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,...

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.

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!

Missing covariates (baseline data) only

- Missing baseline value of the outcome (and other covariates) is a common situation
- MMRM not efficient or potential biased estimates as subjects with missing covariates are excluded from the analysis
- (Kayembe and Breukelen 2022) compared different methods e.g. unadjusted analysis, complete case, mean imputation, MI: mean imputation seems to be appropriate as long as the covariates are measured before randomization (produces unbiased treatment effect estimates with good coverage, easy to implement)

Now, we consider the situation as in our data sets: baseline observed, no intermittent missing values, drop-outs = monotone missing pattern

Sensitivity analyses - Simple approaches

In general, these simple approaches are not recommended for use. Methods are of historic interest and provide a useful starting point. Here, we consider two simple approaches. We will apply these two methods in the practical part to compare results.

Last observation carried forward (LOCF)

For each subject, LOCF imputes missing values using the last observed value for that subject. Typically, under LOCF the repeated masures nature of the data is ignored and a single outcome for each subject is analyzed.

LOCF was used in the past, justified as it was thought that it provides conservative estimates. However, conditions under which LOCF yield conservative estimates and maintain control of

Type I error rates are not straightforward and cannot be assured at the beginning of the trial. For example, LOCF is likely to overestimate treatment benefit if dop-out in the control gorup is more frequent.

Complete case (CC)

Other names: observed case/ completers analysis

Reduce the data set selecting only those subjects with observed outcome value(s).

Completers analysis may create selection bias, may cause overestimation of within group effects particularly at the last scheduled visit.

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 possibilities e.g. class of pattern-mixture models. The pattern-mixture model allows missing outcomes to be imputed under a chosen scenario and in this way can be used to complete the data set and apply the primary analysis to this completed data set.
- 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 relevant scenario for the unobserved data. The imputed data sets are then analysed as with standard MI.

Reference-based multiple imputation

- Has received 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.

- Remember: MI under MAR assumes that the outcome distribution of patients with missing data is the same as the outcome distribution of patients with complete data, conditional on relevant covariates. However, if most patients withdraw from the study after treatment discontinuation, then this is not plausible, as patients who withdraw from the study treatment are expected to have a worse otucome than patients who stay on study treatment. Thus, addressing missing data under a MAR assumption estimates a hypothetical estimand and not a treatment policy estimand.
- Different options to handle missing outcome data for reference-based imputation were described (Carpenter and Kenward 2013): e.g. jump to reference (J2R), copy reference (CR), copy increments in reference (CIR)

Jump to reference J2R assumes that after treatment discontinuation, the patient's mean outcome distribution is that of a reference group, usually the control group. This is a very extreme assumption, as this implies that any efficacy of the drug vanishes immediately after discontinuation - may be plausible for symptomatic treatments.

Copy reference CR assumes that the patient's outcome distribution both before and after treatment discontinuation is the same as the distribution of the reference group. This has a milder effect than J2R: If a treatment-group patient has an outcome that is better than the reference group mean before treatment discontinuation, their imputed values after treatment discontinuation will also be better than the reference group mean.

Copy increments in reference CIR assumes that after treatment discontinuation, the increments are the same as those from the reference group. This is much milder than J2R and CR and implies that benefit gained from the treatment before discontinuation is not lost.

The conventional approach to analyse data using these reference based approaches is MI, following the same steps as MI under MAR.

Software, R: e.g. the rbmi package supports reference-based strategies (Gower-Page and Wolbers 2022)

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 δ to the expected value of the e.g. 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 δ: e.g. selection by content experts.
- Steps: 1. missing values are imputed using standard MI procedure e.g. under MAR (but can also be under MNAR e.g. combinded with copy reference approach), 2. imputed values are shifted by adding some fixed value δ to reflect the MNAR mechanism, 3. analysis with standard statistical methods including Rubin's rule to combine results

Practical part

- Take the (*all2*) *high2* data set
- Look at the MMRM and at the complete case (CC) analysis (refer to section Missing Data for the *all2* data set).
- Apply additionally LOCF and compare results.
- Try MNAR method reference-based MI with J2R and CIR by using the rbmi package. Compare with the other results.

