aes(x = factor(mast_cat2, level = level_order1), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat2, y = conf.low, xend = mast_cat2, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité de revêlage") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))

```



```

parit_pred1 <- expand_grid(
  parKat = factor(1:4),
  milk_early_max = 0
) |>
mutate(

```

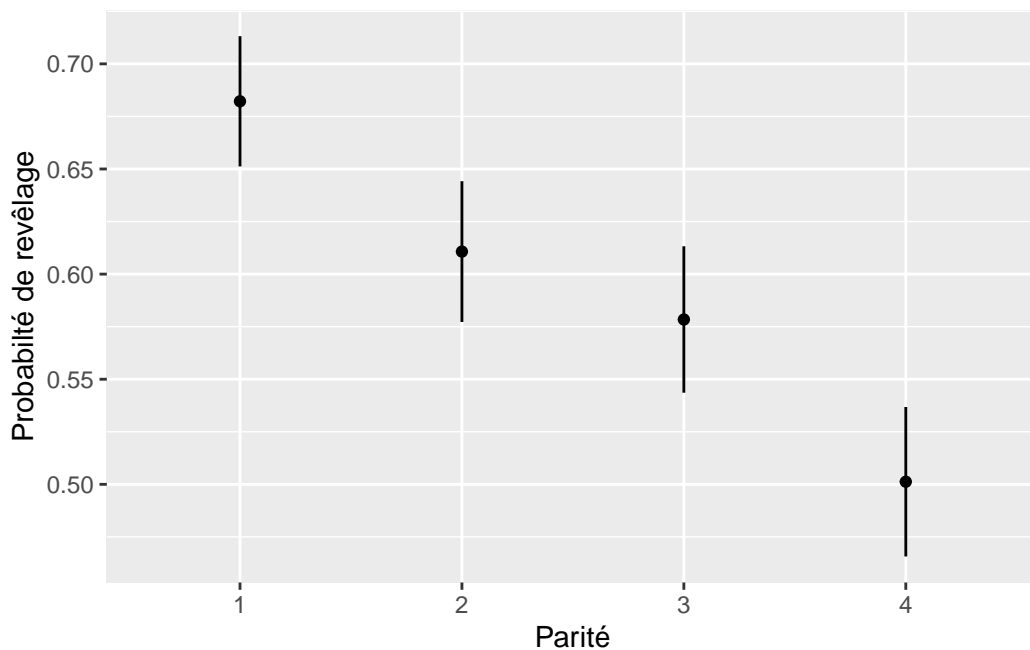
```

milk_early_max_scaled_p1 = 0,
milk_early_max_scaled_p2 = 0,
milk_early_max_scaled_p3 = 0,
milk_early_max_scaled_p4 = 0,
mast_cat2 = factor(ifelse(parKat == 1, "primi_healthy", "multi_healthy"))
)

prd <- marginaleffects::predictions(model = repro_mast_cat2,
                                   newdata = parit_pred1,
                                   re.form = NA, type = "response")

ggplot(prd, aes(x = factor(parKat), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = parKat, y = conf.low, xend = parKat, yend = conf.high)) +
  xlab("Parité") +
  ylab("Probabilité de revêlage")

```



```

milk_pred1 <- expand_grid(
  parKat = factor(1:4),
  milk_early_max = seq(10, 50, by = .1)
) |>
mutate(

```

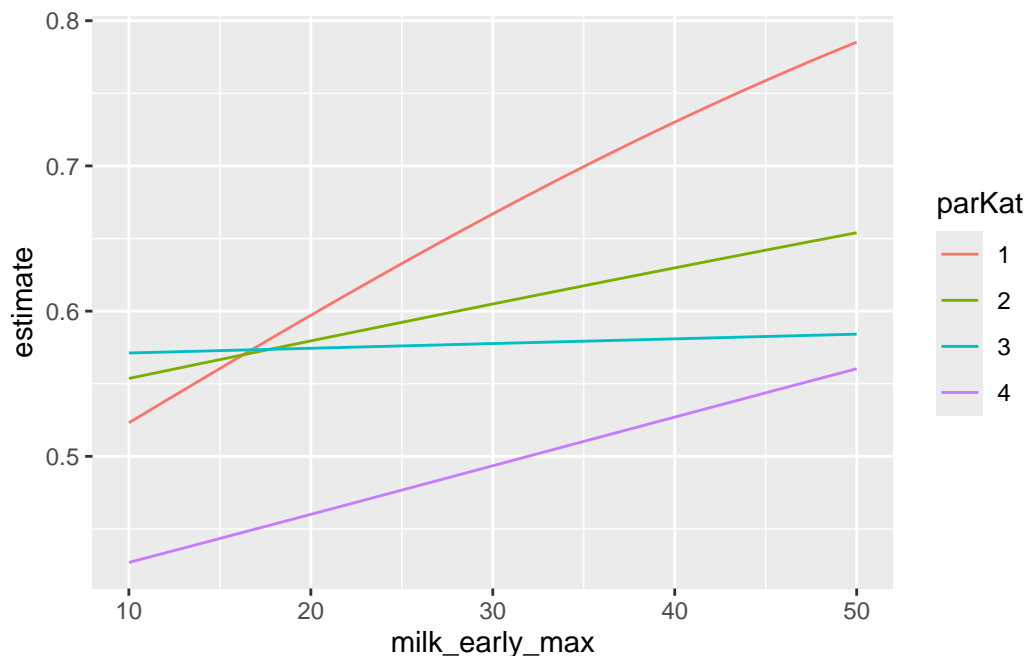
```

milk_early_max_scaled = (milk_early_max - milk_early_max_mean) /
  milk_early_max_sd,
milk_early_max_scaled_p1 = ifelse(parKat == 1, milk_early_max_scaled, 0),
milk_early_max_scaled_p2 = ifelse(parKat == 2, milk_early_max_scaled, 0),
milk_early_max_scaled_p3 = ifelse(parKat == 3, milk_early_max_scaled, 0),
milk_early_max_scaled_p4 = ifelse(parKat == 4, milk_early_max_scaled, 0),
mast_cat2 = factor(ifelse(parKat == 1, "primi_healthy", "multi_healthy"))
)

prd <- marginaleffects::predictions(model = repro_mast_cat2,
  newdata = milk_pred1,
  re.form = NA, type = "response")

ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +
  geom_line()

```



7 Probabilité de réussite de la première IA

Une variable succès à la première IA est construite. Un succès à la première insémination est défini de la manière suivante :

- la date de première insémination étant également la date de la dernière insémination
- la vache inséminée revê

```
lac <- lac |>
mutate(ai1_success = case_when(
  is.na(n_ai) ~ NA,
  ai_first == ai_last & !is.na(calv_date_nxt) ~ 1,
  ai_first != ai_last ~ 0
))
```

Premier modèle contenant uniquement un intercept.

```
ai1_success_null <- glmer(ai1_success ~ 1 + (1 | herd_id),
  data = lac,
  subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0,
  family = binomial(link = "logit"))

summary(ai1_success_null)
```

Generalized linear mixed model fit by maximum likelihood (Laplace

```
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: ai1_success ~ 1 + (1 | herd_id)
Data: lac
Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0
```

```
      AIC      BIC   logLik deviance df.resid
125578.5 125597.4 -62787.2 125574.5     96517
```

Scaled residuals:

```
      Min       1Q   Median       3Q      Max
-1.3394 -0.8048 -0.6448  1.0892  2.7414
```

Random effects:

```
Groups   Name             Variance Std.Dev.
herd_id (Intercept) 0.5152    0.7178
Number of obs: 96519, groups: herd_id, 231
```

Fixed effects:

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.68947    0.04951  -13.93   <2e-16 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```

ai1_success_mast_cat1 <- glmer(ai1_success ~ parKat +
                                milk_early_max_scaled_p1 +
                                milk_early_max_scaled_p2 +
                                milk_early_max_scaled_p3 +
                                milk_early_max_scaled_p4 +
                                mast_cat1 + (1 | herd_id),
                                data = lac,
                                subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0,
                                family = binomial(link = "logit"))

```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(ai1_success_mast_cat1)
```

Generalized linear mixed model fit by maximum likelihood (Laplace

Approximation) [glmerMod]

Family: binomial (logit)

Formula:

```

ai1_success ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
              milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +
              (1 | herd_id)

```

Data: lac

Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
125149.2	125300.8	-62558.6	125117.2	96503

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.4913	-0.8125	-0.6232	1.0937	3.3161

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.5344	0.731

Number of obs: 96519, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.632539	0.052198	-12.118	< 2e-16 ***
parKat1	0.011119	0.023438	0.474	0.635232
parKat3	-0.005357	0.023149	-0.231	0.816981

```

parKat4                -0.190077    0.022854  -8.317  < 2e-16 ***
milk_early_max_scaled_p1 -0.058320    0.018233  -3.199  0.001381 **
milk_early_max_scaled_p2 -0.151878    0.016193  -9.379  < 2e-16 ***
milk_early_max_scaled_p3 -0.163435    0.018948  -8.625  < 2e-16 ***
milk_early_max_scaled_p4 -0.067033    0.017415  -3.849  0.000119 ***
mast_cat1dpHigh_cont     -0.126176    0.038686  -3.262  0.001108 **
mast_cat1dpHigh_env      -0.069031    0.035043  -1.970  0.048850 *
mast_cat1dpHigh_healthy  -0.040161    0.022175  -1.811  0.070133 .
mast_cat1dpLow_cont      -0.040168    0.041314  -0.972  0.330917
mast_cat1dpLow_env       -0.049705    0.030403  -1.635  0.102075
mast_cat1primi_cont      -0.031164    0.039837  -0.782  0.434043
mast_cat1primi_env       0.004007    0.028976   0.138  0.890005
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or
`vcov(x)` if you need it

fit warnings:

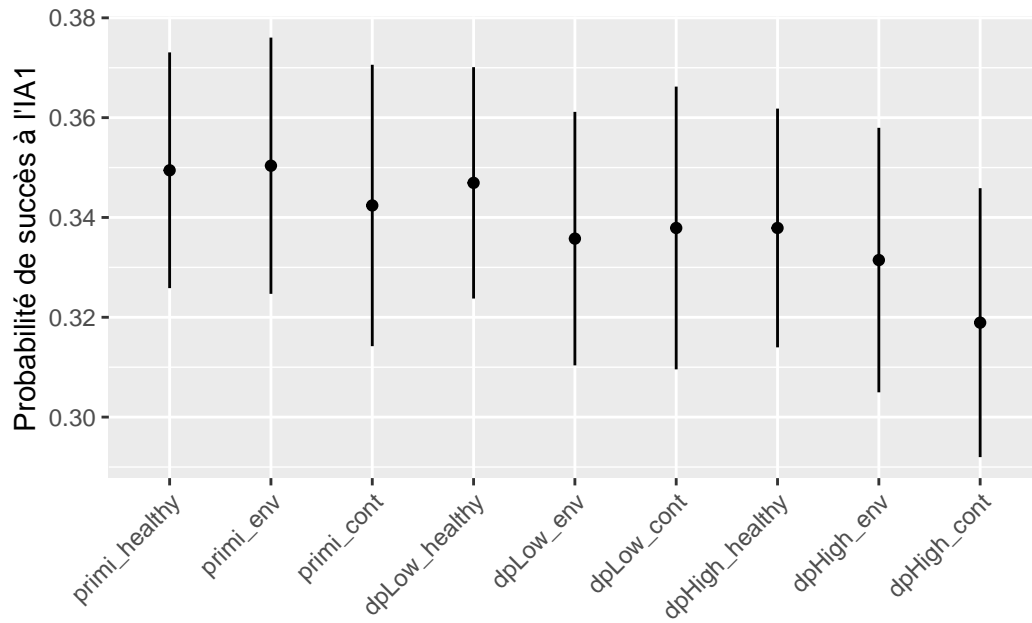
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```

prd <- marginalesffects::predictions(model = ail_success_mast_cat1,
                                     newdata = mast_pred,
                                     re.form = NA, type = "response")

