Main outcomes

S.P.L.M., Gustavo

2025-05-20

Table of contents


```
rm(list = ls())
graphics.off()
cat("\014")  # Clear any pending RStudio sessions or temporary files
```

```r
# Load functions from external script
source("helper_functions.R")

## Load necessary libraries
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4.9000     v readr     2.1.5
v forcats   1.0.0          v stringr   1.5.1
v ggplot2   3.5.1          v tibble    3.2.1
v lubridate 1.9.4          v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```r
library(readxl)
library(lubridate)
library(stringr)
library(purrr)
library(here)
```

```
here() starts at /Users/gustavosplmoura/Library/Mobile Documents/com~apple~CloudDocs/Medicina
```

```r
library(lme4)
```

```
Loading required package: Matrix

Attaching package: 'Matrix'

The following objects are masked from 'package:tidyr':

    expand, pack, unpack
```

```r
library(lmerTest)
```

```

Attaching package: 'lmerTest'
```

```
The following object is masked from 'package:lme4':

    lmer

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

    step
```

```
library(skimr)

# Read Files ----
## Codebooks
codebook_dvep <- read_excel(
    "Codebooks/codebook_dvep.xlsx",
    col_names = TRUE,
    col_types = NULL,
    na = c("", "NA", "NI", "UNK", "NASK", "ASKU", "INV"),
    trim_ws = TRUE,
    skip = 0, # Number of lines to skip before reading data
    n_max = Inf, # Maximum number of lines to read.
    guess_max = 1000
) %>%
    arrange(index)

codebook_structure  <- read_csv(
    "Codebooks/codebook_structure.csv",
    col_names = TRUE)
```

```
Rows: 34 Columns: 9
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): form_name_en, form_name_pt
dbl (7): repeating, eleg, V1, V2, V3, order, variable_count

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
codebook_ncit  <- read_csv(
    "Codebooks/codebook_ncit.csv",
    col_names = TRUE)
```

```
Rows: 330 Columns: 4
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): ncit_code, descriptive
dbl (2): type, medicamentos_comorbidades_complete

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
codebook_bia <- read_excel(
    "Codebooks/codebook_bia.xlsx",
    col_names = TRUE,
    col_types = NULL,
    na = c("", "NA", "NI", "UNK", "NASK", "ASKU", "INV"),
    trim_ws = TRUE,
    skip = 0, # Number of lines to skip before reading data
    n_max = Inf, # Maximum number of lines to read.
    guess_max = 1000
) %>%
    arrange(index)

## Data
data <- readRDS("Data_processed/data.rds")
data_bia <- readRDS("Data_processed/data_bia.rds")
data_bia_D1 <- readRDS("Data_processed/data_bia_D1.rds")
data_bia_D1_mean <- readRDS("Data_processed/data_bia_D1_mean.rds")
data_bia_D1_raw <- readRDS("Data_processed/data_bia_D1_raw.rds")
data_bia_D3 <- readRDS("Data_processed/data_bia_D3.rds")
data_bia_D3_mean <- readRDS("Data_processed/data_bia_D3_mean.rds")
data_bia_D3_raw <- readRDS("Data_processed/data_bia_D3_raw.rds")
data_bia_mean <- readRDS("Data_processed/data_bia_mean.rds")
data_d1_exclusive <- readRDS("Data_processed/data_d1_exclusive.rds")
data_filtered <- readRDS("Data_processed/data_filtered.rds")
data_filtered_seca <- readRDS("Data_processed/data_filtered_seca.rds")
I21_conditions_R <- readRDS("Data_processed/I21_conditions_R.rds")
I22_drugs_R <- readRDS("Data_processed/I22_drugs_R.rds")
I27_labs_R <- readRDS("Data_processed/I27_labs_R.rds")
I29_compliance_new <- readRDS("Data_processed/I29_compliance_new.rds")
I30_events_R <- readRDS("Data_processed/I30_events_R.rds")

## SUPERTIBBLE
data_instruments <- readRDS("Data_instruments/data_instruments.rds")
```

Variáveis de interesse

- (1 | record_id) + visit + allocation_group + age + sex + race + education +
- Comorbidities: hypertension + hypercholesterolemia + hypertrigliceridemia + insulin + drugs_w_loss + drugs_w_gain
- Anthro: abdomen + bmi + mean_bp_mean
- Questionnaires: whoqol_score_overall + ecap_score + evs_score + dass_score_stress + dass_score_anxiety + dass_score_depression + alcohol_significant + smoke_history + carbs_kcal + protein_kcal + fat_kcal + drugs_dose_change_yn +
- Adesão: completed_intervention, intervention_duration, education_years? cp_taking_as_directed_yn, cp_missed_dose_yn, cp_missed_dose_count, cp_discontinued_yn, cp_discontinued_n_days, cp_ran_out_of_drug_yn, cp_medication_confidence_sca

REDUCE TIBBLE FOR MODELLING

```
data_model <- data_filtered %>%
    select(
        record_id:sex,
        hypertension:ecap_score,
        abdomen, bmi, mean_bp_mean,
        resistance:evs_score,
        alcohol_dose,
        carbs_kcal, protein_kcal, fat_kcal, kcal,
        labs_crp:labs_alkp,
        labs_cholesterol:labs_quick_index
    )

saveRDS(
    data_model,
    "Data_processed/data_model.rds")

vars_to_keep <- names(data_model)

# Step 2: filter codebook_dvep
codebook_data_model <- codebook_dvep %>%
    filter(variable %in% vars_to_keep) %>%
    select(
        variable, label_pt, field_type, choices)

saveRDS(
```

```
    codebook_data_model,
    "Data_processed/codebook_data_model.rds")
```

SCALING

$$QUICKI = \frac{1}{log(insulin) + log(glucose)}$$

$$HOMA - IR = \frac{insulin * glucose}{405}$$

```
data_model_scaled <- data_model %>%
  mutate(across(
    .cols = c(
        duration_difference, age,
        whoqol_score_overall, dass_score_depression, dass_score_anxiety,
        dass_score_stress, ecap_score,
        abdomen, bmi, mean_bp_mean,
        resistance, reactance,
        handgrip, evs_score, alcohol_dose,
        carbs_kcal, protein_kcal, fat_kcal, kcal,
        labs_crp, labs_ast, labs_alt, labs_ggt, labs_alkp,
        labs_cholesterol, labs_ldl, labs_hba1c, labs_triglycerides,
        labs_hdl, labs_glucose, labs_insulin, labs_homa_ir
    ),
    .fns = ~ as.numeric(scale(.))
  ))

scaling_params <- data_model %>%
  summarise(across(
    .cols = c(
      duration_difference, age,
      whoqol_score_overall, dass_score_depression, dass_score_anxiety,
      dass_score_stress, ecap_score,
      abdomen, bmi, mean_bp_mean,
      resistance, reactance,
      handgrip, evs_score, alcohol_dose,
      carbs_kcal, protein_kcal, fat_kcal,
      labs_crp, labs_ast, labs_alt, labs_ggt, labs_alkp,
      labs_cholesterol, labs_ldl, labs_hba1c, labs_triglycerides,
      labs_hdl, labs_glucose, labs_insulin, labs_homa_ir
```

```
    ),
    list(mean = ~mean(., na.rm = TRUE), sd = ~sd(., na.rm = TRUE))
  )) %>%
  pivot_longer(everything(),
               names_to = c("variable", ".value"),
               names_sep = "_(?=[^_]+$)",  #   this fixes the splitting
               names_transform = list(.value = as.character))
```

```
# THEN, FOR NEW DATA
scale_with_params <- function(new_data, params) {
  for (i in seq_len(nrow(params))) {
    var <- params$variable[i]
    mean_val <- params$mean[i]
    sd_val <- params$sd[i]
    if (var %in% names(new_data)) {
      new_data[[var]] <- (new_data[[var]] - mean_val) / sd_val
    }
  }
  return(new_data)
}

new_data_scaled <- scale_with_params(new_data, scaling_params)
```

ÂNGULO DE FASE

Filtrando D1 and D3

```
pha_redcap <- data_model_scaled %>%
    filter(
        visit %in% c(1, 3)
    ) %>%
    mutate(
        compliance_score_visit = case_when(
            visit == 3 & completed_intervention == "Não" ~ 0,
            TRUE ~ compliance_score_visit
        )
    )
```