Set-up to use rbmi

Have a short look at the `rbmi()` package first.

```
library(rbmi)
?rbmi

vignette(topic = "quickstart", package = "rbmi")
```

```
starting httpd help server ... done
```

The workflow is based on 4 core functions:

- draws() - fits the imputation models, different methods possible, we will use method_bayes() for MI based on Bayesian posterior parameter draws from MCMC sampling
- impute() - creates multiple imputed data sets
- analyse() - analyses each of the multiple imputed data sets, default = ancova, other options possible
- pool() - combines the results across imputed data sets, for method_bayes (see above) pooling and inference is based on Rubin's rule

Implemented imputation strategies in rbmi:

- Missing at Random (MAR)
- Jump to Reference (JR)
- Copy Reference (CR)
- Copy Increments in Reference (CIR)

I will show how it looks like for the *all2* data set and you will then explore the methods using the *high2* data set.

Plenum - Solution for all2 data set

1. Complete case

Table: Adjusted means for the complete case data set (all2 data with drop-out, select completer)

```
model_lsmeans_cc
```

```
 trt avisit emmean   SE df lower.CL upper.CL
 1   Week 2  -4.33 1.12 34    -6.61    -2.06
 2   Week 2  -5.47 1.09 34    -7.69    -3.26
 1   Week 4  -6.98 1.09 34    -9.19    -4.77
 2   Week 4  -9.12 1.06 34   -11.27    -6.97
 1   Week 8 -10.17 1.21 34   -12.63    -7.71
 2   Week 8 -13.10 1.18 34   -15.49   -10.71

Confidence level used: 0.95
```

2. MMRM, unstructured covariance

Table: Adjusted means for the all2 data set with drop-out analysed with MMRM

```
model_lsmeans_mmrm
```

```
 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
```

3. LOCF

```
all2.locf <- all2 %>% filter(!is.na(chgdrop)) %>%
  dplyr::group_by(subject) %>%
  dplyr::mutate( drop=max(week) )

all2.locf<-all2.locf %>% dplyr::filter(week==drop)
```

```r
ancova <- aov(change ~ basval + trt, data = all2.locf)
summary(ancova)
```

```
            Df Sum Sq Mean Sq F value Pr(>F)  
basval       1    1.8    1.82   0.053 0.8185  
trt          1  114.2  114.20   3.342 0.0739 .
Residuals   47 1606.1   34.17                 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
ancova$coefficients
```

```
(Intercept)      basval        trt2 
-8.69460273  0.02497994 -3.02800963 
```

Table: Mean values for change from baseline of LOCF analysis

Characteristic	**Arm 1**, N = 25	**Arm 2**, N = 25
change	-8.20 (5.50)	-11.24 (6.06)

4. J2R imputation

```r
# Define the names of key variables in the data set
set_mi<-set_vars(
  subjid = "subject",
  visit = "avisit",
  outcome = "chgdrop",
  group = "group",
  covariates = c("basval * avisit", "group * avisit")
)

vars_an<-set_mi
vars_an$covariates <- "basval"

# Define the imputation strategy for each subject with at least one missing observation
dat_ice <- all2 %>%
  arrange(subject, avisit) %>%
  filter(is.na(chgdrop)) %>%
```

```r
  group_by(subject) %>%
  slice(1) %>%
  ungroup() %>%
  select(subject, avisit) %>%
  mutate(strategy = "JR")

# Define the imputation method
method <- method_bayes(
  burn_in = 200,
  burn_between = 5,
  n_samples = 100,
  seed = 072407
)

draw_all2<-draws(data=all2, data_ice = dat_ice, vars=set_mi, method=method, ncores = 1, qu

imputeObj <- impute(
  draw_all2,
  references = c("Arm 1" = "Arm 1", "Arm 2" = "Arm 1")
)

imputed_all2 <- extract_imputed_dfs(imputeObj)

anaObj <- analyse(
  imputeObj,
  vars = vars_an
)
```