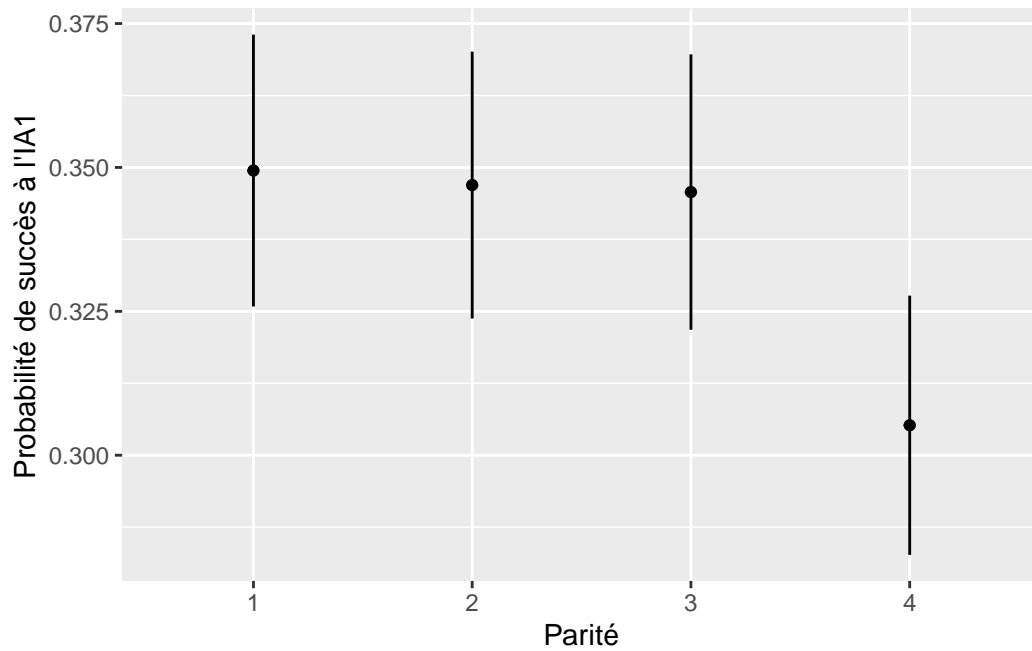
ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité de succès à l'IA1") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))

```

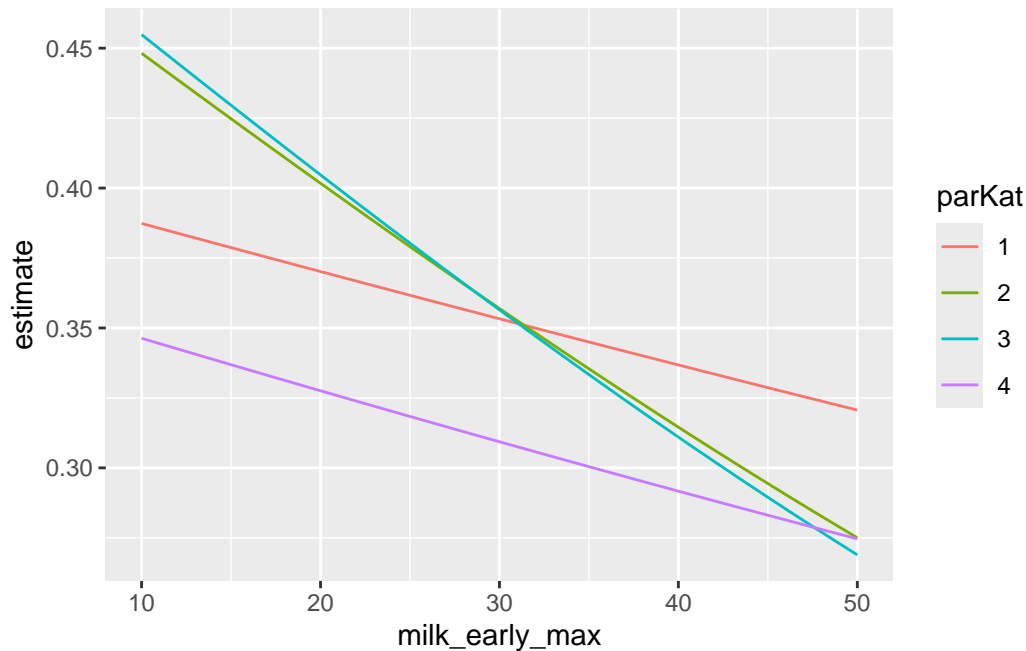


```
prd <- marginales::predictions(model = ai1_success_mast_cat1,
                               newdata = parit_pred,
                               re.form = NA, type = "response")

ggplot(prd, aes(x = factor(parKat), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = parKat, y = conf.low, xend = parKat, yend = conf.high)) +
  xlab("Parité") +
  ylab("Probabilité de succès à l'IA1")
```



```
prd <- marginales::predictions(model = ai1_success_mast_cat1,  
                               newdata = milk_pred,  
                               re.form = NA, type = "response")  
  
ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +  
  geom_line()
```



8 Nombre d'inséminations artificielles dans une lactation

On ne considère ici que les lactations pour les quelles il y a eu au moins une insémination artificielle et pour lesquelles la vache n'est pas morte pendant la lactation. Une méthode qui semble adaptée pour modéliser ce type de données est la régression de Poisson.