All variables

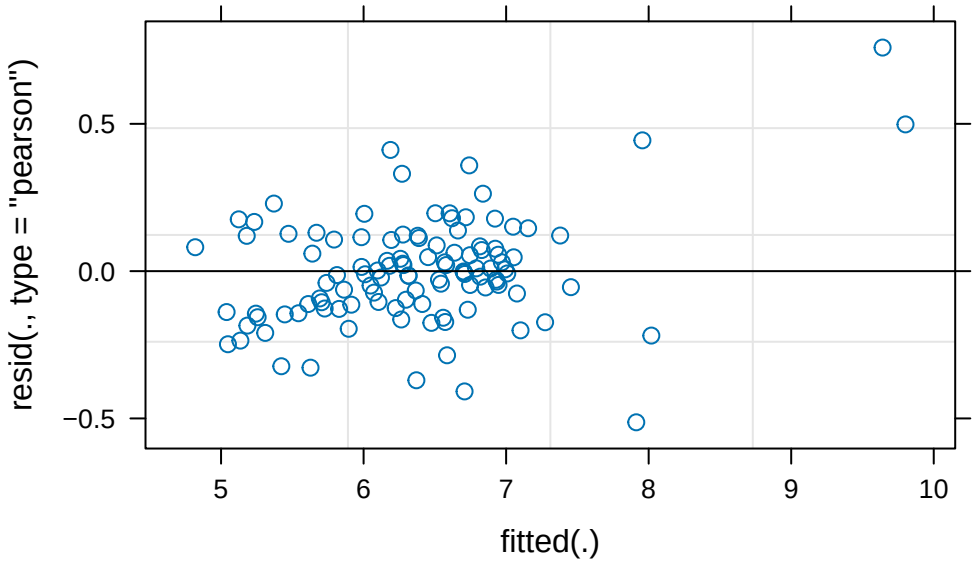
```
library(lme4)
library(lmerTest) # Adds p-values to summary()
```

```
pha_1 <- lmer(phase_angle ~ (1 | record_id) + visit + allocation_group + completed_interventi

summary(pha_1)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula:
phase_angle ~ (1 | record_id) + visit + allocation_group + completed_intervention +
    duration_difference + age + sex + hypertension + hypercholesterolemia +
    hypertrigliceridemia + drugs_w_loss + drugs_w_gain + mean_bp_mean +
    handgrip + evs_score + alcohol_dose + kcal + labs_crp + labs_alt +
    labs_ggt + labs_ldl + labs_triglycerides + labs_hdl + labs_quick_index
   Data: pha_redcap

REML criterion at convergence: 267.6

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.56730 -0.34461 -0.02411  0.32517  2.31790

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.7246   0.8513
 Residual              0.1072   0.3274
Number of obs: 111, groups:  record_id, 73

Fixed effects:
                           Estimate Std. Error       df t value Pr(>|t|)
(Intercept)                 6.80702    1.14545 54.72940   5.943 2.02e-07 ***
visit                       0.04367    0.05999 39.61741   0.728 0.470955
allocation_groupGrupo B    -0.08294    0.23322 58.16739  -0.356 0.723419
completed_interventionSim   0.13431    0.29025 62.80942   0.463 0.645154
duration_difference         0.06796    0.07531 35.47525   0.902 0.372926
age                        -0.15807    0.13498 60.83001  -1.171 0.246148
sexMasculino                0.09460    0.47396 83.73697   0.200 0.842285
hypertension1              -0.41606    0.30355 64.58323  -1.371 0.175222
```

```
hypercholesterolemia1      0.19062    0.27727 69.89339   0.687 0.494049
hypertrigliceridemia1     -0.34038    0.27433 76.54222  -1.241 0.218477
drugs_w_loss1             -0.25530    0.26664 55.83333  -0.957 0.342447
drugs_w_gain1             -0.66202    0.56551 57.08713  -1.171 0.246602
mean_bp_mean               0.18737    0.09217 82.46459   2.033 0.045275 *
handgrip                   0.02906    0.12915 84.31928   0.225 0.822503
evs_score                  0.02820    0.06771 51.41295   0.417 0.678762
alcohol_dose               0.25024    0.06910 55.31084   3.621 0.000637 ***
kcal                       0.08493    0.07469 64.71213   1.137 0.259686
labs_crp                   0.12545    0.05608 35.78386   2.237 0.031613 *
labs_alt                  -0.07563    0.08133 60.52700  -0.930 0.356110
labs_ggt                   0.02983    0.10259 78.08587   0.291 0.771980
labs_ldl                  -0.11607    0.09391 52.71421  -1.236 0.221990
labs_triglycerides         0.04481    0.07352 35.85978   0.610 0.545998
labs_hdl                  -0.02374    0.08890 81.65491  -0.267 0.790104
labs_quick_index          -1.07713    3.09369 48.37580  -0.348 0.729223
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Correlation matrix not shown by default, as p = 24 > 12.
Use print(x, correlation=TRUE)  or
    vcov(x)        if you need it
```

```r
plot(pha_1)
```



A modelagem estatística foi realizada por meio de modelos lineares mistos com intercepto aleatório por participante (record_id). Todas as variáveis contínuas foram previamente padronizadas (média = 0, desvio-padrão = 1), exceto o índice QUICKI, mantido em sua unidade original para fins de interpretabilidade clínica. A variável duration_difference, que representa o desvio absoluto em dias da duração planejada da intervenção (90 dias), foi ajustada para zero na visita basal e posteriormente padronizada. O modelo pha_2 incluiu 24 preditores fixos e foi ajustado utilizando o método de máxima verossimilhança restrita (REML), com inclusão da variância intraindividual no componente aleatório.

Neste modelo, observou-se que a pressão arterial média (mean_bp_mean) foi positivamente associada ao ângulo de fase (= 0.19, p = 0.045), sugerindo que níveis mais elevados de pressão podem estar relacionados a um melhor estado funcional da membrana celular. A dose de álcool (alcohol_dose) foi o preditor com maior significância estatística (= 0.25, p < 0.001), com associação positiva ao ângulo de fase; esse achado deve ser interpretado com cautela, pois pode refletir fatores comportamentais ou nutricionais não capturados diretamente no modelo. Além disso, a proteína C reativa (labs_crp) apresentou associação positiva marginalmente significativa (= 0.13, p = 0.031), o que pode indicar uma complexa interação entre inflamação subclínica e integridade da membrana celular.

Outros preditores como visit, allocation_group, completed_intervention, duration_difference e variáveis laboratoriais como labs_ldl, labs_triglycerides e labs_quick_index não apresentaram associações estatisticamente significativas. A variância do intercepto aleatório por record_id permaneceu elevada (0.72), reforçando a relevância da estrutura longitudinal dos dados. O critério de REML obtido foi 267.6, indicando um ajuste superior ao modelo anterior

(REML = 275), e sugerindo avanço na parcimônia e na qualidade do modelo após a redução de preditores.

Próximos passos sugeridos: - Considerar remoção de variáveis com p > 0.5 e sem base teórica. - Avaliar o impacto de variáveis correlacionadas (e.g., `bmi`, `abdomen`). - Testar modelos reduzidos orientados por AIC ou p-valor.

Interpretation clarity - With all predictors scaled, effect sizes are in SD units. labs_quick_index still in raw units: its coefficient = change in phase angle per unit change in QUICKI, which is very interpretable (and appropriate).

Random effects - You can tell that including a random intercept was appropriate by comparing the random effect variance to the residual variance. The random intercept variance for record_id is 0.6846, which represents the variability between participants. The residual variance is 0.1188, which reflects variability within participants (i.e., unexplained by the model and random noise). Now, the random intercept variance is much larger than the residual variance. This indicates that a substantial portion of the total variance in your outcome (phase_angle) is due to differences between individuals, not just random fluctuation within the same person over time. Hence, accounting for individual-level differences via a random intercept helps the model better estimate the effects of your fixed variables by controlling for this inter-individual baseline shift. Summary: Since Var(Intercept for record_id) > Var(Residual), this suggests strong subject-level variation, meaning including (1 | record_id) was a statistically sound choice.

Refinements

1. Check for multicollinearity
2. High number of predictors (p = 30+) vs. N = 111 to your sample size. This increases risk of: Overfitting, Inflated standard errors, Instability in estimates. Suggestion: Run a stepwise reduction or penalized regression (e.g., LASSO via glmnet) to select the most stable subset.
3. Check model fit and explained variance.
    - Use performance::check_model() to assess residuals, normality, and homoscedasticity.
    - Consider using marginal and conditional R^2 to evaluate model fit.
4. Plot residuals and check assumptions: Homoscedasticity, Normality of residuals, Influential observations.
5. Refine the role of visit. Currently, visit is not significant, but the study is longitudinal. So ask: Is visit best treated as a linear trend (numeric)? Would a factor with interaction be more appropriate? Would a random slope for visit help capture subject-specific trajectories?

1. Check for multicollinearity

Because you have many predictors (30+), and some may be redundant or highly correlated, collinearity is your first enemy. You've handled insulin/glucose/HOMA/QUICKI well. But still check VIFs for collinearity among other variables (like BMI and abdomen, or macronutrients - total energy intake). High collinearity can: (1) Inflate standard errors (making real effects look non-significant), (2) Obscure model interpretation, (3) Affect convergence and coefficient stability. How? Use performance::check_collinearity():

```
library(performance)
check_collinearity(pha_1)
```

```
# Check for Multicollinearity

Low Correlation

                   Term  VIF   VIF 95% CI Increased SE Tolerance
                  visit 2.64 [2.13, 3.38]         1.63      0.38
      allocation_group 1.24 [1.10, 1.57]         1.11      0.81
 completed_intervention 1.54 [1.31, 1.93]         1.24      0.65
    duration_difference 2.07 [1.71, 2.63]         1.44      0.48
                    age 1.67 [1.40, 2.10]         1.29      0.60
                    sex 2.41 [1.96, 3.07]         1.55      0.41
           hypertension 1.51 [1.29, 1.90]         1.23      0.66
  hypercholesterolemia 1.64 [1.39, 2.07]         1.28      0.61
  hypertrigliceridemia 1.71 [1.44, 2.15]         1.31      0.58
           drugs_w_loss 1.20 [1.08, 1.54]         1.10      0.83
           drugs_w_gain 1.13 [1.03, 1.52]         1.06      0.88
           mean_bp_mean 1.84 [1.53, 2.32]         1.36      0.54
               handgrip 1.94 [1.61, 2.46]         1.39      0.51
              evs_score 1.31 [1.15, 1.65]         1.14      0.76
           alcohol_dose 1.48 [1.27, 1.85]         1.22      0.68
                   kcal 1.67 [1.41, 2.10]         1.29      0.60
               labs_crp 1.22 [1.09, 1.56]         1.10      0.82
               labs_alt 1.48 [1.27, 1.85]         1.22      0.68
               labs_ggt 1.65 [1.39, 2.07]         1.28      0.61
               labs_ldl 1.69 [1.42, 2.12]         1.30      0.59
    labs_triglycerides 1.48 [1.27, 1.86]         1.22      0.67
              labs_hdl 1.32 [1.15, 1.66]         1.15      0.76
      labs_quick_index 1.23 [1.09, 1.56]         1.11      0.82
 Tolerance 95% CI
     [0.30, 0.47]
```

```
[0.64, 0.91]
[0.52, 0.76]
[0.38, 0.59]
[0.48, 0.71]
[0.33, 0.51]
[0.53, 0.77]
[0.48, 0.72]
[0.46, 0.70]
[0.65, 0.93]
[0.66, 0.97]
[0.43, 0.65]
[0.41, 0.62]
[0.61, 0.87]
[0.54, 0.79]
[0.48, 0.71]
[0.64, 0.92]
[0.54, 0.79]
[0.48, 0.72]
[0.47, 0.70]
[0.54, 0.79]
[0.60, 0.87]
[0.64, 0.92]
```

```
# r2(pha_1)  # Marginal and conditional R²
```

Interpretation of collinearity results: No concerning multicollinearity (all VIFs < 3).

Variance Inflation Factor (VIF) measures how much the variance (i.e., the standard error squared) of a regression coefficient is inflated due to collinearity with other predictors. In other words, it tells you how strongly one predictor is linearly related to the others.

VIF Value	Interpretation
1	No correlation with other variables
1–2	Low correlation, no concern
2–5	Moderate correlation — **keep an eye**
5–10	High correlation — **potential problem**
>10	Severe multicollinearity — **likely an issue**

2. Reduce the model

Your model currently includes 30 fixed effects, which may limit statistical power and interpretability due to overfitting.

A abordagem mais sensata para reduzir o modelo depende diretamente do seu objetivo principal.