Table: Estimates from jump to reference J2R imputation

```r
poolObj <- pool(anaObj)
as.data.frame(poolObj)
```

```
      parameter       est        se        lci        uci         pval
1     trt_Week 2  -1.189928 1.2864325  -3.780746  1.4008900 3.598958e-01
2 lsm_ref_Week 2  -4.125036 0.9088264  -5.955372 -2.2947002 4.178855e-05
3 lsm_alt_Week 2  -5.314964 0.9088264  -7.145300 -3.4846279 5.199440e-07
4     trt_Week 4  -1.920738 1.3711052  -4.689028  0.8475529 1.687137e-01
5 lsm_ref_Week 4  -6.404449 0.9777849  -8.379818 -4.4290798 7.521493e-08
6 lsm_alt_Week 4  -8.325187 0.9654627 -10.274069 -6.3763039 8.265733e-11
7     trt_Week 8  -2.211225 1.6967275  -5.649685  1.2272346 2.005871e-01
8 lsm_ref_Week 8  -9.656881 1.2244745 -12.142348 -7.1714149 2.770935e-09
```

```
9 lsm_alt_Week 8 -11.868107 1.1717638 -14.238826 -9.4973871 1.948658e-12
```

5. Change from J2R to CIR Use the additional argument update_strategies in the impute function.

```
dat_ice_CIR <- dat_ice %>%
  mutate(strategy = ifelse(strategy == "JR", "CIR", strategy))

imputeObj_CIR <- impute(
  draw_all2,
  references = c("Arm 1" = "Arm 1", "Arm 2" = "Arm 1"),
  update_strategy = dat_ice_CIR
)

anaObj_CIR <- analyse(
  imputeObj_CIR,
  vars = vars_an
)
```

Table: Estimates from copy increments in reference CIR imputation

```
poolObj_CIR <- pool(anaObj_CIR)
as.data.frame(poolObj_CIR)
```

```
      parameter        est        se        lci        uci         pval
1    trt_Week 2  -1.189928 1.2864325  -3.780746  1.4008900 3.598958e-01
2 lsm_ref_Week 2  -4.125036 0.9088264  -5.955372 -2.2947002 4.178855e-05
3 lsm_alt_Week 2  -5.314964 0.9088264  -7.145300 -3.4846279 5.199440e-07
4    trt_Week 4  -2.014793 1.3710976  -4.782582  0.7529964 1.492281e-01
5 lsm_ref_Week 4  -6.389437 0.9827301  -8.375095 -4.4037784 8.988973e-08
6 lsm_alt_Week 4  -8.404230 0.9687489 -10.359825 -6.4486346 7.096096e-11
7    trt_Week 8  -2.609022 1.6166710  -5.879658  0.6616141 1.146759e-01
8 lsm_ref_Week 8  -9.646948 1.1739906 -12.026748 -7.2671472 8.012076e-10
9 lsm_alt_Week 8 -12.255969 1.1568184 -14.598416 -9.9135228 7.351873e-13
```

Now, you can first of all repeat the analysis on the *all2* data set to see if you can manage it. Or you go directly to the next step and apply methods to the *high2* data set.

One starting point for the *high2* data set as the structure is a little bit different:

First, fill in missing visits. This was not necessary in the *all2* data set. This can be done with the expand_locf function of the rbmi package. Note, *change* is the outcome variable and not *chgdrop* as in *all2*

```
high2 <- high2 %>% ungroup()

high2_expand <- expand_locf(
  high2,
  subject = levels(high2$subject),
  avisit = levels(high2$avisit),
  vars = c("basval","trt","group"),
  group = c("subject"),
  order = c("subject", "avisit")
)
```

Solution for high2 data set

First, fill in missing visits. This was not necessary in the *all2* data set. Note, *change* is the outcome variable and not *chgdrop* as in *all2*


```
high2 <- high2 %>% ungroup()

high2_expand <- expand_locf(
  high2,
  subject = levels(high2$subject),
  avisit = levels(high2$avisit),
  vars = c("basval","trt","group"),
  group = c("subject"),
  order = c("subject", "avisit")
)
```


