# Longitudinal Data Modeling

Dr. Marietta Kirchner Dr. Alexandra Lauer

September 24, 2024

# Table of contents

# Preface

# 1 Introduction

## 1.1 Workshop Structure

This class focuses on the longitudinal modeling of data from Patient Reported Outcomes (PROs). It is meant to be hands-on class with applications in R.

Content and structure follow the book by (Mallinckrodt 2016). We would like to extend our warmest gratitude towards Dr. Mallinckrodt for providing the example data for the workshop.

The following topics will be covered:

- Welcome and Introduction (WS session 1)
- Exploration and visualization of longitudinal data (WS session 1/2)
- Inferences from longitudinal data (WS session 3 + 4)
- Assessment of missingness patterns (WS session 5)
- Sensitivity analyses to assess the impact of missingness (WS session 6)
- Annex: Inferences from longitudinal binary data (WS session 7)

## 1.2 Longitudinal Data

This workshop focuses on the analysis of data observed in randomized clinical trials (RCTs). Here, patients have assessments taken at the start of their treatment and then subsequently throughout the course of the trial based on a pre-specified schedule of assessments. The measurement at the start of the treatment is usually referred to as the baseline.

Researchers can be interested in

1. the occurrence of a certain event during the course of the trial, e.g. death or a cardiac event, or the time to the occurrence of such an event, or
2. the longitudinal profile from multiple repeated measurements taken, with a focus on either estimates at a landmark visit or across several time points.

The outcomes under point 1. can be handled via a comparison of the percentages of patients with events between treatment arms, or a time-to-event analysis. Both are out of scope of this workshop.

## 1.3 Basics about RStudio (pre-read)

Alex to add pre-read (YouTube + Cheat Sheets)

## 1.4 Example data

# 2 Longitudinal Data Exploration and Visualization

## 2.1 Introduction

- Data on individuals followed over time with information collected at several time points.
- Clusters are the individuals who are followed over time.
- Repeated observations may or may not be taken at regular times (balanced, fixed occasions, do not differ between subjects).
- Our interest is in the change from baseline.

Datasets used in this course: - Example data is taken from (Mallinckrodt 2016). The authors generated data sets based on two nearly identically designed antidepressant clinical trials by randomly selecting subjects from the original data. - Contain data on the continuous variable HAMD17 (Hamilton 17-item rating scale for depression). - Two treatement arms are included: placebo (arm 1) vs. drug (arm 2). - Assessments were taken at baseline and weeks 1, 2, 4, 6, and 8.

There are 3 data sets created from the original data: - Data *all2* = Subsample of the large dataset with n=50, visits: weeks 2, 4, 8 - Data *high2* = Large dataset with n=100, high dropout = 70% (drug), 60% (placebo) - Data *low2* = Large dataset with n=100, low dropout = 18%

## 2.2 Data set all2

- Small data set with n=50 subjects.
- 1st version: complete data where all subjects adhered to the originally assigend study medication, variable *change*
- 2nd version = missing data: identical to the first except some data were missing (dropout), variable *chgdrop*

Looking at the variables in the data set

```
head(all2)
```

```
# A tibble: 6 x 14
  subject  time chgdrop trt   basval change pgiimp gender chgrescue dropout_grp
  <fct>   <dbl>   <dbl> <chr>  <dbl>  <dbl>  <dbl> <chr>      <dbl> <chr>
1 1           1     -11 2         24    -11      3 F            -11 Week 2 Drop~
2 1           2      NA 2         24    -16      2 F            -26 Week 2 Drop~
3 1           3      NA 2         24    -24      2 F            -34 Week 2 Drop~
4 2           1      -6 1         20     -6      4 F             -6 Week 2 Drop~
5 2           2      NA 1         20     -8      4 F            -18 Week 2 Drop~
6 2           3      NA 1         20     -5      5 F            -15 Week 2 Drop~
# i 4 more variables: aval <dbl>, avisit <fct>, week <dbl>, group <fct>
```

### 2.2.1 Task 1 - Exploration of data set all2 - 15 minutes working time

Only consider the complete data, variable *change* - Are the data balanced and equally spaced? - Number of observations by week? - Summary statistics for HAMD17 (change from baseline) by week. - Plot trajectories for each individual, different colors for each treatment group (or panels). Add mean to your plot. ODER Mean plot separat evtl. mit CI - Generate and interpret the group-wise boxplots of the change from baseline. - Plot trajectories separately by gender. Comment on the plots.

### 2.2.2 Task 1 Discussion, possible solution

Table: Summary statistics for HAMD17 by treatment and week in the all2 data set

```
all2 %>%
  select(change, group, avisit) %>%
  tbl_strata(strata=group,
             ~.x %>%
               tbl_summary(by = avisit,
                           statistic = list(
      all_continuous() ~ "{mean} ({sd})"),
      digits = all_continuous() ~ 2 )
)
```

```
Table printed with `knitr::kable()`, not {gt}. Learn why at
https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
To suppress this message, include `message = FALSE` in code chunk header.
```

| Characteristic | Week 2, N = 25 | Week 4, N = 25 | Week 8, N = 25 | Week 2, N = 25 | Week 4, N = 25 | Week 8, N = 25 |
|---|---|---|---|---|---|---|
| change | -4.20 (3.66) | -6.80 (4.25) | -9.88 (4.85) | -5.24 (5.49) | -8.60 (5.39) | -13.24 (5.54) |

Figure: trajectories

```
ggplot(data = all2, aes(x = week, y = change, group=subject)) +
  geom_point() + geom_line() + facet_grid(.~group) + ylab("Change from baseline HAMD17") +
  scale_x_continuous(name="Visit [week]", breaks=c(2,4,8))
```

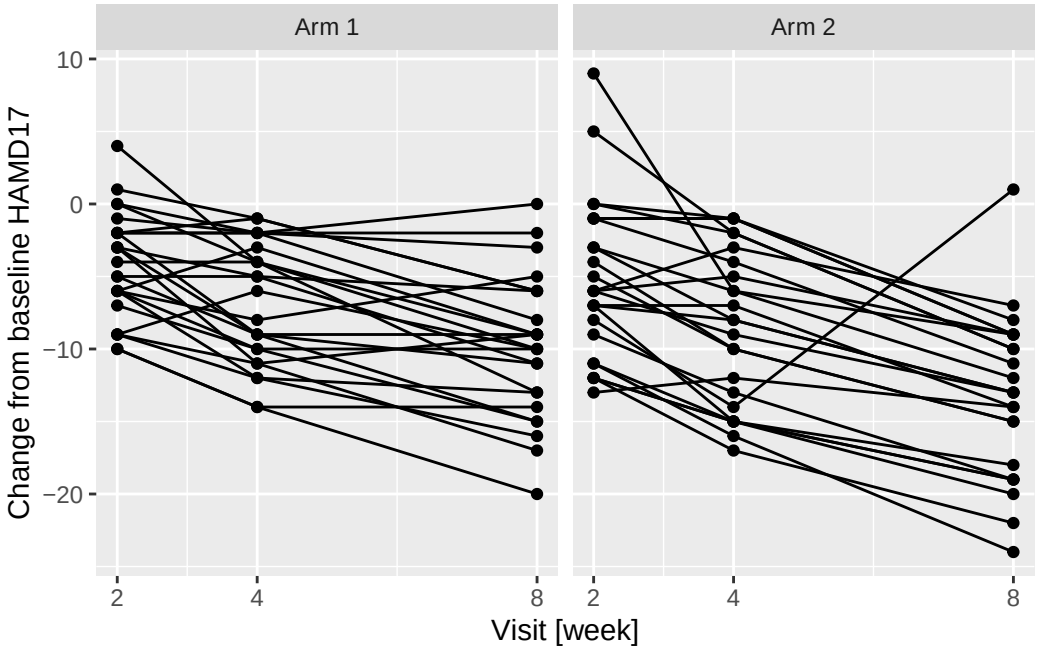



Figure 2.1: Individual trajectories of HAMD17 by treatment group

```
ggplot(data = all2, aes(x = week, y = change)) +
  geom_point(aes(colour=factor(group))) + ylab("Change from baseline HAMD17") +
  scale_x_continuous(name="Visit [week]", breaks=c(2,4,8)) +
  stat_summary(aes(group = group, colour=factor(group)), geom = "line", fun.y = mean,
               size = 1) +
  stat_summary(aes(group = group, colour=factor(group)), geom = "point", fun.y = mean,
               shape=17,size = 2)
```

```
Warning: The `fun.y` argument of `stat_summary()` is deprecated as of ggplot2 3.3.0.
i Please use the `fun` argument instead.
```