Se o seu foco for **predição ou performance do modelo**, então a estratégia **data-driven** é mais apropriada, como:

- **(1) Seleção stepwise backward** com base em critérios de informação como AIC ou BIC (AIC favorece modelos com melhor ajuste, BIC penaliza mais a complexidade).
- **LASSO (via `glmnet`)**, que impõe uma penalização e tende a selecionar um subconjunto estável de variáveis, pode ser especialmente útil quando o número de preditores é alto e há colinearidade moderada.

Entretanto, se o objetivo for **interpretação e inferência causal ou explicativa** (como é comum em ensaios clínicos e estudos de intervenção), a melhor abordagem é:

- **(2) Simplificação orientada pela teoria e plausibilidade clínica**, removendo variáveis que claramente não contribuem de forma significativa, que se sobrepõem a outras medidas (ex: manter apenas `quick_index` e excluir glicemia/insulina), ou que apresentem comportamento instável nos modelos (como efeitos colineares ou variações negativas no sinal da estimativa ao longo das versões do modelo).

Quando eu menciono **"variações negativas no sinal da estimativa ao longo das versões do modelo"**, estou me referindo ao comportamento instável dos coeficientes de algumas variáveis à medida que você modifica o modelo — por exemplo, adicionando ou removendo preditores.

Imagine que, em um modelo mais simples, a variável `bmi` tem um coeficiente **positivo**, sugerindo que quanto maior o IMC, maior o ângulo de fase. Mas ao incluir outras variáveis correlacionadas (como `abdomen` ou `resistance`), o coeficiente de `bmi` se torna **negativo** ou **não significativo**. Essa mudança de sinal pode indicar:

- **Colinearidade**: `bmi` e `abdomen`, por exemplo, podem estar explicando o mesmo componente corporal.
- **Falta de robustez**: a interpretação do efeito da variável muda conforme outras são incluídas, dificultando conclusões consistentes.
- **Overfitting ou ajuste instável**, especialmente em amostras pequenas.

Detectar esse comportamento é um sinal de que a variável pode estar redundante ou que há necessidade de ajustes — por exemplo, usar apenas uma das variáveis correlacionadas ou aplicar técnicas de regularização.

Portanto, **"variação no sinal da estimativa"** não é sobre valor negativo em si, mas sobre **mudança de direção do efeito estimado**, o que enfraquece a confiança na interpretação daquele preditor.

Uma boa estratégia híbrida é:

1. **Fixar um conjunto mínimo de variáveis-chave teóricas** (ex: sexo, idade, grupo, tempo).
2. **Aplicar redução stepwise nos demais termos**, guiando-se por BIC (se você deseja maior parcimônia) ou por LASSO.
3. **Comparar modelos com ANOVA** e gráficos de resíduos para garantir que a simplificação não deteriora o ajuste.

Essa abordagem permite alcançar um equilíbrio entre interpretabilidade e estabilidade do modelo.

3. Check model fit

A avaliação dos pressupostos do modelo pha_1 foi realizada por meio da função check_model() do pacote performance, a qual indicou que, em linhas gerais, o modelo apresenta adequação estatística razoável. O gráfico de Posterior Predictive Check mostra boa sobreposição entre a densidade dos valores observados e os valores preditos pelo modelo, indicando adequada capacidade preditiva.

A verificação da linearidade dos resíduos frente aos valores ajustados revelou um desvio leve da horizontalidade na extremidade superior, sugerindo possível não linearidade em valores mais altos do desfecho. A homogeneidade da variância (homoscedasticidade) também apresentou leve violação, com aumento da variância dos resíduos em valores preditos mais altos — característica que pode afetar a precisão das estimativas nessas faixas.

A análise de observações influentes (gráfico de alavancagem vs. resíduos padronizados) identificou algumas observações com leve influência (por exemplo, IDs 10, 69, 70, 75, 109), mas nenhuma ultrapassando os limiares clássicos de alavancagem ou resíduos padronizados extremos.

A colinearidade foi considerada baixa, com todos os fatores de inflação da variância (VIF) abaixo de 5, o que indica ausência de multicolinearidade severa entre os preditores. O gráfico de normalidade dos resíduos mostra aderência razoável à distribuição normal, com pequenas desvios nas caudas, o que é aceitável para modelos mistos com tamanho amostral moderado. Por fim, a distribuição dos efeitos aleatórios (record_id) se aproximou da normalidade, com leve assimetria nas extremidades, reforçando que o intercepto aleatório foi uma escolha apropriada para capturar a variabilidade entre indivíduos.

5. Refine the role of visit

lmer(phase_angle ~ visit + (visit | record_id) + …

Error: number of observations (=111) <= number of random effects (=146) for term (visit | record_id)

lmer(phase_angle ~ visit + (1 + visit || record_id) + …

boundary (singular) fit: see help('isSingular')

Reduce the Model

pha_2

```
pha_2 <- lmer(phase_angle ~ (1 | record_id) + visit + allocation_group + age + sex + bmi + me

summary(pha_2)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: phase_angle ~ (1 | record_id) + visit + allocation_group + age +
    sex + bmi + mean_bp_mean + handgrip + evs_score + kcal +
    labs_crp + labs_alt + labs_ggt + labs_ldl + labs_triglycerides +
    labs_hdl + labs_quick_index
   Data: pha_redcap

REML criterion at convergence: 275.6

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.1820 -0.3023 -0.0278  0.2559  3.6054

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.5811   0.7623
 Residual              0.1780   0.4219
Number of obs: 111, groups:  record_id, 73

Fixed effects:
                          Estimate Std. Error        df t value Pr(>|t|)
(Intercept)               7.082490   1.176374 77.136367   6.021 5.51e-08 ***
visit                    -0.004178   0.060460 47.468510  -0.069   0.9452
allocation_groupGrupo B -0.252382   0.214787 64.288710  -1.175   0.2443
age                      -0.187470   0.105337 65.806508  -1.780   0.0797 .
sexMasculino              0.335406   0.456188 84.261799   0.735   0.4642
bmi                       0.010733   0.097621 89.559042   0.110   0.9127
mean_bp_mean              0.174320   0.096937 93.792525   1.798   0.0753 .
```

```
handgrip                 0.077031   0.130670 92.345158   0.590   0.5570
evs_score                0.035996   0.073279 74.994735   0.491   0.6247
kcal                     0.026568   0.080627 77.299814   0.330   0.7427
labs_crp                 0.080876   0.065164 51.402275   1.241   0.2202
labs_alt                -0.001230   0.085298 88.062468  -0.014   0.9885
labs_ggt                 0.026733   0.096885 82.285184   0.276   0.7833
labs_ldl                -0.064943   0.090663 92.475432  -0.716   0.4756
labs_triglycerides       0.015440   0.079832 62.011443   0.193   0.8473
labs_hdl                 0.002898   0.092120 93.997884   0.031   0.9750
labs_quick_index        -1.886146   3.435802 75.554696  -0.549   0.5846
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Correlation matrix not shown by default, as p = 17 > 12.
Use print(x, correlation=TRUE)  or
    vcov(x)        if you need it
```

AIC(pha_1, pha_2) df AIC pha_1 26 319.5554 pha_2 19 313.6283

BIC(pha_1, pha_2) df BIC pha_1 26 390.0032 pha_2 19 365.1094

A comparação entre os modelos pelo critério de informação de Akaike (AIC) e o critério bayesiano de informação (BIC) favoreceu o modelo reduzido (pha_2), que apresentou valores mais baixos de AIC (313,6 vs. 319,6) e BIC (365,1 vs. 390,0) em relação ao modelo completo (pha_1). Isso sugere que a simplificação do modelo resultou em melhor equilíbrio entre ajuste e complexidade, mesmo com a perda de significância estatística de algumas covariáveis previamente relevantes.

pha_3

```
pha_3 <- lmer(phase_angle ~ (1 | record_id) + visit + allocation_group + completed_interventi

summary(pha_3)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula:
phase_angle ~ (1 | record_id) + visit + allocation_group + completed_intervention +
    duration_difference + age + sex + hypertension + hypercholesterolemia +
    hypertrigliceridemia + drugs_w_loss + drugs_w_gain + mean_bp_mean +
```

```
    handgrip + evs_score + alcohol_dose + kcal + labs_crp + labs_ldl +
    labs_quick_index
   Data: pha_redcap

REML criterion at convergence: 256.2

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.67542 -0.37054 -0.01038  0.28131  2.47511

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.6800   0.8246
 Residual              0.1089   0.3300
Number of obs: 111, groups:  record_id, 73

Fixed effects:
                           Estimate Std. Error       df t value Pr(>|t|)
(Intercept)                 6.73199    1.12618 62.54202   5.978 1.18e-07 ***
visit                       0.04629    0.05831 39.70014   0.794  0.43198
allocation_groupGrupo B    -0.08824    0.22558 60.66392  -0.391  0.69704
completed_interventionSim   0.09468    0.27813 63.80928   0.340  0.73466
duration_difference         0.06221    0.07418 37.43891   0.839  0.40701
age                        -0.14627    0.13082 63.16447  -1.118  0.26776
sexMasculino                0.05360    0.44022 84.14001   0.122  0.90339
hypertension1              -0.37768    0.29365 65.31066  -1.286  0.20294
hypercholesterolemia1       0.13976    0.26480 72.33121   0.528  0.59926
hypertrigliceridemia1      -0.21111    0.23292 62.70194  -0.906  0.36822
drugs_w_loss1              -0.25216    0.25896 58.72682  -0.974  0.33419
drugs_w_gain1              -0.64500    0.54712 60.02599  -1.179  0.24308
mean_bp_mean                0.18141    0.08674 82.54798   2.092  0.03955 *
handgrip                    0.04801    0.12342 88.77623   0.389  0.69823
evs_score                   0.03655    0.06602 54.80914   0.554  0.58207
alcohol_dose                0.24846    0.06783 59.87919   3.663  0.00053 ***
kcal                        0.09010    0.07346 72.69290   1.227  0.22393
labs_crp                    0.11992    0.05502 40.88103   2.180  0.03509 *
labs_ldl                   -0.09732    0.08787 60.12632  -1.108  0.27247
labs_quick_index           -0.90144    3.03633 57.49316  -0.297  0.76762
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Correlation matrix not shown by default, as p = 20 > 12.
Use print(x, correlation=TRUE)  or
    vcov(x)        if you need it
```

```
check_collinearity(pha_3)
```

```
# Check for Multicollinearity

Low Correlation

                   Term  VIF   VIF 95% CI Increased SE Tolerance
                  visit 2.46 [1.98, 3.18]         1.57      0.41
       allocation_group 1.22 [1.08, 1.58]         1.11      0.82
 completed_intervention 1.49 [1.27, 1.89]         1.22      0.67
    duration_difference 1.99 [1.63, 2.54]         1.41      0.50
                    age 1.65 [1.39, 2.11]         1.29      0.60
                    sex 2.20 [1.79, 2.83]         1.48      0.45
           hypertension 1.50 [1.27, 1.90]         1.22      0.67
   hypercholesterolemia 1.58 [1.34, 2.02]         1.26      0.63
   hypertrigliceridemia 1.30 [1.14, 1.67]         1.14      0.77
           drugs_w_loss 1.20 [1.07, 1.57]         1.10      0.83
           drugs_w_gain 1.12 [1.03, 1.56]         1.06      0.89
           mean_bp_mean 1.65 [1.39, 2.10]         1.29      0.61
               handgrip 1.84 [1.52, 2.35]         1.36      0.54
              evs_score 1.25 [1.11, 1.62]         1.12      0.80
           alcohol_dose 1.44 [1.23, 1.83]         1.20      0.69
                   kcal 1.62 [1.36, 2.06]         1.27      0.62
               labs_crp 1.17 [1.05, 1.55]         1.08      0.86
               labs_ldl 1.50 [1.28, 1.91]         1.23      0.67
       labs_quick_index 1.19 [1.07, 1.56]         1.09      0.84
 Tolerance 95% CI
     [0.31, 0.51]
     [0.63, 0.92]
     [0.53, 0.79]
     [0.39, 0.61]
     [0.48, 0.72]
     [0.35, 0.56]
     [0.53, 0.78]
     [0.50, 0.75]
     [0.60, 0.88]
     [0.64, 0.93]
     [0.64, 0.98]
```

```
      [0.48, 0.72]
      [0.42, 0.66]
      [0.62, 0.90]
      [0.55, 0.81]
      [0.48, 0.73]
      [0.65, 0.95]
      [0.52, 0.78]
      [0.64, 0.94]
```

```
AIC(pha_1, pha_2, pha_3)
```

```
      df      AIC
pha_1 26 319.5554
pha_2 19 313.6283
pha_3 22 300.2185
```

```
BIC(pha_1, pha_2, pha_3)
```

```
      df      BIC
pha_1 26 390.0032
pha_2 19 365.1094
pha_3 22 359.8281
```