1. Complete case

```
high2.cc<- high2 %>% dplyr::filter(drop==8)

fit_cc <- mmrm::mmrm(
  formula = change ~ basval*avisit + trt*avisit + us(avisit | subject),
  data = high2.cc,
  control = mmrm_control(method = "Kenward-Roger")
)

summary(fit_cc)
```

```
mmrm fit
```

```
Formula:     change ~ basval * avisit + trt * avisit + us(avisit | subject)
Data:        high2.cc (used 649 observations from 130 subjects with maximum 5
timepoints)
Covariance:  unstructured (15 variance parameters)
Method:      Kenward-Roger
Vcov Method: Kenward-Roger
Inference:   REML

Model selection criteria:
     AIC      BIC   logLik deviance
  3693.8   3736.9  -1831.9   3663.8

Coefficients:
                     Estimate Std. Error        df t value Pr(>|t|)
(Intercept)           3.26341    1.33919 127.00000   2.437   0.0162 *
basval               -0.29475    0.07287 127.00000  -4.045 9.03e-05 ***
avisitWeek 2         -0.11631    1.40264 127.74000  -0.083   0.9340
avisitWeek 4         -0.77525    1.56814 127.01000  -0.494   0.6219
avisitWeek 6         -3.27487    1.59347 127.01000  -2.055   0.0419 *
avisitWeek 8         -3.94403    1.69295 127.01000  -2.330   0.0214 *
trt2                 -0.06433    0.81571 127.00000  -0.079   0.9373
basval:avisitWeek 2  -0.11719    0.07647 128.03000  -1.532   0.1279
basval:avisitWeek 4  -0.18029    0.08533 127.01000  -2.113   0.0366 *
basval:avisitWeek 6  -0.10859    0.08671 127.01000  -1.252   0.2127
basval:avisitWeek 8  -0.06299    0.09212 127.01000  -0.684   0.4954
avisitWeek 2:trt2    -0.32364    0.85100 127.17000  -0.380   0.7043
avisitWeek 4:trt2    -1.07631    0.95516 127.01000  -1.127   0.2619
avisitWeek 6:trt2    -1.35403    0.97059 127.01000  -1.395   0.1654
avisitWeek 8:trt2    -1.65323    1.03118 127.01000  -1.603   0.1114
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Covariance estimate:
        Week 1  Week 2  Week 4  Week 6  Week 8
Week 1 21.0337 16.0819 13.3827 12.2436 13.2340
Week 2 16.0819 34.0261 22.4388 20.5555 20.0298
Week 4 13.3827 22.4388 34.9214 27.3814 25.0059
Week 6 12.2436 20.5555 27.3814 33.6239 30.2604
Week 8 13.2340 20.0298 25.0059 30.2604 39.6554
```

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

```
 trt avisit emmean    SE  df lower.CL upper.CL
 1   Week 1  -1.91 0.595 127    -3.08   -0.731
 2   Week 1  -1.97 0.550 127    -3.06   -0.885
 1   Week 2  -4.08 0.756 126    -5.58   -2.585
 2   Week 2  -4.47 0.702 127    -5.86   -3.080
 1   Week 4  -5.85 0.764 127    -7.36   -4.336
 2   Week 4  -6.99 0.706 127    -8.38   -5.591
 1   Week 6  -7.09 0.749 127    -8.57   -5.607
 2   Week 6  -8.51 0.692 127    -9.88   -7.138
 1   Week 8  -6.96 0.811 127    -8.56   -5.352
 2   Week 8  -8.67 0.750 127   -10.16   -7.191

Confidence level used: 0.95
```