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

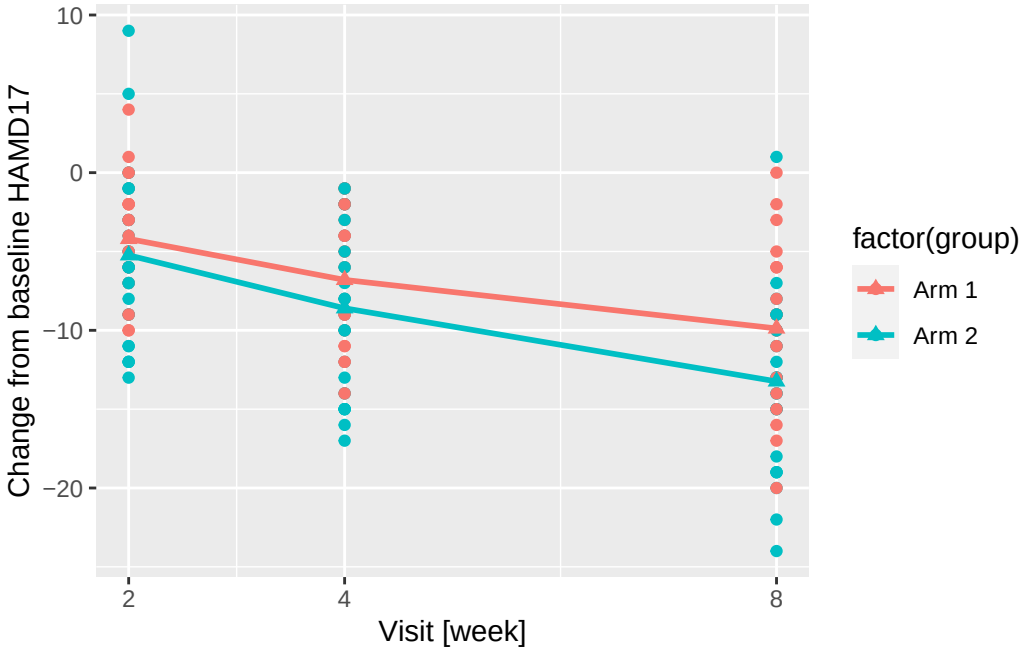



Figure 2.2: Mean HAMD17 change from baseline by treatment group

## 2.3 Data set high2

- Large data set with n=100 subjects.
- Note that we have no intermittent missing values but drop-outs.

Looking at the variables in the data set.

```
head(high2)
```

```
# A tibble: 6 x 14
# Groups:   patient [2]
  patient trt   poolinv basval  week change pgiimp   age gender  aval  drop
```

```
    <dbl> <chr> <chr>    <dbl> <dbl>  <dbl>  <dbl> <dbl> <chr>  <dbl> <dbl>
1    1401 1     005         19     1     -7      3  44.5 F         12     2
2    1401 1     005         19     2     -4      3  44.5 F         15     2
3    1411 2     005         17     1      0      3  35.7 F         17     8
4    1411 2     005         17     2     -2      3  35.7 F         15     8
5    1411 2     005         17     4      2      3  35.7 F         19     8
6    1411 2     005         17     6     -3      2  35.7 F         14     8
# i 3 more variables: group <fct>, avisit <fct>, dropgr <fct>
```

### 2.3.1 Task 2 - Exploration of data set high2 - 15 minutes working time

- drop-outs, last visit for each subject
- trajectories for different drop-out groups

### 2.3.2 Task 2 Discussion, possible solution

```
ggplot(data = high2, aes(x = week, y = change, group=patient)) +
  geom_point() + geom_line() + facet_grid(.~group) +
  ylab("Change from baseline HAMD17") + scale_x_continuous(name="Visit [week]", breaks=c(1
```

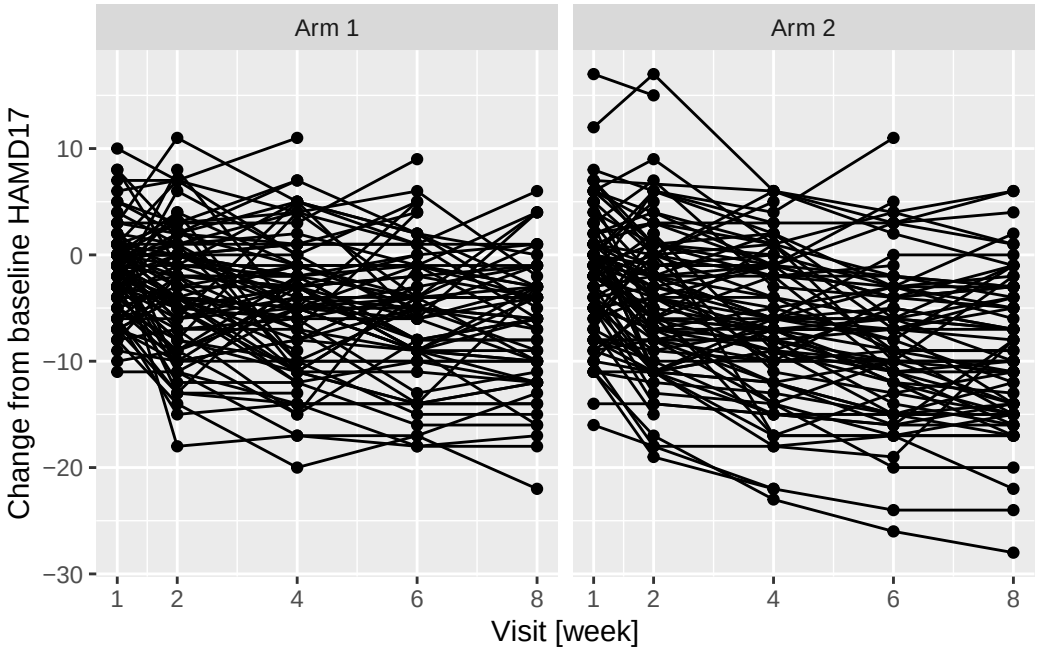



Figure 2.3: Individual trajectories of HAMD17 by treatment group

```
ggplot(data = high2, aes(x = week, y = change, group=patient)) +
  geom_point(col="lightgray") + geom_line(col="lightgray") + facet_grid(.~group) +
  ylab("Change from baseline HAMD17") + scale_x_continuous(name="Visit [week]", breaks=c(1
  stat_summary(aes(group = dropgr, colour=factor(dropgr)), geom = "line", fun.y = mean,
               size = 1) +
  stat_summary(aes(group = dropgr, colour=factor(dropgr)), geom = "point", fun.y = mean,
               shape=17,size = 2)
```