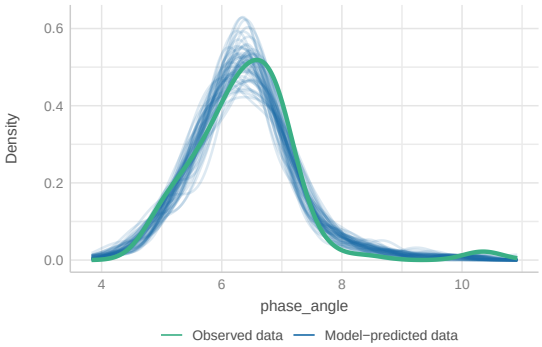
```
r2(pha_3)
```

```
# R2 for Mixed Models

  Conditional R2: 0.896
     Marginal R2: 0.250
```

```
plots <- performance::check_model(pha_3)
print(plots)
```

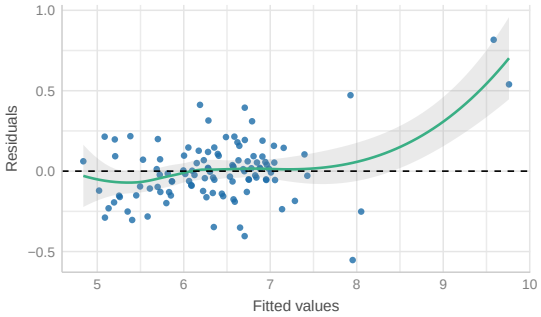
Posterior Predictive Check

Model−predicted lines should resemble observed data line



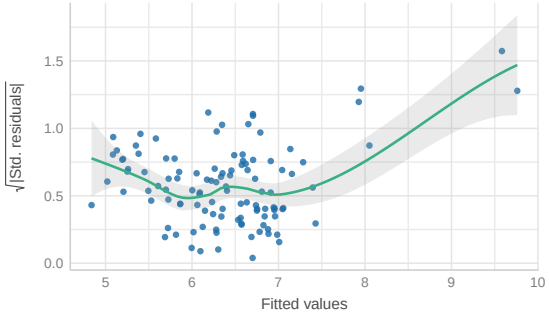
Linearity

Reference line should be flat and horizontal



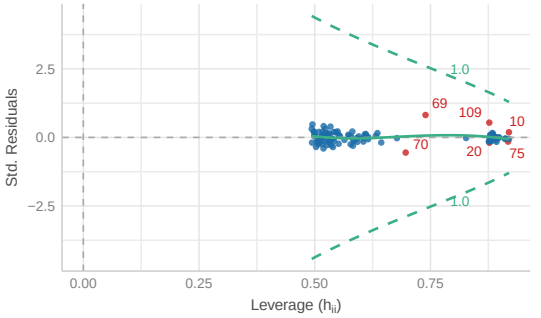
Homogeneity of Variance

Reference line should be flat and horizontal



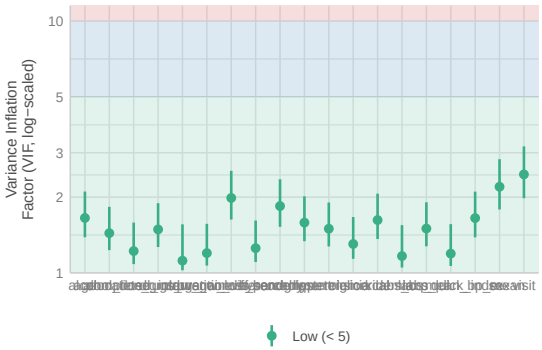
Influential Observations

Points should be inside the contour lines



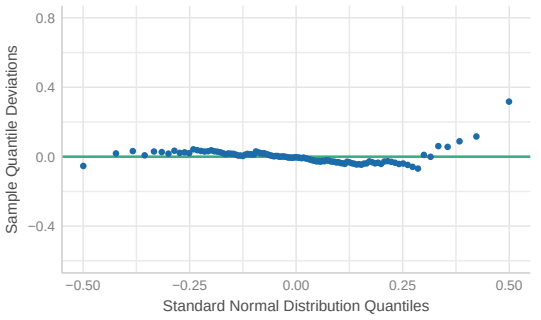
Collinearity

High collinearity (VIF) may inflate parameter uncertainty



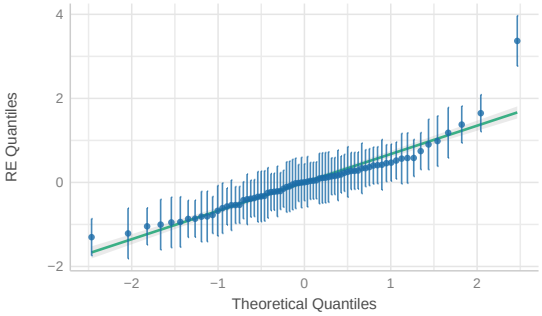
Normality of Residuals

Dots should fall along the line



Normality of Random Effects (record_id)

Dots should be plotted along the line



Três modelos hierárquicos foram comparados utilizando os critérios de informação AIC e BIC. O modelo pha_3, contendo 22 preditores fixos, apresentou os menores valores de AIC (300,2) e BIC (359,8), superando tanto o modelo completo pha_1 (AIC = 319,6; BIC = 390,0) quanto o modelo reduzido pha_2 (AIC = 313,6; BIC = 365,1). Esses resultados indicam que pha_3 oferece o melhor equilíbrio entre qualidade de ajuste e complexidade do modelo, sendo, portanto, o modelo preferido para interpretação final dos resultados.

Uma versão reduzida do modelo foi desenvolvida com o objetivo de aprimorar o equilíbrio entre parcimônia e desempenho preditivo. A seleção das variáveis foi orientada tanto por critérios teóricos quanto estatísticos, priorizando preditores com plausibilidade clínica e contribuições informativas nas versões anteriores. O novo modelo (pha_final) manteve o intercepto aleatório por participante (record_id) e incluiu 20 preditores fixos, resultando em um critério de máxima verossimilhança restrita (REML) inferior ao dos modelos anteriores (REML = 256,2), indicando melhora no ajuste.

As variáveis mean_bp_mean ($p = 0.040$), alcohol_dose ($p < 0.001$) e labs_crp ($p = 0.035$) mantiveram associação estatisticamente significativa com o ângulo de fase, mesmo após o ajuste multivariado, o que reforça a robustez dessas associações. A variável visit foi mantida como efeito fixo para controle do efeito temporal, embora não tenha mostrado significância estatística ($p = 0.432$). A variância do intercepto aleatório ($^2 = 0.68$) permaneceu relevante, o que confirma a presença de heterogeneidade entre os participantes e a adequação do uso de um modelo misto.

A análise multicolinearidade mostrou valores de VIF < 3 para todos os preditores, descartando problemas relevantes de colinearidade. Os diagnósticos do modelo indicaram distribuição adequada dos resíduos, ausência de observações altamente influentes e normalidade satisfatória dos efeitos aleatórios, corroborando a adequação do modelo ajustado.

A avaliação dos pressupostos do modelo pha_final foi realizada com base em gráficos diagnósticos. O gráfico de densidade ("Posterior Predictive Check") indicou que os valores preditos se alinharam adequadamente com a distribuição observada do ângulo de fase. Os resíduos padronizados apresentaram distribuição aproximadamente normal, conforme demonstrado no gráfico de quantis teóricos dos efeitos aleatórios e no gráfico de normalidade dos resíduos, reforçando a adequação do modelo misto com intercepto aleatório por participante.

A homogeneidade da variância apresentou leve tendência de heterocedasticidade nas extremidades do ajuste, mas sem padrão grave de violação. A linearidade foi, em geral, respeitada, embora a tendência suavizada indique alguma curvatura para valores mais altos do desfecho. O gráfico de observações influentes mostrou que todos os pontos estão dentro dos limites de influência padronizada, indicando ausência de outliers com alavancagem elevada.

Por fim, a análise de colinearidade mostrou que todos os preditores apresentaram VIF abaixo de 3, descartando preocupações com multicolinearidade. Dessa forma, os pressupostos do modelo foram globalmente atendidos, validando a interpretação dos coeficientes estimados.

O modelo apresentou R^2 marginal de 0,250, indicando que as variáveis fixas explicam 25% da variância do desfecho, e R^2 condicional de 0,896, refletindo a alta variabilidade explicada quando se considera a estrutura aleatória intra-individual, compatível com o delineamento longitudinal do estudo.