```
nb_ai_null <- glmer(n_ai ~ 1 +
  (1 | herd_id),
  data = lac,
  subset = !is.na(n_ai) & n_ai > 0 & death == 0,
  family = poisson)

summary(nb_ai_null)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: poisson ( log )
Formula: n_ai ~ 1 + (1 | herd_id)
Data: lac
Subset: !is.na(n_ai) & n_ai > 0 & death == 0
```

AIC	BIC	logLik	deviance	df.resid
335957.4	335976.3	-167976.7	335953.4	97433

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.9892	-0.7055	-0.3793	0.3776	18.7368

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.06669	0.2582

Number of obs: 97435, groups: herd_id, 231

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.83261	0.01732	48.07	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
nb_ai_full <- glmer(n_ai ~ parKat +
  milk_early_max_scaled_p1 +
  milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 +
  milk_early_max_scaled_p4 +
  mast_cat1 +
  (1 | herd_id),
  data = lac,
  subset = !is.na(n_ai) & n_ai > 0 & death == 0,
  family = poisson)
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(nb_ai_null)
```

Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]
 Family: poisson (log)
 Formula: n_ai ~ 1 + (1 | herd_id)
 Data: lac
 Subset: !is.na(n_ai) & n_ai > 0 & death == 0

AIC	BIC	logLik	deviance	df.resid
-----	-----	--------	----------	----------

335957.4 335976.3 -167976.7 335953.4 97433

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.9892	-0.7055	-0.3793	0.3776	18.7368

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.06669	0.2582

Number of obs: 97435, groups: herd_id, 231

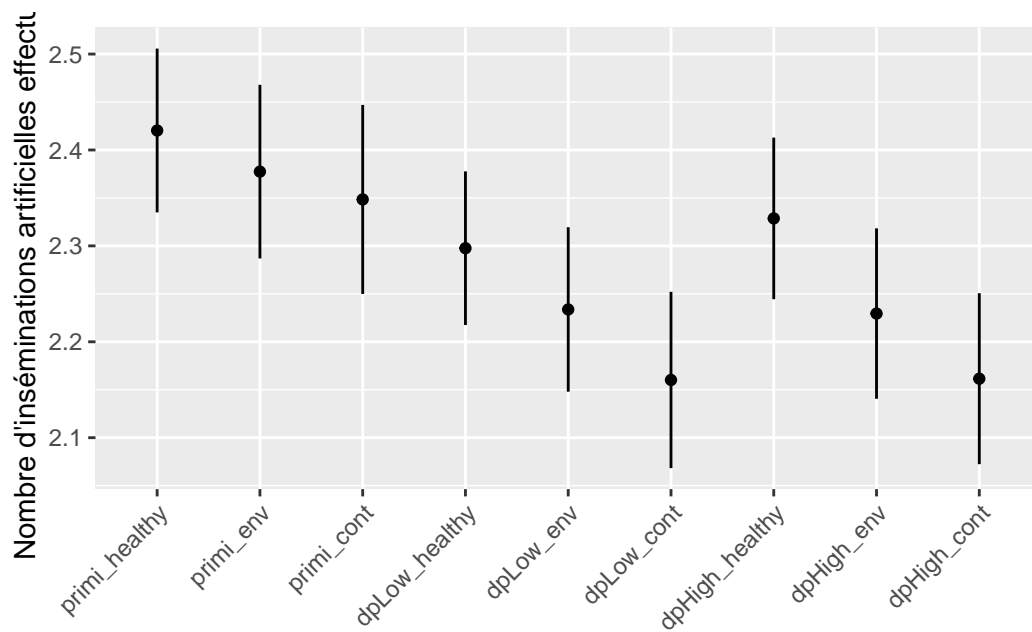
Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	0.83261	0.01732	48.07	<2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
prd <- marginalesffects::predictions(model = nb_ai_full,
                                     newdata = mast_pred,
                                     re.form = NA, type = "response")

ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Nombre d'inséminations artificielles effectuées") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```

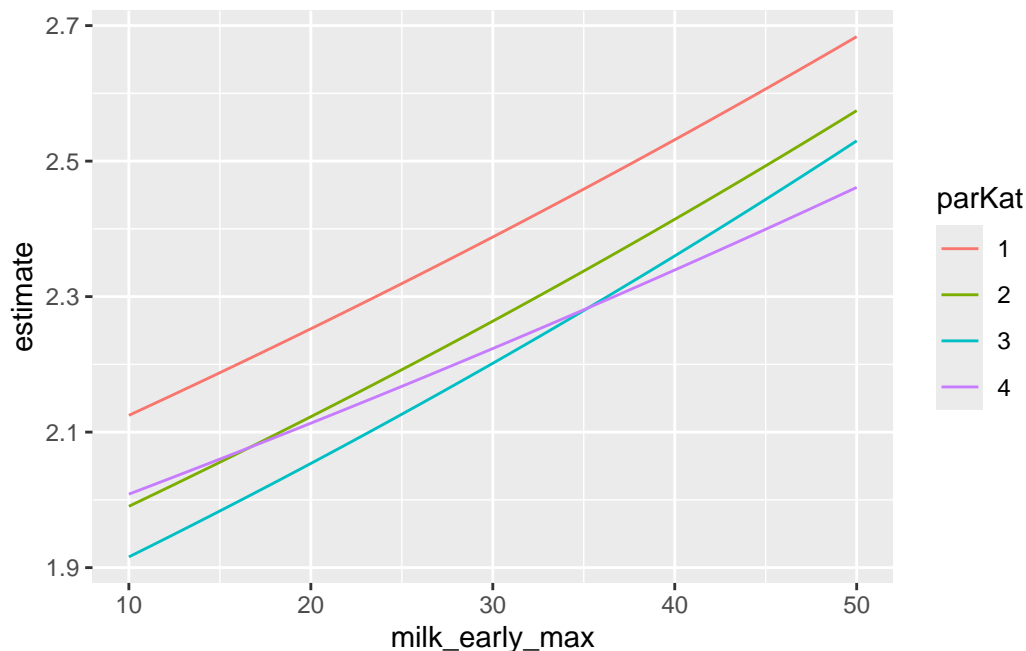



```

prd <- marginales::predictions(model = nb_ai_full,
                               newdata = milk_pred,
                               re.form = NA, type = "response")

ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +
  geom_line()

