Some Applications of Marginally Interpretable Linear Transformation Models for Clustered Observations

Luisa Barbanti
Universität Zürich

Torsten Hothorn
Universität Zürich

Abstract

Owing to their generality, transformation models can be used to set-up and compute many interesting regression models for discrete and continuous responses. This document focuses on the analysis of clustered observations. Marginal predictive distributions are defined by transformation models and their joint normal distribution depends on a structured covariance matrix. Applications with skewed, bounded, and survival continuous outcomes as well as binary and ordered categorical responses are presented. Data is analysed by a proof-of-concept implementation of parametric linear transformation models for clustered observations available in the **tram** add-on package to the R system for statistical computing.

Keywords: conditional mixed models, marginal models, marginal predictive distributions, survival analysis, categorical data analysis.

1. Introduction

The purpose of this document is to compare marginally interpretable linear transformation models for clustered observations (Barbanti and Hothorn 2024) to conventional conditional formulations of mixed-effects models where such an overlap exists. In addition, novel transformation models going beyond the capabilities of convential mixed-effects models are estimated and interpreted. A proof-of-concept implementation available in package **tram** (Hothorn *et al.*) is applied. The results presented in this document can be reproduced from the `mtram` demo

```
R> install.packages("tram")
R> demo("mtram", package = "tram")
```

2. Normal and Non-normal Mixed-effects Models

First we consider mixed-effects models for reaction times in the sleep deprivation study (Be-

Please cite this document as: Luisa Barbanti and Torsten Hothorn () Some Applications of Marginally Interpretable Linear Transformation Models for Clustered Observations. R package vignette version 1.4-8, URL https://doi.org/10.32614/CRAN.package.tram.

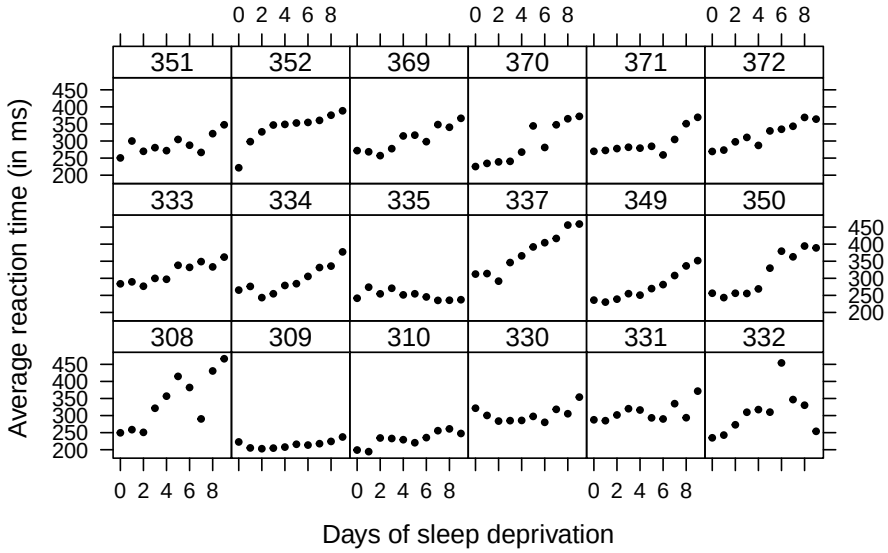


Figure 1: Sleep deprivation: Average reaction times to a specific task over several days of sleep deprivation for 18 subjects from Belenky *et al.* (2003).

lenky *et al.* 2003). The average reaction times to a specific task over several days of sleep deprivation are given for $i = 1, \ldots, N = 18$ subjects in Figure 1. The data are often used to illustrate conditional normal linear mixed-effects models with correlated random intercepts and slopes.