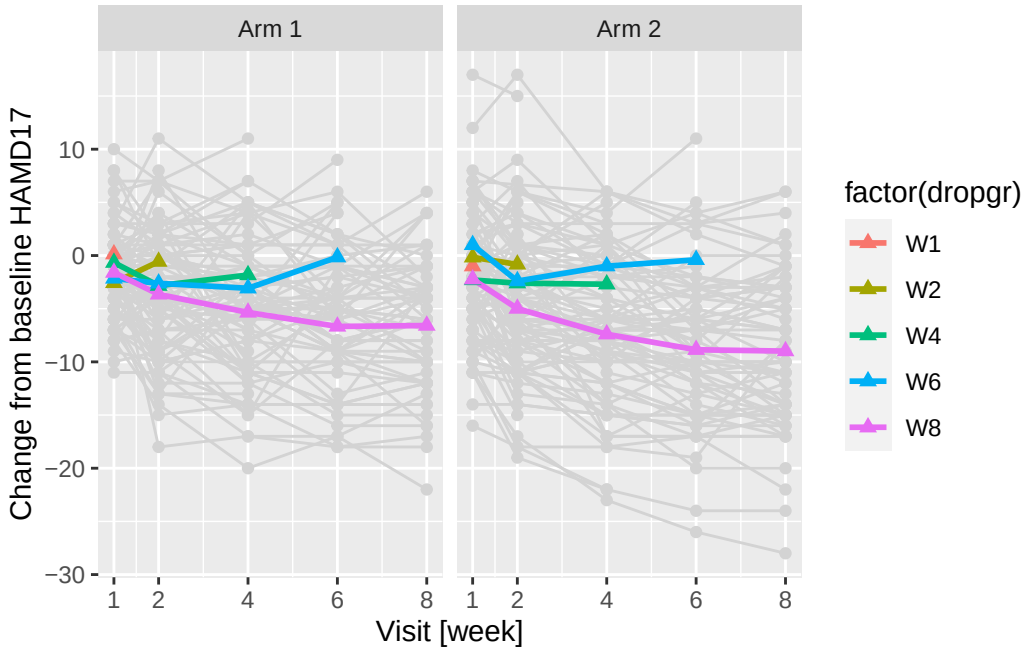



Figure 2.4: Visit-wise mean HAMD17 changes from baseline by treatment group and drop-out

# 3 Correlation structure, covariance matrices

- Longitudinal data allows to exploit the correlation between otucomes wihtin subjects regardless of whether or not focues is on a single landmark time point.
- Model within-subject error correlation
- Different residual covariance structures can be implemented

## 3.1 Overview - different covariance matrices

- Variance components (VC) independence structure
- Compound symmetry (CS) also known as exchangeable
- Toeplitz (TOEP)
- First order auto regressive (AR(1))
- Unstructured (UN)

Selected covariance structures for data with three assessment times (t=3) are shown below. Note that with three assessment times, the number of parameters estimated for the various structures did not differ as much as would be the case with more assessment times. Thus, results from different covariance structures are more similar than would be the case with more assessment times.

### 3.1.1 Independence structure (VC)

Constant variance. It is assumed to be no correlation between assessments (residuals are independent across time).

$$\begin{bmatrix} \sigma^2 & 0 & 0 \\ 0 & \sigma^2 & 0 \\ 0 & 0 & \sigma^2 \end{bmatrix}$$

### 3.1.2 Compound symmetry (CS)

Constant variance and constant covariance across all assessments. Also known as exchangeable. It requires two parameter estimates. Most simplest repeated measures (i.e., correlated errors) structure.

$$R = \begin{bmatrix} \sigma^2 + \sigma_1 & \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma^2 + \sigma_1 & \sigma_1 \\ \sigma_1 & \sigma_1 & \sigma^2 + \sigma_1 \end{bmatrix}$$

### 3.1.3 Unstructured (UN)

This is the most general (saturated) model. It has t + [t(t-1)/2] parameters to be estimated. Here it is 3 + 3 = 6 parameters.

$$R = \begin{bmatrix} \sigma_1^2 & \sigma_{21} & \sigma_{31} \\ \sigma_{21} & \sigma_2^2 & \sigma_{32} \\ \sigma_{31} & \sigma_{32} & \sigma_3^2 \end{bmatrix}$$

### 3.1.4 Toeplitz structure (TOEP)

Homogenous variances and heterogenous correlations. Same correlation value is used whenever the degree of adjacency is the same e.g. correlation between times 1 and 2 = correlation between times 2 and 3. Repeated measurements are assumed to be equally spaced. TOEP requries t parameter estimates so here we have t=3 parameter.

$$R = \begin{bmatrix} \sigma^2 & \sigma_1^2 & \sigma_2^2 \\ \sigma_1^2 & \sigma^2 & \sigma_1^2 \\ \sigma_2^2 & \sigma_1^2 & \sigma^2 \end{bmatrix}$$

### 3.1.5 Autoregressive structure (AR(1))

Correlation decreases as time between observations increases. Assumtpion of equal spacing between each repeated measurement must be reasonably applicable. This structure requires the estimation of two parameters.

$$R = \begin{bmatrix} \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 \\ \sigma^2\rho & \sigma^2 & \sigma^2\rho \\ \sigma^2\rho^2 & \sigma^2\rho & \sigma^2 \end{bmatrix}$$

### 3.1.6 Spatial Power (SP)

Spatial covariance structures does not require equal spacing between measurements. Instead, as long as the distance between visits can be quantified in terms of time and/or other coordinates, the spatial covariance structure can be applied. Covariances are mathematical functions of Euclidean distances between observed measurements. Again, two parameters need to be estimated.

For spatial exponential, the covariance structure is defined as follows:

$$R = \begin{bmatrix} \sigma^2 & \sigma^2\rho_{12} & \sigma^2\rho_{13} \\ \sigma^2\rho_{21} & \sigma^2 & \sigma^2\rho_{23} \\ \sigma^2\rho_{31} & \sigma^2\rho_{32} & \sigma^2 \end{bmatrix}$$

with

$$\rho_{ij} = \rho^{d_{ij}}$$

where

$$d_{ij}$$

is the distance between time point i and time point j e.g. distance in weeks.

## 3.2 Selecting the covariance structure

There are a variety of considerations when selecting the covariance structure: - number of parameters - interpretation of the structure - model fit

UN is the most flexible (complex) structure and can fail to run especially if one has many repeated measures. Choose a reasonable covaraiance structure which is the best compromise between model fit and complexity. E.g. use AIC as it penalises more complex models.

## 3.3 Task 3 - Exploration of correlation in the data

- Compute the empirical correlations between measurement timepoints in the all2 data set (e.g. correlation between baseline and post-baseline changes).
- Looking at these correlations + using your knowledge of the experiment (e.g., spacing of measurements), comment on the suitability of the correlation structures VC, CS, UN, AR(1).

## 3.4 Task 3 - Discussion and possible solution

Table: Correlation and covariance matrix

```
all2 %>% pivot_wider(id_cols=subject,names_from = time, values_from = c(basval,change)) %>
  select(-c(basval_2,basval_3)) -> all2.w

cor(all2.w[-1])
```

```
            basval_1   change_1   change_2    change_3
basval_1  1.00000000 -0.2636447 -0.3165711 -0.02915138
change_1 -0.26364471  1.0000000  0.7557078  0.51502724
change_2 -0.31657106  0.7557078  1.0000000  0.71298768
change_3 -0.02915138  0.5150272  0.7129877  1.00000000
```

```
cov(all2.w[-1])
```

```
           basval_1  change_1  change_2   change_3
basval_1 16.3330612 -4.955918 -6.253061 -0.6391837
change_1 -4.9559184 21.634286 17.179592 12.9967347
change_2 -6.2530612 17.179592 23.887755 18.9061224
change_3 -0.6391837 12.996735 18.906122 29.4351020
```

# 4 Inference from Longitudinal Data

This section will focus on the application of Mixed Model with Repeated Measures (MMRMs). Our main focus will be the modeling of the means of the data. MMRMs are generalizations of standard linear models in the way that data is allowed to be correlated between subsequent measurements from the same subject and exhibit non-constant variability.

The primary assumptions for MMRMs are:

- The data are normally distributed
- The means (expected values) of the data are linear in terms of a certain set of parameters.
- The variances and covariances of the data are in terms of a different set of parameters, and they exhibit a structure matching one of those outlined in the former chapter.