PCR

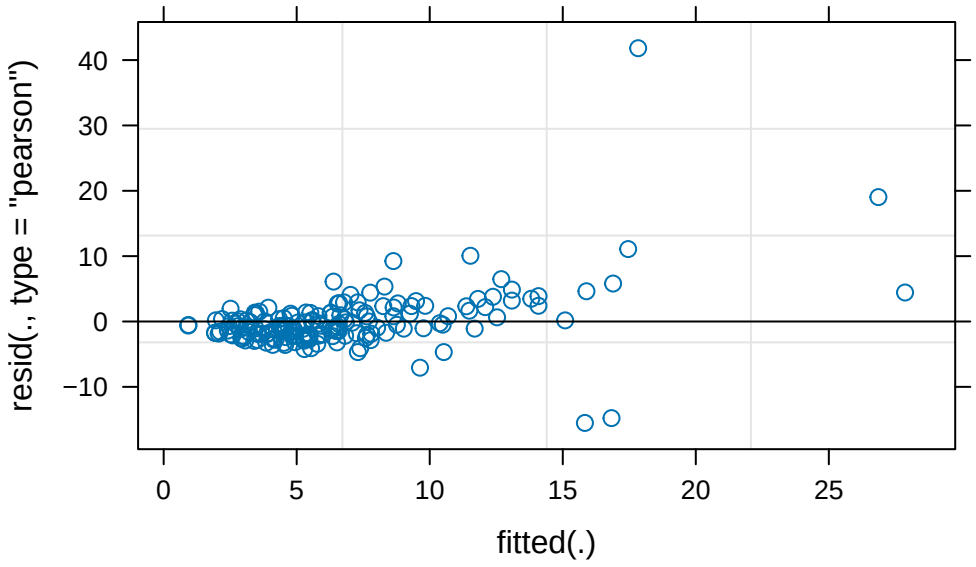
1

```r
model3 <- lmer(labs_crp ~ visit + allocation_group + (1 | record_id), data = data_filtered)

summary(model3)
```

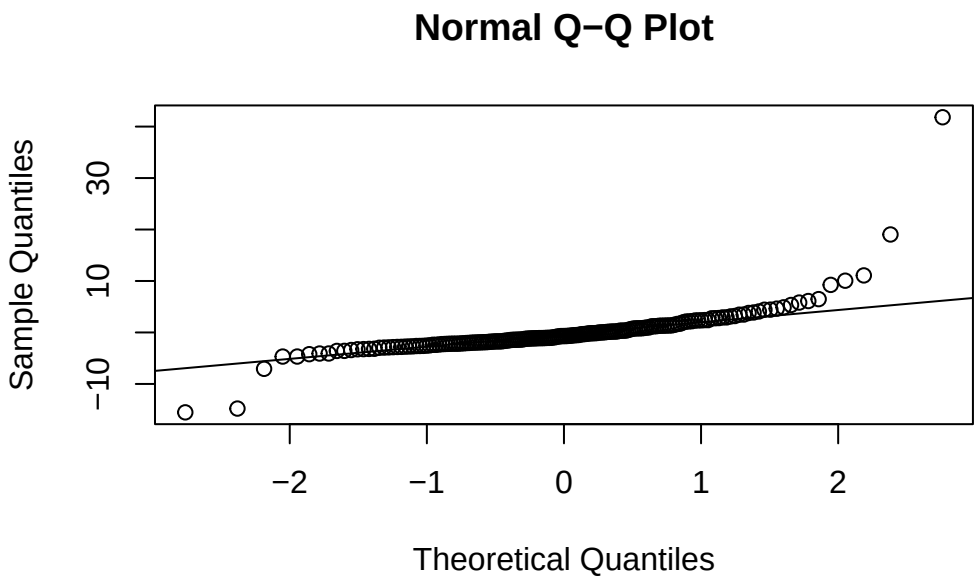
```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: labs_crp ~ visit + allocation_group + (1 | record_id)
   Data: data_filtered

REML criterion at convergence: 1163.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-2.8000 -0.3580 -0.1254  0.2196  7.5431

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 27.00    5.196
 Residual              30.73    5.544
Number of obs: 174, groups:  record_id, 75

Fixed effects:
                        Estimate Std. Error       df t value Pr(>|t|)
(Intercept)               8.2417     1.4442 159.2172   5.707 5.48e-08 ***
visit                    -1.0011     0.5504 109.9562  -1.819   0.0716 .
allocation_groupGrupo B   0.4793     1.4930  68.5534   0.321   0.7492
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) visit
```

```
visit       -0.689
allctn_grGB -0.543  0.051
```

```
plot(model3)  # Residuals vs. fitted
```




```
qqnorm(resid(model3)); qqline(resid(model3))  # Normality check
```



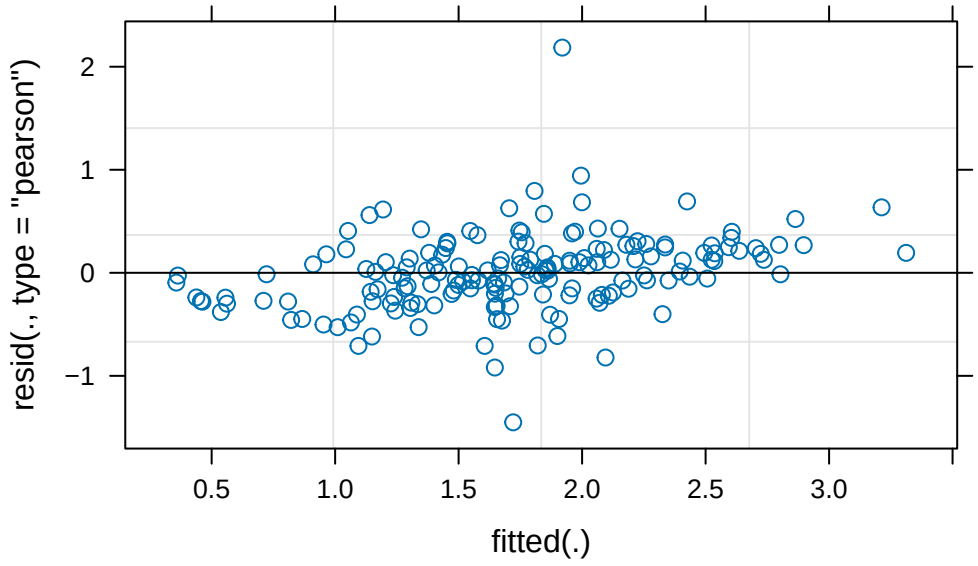
.

```
model3_log <- lmer(log1p(labs_crp) ~ visit + allocation_group + (1 | record_id), data = data_
summary(model3_log)
```

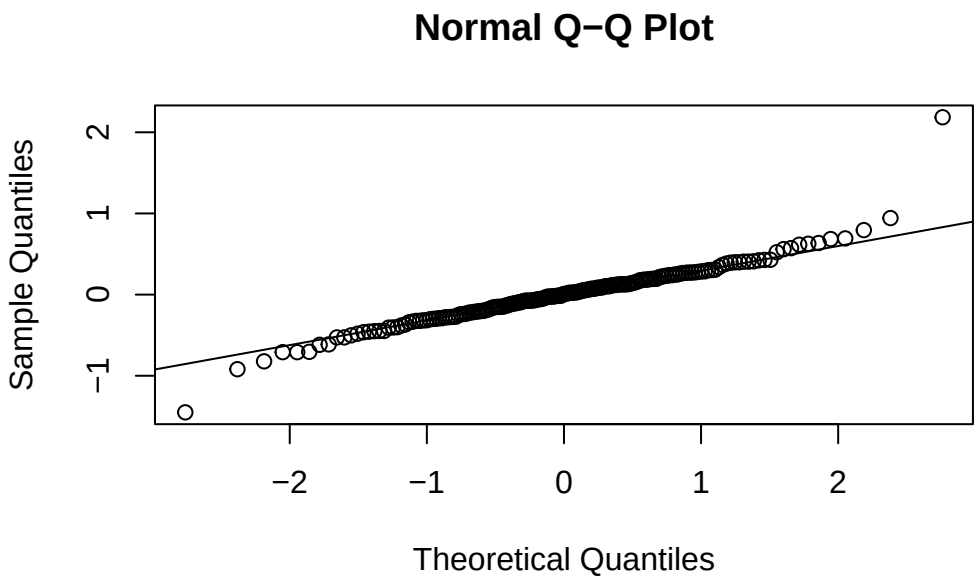
```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: log1p(labs_crp) ~ visit + allocation_group + (1 | record_id)
   Data: data_filtered

REML criterion at convergence: 356.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.1250 -0.4690  0.0144  0.4178  4.7068

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.4128   0.6425
 Residual              0.2156   0.4643
Number of obs: 174, groups:  record_id, 75

Fixed effects:
```

```
                          Estimate Std. Error        df t value Pr(>|t|)
(Intercept)                1.82377    0.14420 134.87011  12.647   <2e-16 ***
visit                     -0.09940    0.04678 106.74492  -2.125   0.0359 *
allocation_groupGrupo B    0.13948    0.16652  72.04671   0.838   0.4050
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) visit
visit       -0.580
allctn_grGB -0.598  0.040
```

```
plot(model3_log)
```




```
qqnorm(resid(model3_log)); qqline(resid(model3_log))  # Normality check
```



log+1 transformation to the skewed CRP variable, and the results show clear improvement.

Term	Estimate	p-value	Interpretation
Intercept	1.82	<0.001	Mean **log(CRP + 1)** at baseline in Grupo A
Visit	–0.099	0.036	**CRP decreases significantly over time**
Grupo B (vs A)	+0.139	0.405	No significant difference at baseline

The **effect of visit became statistically significant** ($p = 0.036$), whereas it was borderline before ($p = 0.072$).

Diagnostic Plots Residuals vs. Fitted

- More **symmetrical and homoscedastic** than before.
- No clear fan shape or funnel — much better than untransformed.

Q-Q Plot

- **Much closer to the line**, indicating that residuals are approximately **normally distributed**.

- A few expected mild deviations at the tails, but very acceptable.

What does this mean in original CRP scale?

Let's back-transform the time effect:

- Estimate for visit = –0.099
- Since you're modeling log1p(CRP), to interpret in original scale:

$$textexp(-0.099) = 0.9056$$

This means: **each visit is associated with ~9.4% decrease** in CRP over time, **on average**.