2. MMRM

```r
fit_mmrm <- mmrm::mmrm(
  formula = change ~ basval*avisit + trt*avisit + us(avisit | subject),
  data = high2,
  control = mmrm_control(method = "Kenward-Roger")
)

summary(fit_mmrm)
```

```
mmrm fit

Formula:     change ~ basval * avisit + trt * avisit + us(avisit | subject)
Data:        high2 (used 830 observations from 200 subjects with maximum 5
timepoints)
Covariance:  unstructured (15 variance parameters)
Method:      Kenward-Roger
Vcov Method: Kenward-Roger
Inference:   REML

Model selection criteria:
     AIC      BIC   logLik deviance 
  4779.1   4828.6  -2374.6   4749.1 
```

```
Coefficients:
                     Estimate Std. Error        df t value Pr(>|t|)    
(Intercept)           3.33421    1.12651 196.97000   2.960  0.00346 ** 
basval               -0.27934    0.05962 196.97000  -4.685  5.2e-06 ***
avisitWeek 2         -0.15400    1.17265 181.53000  -0.131  0.89566    
avisitWeek 4         -1.00849    1.35934 172.12000  -0.742  0.45916    
avisitWeek 6         -3.27037    1.53582 166.05000  -2.129  0.03470 *  
avisitWeek 8         -3.93835    1.65523 140.95000  -2.379  0.01868 *  
trt2                 -0.04273    0.64969 196.97000  -0.066  0.94763    
basval:avisitWeek 2  -0.08292    0.06254 181.91000  -1.326  0.18659    
basval:avisitWeek 4  -0.10700    0.07290 173.67000  -1.468  0.14396    
basval:avisitWeek 6  -0.01321    0.08198 165.55000  -0.161  0.87216    
basval:avisitWeek 8   0.01778    0.08902 143.32000   0.200  0.84197    
avisitWeek 2:trt2    -0.61015    0.69414 181.41000  -0.879  0.38057    
avisitWeek 4:trt2    -1.41851    0.81728 175.52000  -1.736  0.08438 .  
avisitWeek 6:trt2    -2.31835    0.91503 165.19000  -2.534  0.01222 *  
avisitWeek 8:trt2    -2.47738    0.99465 143.57000  -2.491  0.01389 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Covariance estimate:
        Week 1  Week 2  Week 4  Week 6  Week 8
Week 1 20.9961 17.1332 15.4142 15.3503 15.8717
Week 2 17.1332 35.2157 25.8380 25.5499 24.3926
Week 4 15.4142 25.8380 38.8771 33.0523 30.1128
Week 6 15.3503 25.5499 33.0523 43.7638 39.3236
Week 8 15.8717 24.3926 30.1128 39.3236 47.7371
```

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

```
 trt avisit emmean    SE  df lower.CL upper.CL
 1   Week 1  -1.61 0.458 197    -2.52   -0.711
 2   Week 1  -1.66 0.459 197    -2.56   -0.752
 1   Week 2  -3.24 0.609 191    -4.44   -2.036
 2   Week 2  -3.89 0.613 193    -5.10   -2.681
 1   Week 4  -4.52 0.656 182    -5.81   -3.223
 2   Week 4  -5.98 0.656 182    -7.27   -4.684
 1   Week 6  -5.12 0.718 168    -6.53   -3.701
 2   Week 6  -7.48 0.715 166    -8.89   -6.067
 1   Week 8  -5.24 0.785 149    -6.79   -3.686
```

```
 2   Week 8  -7.76 0.762 139    -9.26   -6.251

Confidence level used: 0.95
```

3. LOCF

```r
high2.locf<-high2 %>% dplyr::filter(week==drop)

ancova <- aov(change ~ basval + trt, data = high2.locf)
summary(ancova)
```

```
             Df Sum Sq Mean Sq F value  Pr(>F)
basval        1    483   483.3   9.709 0.00211 **
trt           1    241   241.4   4.851 0.02880 *
Residuals   197   9805    49.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
ancova$coefficients
```

```
(Intercept)      basval        trt2
  0.3536399  -0.2648315  -2.2086854
```

```r
high2.locf %>% ungroup() %>%
  select(change, group) %>%
  tbl_summary(by = group,
              statistic = list(
                all_continuous() ~ "{mean} ({sd})"),
              digits = all_continuous() ~ 2 )
```

Characteristic	**Arm 1**, N = 100	**Arm 2**, N = 100
change	-4.22 (6.38)	-6.72 (7.90)