[Alex to add reference to PROC MIXED]

The mixed linear model can be described via the following formula

$$y_i = X_i\beta + Z_i\gamma_i + \varepsilon_i\,,\; i = 1, \ldots, N$$

where $y$ is the vector of responses (observed data, dependent variable), $\beta$ is an unknown vector of fixed effects with known design matrix $X$, $\gamma$ is an unknown vector of random effects with known design matrix $Z$, and $\varepsilon$ is an unknown random error vector. Furthermore $N$ denotes the total number of subjects in our analysis. For the sake of readability, we will omit the subject index and simplify the above formula to

$$y = X\beta + Z\gamma + \varepsilon\,.$$

We will further assume that $\gamma$ and $\varepsilon$ are uncorrelated Gaussian random variables with expectation 0 and variances $G$ and $R$, respectively. Then the variance-covariance matrix of $y$ is given by

$$\mathrm{Var}(y) := V = ZGZ' + R\,.$$

In this case $ZGZ'$ comprises the random effects component, and $R$ is the within-subject component.

In this workshop we will focus on the case where only the within-subject component is accounted for, via modeling of the $R$ matrix. The random effects component $Z\gamma$ will be omitted. In this case we will have $\text{Var}(y) = V = R$, resulting in a model given by

$$y = X\beta + \varepsilon\,.$$

## 4.1 Categorical Time

You can start and familiarise yourself with the main function `mmrm()` using the command

```
library(mmrm)
?mmrm
```

Two inputs are strictly required to get `mmrm()` to work:

- A model formula
- The dataset, containing the response, as well as all fixed effects and variables in the covariance matrix.

**Exercise**: Fit a model `fit_cat_time` using the dataset `all2`, with change as dependent variable, baseline value, visit, baseline by visit interaction and treatment by visit interaction as fixed effects and an unstructured covariance matrix for visits within each subject.

- How do different choices for covariance matrices change the results? What is the difference on the estimation procedure?
- You can obtain a summary of the fit results via `summary(fit_cat_time)`. How do you interpret the fit summary?
- Look at the structure of the fit summary and try to extract the estimate of the $R$ matrix.

```
fit_cat_time <- mmrm::mmrm(
  formula = change ~ basval*avisit + trt*avisit + us(avisit | subject),
  data = all2,
  control = mmrm_control(method = "Kenward-Roger")
)

summary(fit_cat_time)
```

```
mmrm fit

Formula:     change ~ basval * avisit + trt * avisit + us(avisit | subject)
Data:        all2 (used 150 observations from 50 subjects with maximum 3
timepoints)
Covariance:  unstructured (6 variance parameters)
Method:      Kenward-Roger
Vcov Method: Kenward-Roger
Inference:   REML

Model selection criteria:
     AIC      BIC   logLik deviance
   822.4    833.9   -405.2    810.4

Coefficients:
                     Estimate Std. Error        df t value Pr(>|t|)
(Intercept)           1.98452    3.27479  47.00000   0.606  0.54743
basval               -0.31235    0.15905  47.00000  -1.964  0.05548 .
avisitWeek 4         -0.90862    2.39866  47.00000  -0.379  0.70654
avisitWeek 8        -10.58630    3.45922  47.00000  -3.060  0.00365 **
trt2                 -1.18993    1.27265  47.00000  -0.935  0.35457
basval:avisitWeek 4  -0.08542    0.11650  47.00000  -0.733  0.46704
basval:avisitWeek 8   0.24779    0.16801  47.00000   1.475  0.14691
avisitWeek 4:trt2    -0.80100    0.93217  47.00000  -0.859  0.39454
avisitWeek 8:trt2    -2.20106    1.34432  47.00000  -1.637  0.10825
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Covariance estimate:
        Week 2  Week 4  Week 8
Week 2 20.6112 15.3034 12.2766
Week 4 15.3034 21.3565 17.6648
Week 8 12.2766 17.6648 27.6127
```

We can assess the structure of the fit summary via

```
str(summary(fit_cat_time))
```

```
List of 15
 $ cov_type        : chr "us"
 $ reml            : logi TRUE
 $ n_groups        : int 1
```

```
 $ n_theta         : int 6
 $ n_subjects      : int 50
 $ n_timepoints    : int 3
 $ n_obs           : int 150
 $ beta_vcov       : num [1:9, 1:9] 10.724 -0.501 -2.675 -4.267 -1.047 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:9] "(Intercept)" "basval" "avisitWeek 4" "avisitWeek 8" ...
  .. ..$ : chr [1:9] "(Intercept)" "basval" "avisitWeek 4" "avisitWeek 8" ...
 $ varcor          : num [1:3, 1:3] 20.6 15.3 12.3 15.3 21.4 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:3] "Week 2" "Week 4" "Week 8"
  .. ..$ : chr [1:3] "Week 2" "Week 4" "Week 8"
 $ method          : chr "Kenward-Roger"
 $ vcov            : chr "Kenward-Roger"
 $ coefficients    : num [1:9, 1:5] 1.985 -0.312 -0.909 -10.586 -1.19 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:9] "(Intercept)" "basval" "avisitWeek 4" "avisitWeek 8" ...
  .. ..$ : chr [1:5] "Estimate" "Std. Error" "df" "t value" ...
 $ n_singular_coefs: int 0
 $ aic_list        :List of 4
  ..$ AIC     : num 822
  ..$ BIC     : num 834
  ..$ logLik  : num -405
  ..$ deviance: num 810
 $ call            : language mmrm::mmrm(formula = change ~ basval * avisit + trt * avisit +
 - attr(*, "class")= chr "summary.mmrm"
```

and then extract the covariance matrix

```
summary(fit_cat_time)$varcor
```

```
         Week 2   Week 4   Week 8
Week 2 20.61117 15.30339 12.27661
Week 4 15.30339 21.35648 17.66478
Week 8 12.27661 17.66478 27.61271
```

## 4.2 Continuous Time

Time as continuous effect -> single df for time and trt-by-time interaction

Modeling: - Need avisit for structure of covariance matrix - Implicit assumption is for the covariance between values for two timepoints to be equal, regardless of the specific timing

```
fit_cont_time <- mmrm::mmrm(
  formula = change ~ basval*time + trt*time + us(avisit | subject),
  weights = all2$time,
  data = all2,
  control = mmrm_control(method = "Kenward-Roger")
)
```

Quadratic trend

```
all2$timesq <- all2$time^2

fit_cont_timesq <- mmrm::mmrm(
  formula = change ~ basval*timesq + trt*timesq + us(avisit | subject),
  weights = all2$time,
  data = all2,
  control = mmrm_control(method = "Kenward-Roger")
)
```

model checks - residuals per time point

## 4.3 Baseline as a Response (cLDA + LDA)

## 4.4 (Adjusted) LS Means from MMRMs

LS Means are means of the dependent variable adjusted for covariates in the statistical model. We can obtain LS Means estimates and contrasts allowing for a treatment comparison using the `emmeans` package.