Summary

Point	**Result**
Residuals	Look better: less heteroscedasticity
Q-Q plot	Much closer to normal
Time effect	Now statistically significant (p = 0.036)
Log transformation	Successfully improved model performance

The idea is to explore the antiinflamatory effect of the intervention. Currently, the model assumes parallel time trends for both groups, i.e., it estimates:

- A main effect of time (CRP changes over time),
- A main effect of group (baseline difference),
- But no interaction (i.e., it assumes both groups change equally over time).

Why this is not enough

If the intervention is effective, we expect:

- CRP to decrease faster in the intervention group (Grupo B),
- Which means there should be a significant interaction between visit and allocation_group.

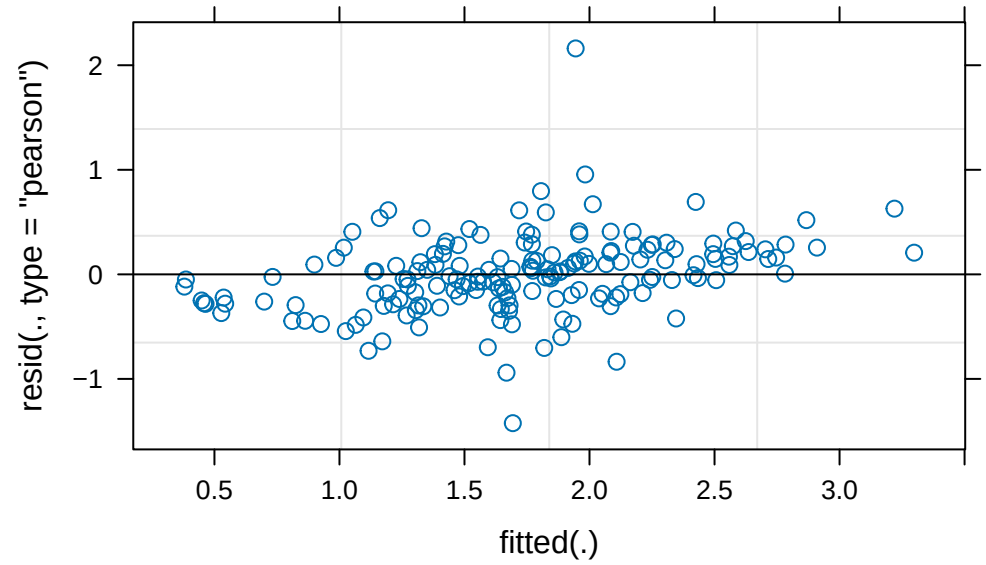
Model with interaction:

- Adds visit:allocation_groupGrupo B as an interaction term,
- Tests whether CRP changes differently over time in Grupo B vs Grupo A.

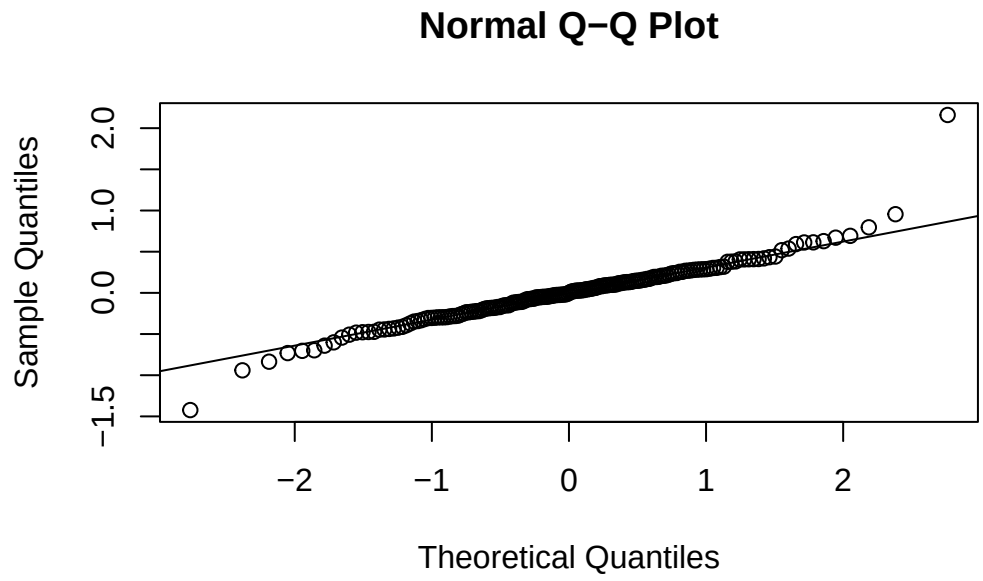
```r
model3_log_inter <- lmer(log1p(labs_crp) ~ visit * allocation_group + (1 | record_id), data =

summary(model3_log_inter)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: log1p(labs_crp) ~ visit * allocation_group + (1 | record_id)
   Data: data_filtered

REML criterion at convergence: 359

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.0555 -0.4762 -0.0015  0.4387  4.6384

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.413    0.6426
 Residual              0.217    0.4658
Number of obs: 174, groups:  record_id, 75

Fixed effects:
                               Estimate Std. Error        df t value Pr(>|t|)
(Intercept)                     1.78653    0.16260 162.42929  10.987   <2e-16
visit                          -0.07858    0.06281 103.85234  -1.251    0.214
allocation_groupGrupo B         0.21990    0.23207 164.19818   0.948    0.345
visit:allocation_groupGrupo B  -0.04706    0.09450 106.26543  -0.498    0.620

(Intercept)                    ***
visit
allocation_groupGrupo B
visit:allocation_groupGrupo B
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) visit  all_GB
visit       -0.691
allctn_grGB -0.701  0.484
vst:llct_GB  0.459 -0.665 -0.696
```

```r
plot(model3_log_inter)
```




```r
qqnorm(resid(model3_log_inter)); qqline(resid(model3_log_inter))  # Normality check
```



Key Result: No Significant Interaction • The term visit:allocation_groupGrupo B has p = 0.620, meaning: There is no statistical evidence that the intervention led to a greater reduction in CRP over time compared to control. • The trend for CRP decrease over time is similar in both groups.

Interpretation

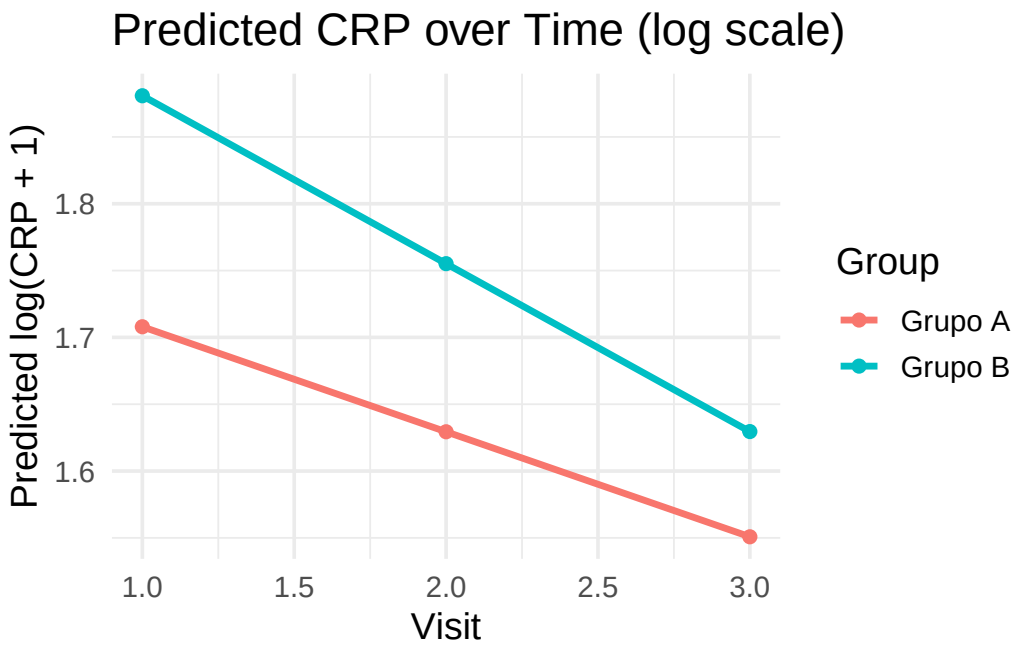
Despite applying a more appropriate transformation and including the interaction: • Time still shows a mild (non-significant) decreasing trend in CRP. • No baseline difference between groups. • No enhanced effect in the intervention group.

This means that, based on your data, the intervention did not show a measurable anti-inflammatory effect on CRP.

PLOT plot shows: • Predicted log(CRP + 1) at each visit for each group. • Makes it easy to compare time trends across intervention and control groups on the same scale used in the model. • Useful for statistical interpretation and checking for meaningful differences.

```
# Create a new data frame for prediction: all combinations of visit × group
new_data <- expand.grid(
  visit = unique(data_filtered$visit),
  allocation_group = unique(data_filtered$allocation_group)
)

# Predict fixed effects (marginal means, no random effects)
new_data$pred_log_crp <- predict(model3_log_inter, newdata = new_data, re.form = NA)

# Plot predicted log(CRP + 1)
ggplot(new_data, aes(x = visit, y = pred_log_crp, color = allocation_group, group = allocati
  geom_line(size = 1.2) +
  geom_point(size = 2) +
  labs(
    title = "Predicted CRP over Time (log scale)",
    y = "Predicted log(CRP + 1)",
    x = "Visit",
    color = "Group"
  ) +
  theme_minimal(base_size = 14)
```

```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```



Visual Insights

- Both groups show a downward trend in CRP over time, suggesting a general anti-inflammatory progression.
- Grupo B starts at a higher baseline and appears to have a slightly steeper decline in log(CRP), although the interaction term was not statistically significant ($p = 0.620$).

This means that although Grupo B appears to improve more, this difference in slopes is not statistically supported.

Grupo B's Higher Baseline

Baseline imbalance may be a concern, particularly when:

1. The groups were supposed to be randomized and equivalent at baseline, and
2. The outcome variable (CRP, in this case) is already higher in one group before the intervention starts.

If Grupo B starts with higher CRP, then:

- Any greater absolute reduction over time may be due to regression to the mean, not the intervention.
- It violates the assumption that both groups are comparable at baseline, which undermines causal inference.

Adjust for baseline differences:

What this model does:

- Adjusts for baseline CRP directly, reducing bias from initial imbalance.
- The coefficient for allocation_groupGrupo B now reflects the difference at follow-up, controlling for baseline.
- The time effect (visit) still captures change over time.
- The model tests whether the intervention group had lower CRP over time than expected based on their higher starting levels.

If log1p_baseline_crp is significant, it means initial inflammation strongly predicts future levels — expected in longitudinal biomarkers.

If allocation_groupGrupo B or the visit:group interaction becomes significant after adjustment, that strengthens the case for a true treatment effect.

Let me know if you'd like to: • Visualize adjusted predictions • Back-transform to original CRP scale • Handle this in a subset of visits (e.g. only V1 and V3)