4. J2R

```r
set_mi<-set_vars(
  subjid = "subject",
  visit = "avisit",
```

```
  outcome = "change",
  group = "group",
  covariates = c("basval * avisit", "group * avisit")
)

vars_an<-set_mi
vars_an$covariates <- "basval"

dat_ice <- high2_expand %>%
  arrange(subject, avisit) %>%
  filter(is.na(change)) %>%
  group_by(subject) %>%
  slice(1) %>%
  ungroup() %>%
  select(subject, avisit) %>%
  mutate(strategy = "JR")

method <- method_bayes(
  burn_in = 200,
  burn_between = 5,
  n_samples = 100,
  seed = 072407
)

draw_high2<-draws(data=high2_expand, data_ice = dat_ice, vars=set_mi, method=method, ncore

imputeObj <- impute(
  draw_high2,
  references = c("Arm 1" = "Arm 1", "Arm 2" = "Arm 1")
)

imputed_high2 <- extract_imputed_dfs(imputeObj)

anaObj <- analyse(
  imputeObj,
  vars = vars_an
)
```

Table: Estimates from copy jump to reference J2R imputation

```r
poolObj <- pool(anaObj)
as.data.frame(poolObj)
```

```
       parameter         est        se       lci        uci         pval
1      trt_Week 1 -0.04272539 0.6513099 -1.327240  1.2417894 9.477641e-01
2  lsm_ref_Week 1 -1.64363730 0.4593710 -2.549610 -0.7376649 4.367614e-04
3  lsm_alt_Week 1 -1.68636270 0.4593710 -2.592335 -0.7803903 3.118171e-04
4      trt_Week 2 -0.56805913 0.8694895 -2.283587  1.1474688 5.143679e-01
5  lsm_ref_Week 2 -3.29966757 0.6139776 -4.511086 -2.0882493 2.330387e-07
6  lsm_alt_Week 2 -3.86772670 0.6154448 -5.082087 -2.6533665 2.383863e-09
7      trt_Week 4 -1.23794505 0.9341607 -3.081702  0.6058116 1.868456e-01
8  lsm_ref_Week 4 -4.56691815 0.6705840 -5.890893 -3.2429436 1.703692e-10
9  lsm_alt_Week 4 -5.80486320 0.6575761 -7.102678 -4.5070482 1.090291e-15
10     trt_Week 6 -1.69994606 1.0170780 -3.707618  0.3077255 9.647476e-02
11 lsm_ref_Week 6 -5.21952895 0.7327090 -6.666472 -3.7725863 3.306987e-11
12 lsm_alt_Week 6 -6.91947501 0.7414260 -8.383999 -5.4549508 9.864655e-17
13     trt_Week 8 -1.72945373 1.1031354 -3.908286  0.4493783 1.189418e-01
14 lsm_ref_Week 8 -5.31618579 0.8094870 -6.916486 -3.7158856 9.139985e-10
15 lsm_alt_Week 8 -7.04563951 0.8266465 -8.680733 -5.4105457 2.995529e-14
```

Summary of results

Take the figure from the visualization part from day 1 to better understand what we have found here.

Table 3: Estimates for all methods

	Mean	Diff
Completers, Arm 1	-6.96	
Completers, Arm 2	-8.67	-1.71
MMRM, Arm 1	-5.24	
MMRM, Arm 2	-7.76	-2.52
LOCF, Arm 1	-4.22	
LOCF, Arm 1	-6.72	-2.50
J2R , Arm 1	-5.32	
J2R , Arm 1	-7.06	-1.74

Carpenter, Roger, J. R., and M. G. Kenward. 2013. “Analysis of Longitudinal Trials with Protocol Deviation: A Framework for Relevant, Accessible Assumptions, and Inference via Multiple Imputation.” *Journal of Biopharmaceutical Statistics*, October. https://doi.org/10.1080/10543406.2013.834911.

Gower-Page, Noci, C., and M. Wolbers. 2022. "Rbmi: A r Package for Standard and Reference-Based Multiple Imputation Methods." *Journal of Open Source Software*, June. https://doi.org/10.21105/joss.04251.

Kayembe, Jolani, M. T., and G. J. P. van Breukelen. 2022. "Imputation of Missing Covariates in Randomized Controlled Trials with Continuous Outcomes: Simple, Unbiased and Efficient Methods." *Journal of Biopharmaceutical Statistics*, September. https://doi.org/10.1080/10543406.2021.2011898.