**Example**: Calculate the observed (raw) means of changes along with number of patients by treatment group from the dataset `all2` overall and by visit. Then take the model `fit_cat_time` and derive the respective LS Means from the model. What do you observe?

```
# Raw means

all2 %>%
  dplyr::group_by(group) %>%
  dplyr::summarise(
    N = dplyr::n(),
```

```
    Mean = mean(change),
    .groups = "drop"
  )
```

```
# A tibble: 2 x 3
  group     N  Mean
  <fct> <int> <dbl>
1 Arm 1    75 -6.96
2 Arm 2    75 -9.03
```

```
all2 %>%
  dplyr::group_by(group, avisit) %>%
  dplyr::summarise(
    N = dplyr::n(),
    Mean = mean(change),
    .groups = "drop"
  )
```

```
# A tibble: 6 x 4
  group avisit     N   Mean
  <fct> <fct>  <int>  <dbl>
1 Arm 1 Week 2    25  -4.2
2 Arm 1 Week 4    25  -6.8
3 Arm 1 Week 8    25  -9.88
4 Arm 2 Week 2    25  -5.24
5 Arm 2 Week 4    25  -8.6
6 Arm 2 Week 8    25 -13.2
```

The respective LS Means from the model with time as a fixed factor yields the following estimates:

```
library(emmeans)

emmeans::ref_grid(fit_cat_time)
```

```
'emmGrid' object with variables:
    basval = 19.56
    avisit = Week 2, Week 4, Week 8
    trt = 1, 2
```

```
emmeans(fit_cat_time, ~trt)
```

```
NOTE: Results may be misleading due to involvement in interactions
```

```
 trt emmean    SE df lower.CL upper.CL
 1    -6.90 0.836 47    -8.58    -5.22
 2    -9.09 0.836 47   -10.77    -7.41

Results are averaged over the levels of: avisit
Confidence level used: 0.95
```

```
emmeans(fit_cat_time, ~trt*avisit)
```

```
 trt avisit emmean    SE df lower.CL upper.CL
 1   Week 2  -4.13 0.899 47    -5.93    -2.32
 2   Week 2  -5.31 0.899 47    -7.12    -3.51
 1   Week 4  -6.70 0.916 47    -8.55    -4.86
 2   Week 4  -8.70 0.916 47   -10.54    -6.85
 1   Week 8  -9.86 1.033 47   -11.94    -7.79
 2   Week 8 -13.26 1.033 47   -15.33   -11.18

Confidence level used: 0.95
```

### 4.4.1 Observed vs. balanced margins

In the example above we have used the standard option for the weights in the calculation of LS Means. We will delve deeper into the following two options and will try to understand the difference:

- `weights = "equal"`: Each stratum induced by covariate levels is assigned the same weight in the calculation of the LS Means. This is the default option.
- `weights = "proportional"`: Each stratum induced by covariate levels is assigned a weight according to their observed proportion in the calculation of the LS Mean. This option gives each stratum a weight corresponding to its size. Estimates using this option are reflective of the balance of covariates in the data.

**Exercise:** Based on the `fit_cat_time` model, compare the LS Means for the change in the response variable by treatment overall and treatment by visit interaction using the different

options for `weight`. Compare the results for the two LS Means options to the observed means and to one another.

Discuss the following points:

- Why is there no difference between LS Means estimates for the overall treatment effect and the treatment by visit interaction? (Hint: Create a frequency table)

Now update the `fit_cat_time` model to `fit_cat_time2`, and include the covariate `gender`. Estimate the same LS Means for the change in the response variable by treatment (overall) and treatment by visit interaction.

- Why is there a difference now between results from the different LS Means options? (Hint: another frequency table can help)
- What effect could missing data have on the estimation, even in the case of `fit_cat_time`? I.e. what would happen if this data was not complete but subject to missingness, with the degree of missing data increasing over time and being disproportionate between treatment arms?

**Solution**:

We first calculate the LS Means, using the different `weights` options and find they are indeed identical.

```
# These will yield the same results:
emmeans(fit_cat_time, ~trt, weights = "equal")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

```
 trt emmean    SE df lower.CL upper.CL
 1    -6.90 0.836 47    -8.58    -5.22
 2    -9.09 0.836 47   -10.77    -7.41

Results are averaged over the levels of: avisit
Confidence level used: 0.95
```

```
emmeans(fit_cat_time, ~trt, weights = "proportional")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

```
 trt emmean    SE df lower.CL upper.CL
 1    -6.90 0.836 47    -8.58    -5.22
 2    -9.09 0.836 47   -10.77    -7.41

Results are averaged over the levels of: avisit
Confidence level used: 0.95
```

```
emmeans(fit_cat_time, ~trt*avisit, weights = "equal")
```

```
 trt avisit emmean    SE df lower.CL upper.CL
 1   Week 2  -4.13 0.899 47    -5.93    -2.32
 2   Week 2  -5.31 0.899 47    -7.12    -3.51
 1   Week 4  -6.70 0.916 47    -8.55    -4.86
 2   Week 4  -8.70 0.916 47   -10.54    -6.85
 1   Week 8  -9.86 1.033 47   -11.94    -7.79
 2   Week 8 -13.26 1.033 47   -15.33   -11.18

Confidence level used: 0.95
```

```
emmeans(fit_cat_time, ~trt*avisit, weights = "proportional")
```

```
 trt avisit emmean    SE df lower.CL upper.CL
 1   Week 2  -4.13 0.899 47    -5.93    -2.32
 2   Week 2  -5.31 0.899 47    -7.12    -3.51
 1   Week 4  -6.70 0.916 47    -8.55    -4.86
 2   Week 4  -8.70 0.916 47   -10.54    -6.85
 1   Week 8  -9.86 1.033 47   -11.94    -7.79
 2   Week 8 -13.26 1.033 47   -15.33   -11.18

Confidence level used: 0.95
```

Now we can update the model to include the covariate `gender`. We can specify this a new model using the `mmrm()` function again, or simply use `update()` to add the new covariate to the model. Either way is fine, and a look into the model formula from the fit summary shows the two approaches work interchangeably.

```
fit_cat_time2 <- update(fit_cat_time, . ~ . + gender)
summary(fit_cat_time2)
```

```
mmrm fit

Formula:
change ~ basval + avisit + trt + (us(avisit | subject)) + gender +
    basval:avisit + avisit:trt
Data:        all2 (used 150 observations from 50 subjects with maximum 3
timepoints)
Covariance:  unstructured (6 variance parameters)
Method:      Kenward-Roger
Vcov Method: Kenward-Roger
Inference:   REML

Model selection criteria:
     AIC      BIC   logLik deviance
   817.0    828.5   -402.5    805.0

Coefficients:
                     Estimate Std. Error        df t value Pr(>|t|)
(Intercept)           0.47589    3.23944  46.14000   0.147  0.88385
basval               -0.30674    0.15200  45.44000  -2.018  0.04951 *
avisitWeek 4         -0.90862    2.39786  47.00000  -0.379  0.70645
avisitWeek 8        -10.58630    3.45626  47.00000  -3.063  0.00362 **
trt2                 -0.34868    1.30287  46.74000  -0.268  0.79016
genderM               2.32931    1.29556  45.99000   1.798  0.07876 .
basval:avisitWeek 4  -0.08542    0.11646  47.00000  -0.734  0.46689
basval:avisitWeek 8   0.24779    0.16786  47.00000   1.476  0.14657
avisitWeek 4:trt2    -0.80100    0.93186  47.00000  -0.860  0.39439
avisitWeek 8:trt2    -2.20106    1.34318  47.00000  -1.639  0.10795
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Covariance estimate:
        Week 2  Week 4  Week 8
Week 2 18.8295 14.3160 12.0002
Week 4 14.3160 21.1623 18.1813
Week 8 12.0002 18.1813 28.8384
```

A look into the reference grid shows us the new factor levels for `gender`. Note that `gender` itself will not be included in the `emmeans()` statement, but the output indicates the averaging over its levels (same for the levels of `avisit`)

```r
# Reference grid shows us the new levels
emmeans::ref_grid(fit_cat_time2)
```

```
'emmGrid' object with variables:
    basval = 19.56
    avisit = Week 2, Week 4, Week 8
    trt = 1, 2
    gender = F, M
```

```r
# These two won't yield the same results
emmeans(fit_cat_time2, ~trt, weights = "equal")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

```
 trt emmean    SE   df lower.CL upper.CL
 1    -7.13 0.840 45.8    -8.82    -5.44
 2    -8.48 0.896 46.4   -10.28    -6.68

Results are averaged over the levels of: avisit, gender
Confidence level used: 0.95
```

```r
emmeans(fit_cat_time2, ~trt*avisit, weights = "proportional")
```

```
 trt avisit emmean    SE   df lower.CL upper.CL
 1   Week 2  -4.55 0.890 46.2    -6.34    -2.75
 2   Week 2  -4.89 0.890 46.2    -6.69    -3.10
 1   Week 4  -7.13 0.941 47.1    -9.02    -5.23
 2   Week 4  -8.27 0.941 47.1   -10.17    -6.38
 1   Week 8 -10.29 1.081 46.6   -12.46    -8.11
 2   Week 8 -12.83 1.081 46.6   -15.01   -10.66

Results are averaged over the levels of: gender
Confidence level used: 0.95
```

The following frequency table shows the imbalance in the distribution of the `gender` variable. We can see that Treatment 1 has more men than women, whereas Treatment 2 has more women than men.