```



9 Dynamique des infections intra-mammaire lors de la période tarie suivant la mise à la reproduction

9.1 Probabilité d'une CCS élevée au dernier contrôle avant tarissement

```
inf_end_lac_full <- glmer(factor(infect_end_lac) ~ parKat +
  milk_early_max_scaled_p1 +
  milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 +
  milk_early_max_scaled_p4 +
  mast_cat1 +
  (1 | herd_id),
  family = binomial(link = "logit"),
  data = lac,
  subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0 & !is.na(infect_end_lac))
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

Warning in checkConv(attr(opt, "derivs"), opt\$par, ctrl = control\$checkConv, :
Model failed to converge with max|grad| = 0.00413046 (tol = 0.002, component 1)

```
summary(inf_end_lac_full)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace  
Approximation) [glmerMod]
```

```
Family: binomial (logit)
```

```
Formula:
```

```
factor(infct_end_lac) ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +  
  milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +  
  (1 | herd_id)
```

```
Data: lac
```

```
Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0 & !is.na(infct_end_lac)
```

AIC	BIC	logLik	deviance	df.resid
81209.2	81356.3	-40588.6	81177.2	72580

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-4.3276	-0.6406	-0.4413	0.7760	3.8664

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.216	0.4647

```
Number of obs: 72596, groups: herd_id, 217
```

```
Fixed effects:
```

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-1.28032	0.03857	-33.197	< 2e-16 ***
parKat1	-0.35399	0.03176	-11.147	< 2e-16 ***
parKat3	0.41635	0.02820	14.765	< 2e-16 ***
parKat4	0.77485	0.02816	27.518	< 2e-16 ***
milk_early_max_scaled_p1	0.04357	0.02412	1.806	0.0709 .
milk_early_max_scaled_p2	0.11020	0.02006	5.493	3.95e-08 ***
milk_early_max_scaled_p3	0.11878	0.02261	5.253	1.50e-07 ***
milk_early_max_scaled_p4	0.02709	0.02163	1.252	0.2104
mast_cat1dpHigh_cont	2.35456	0.05900	39.907	< 2e-16 ***
mast_cat1dpHigh_env	1.56029	0.04330	36.033	< 2e-16 ***
mast_cat1dpHigh_healthy	0.93902	0.02588	36.286	< 2e-16 ***
mast_cat1dpLow_cont	1.46248	0.05003	29.231	< 2e-16 ***
mast_cat1dpLow_env	0.62248	0.03538	17.593	< 2e-16 ***
mast_cat1primi_cont	2.03849	0.04714	43.243	< 2e-16 ***
mast_cat1primi_env	0.92463	0.03561	25.965	< 2e-16 ***

```
---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or

`vcov(x)` if you need it

fit warnings:

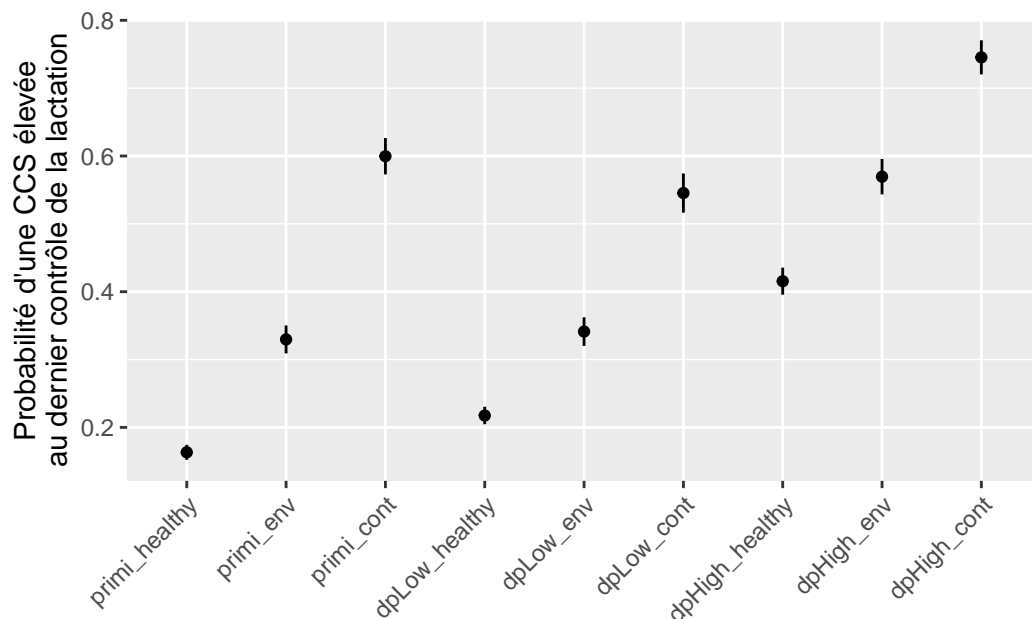
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

optimizer (Nelder-Mead) convergence code: 0 (OK)