```
# Adjusting for Baseline CRP in the Mixed Model (on log scale)

# Step 1: Create a baseline CRP variable (log-transformed)
# You should ensure that baseline CRP is correctly identified from your data

# Example: assuming visit 1 is baseline
data_filtered <- data_filtered %>%
  group_by(record_id) %>%
  mutate(log1p_baseline_crp = first(log1p(labs_crp[visit == 1]))) %>%
  ungroup()

# Step 2: Fit the adjusted model
model3_log_adj <- lmer(
  log1p(labs_crp) ~ visit + allocation_group + log1p_baseline_crp + (1 | record_id),
  data = data_filtered
)

# Step 3: Summarize the results
summary(model3_log_adj)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
```

```
Formula: log1p(labs_crp) ~ visit + allocation_group + log1p_baseline_crp +
    (1 | record_id)
   Data: data_filtered

REML criterion at convergence: 250.9

Scaled residuals:
    Min      1Q  Median      3Q     Max
-5.6658 -0.3818 -0.0163  0.4173  3.2431

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.03532  0.1879
 Residual              0.19927  0.4464
Number of obs: 174, groups:  record_id, 75

Fixed effects:
                          Estimate Std. Error         df t value Pr(>|t|)
(Intercept)               0.529463   0.128682 138.565919   4.114 6.62e-05 ***
visit                    -0.105934   0.043048 129.081368  -2.461   0.0152 *
allocation_groupGrupo B  -0.008376   0.082322  75.417846  -0.102   0.9192
log1p_baseline_crp        0.771625   0.050795  76.008799  15.191  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) visit  all_GB
visit       -0.595
allctn_grGB -0.264  0.069
lg1p_bsln_c -0.653 -0.035 -0.118
```

This model directly controls for **baseline differences in CRP**, correcting the bias introduced by the fact that **Grupo B started with higher inflammation**.

Fixed Effects Summary

Term	Estimate	p-value	Interpretation
(Intercept)	0.529	<0.001	Estimated log(CRP+1) for Grupo A at baseline when baseline CRP is 0
visit	–0.106	**0.015**	CRP significantly **decreases over time**

Term	Estimate	p-value	Interpretation
Grupo B (vs Grupo A)	–0.008	0.919	No difference between groups **after adjusting for baseline**
log1p_baseline_crp	0.772	<0.001	Strong predictor: higher baseline CRP leads to higher follow-up CRP

Key Takeaways

- **Time effect (visit) is now significant (p = 0.015)**:

  CRP decreases over time **even after accounting for baseline**.

- **Grupo B effect disappears (p = 0.919)**:

  Once you adjust for baseline CRP, there's **no evidence the intervention had a distinct effect** on CRP reduction compared to control.

- **Baseline CRP is a major driver** of later CRP (= 0.77, p < 0.001).

Interpretation

The original difference in CRP trends between groups was likely due to **baseline imbalance**, not the intervention itself.

This adjusted model is **more reliable**, and the results suggest:

- CRP decreases over time for all participants,
- But the intervention **did not produce a differential anti-inflammatory effect**.

Dropout Influence

```
#Step 1: Check the distribution of visits per subject
# Count number of observations per subject
dropout_check <- data_filtered %>%
  group_by(record_id) %>%
  summarize(n_visits = n_distinct(visit)) %>%
  count(n_visits)


#Step 2: Compare dropout by group
## Last visit per subject
last_visit_by_group <- data_filtered %>%
  group_by(record_id, allocation_group) %>%
```

```
  summarize(last_visit = max(visit)) %>%
  ungroup()
```

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

```
# Table: proportion reaching visit 3
table(last_visit_by_group$allocation_group, last_visit_by_group$last_visit)
```

```
          1  2  3
  Grupo A  6  4 27
  Grupo B  8  4 26
```

```
# This checks whether Grupo B had more missing data at later visits than Grupo A - which cou

#Step 3: Is dropout related to baseline CRP?
# If participants who dropped out had higher baseline CRP, your results may be biased due to

# Use the baseline CRP and check whether it's different in dropouts
baseline_dropout <- data_filtered %>%
  group_by(record_id) %>%
  mutate(last_visit = max(visit)) %>%
  filter(visit == 1) %>%
  mutate(dropped_out = last_visit < 3)

# Compare baseline CRP by dropout status
t.test(log1p(labs_crp) ~ dropped_out, data = baseline_dropout)
```

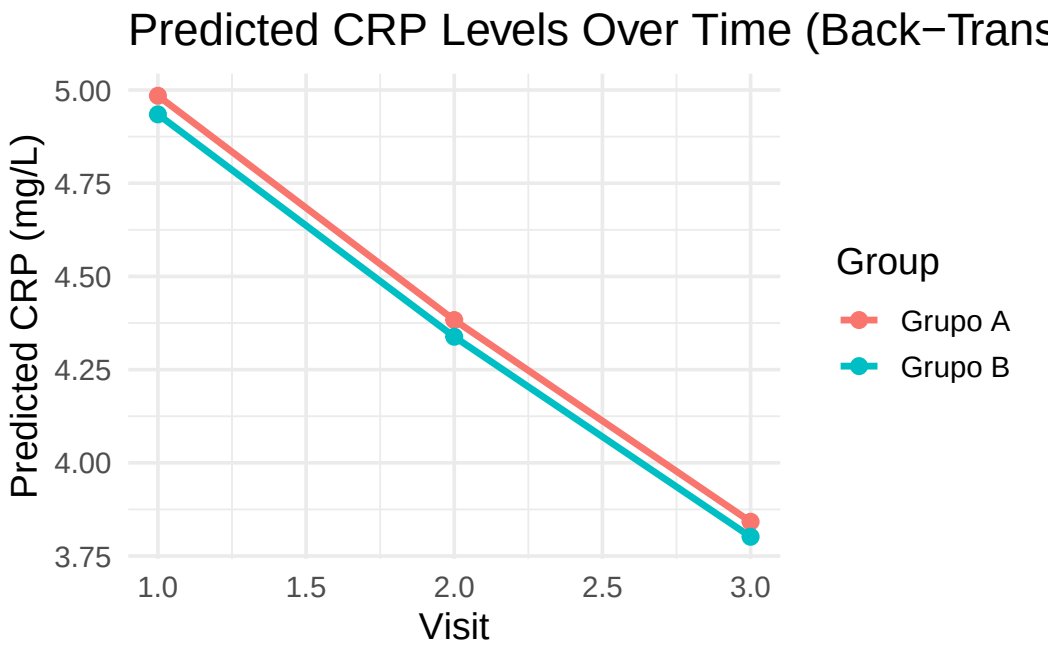
```
    Welch Two Sample t-test

data:  log1p(labs_crp) by dropped_out
t = 1.3607, df = 39.345, p-value = 0.1813
alternative hypothesis: true difference in means between group FALSE and group TRUE is not e
95 percent confidence interval:
 -0.1355451  0.6932685
sample estimates:
mean in group FALSE  mean in group TRUE
           1.867912            1.589051
```

- Participants who dropped out had slightly lower CRP at baseline, but the difference is no
- There's no evidence that dropout was related to baseline inflammation.
- Dropout appears to be random with respect to baseline CRP, which supports the Missing at

Because: • The mixed-effects model (LMM) is valid under MAR, • There's no significant baseline CRP difference between those who stayed and those who dropped out,

Your model results are likely unbiased with respect to dropout.

```
# Back-transforming predicted log(CRP + 1) values to CRP (mg/L)

# Step 1: Create a prediction grid for all combinations of visit × group
# Use the median baseline CRP for prediction (in log1p scale)
baseline_crp_median <- median(log1p(baseline_dropout$labs_crp), na.rm = TRUE)

# Create grid of new data
new_data <- expand.grid(
  visit = unique(data_filtered$visit),
  allocation_group = unique(data_filtered$allocation_group)
) %>%
  mutate(log1p_baseline_crp = baseline_crp_median)

# Step 2: Predict from the adjusted model (fixed effects only)
new_data$pred_log <- predict(model3_log_adj, newdata = new_data, re.form = NA)

# Step 3: Back-transform
new_data$pred_crp <- exp(new_data$pred_log) - 1

# Step 4: Plot back-transformed predictions
ggplot(new_data, aes(x = visit, y = pred_crp, color = allocation_group, group = allocation_gr
  geom_line(linewidth = 1.2) +
  geom_point(size = 2.5) +
  labs(
    title = "Predicted CRP Levels Over Time (Back-Transformed)",
    x = "Visit",
    y = "Predicted CRP (mg/L)",
    color = "Group"
  ) +
  theme_minimal(base_size = 14)
```



What this plot shows

- `Predicted CRP levels in mg/L, adjusted for the median baseline CRP.`
- `Makes the model output clinically interpretable.`

You can clearly see: • The overall downward trend in CRP over time. • That Grupo B does not differ from Grupo A in rate of CRP decline after adjusting for baseline.