```r
table(all2$trt, all2$gender)
```

```
   F  M
1 30 45
2 57 18
```

The data is no longer balanced across the covariates in the model. The `weights = "equal"` option is agnostic to this imbalance and assigns all levels equal weights, whereas the `weights = "proportional"` assigns a weight reflecting the proportional size of the stratum over which the average is taken.

### 4.4.2 Contrasts

## 4.5 Fit diagnostics

The following section closely follows the content in Chapter 10 in (Fitzmaurice 2011).

Our analysis should be concluded with a look into the fit diagnostics, more specifically, the residuals. Residuals are defined by the difference between the true responses and the fitted values from the model:

$$r := y - X\hat{\beta}\,,$$

where $\hat{\beta}$ are the estimated coefficients from our model. Residuals provide an estimate of the true vector of random errors

$$\varepsilon = y - X\beta\,.$$

As per our modeling assumptions, $\varepsilon$ should follow a normal distribution with mean zero. The mean of the residuals is zero and therefore identical with the mean of the error term. For the covariance of the residuals however, the variance-covariance matrix of $\varepsilon$ only serves us as an approximation (as suggested by (Fitzmaurice 2011) for all 'practical applications'):

$$Cov(r) \approx Cov(\varepsilon) = R\,.$$

This assumption has several implications on the residual diagnostics:

- The variance is not necessarily constant. Plotting the fitted values versus the residuals might therefore lead to a non-constant range. An examination of the residual variance or autocorrelation among residuals is therefore not very meaningful.

- Residuals from analyses of longitudinal data can exhibit correlation with the covariates. Scatterplots of residuals versus selected covariates can therefore reveal systematic trends (which normally should not be the case).

A transformation of residuals to achieve constant variance and zero correlation is therefore often useful. This transformation uses the so-called *Cholesky decomposition* of the variance-covariance matrix $R$. Let $L$ be a lower triangular matrix, such that

$$R = L\,L'\,,$$

then the transformed residuals are given by

$$r^* = L^{-1}(y - X\beta)\,.$$

In the `mmrm` package, transformed residuals can be derived using the `type = "normalized"` option.

Residual plots overall and by visit

- histogram of transformed residuals
- predicted vs. actual values
- predicted vs. residuals
- standardized residuals vs. theoretic quantiles

**Exercise**: Which visualisations can you think of that make sense to assess the goodness of fit here? Create a new `tibble` (or `data.frame`) containing the variables of importance and try plotting them in a meaningful way. Discuss the results within your group.

**Solution**:

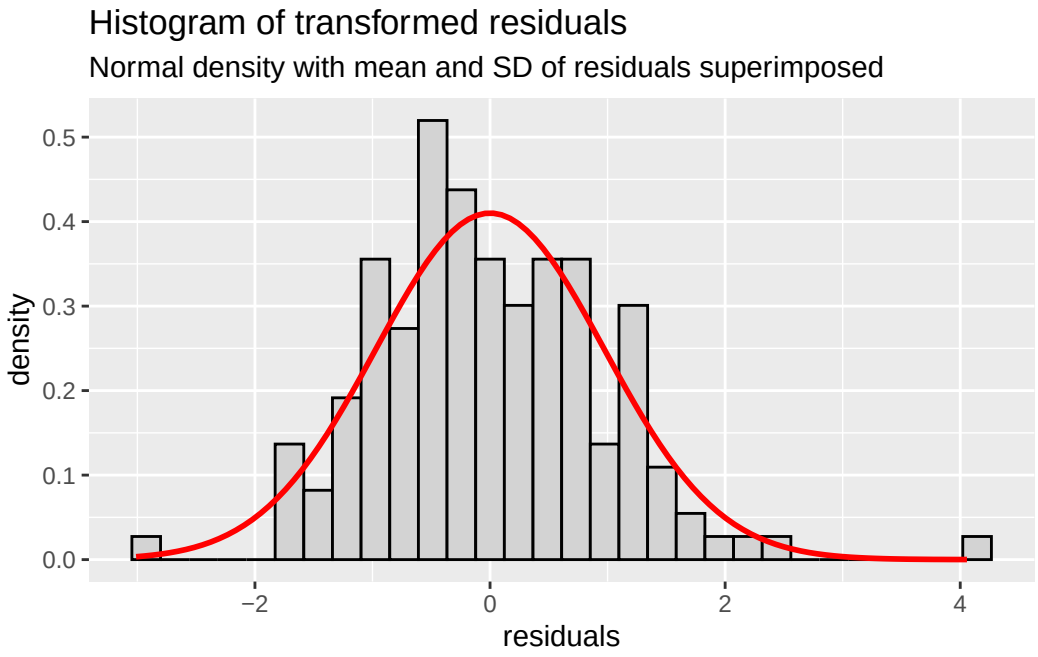
To avoid repetition, let us first save the important variables to perform fit diagnostics in a `tibble`.

```
df_residuals <- dplyr::tibble(
  residuals = residuals(fit_cat_time, type = "normalized"),
  predictions = fitted(fit_cat_time),
  all2
)
```

We can firstly look into a histogram of transformed residuals. The shape should resemble the density function of normal distribution with mean zero and positive variance. Superimposing the density function with mean and SD derived from the model residuals, let's us see that this is indeed the case. We can also detect a slight skewness to the right.

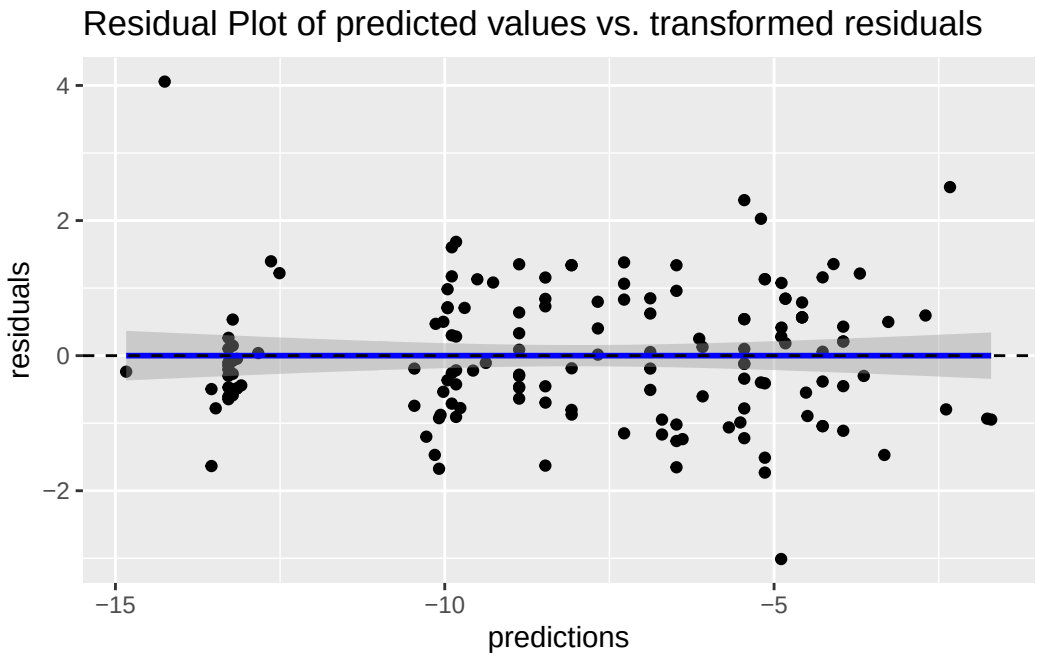
```r
library(ggplot2)