Model failed to converge with $\max|\text{grad}| = 0.00413046$ (tol = 0.002, component 1)

```
prd <- marginalesffects::predictions(model = inf_end_lac_full,
                                     newdata = mast_pred,
                                     re.form = NA, type = "response")

ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité d'une CCS élevée\nau dernier contrôle de la lactation") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```

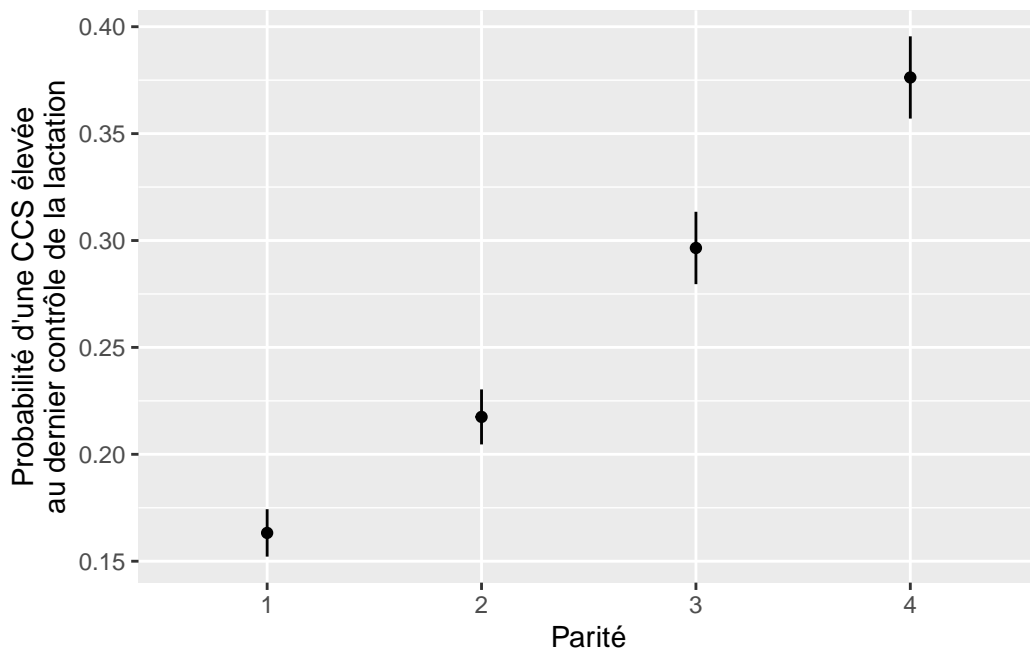


```

prd <- marginales::predictions(model = inf_end_lac_full,
                              newdata = parit_pred,
                              re.form = NA, type = "response")

ggplot(prd, aes(x = factor(parKat), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = parKat, y = conf.low, xend = parKat, yend = conf.high)) +
  xlab("Parité") +
  ylab("Probabilité d'une CCS élevée\nau dernier contrôle de la lactation")

```

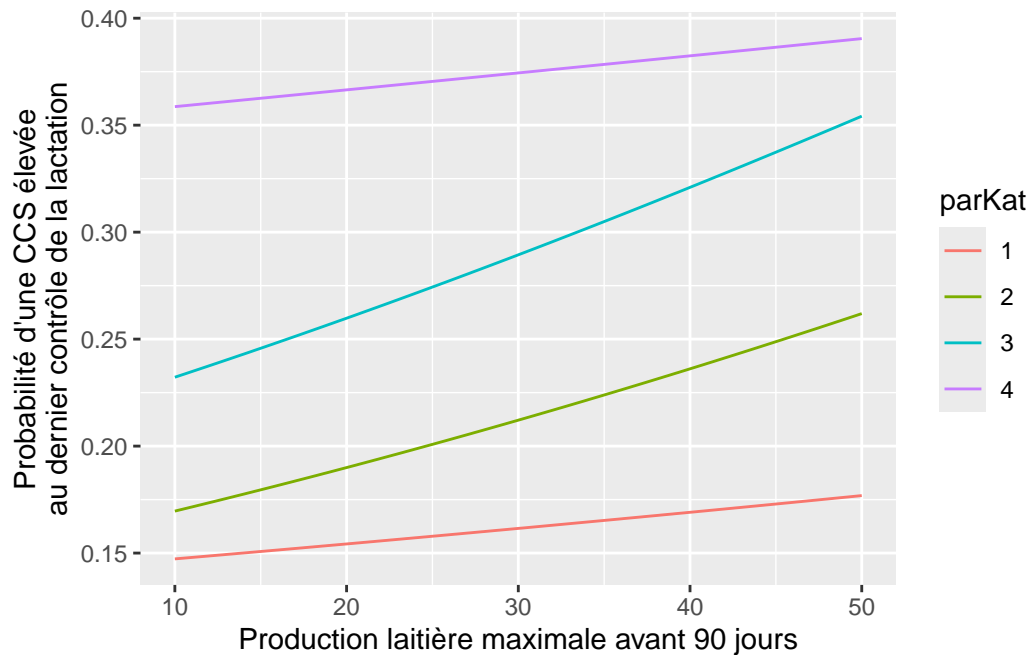


```

prd <- marginales::predictions(model = inf_end_lac_full,
                              newdata = milk_pred,
                              re.form = NA, type = "response")

ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +
  geom_line() +
  ylab("Probabilité d'une CCS élevée\nau dernier contrôle de la lactation") +
  xlab("Production laitière maximale avant 90 jours")

```



9.2 Probabilité de guérison

On utilise la variable statut mammite comme prédicteur de ce qui va se passer en fin de lactation en lors de la lactation suivante.

```
nxt_cure_full <- glmer(nxt_cure ~ parKat +
  milk_early_max_scaled_p1 +
  milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 +
  milk_early_max_scaled_p4 +
  mast_cat1 +
  (1 | herd_id),
  family = binomial(link = "logit"),
  data = lac,
  subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0)
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

Warning in checkConv(attr(opt, "derivs"), opt\$par, ctrl = control\$checkConv, :
Model failed to converge with max|grad| = 0.00243732 (tol = 0.002, component 1)

```
summary(nxt_cure_full)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula:
nxt_cure ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +
  (1 | herd_id)
Data: lac
Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0
```

AIC	BIC	logLik	deviance	df.resid
29004.6	29135.4	-14486.3	28972.6	26245

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.9817	-0.9052	0.4848	0.5945	1.9748