```
# add confidence intervals to predicted CRP plot (back-transformed)

# Load required packages
library(ggplot2)
library(dplyr)
library(merTools)  # for predictInterval
```

```
Loading required package: arm
```

```
Loading required package: MASS
```

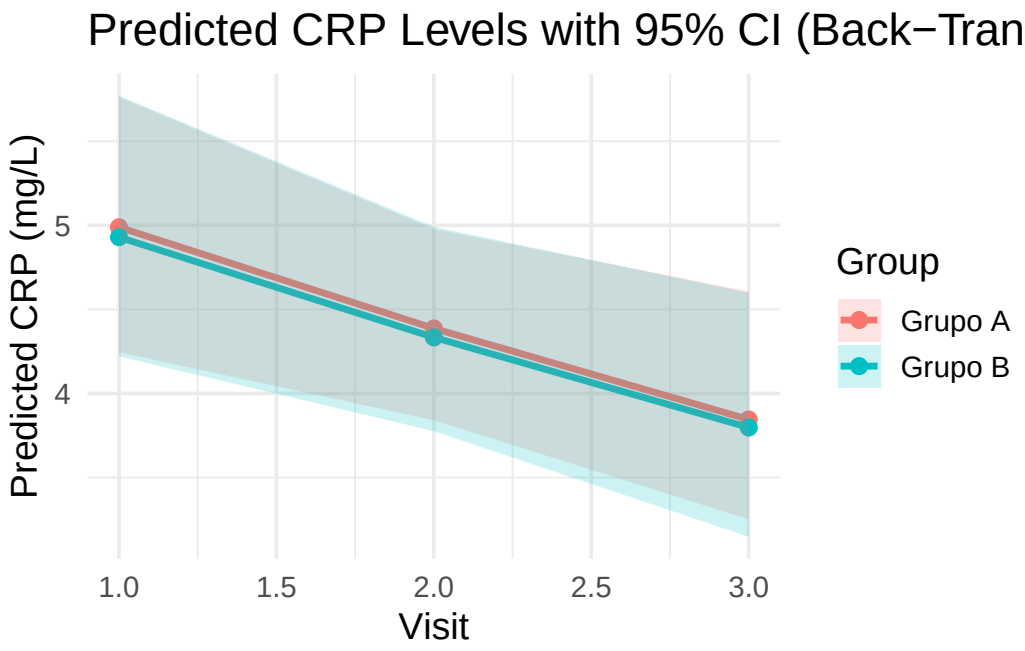
```

Attaching package: 'MASS'
```

```
The following object is masked from 'package:dplyr':

    select


arm (Version 1.14-4, built: 2024-4-1)

Working directory is /Users/gustavosplmoura/Library/Mobile Documents/com~apple~CloudDocs/Medi


Attaching package: 'arm'

The following object is masked from 'package:performance':

    display
```

```r
# Step 1: Create new data with baseline CRP held at median
baseline_crp_median <- median(log1p(baseline_dropout$labs_crp), na.rm = TRUE)

new_data <- expand.grid(
  visit = unique(data_filtered$visit),
  allocation_group = unique(data_filtered$allocation_group)
) %>%
  mutate(
    log1p_baseline_crp = baseline_crp_median,
    record_id = "dummy"  # Add a dummy ID to match model's grouping variable
  )

# Step 2: Get prediction intervals on log scale (includes uncertainty)
set.seed(123)  # for reproducibility
pred_int <- predictInterval(
  model3_log_adj,
  newdata = new_data,
  level = 0.95,
  n.sims = 1000,
  stat = "mean",
  type = "linear.prediction",
  include.resid.var = FALSE
)
```

```
Warning:      The following levels of record_id from newdata
 -- dummy -- are not in the model data.
     Currently, predictions for these values are based only on the
 fixed coefficients and the observation-level error.
```

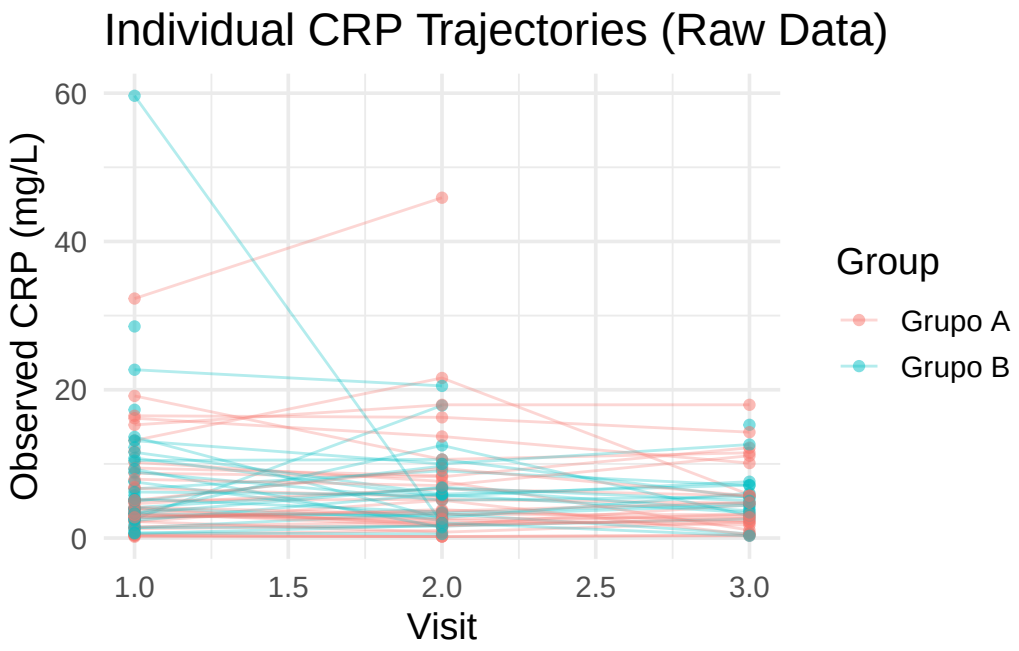
```r
# Combine with original new_data
new_data <- bind_cols(new_data, pred_int)

# Step 3: Back-transform
new_data <- new_data %>%
  mutate(
    fit = exp(fit) - 1,
    lwr = exp(lwr) - 1,
    upr = exp(upr) - 1
  )

# Step 4: Plot with ribbons (CI)
ggplot(new_data, aes(x = visit, y = fit, color = allocation_group, group = allocation_group)
  geom_line(linewidth = 1.2) +
  geom_point(size = 2.5) +
  geom_ribbon(aes(ymin = lwr, ymax = upr, fill = allocation_group), alpha = 0.2, color = NA)
  labs(
    title = "Predicted CRP Levels with 95% CI (Back-Transformed)",
    y = "Predicted CRP (mg/L)",
    x = "Visit",
    color = "Group",
    fill = "Group"
  ) +
  theme_minimal(base_size = 14)
```




```
# Optional: Line plot of individual trajectories
ggplot(data_filtered, aes(x = visit, y = labs_crp, group = record_id, color = allocation_grou
  geom_line(alpha = 0.3) +
  geom_point(alpha = 0.5) +
  labs(
    title = "Individual CRP Trajectories (Raw Data)",
    y = "Observed CRP (mg/L)",
    x = "Visit",
    color = "Group"
  ) +
  theme_minimal(base_size = 14)
```

```
Warning: Removed 12 rows containing missing values or values outside the scale range
(`geom_line()`).
```

```
Warning: Removed 15 rows containing missing values or values outside the scale range
(`geom_point()`).
```



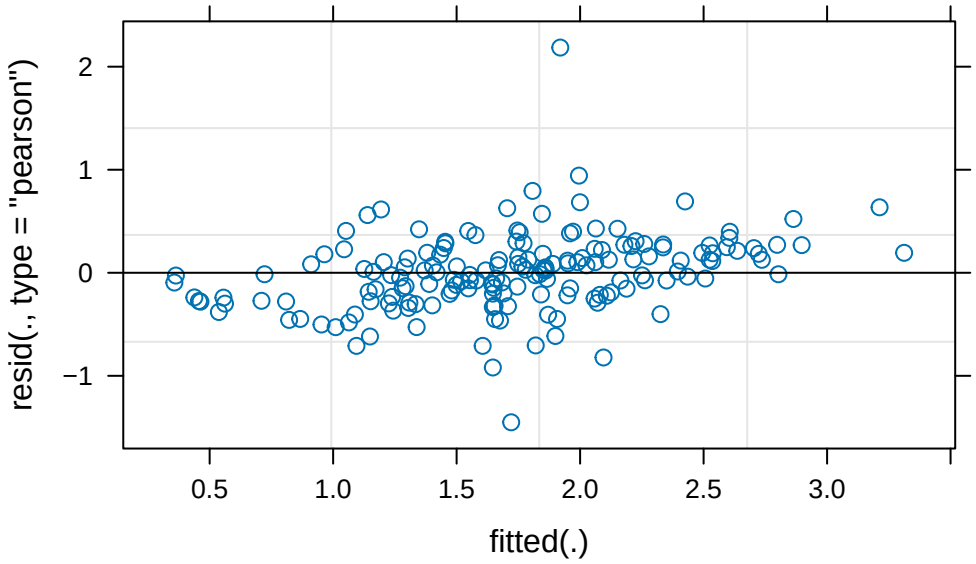
2

```
data_filtered_lmm <- data_filtered %>%
  group_by(record_id) %>%
  filter(n() > 1) %>%
  ungroup()

model4_log <- lmer(log1p(labs_crp) ~ visit + bmi + age + sex + dass_score_stress +
     labs_hba1c + labs_ggt + labs_triglycerides + allocation_group +
     (1 | record_id), data = data_filtered_lmm)

summary(model4_log)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: log1p(labs_crp) ~ visit + bmi + age + sex + dass_score_stress +
    labs_hba1c + labs_ggt + labs_triglycerides + allocation_group +
    (1 | record_id)
   Data: data_filtered_lmm

REML criterion at convergence: 260.1
```

```
Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.0965 -0.4761  0.0079  0.4576  3.9622

Random effects:
 Groups    Name        Variance Std.Dev.
 record_id (Intercept) 0.3083   0.5553
 Residual              0.2502   0.5002
Number of obs: 103, groups:  record_id, 61

Fixed effects:
                           Estimate Std. Error         df t value Pr(>|t|)
(Intercept)              -2.9172883  1.4955248 81.3986059  -1.951  0.05454 .
visit                    -0.1034030  0.0550179 41.8942409  -1.879  0.06715 .
bmi                       0.1130495  0.0410467 79.3316852   2.754  0.00729 **
age                      -0.0148058  0.0106836 59.1960337  -1.386  0.17099
sexMasculino             -0.5116809  0.2786086 54.6626748  -1.837  0.07172 .
dass_score_stress         0.0001941  0.0081992 89.7066088   0.024  0.98117
labs_hba1c                0.3224812  0.1110692 68.3963949   2.903  0.00496 **
labs_ggt                  0.0001671  0.0035339 75.5998101   0.047  0.96241
labs_triglycerides       -0.0007923  0.0013128 91.8399906  -0.604  0.54766
allocation_groupGrupo B   0.2106135  0.1802660 49.7850831   1.168  0.24823
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) visit  bmi    age    sxMscl dss_s_ lbs_h1 lbs_gg lbs_tr
visit       -0.150
bmi         -0.896  0.111
age          0.028 -0.042 -0.227
sexMasculin -0.080  0.064  0.036 -0.056
dss_scr_str -0.045  0.212 -0.134  0.287  0.160
labs_hba1c  -0.477 -0.045  0.218 -0.342  0.122 -0.033
labs_ggt     0.056 -0.015 -0.143  0.177 -0.177  0.065 -0.094
lbs_trglycr  0.143 -0.003 -0.135 -0.059 -0.166  0.030 -0.252 -0.151
allctn_grGB -0.067  0.041 -0.005 -0.139  0.156 -0.021  0.152 -0.095 -0.031
```

```r
plot(model3_log)
```




```
qqnorm(resid(model3_log)); qqline(resid(model3_log))  # Normality check```
```