df_residuals %>%
  ggplot(aes(x = residuals)) +
  geom_histogram(aes(y = after_stat(density)), fill='lightgray', col='black') +
  stat_function(fun = dnorm, args = list(mean=mean(df_residuals$residuals), sd=sd(df_resid
  ggtitle(
    label = "Histogram of transformed residuals",
    subtitle = "Normal density with mean and SD of residuals superimposed"
  )
```




Alternatively, we can create a Q-Q-Plot of ...

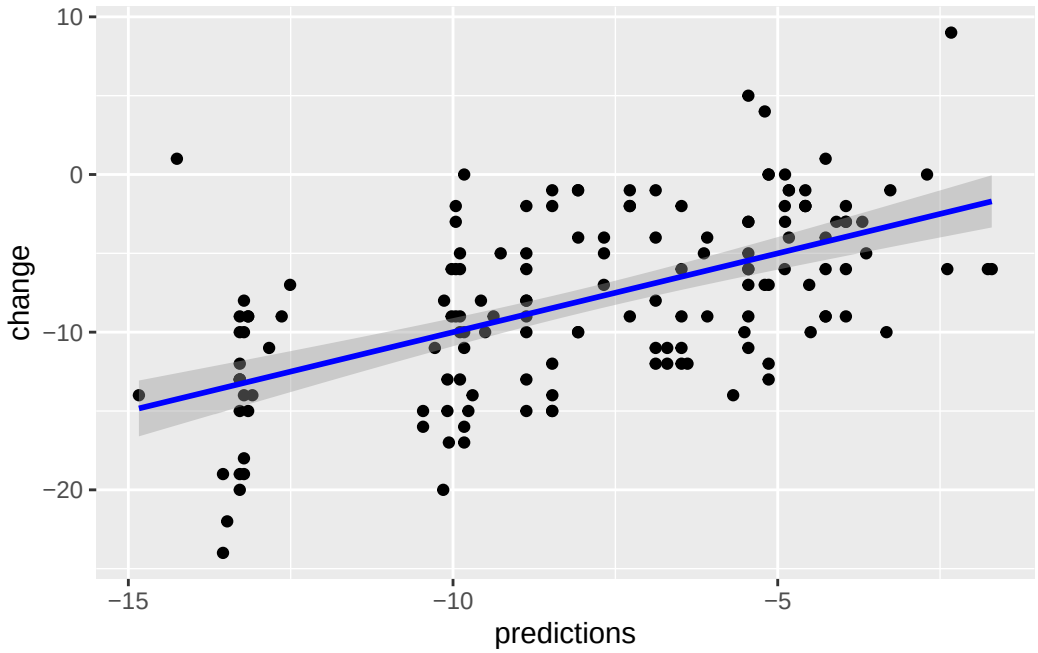
```r
df_residuals %>%
  ggplot(aes(x = predictions, y = residuals)) +
  geom_point() +
  geom_smooth(method = lm, color = "blue") +
  geom_hline(yintercept = 0, show.legend = FALSE, linetype = 2) +
  ggtitle(
    label = "Residual Plot of predicted values vs. transformed residuals"
  )
```




What do we see?

- The points in the plot look well dispersed and symmetric around zero. The fitted line shows no departure from zero.
- There is no systematic trend, but a rather random scatter.

```
df_residuals %>%
  ggplot(aes(x = predictions, y = change)) +
  geom_point() +
  geom_smooth(method = lm, color = "blue")
```




### 4.5.1 Addendum on RS&I Models

Different dosing/ assessment frequency between treatment arms in parallel design -> oncology (chemo with fixed cycles vs immune-therapy)

# 5 Missing Data

So far, we conducted all our analyses on the basis of complete data. This is a blissful, yet highly unusual setting.

Missingness patterns:

- Monotonic/ dropout
- Intermittend

Our dataset contains a second variable `chgdrop`, which is subject to missingness. Let's rerun our initial MMRM with `chgdrop` as dependent variable, baseline value, visit, baseline by visit interaction and treatment by visit interaction as fixed effects and an unstructured covariance matrix for visits within each subject.

This formulation is very similar to the one at the beginning of the former chapter. How do the results differ in terms of LS Means of change from baseline by treatment arm over time?

```
fit_cat_time <- mmrm::mmrm(
  formula = chgdrop ~ basval*avisit + trt*avisit + us(avisit | subject),
  data = all2,
  control = mmrm_control(method = "Kenward-Roger")
)

# summary(fit_cat_time)

model_lsmeans <- emmeans::emmeans(fit_cat_time, ~trt*avisit, weights = "proportional")
model_lsmeans
```

```
 trt avisit emmean    SE   df lower.CL upper.CL
 1   Week 2  -4.10 0.900 47.0    -5.91    -2.29
 2   Week 2  -5.29 0.899 47.0    -7.10    -3.48
 1   Week 4  -6.42 0.974 46.5    -8.38    -4.46
 2   Week 4  -8.52 0.951 44.8   -10.43    -6.60
 1   Week 8  -9.73 1.142 40.4   -12.03    -7.42
 2   Week 8 -12.62 1.114 40.1   -14.88   -10.37

Confidence level used: 0.95
```

```r
emmeans::emmeans(fit_cat_time, ~trt*avisit, weights = "proportional") %>%
  contrast(
    list(
      "Difference in LS Means at Week 8" = c(0, 0, 0, 0, -1, 1),
      "Difference in longitudinal LS Means to Week 8" = c(-1, 1, -1, 1, -1, 1)/3
    )
  )
```