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.1355	0.3682

Number of obs: 26261, groups: herd_id, 216

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	1.50967	0.04812	31.372	< 2e-16 ***
parKat1	0.01693	0.06638	0.255	0.798719
parKat3	-0.08806	0.04727	-1.863	0.062490 .
parKat4	-0.22755	0.04376	-5.200	1.99e-07 ***
milk_early_max_scaled_p1	-0.01731	0.04631	-0.374	0.708569
milk_early_max_scaled_p2	-0.05664	0.03463	-1.636	0.101915
milk_early_max_scaled_p3	-0.11750	0.03432	-3.424	0.000617 ***
milk_early_max_scaled_p4	-0.06436	0.02906	-2.215	0.026768 *
mast_cat1dpHigh_cont	-1.33510	0.05593	-23.869	< 2e-16 ***
mast_cat1dpHigh_env	-0.70474	0.05566	-12.662	< 2e-16 ***
mast_cat1dpHigh_healthy	-0.21968	0.04517	-4.863	1.16e-06 ***
mast_cat1dpLow_cont	-1.03531	0.06516	-15.890	< 2e-16 ***
mast_cat1dpLow_env	-0.43655	0.06074	-7.187	6.63e-13 ***
mast_cat1primi_cont	-0.51182	0.07447	-6.873	6.29e-12 ***
mast_cat1primi_env	-0.08941	0.07504	-1.192	0.233436

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or

`vcov(x)` if you need it

fit warnings:

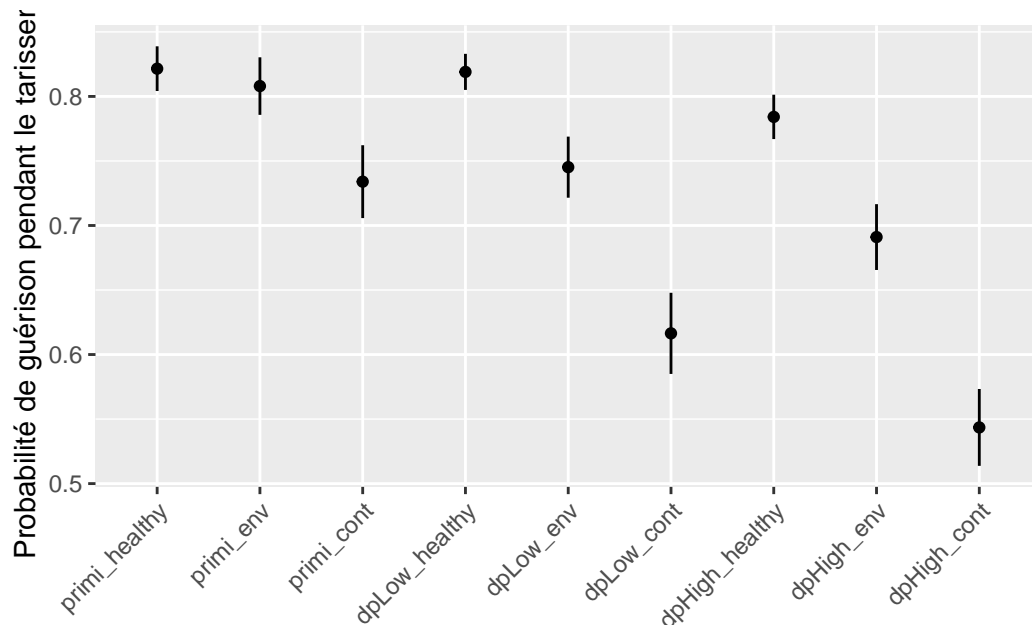
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

optimizer (Nelder-Mead) convergence code: 0 (OK)

Model failed to converge with $\max|\text{grad}| = 0.00243732$ (tol = 0.002, component 1)

```
prd <- marginalesffects::predictions(model = nxt_cure_full,
                                     newdata = mast_pred,
                                     re.form = NA, type = "response")

ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité de guérison pendant le tarissement") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```

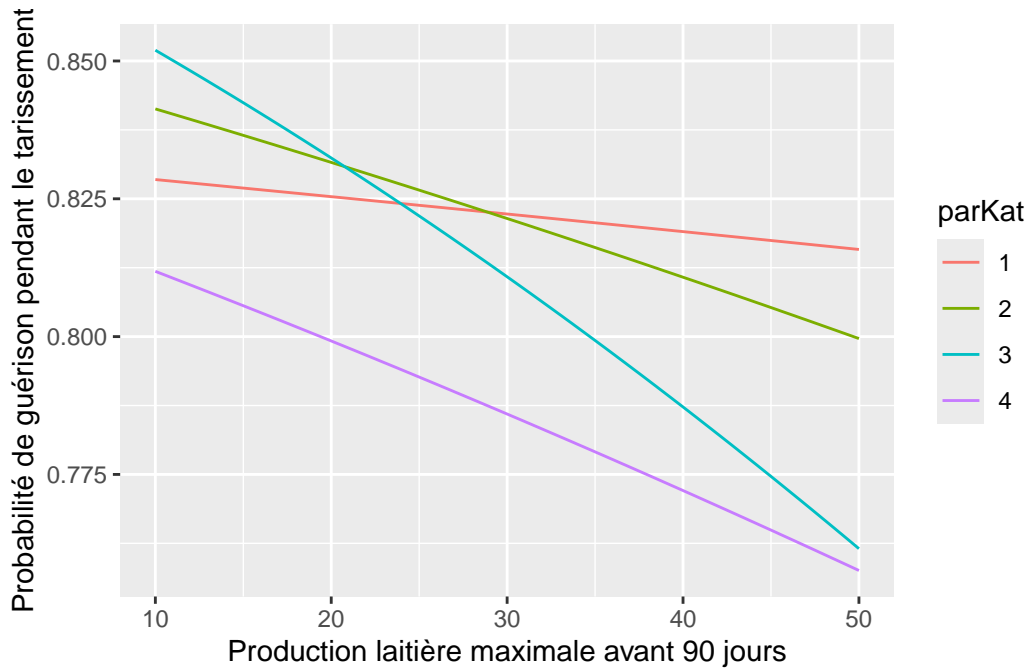



```

prd <- margineffects::predictions(model = nxt_cure_full,
                                newdata = milk_pred,
                                re.form = NA, type = "response")

ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +
  geom_line() +
  ylab("Probabilité de guérison pendant le tarissement") +
  xlab("Production laitière maximale avant 90 jours")

```



9.3 Probabilité de nouvelle infection

Modélisation de la probabilité de nouvelle infection pendant la période tarie.