The classical normal linear random-intercept/random-slope model, treating the study participants as independent observations, is fitted by maximum likelihood to the data using the `lmer()` function from the **lme4** add-on package (Bates *et al.* 2015):

```
R> sleep_lmer <- lmer(Reaction ~ Days + (Days | Subject),
+                     data = sleepstudy, REML = FALSE)
```

The corresponding conditional model for subject i reads

$$\mathbb{P}(\text{Reaction} \leq y \mid \text{day}, i) = \Phi\left(\frac{y - \alpha - \beta\text{day} - \alpha_i - \beta_i\text{day}}{\sigma}\right), \quad (\alpha_i, \beta_i) \sim \text{N}_2(\mathbf{0}, \boldsymbol{G}(\boldsymbol{\gamma}))$$

with $\sigma^{-2}\boldsymbol{G} = \boldsymbol{\Lambda}(\boldsymbol{\gamma})\boldsymbol{\Lambda}(\boldsymbol{\gamma})^\top$ and

$$\boldsymbol{\Lambda}(\boldsymbol{\gamma}) = \begin{pmatrix} \gamma_1 & 0 \\ \gamma_2 & \gamma_3 \end{pmatrix}, \quad \boldsymbol{\gamma} = (\gamma_1, \gamma_2, \gamma_3)^\top.$$

The same model, however using the alternative parameterisation and an independent (of **lme4**, only the `update()` method for Cholesky factors is reused) gradient-based maximisation of the log-likelihood, is estimated in a two-step approach as

```
R> library("tram")
```

```
R> sleep_LM <- Lm(Reaction ~ Days, data = sleepstudy)
R> sleep_LMmer <- mtram(sleep_LM, ~ (Days | Subject), data = sleepstudy)
```

The first call to `Lm()` computes the equivalent of a normal linear regression model parameterised as a linear transformation model *ignoring* the longitudinal nature of the observations. The purpose if to set-up the necessary model infrastructure (model matrices, inverse link functions, etc.) and to compute reasonable starting values for the fixed effects. The second call to `mtram()` specifies the random effects structure (here a correlated pair of random intercept for subject and random slope for days) and optimises the likelihood for all model parameters $\vartheta_1, \tilde{\alpha}, \tilde{\beta}$, and $\boldsymbol{\gamma}$ in the model (here also looking at the conditional model for subject i)

$$\mathbb{P}(\text{Reaction} \leq y \mid \text{day}, i) = \Phi\left(\vartheta_1 y + \tilde{\alpha} - \tilde{\beta}\text{day} - \tilde{\alpha}_i - \tilde{\beta}_i\text{day}\right), \quad (\tilde{\alpha}_i, \tilde{\beta}_i) \sim \mathrm{N}_2(\mathbf{0}, \mathbf{\Lambda}(\boldsymbol{\gamma})\mathbf{\Lambda}(\boldsymbol{\gamma}))$$

that is, all fixed and random effect parameters are divided by the residual standard deviation σ (this is the reparameterisation applied by `Lm()`). Of course, the parameter ϑ_1, the inverse residual standard deviation, is ensured to be positive via an additional constraint in the optimiser maximising the log-likelihood.

The log-likelihoods of the two models fitted by `lmer()` and `mtram()` are very close

```
R> logLik(sleep_lmer)

'log Lik.' -876 (df=6)

R> logLik(sleep_LMmer)

'log Lik.' -876 (df=6)
```

Looking at the model coefficients, the two procedures lead to almost identical inverse residual standard deviations

```
R> (sdinv <- 1 / summary(sleep_lmer)$sigma)

[1] 0.03907

R> coef(sleep_LMmer)["Reaction"]

Reaction
 0.03908
```

and fixed effects (the slope can be interpreted as inverse coefficient of variation)

```
R> fixef(sleep_lmer) * c(-1, 1) * sdinv

(Intercept)        Days
     -9.824       0.409

R> coef(sleep_LMmer)[c("(Intercept)", "Days")]
```

```
(Intercept)        Days
    -9.8252      0.4091
```