```
contrast                                          estimate   SE   df t.ratio
Difference in LS Means at Week 8                     -2.90 1.60 40.3  -1.814
Difference in longitudinal LS Means to Week 8        -2.06 1.23 46.8  -1.671
p.value
 0.0772
 0.1014
```

To understand the nature of the differences between the model using `change` as a response variable and the one with `chgdrop`, we need to look closer into the extent of missing data and understand its nature.

## 5.1 Missing Data Mechanisms

To understand the nature of missing data in our clinical trial, we consider the following taxonomy, introduced by (Roderick JA Little 2019). We differentiate between the following three types of missing data:

- **Missing Completely at Random (MCAR)**: Conditional on all covariates in our analysis, the probability of missingness does not depend on either observed or unobserved values of the response variable.
- **Missing at Random (MAR)**: Conditional on all covariates and observed response values in our analysis, the probability of missingness does not depend on the unobserved values of the response variable.
- **Missing not at Random (MNAR)**: Conditional on all covariates and observed response values in our analysis, the probability of missingness does depend on the unobserved values of the response variable.

(Craig Mallinckrodt 2016) give the following interpretation around the three types of missingness:

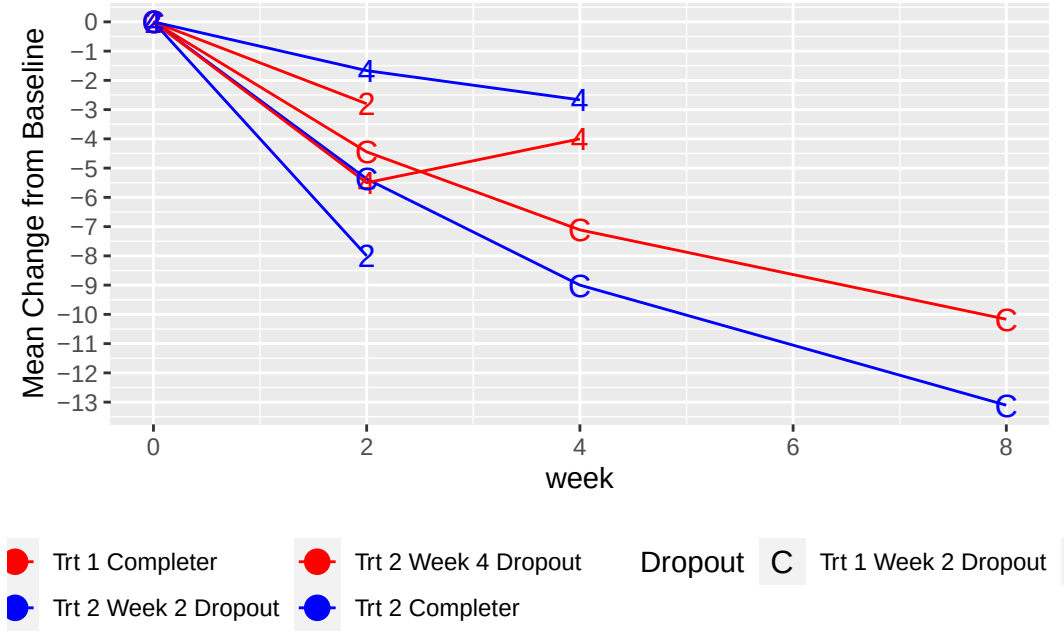
"*With MCAR, the outcome variable is not related to the probability of dropout (after taking into account covariates). In MAR, the observed values of the outcome variable are related*

*to the probability of dropout, but the unobserved outcomes are not (after taking into account covariates and observed outcomes). In MNAR the unobserved outcomes are related to the probability of dropout even after the observed outcomes and covariates have been taken into account.*"

The following two sections outline handling strategies for missing data. However, the best approach to handle missing data is to minimise its extent. While the occurence of missing data can rarely be avoided at all (think about the collection of questionnaire data in oncology studies and the missing data after subjects die), it is important to pursue an "as complete as can be" data collection.

Baseline and screening data are of utmost importance in a pursuit of data completeness. If a screening value is missing, but was meant to be used as a covariate, this subjects' whole data will be dropped from the analysis even if all responses were observed. If the baseline response variable was missing we are unable to compute a change from baseline, which also leads to the loss of this subjects' data in the model (although LDA models are still able to provide an estimate) even if all post-baseline values were observed.

## 5.2 Missing data handling I (descriptive stats + visualisations)



## 5.3 Missing data handling II (analytic approaches)

### 5.3.1 Complete Case Analyses

# 6 Sensitivity Analyses

Purpose: talk about sensitivity analyses with respect to missing data

- MMRM is an appropriate choice for the primary analysis in many longitudinal clinical trials under the missing at random (MAR) assumption.
- MMRM can handle missing values. BUT: need of baseline and at least one post-baseline value.
- No imputation for individual missing values but missing data is implicitly imputed.
- Exploit the correlation between outcomes within subjects.
- MAR: future outcomes for subjects who discontinued are assumed be similar to the future outcomes of subjects who continued if they had the same values of past (observed) outcomes, covariates,...

## 6.1 Purpose of sensitivity analyses

- Consider sensivitiy analyses to check model assumptions e.g. assumption of MAR.
- Comparing results from sensitivity analyses: how much inference rely on the assumptions.
- Here, inference with regard to the treatment effect. Thus, investigate how treatment effects vary depending on assumptions (about missing data).
- Uncertainty from incompleteness cannot be objectively evaluated from observed data so there is a need for missing data sensitivity analyses.

## 6.2 MMRM vs. MI

- Flexibility in modeling treatment effects over time and the within-patient error correlation structure makes MMRM a widely useful analysis.
- MMRM, MI: two major approaches to missing data with good statistical properties. Both rely on MAR assumption (for MI: standard implementation).
- MMRM: missing values implicitly imputed, MI: missing values explicitly imputed.
- MMRM vs. MI: approximately equivalent provided the variables used in the imputation model are the same as those included in the analysis model (level of equivalence will depend on the number of imputations)

- MI: imputation model with at least those variables from the primary model, additional auxiliary variables can be used in the imputation model to improve the accuracy of the missing data prediction.
- Handling missing not at random (MNAR) possible for MI (e.g. reference-based imputation) but not within MMRM.
- MMRM does not work if missing baseline values are present. Missing baseline values can be imputed first. Additionally, at least one post-baseline value has to be observed. Alternative: LDA where baseline is part of the response vector.

Note that, when implemented in similar manners, MI and MMRM have similar assumptions and yield similar results. Thus, MI implemented similarly to MMRM is not a sensitivity analysis!

## 6.3 Missing covariates (baseline data)

- Missing baseline value of the outcome: MI or use of mean imputation (Paper: ),
- MMRM not efficient or potential biased estimates as subjects with missing covariates are excluded from the analysis

## 6.4 Sensitivity analyses - Simple approaches

In general, not recommended for use. Methods are of histroic interest and provide a useful starting point. - Last observation carried forward (LOCF): used in the past, justified as it was thought that it provides conservative estimates. - Complete case (observed case/completers analysis): creates selection bias, may cause overestimation of within group effects particularly at the last scheduled visit.

We apply these two methods here as well to compare results.

## 6.5 Sensitivity analyses - Handling nonignorable missingness (MNAR)

- Assumption of MAR is often reasonable, but possibility of data missing not at random (MNAR) is difficult to rule out.
- Thus, analysis under MNAR needed.
- Analysis under MNAR: these methods are heavily assumption driven and the assumptions are not testable as we do not have the missing data.
- Consider a sensitivity analysis framework allowing assessment of robustness of results to the various assumptions.

- MNAR methods: different possibilies e.g. class of pattern-mixture models
- MI can be used to explore departures from MAR (for analysis under a MNAR assumption). This is referred to as controlled MI and includes delta-based MI and reference-based MI (belong to the class of pattern mixture models). Data is imputed under an alternative MNAR distribution that reflects a contextually relevant scenario for the unobserved data. The imputed data sets are then analysed as with standard MI.

### 6.5.1 Reference-based multiple imputation

- Has recieved increasing attention in clinical trials as it provides an attractive approach for a sensitivity analysis because missing data assumptions are framed in an intuitive way. The departure from MAR is captured in a qualitative way, making the formulation of the problem intuitive.
- For example, a plausible MNAR mechanism in a placebo-controlled trial is to assume that subjects in the experimental arm who dropped out stop taking their treatment and have similar outcomes to those in the placebo arm.
- Different options to handle missing outcome data for reference-based imputation were described: e.g. copy reference, copy increments in reference, jump to reference (Paper Carpenter 2013)

### 6.5.2 Delta-based multiple imputation

- Impute data assuming all unobserved subjects having a poorer or better response than those observed, by adding or subtracting a delta parameter $\delta$ to the expected value of the MAR imputed values.
- Delta can be implemented in all treatment groups, or in only one group, or may vary by treatment group or an alternative specified factor.
- Choice of values for the sensitivity parameter $\delta$: often selected by the analyst. Better: selection by content experts.
- Steps: 1. missing values are imputed using standard MI procedure under MAR, 2. imputed values are shifted by adding some fixed value $\delta$ to reflect the MNAR mechanism, 3. analysis with standard statistical methods including Rubin's rule to combine results

## 6.6 Practical part

- Take the *high2* data set
- Look again at the complete case analysis
- Apply additionally LOCF and compare results
- Try MNAR method reference-based MI and/or delta-based MI. Compare with the other results.

# 7 Inferences from binary longitudinal data

# References


Craig Mallinckrodt, Ilya Lipkovich. 2016. *Analyzing Longitudinal Clinical Trial Data: A Practical Guide*. Vol. 1. USA: Chapman; Hall/CRC. https://doi.org/10.1201/9781315186634.

Fitzmaurice, Laird, G. M. 2011. *Applied Longitudinal Analysis*. Vol. 2. USA: New York, Wiley. https://doi.org/10.1002/9781119513469.

Mallinckrodt, Craig. 2016. *Analyzing Longitudinal Clinical Trial Data*. Chapman; Hall/CRC. https://doi.org/10.1201/9781315186634.

Roderick JA Little, Donald B. Rubin. 2019. *Statistical Analysis with Missing Data*. Vol. 3. USA: New York, Wiley. https://doi.org/10.1002/9781119482260.