```

nxt_ninf_full <- glmer(nxt_ninf ~ parKat +
                      milk_early_max_scaled_p1 +
                      milk_early_max_scaled_p2 +
                      milk_early_max_scaled_p3 +
                      milk_early_max_scaled_p4 +
                      mast_cat1 +
                      (1 | herd_id),
                      family = binomial(link = "logit"),

```

```
data = lac,
subset = n_ctrl_early > 0 & repro == 1 & dp_mis == 0)
```

fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```
summary(nxt_ninf_full)
```

Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]

Family: binomial (logit)

Formula:

```
nxt_ninf ~ parKat + milk_early_max_scaled_p1 + milk_early_max_scaled_p2 +
  milk_early_max_scaled_p3 + milk_early_max_scaled_p4 + mast_cat1 +
  (1 | herd_id)
```

Data: lac

Subset: n_ctrl_early > 0 & repro == 1 & dp_mis == 0

AIC	BIC	logLik	deviance	df.resid
37058.1	37198.0	-18513.0	37026.1	46319

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.2708	-0.4310	-0.3592	-0.3003	4.2155

Random effects:

Groups	Name	Variance	Std.Dev.
herd_id	(Intercept)	0.147	0.3835

Number of obs: 46335, groups: herd_id, 217

Fixed effects:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-1.94738	0.04108	-47.405	< 2e-16 ***
parKat1	-0.18344	0.04529	-4.051	5.11e-05 ***
parKat3	0.13050	0.04687	2.784	0.00536 **
parKat4	0.39458	0.04725	8.350	< 2e-16 ***
milk_early_max_scaled_p1	0.01947	0.03476	0.560	0.57544
milk_early_max_scaled_p2	0.03469	0.03101	1.119	0.26330
milk_early_max_scaled_p3	0.08212	0.03792	2.166	0.03034 *
milk_early_max_scaled_p4	0.01599	0.03708	0.431	0.66629
mast_cat1dpHigh_cont	1.17428	0.11073	10.605	< 2e-16 ***
mast_cat1dpHigh_env	0.79411	0.07797	10.185	< 2e-16 ***

```

mast_cat1dpHigh_healthy    0.27983    0.04585    6.103 1.04e-09 ***
mast_cat1dpLow_cont       0.89186    0.08546   10.436 < 2e-16 ***
mast_cat1dpLow_env        0.48163    0.05558    8.665 < 2e-16 ***
mast_cat1primi_cont       0.94106    0.08241   11.420 < 2e-16 ***
mast_cat1primi_env        0.47451    0.05390    8.804 < 2e-16 ***

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation matrix not shown by default, as $p = 15 > 12$.

Use `print(x, correlation=TRUE)` or

`vcov(x)` if you need it

fit warnings:

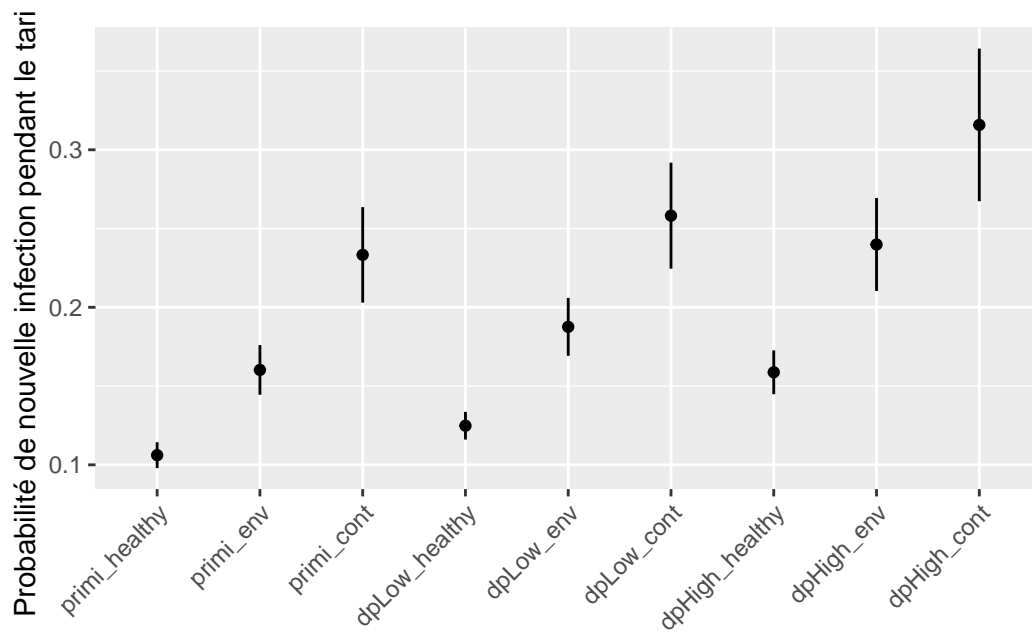
fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

```

prd <- marginalesffects::predictions(model = nxt_ninf_full,
                                     newdata = mast_pred,
                                     re.form = NA, type = "response")

ggplot(prd, aes(x = factor(mast_cat1, level = level_order), y = estimate)) +
  geom_point() +
  geom_segment(aes(x = mast_cat1, y = conf.low, xend = mast_cat1, yend = conf.high)) +
  xlab("") +
  ylab("Probabilité de nouvelle infection pendant le tarissement") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))

```



```

prd <- marginales::predictions(model = nxt_ninf_full,
                               newdata = milk_pred,
                               re.form = NA, type = "response")

ggplot(prd, aes(x = milk_early_max, y = estimate, col = parKat)) +
  geom_line()

```