The random-effect parameters $\boldsymbol{\gamma}$ are also reasonably close

```
R> sleep_lmer@theta

[1] 0.92919 0.01817 0.22264

R> coef(sleep_LMmer)[-(1:3)]

 gamma1  gamma2  gamma3
0.92897 0.01792 0.22290
```

Consequently, the variance-covariance and correlation matrices

```
R> sleep_LMmer$G * (1 / sdinv)^2

      [,1]  [,2]
[1,] 565.2 10.90
[2,]  10.9 32.75

R> cov2cor(sleep_LMmer$G * (1 / sdinv)^2)

        [,1]    [,2]
[1,] 1.00000 0.08014
[2,] 0.08014 1.00000

R> unclass(VarCorr(sleep_lmer))$Subject

            (Intercept)  Days
(Intercept)      565.48 11.06
Days              11.06 32.68
attr(,"class")
[1] "vcmat_us" "matrix"   "array"
attr(,"stddev")
(Intercept)        Days
     23.780       5.717
attr(,"correlation")
            (Intercept)    Days
(Intercept)     1.00000 0.08132
Days            0.08132 1.00000
```

are practically equivalent. This result indicates the correctness of the alternative implementation of normal linear mixed-effects models in the transformation model framework: `mtram()` reuses some infrastructure from **lme4** and **Matrix**, most importantly fast update methods for

Cholesky factors, but the likelihood and corresponding optimisation relies on an independent implementation. So why are we doing this? Because `mtram()` is able to deal with models or likelihoods not available in **lme4**, for example the likelihood for interval-censored observations.

Let's assume that the timing of the reaction times was less accurate than suggested by the numerical representation of the results. The following code

```
R> library("survival")
R> sleepstudy$Reaction_I <- with(sleepstudy, Surv(Reaction - 20, Reaction + 20,
+                                                 type = "interval2"))
R> sleepstudy$Reaction_I[1:5]

[1] [229.6, 269.6] [238.7, 278.7] [230.8, 270.8] [301.4, 341.4]
[5] [336.9, 376.9]
```

converts the outcome to interval-censored values, where each interval has length 40. The above mixed model can now be estimated by maximising the likelihood corresponding to interval-censored observations. We use `BoxCox()` with a linear transformation function, because it is easier to compute reasonable starting values:

```
R> sleep_LM_I <- BoxCox(Reaction_I ~ Days, data = sleepstudy, order = 1)
R> sleep_LMmer_I <- mtram(sleep_LM_I, ~ (Days | Subject), data = sleepstudy)
```

Of course, the log-likelihood changes (because this is a log-probability and not a log-density of a continuous distribution) but the parameter estimates are reasonably close

```
R> logLik(sleep_LMmer_I)

'log Lik.' -213.2 (df=6)

R> coef(sleep_LMmer_I)

Bs1(Reaction_I) Bs2(Reaction_I)            Days          gamma1
       -0.66039         4.72093         0.43675         0.99549
         gamma2          gamma3
        0.02379         0.26312

R> coef(sleep_LMmer)

(Intercept)    Reaction        Days      gamma1      gamma2      gamma3
   -9.82523     0.03908     0.40914     0.92897     0.01792     0.22290
```

The next question is if the normal assumption for reaction times is appropriate. In the transformation world, this assumption is simple to assess because we can easily (theoretically and in-silico) switch to the non-normal linear mixed-effects transformation model

$$\mathbb{P}(\text{Reaction} \leq y \mid \text{day}, i) = \Phi\left(h(y) - \tilde{\beta}\text{day} - \tilde{\alpha}_i - \tilde{\beta}_i\text{day}\right), \quad (\tilde{\alpha}_i, \tilde{\beta}_i) \sim \text{N}_2(\mathbf{0}, \mathbf{\Lambda}(\boldsymbol{\gamma})\mathbf{\Lambda}(\boldsymbol{\gamma}))$$

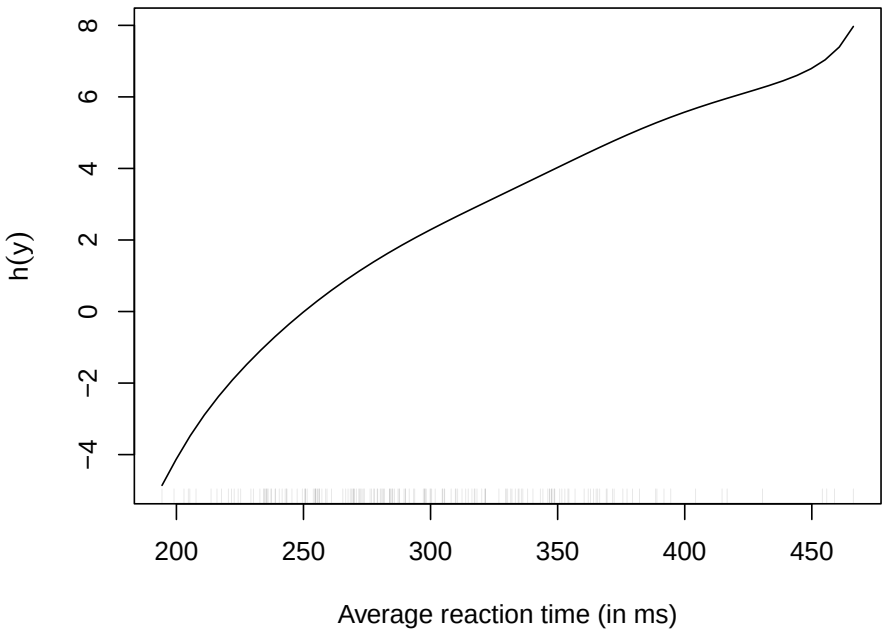


Figure 2: Sleep deprivation: Data-driven transformation $\hat{h}$ of average reaction times to sleep deprivation. The non-linearity induces a non-normal marginal distribution function of reaction times.

where $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$ represents a monotone non-decreasing transformation function. The function implementing such a more flexible model in named in honor of the first paper on the analysis of transformed responses by Box and Cox (1964) but it *does not* simply apply what is known as a Box-Cox transformation. Bernstein polynomials $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$ under suitable constraints (Hothorn *et al.* 2018) are applied instead by

```
R> sleep_BC <- BoxCox(Reaction ~ Days, data = sleepstudy)
R> sleep_BCmer <- mtram(sleep_BC, ~ (Days | Subject), data = sleepstudy)
R> logLik(sleep_BCmer)

'log Lik.' -857.7 (df=11)
```

The increase in the log-likelihood compared to the normal model is not a big surprise. Plotting the transformation function $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$ as a function of reaction time can help to assess deviations from normality because the latter assumption implies a linear transformation function. Figure 2 clearly indicates that models allowing a certain skewness of reaction times will provide a better fit to the data. This might also not come as a big surprise to experienced data analysts.

Such probit-type mixed-effects models have been studied before, mostly by merging a Box-Cox power transformation h with a grid-search over REML estimates (Gurka *et al.* 2006),

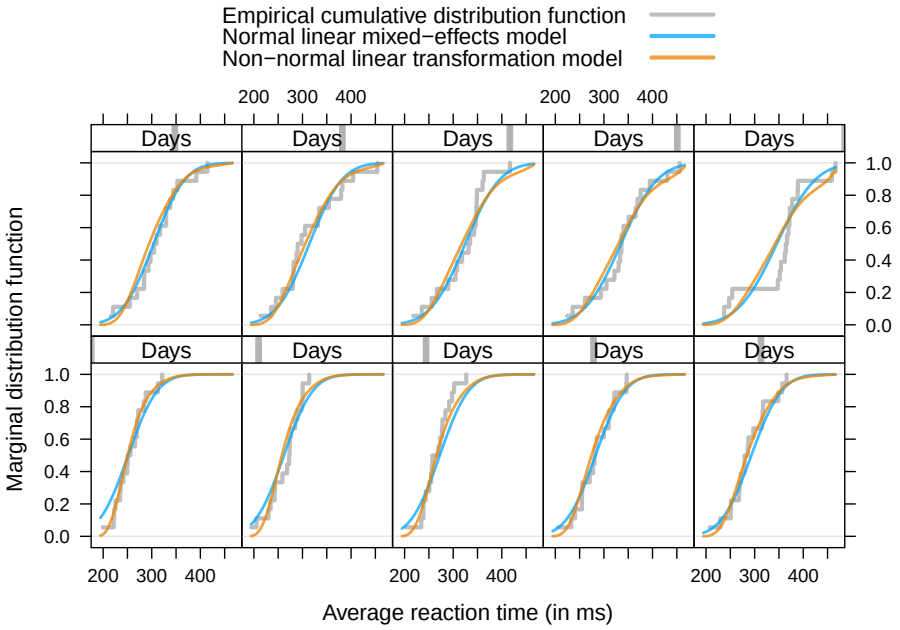


Figure 3: Sleep deprivation: Marginal distribution of reaction times, separately for each day of study participation. The grey step-function corresponds to the empirical cumulative distribution function, the blue line to the marginal cumulative distribution of a normal linear mixed-effects model, and the orange line to a non-normal linear mixed-effects transformation model.

a conditional likelihood (Hutmacher *et al.* 2011), or a grid-search maximising the profile likelihood (Maruo *et al.* 2017). Recently, Tang *et al.* (2018) and Wu and Wang (2019) proposed a monotone spline parameterisation of h in a Bayesian context. The model presented here was estimated by simultaneously maximising the log-likelihood (Barbanti and Hothorn 2024) with respect to the parameters $\boldsymbol{\vartheta}$, β, and $\boldsymbol{\gamma}$. For a linear Bernstein polynomial of order one, the models obtained with this approach and classical maximum likelihood estimation in normal linear mixed-effects models are equivalent (up to reparameterisation of β).

However, what does this finding mean in terms of a direct comparison of the model and the data? Looking at the marginal cumulative distribution functions of average reaction time conditional on days of sleep deprivation in Figure 3 one finds that the non-normal marginal transformation models provided a better fit to the marginal empirical cumulative distribution functions than the normal marginal models. Especially for short reaction times in the first week of sleep deprivation, the orange marginal cumulative distribution is much closer to the empirical cumulative distribution function representing the marginal distribution of reaction times at each single day of study participation.

It should be noted that the small positive correlation between random intercept and random slope observed in the normal linear mixed-effects model turned into a negative correlation in this non-normal model

```
R> cov2cor(sleep_BCmer$G)
```

```
        [,1]    [,2]
[1,]  1.0000 -0.1797
[2,] -0.1797  1.0000
```

What is the uncertainty associated with this parameter? The correlation is a non-linear function of $\boldsymbol{\gamma}$ and therefore the direct computation of confidence intervals questionable. However, we can extract an estimate of the covariance of the estimated model parameters from the model and, relying on the asymptotic normality of the maximum likelihood estimators, we can sample from the asymptotic distribution of the variance of the random intercept $\tilde{\alpha}$, the random slope $\tilde{\beta}$, and their correlation

```
R> library("mvtnorm")
R> VC <- vcov(sleep_BCmer)
R> idx <- (nrow(VC) - 2):nrow(VC)
R> Rcoef <- rmvnorm(1000, mean = coef(sleep_BCmer), sigma = VC)[,idx]
R> ret <- apply(Rcoef, 1, function(gamma) {
+      L <- matrix(c(gamma[1:2], 0, gamma[3]), nrow = 2)
+      V <- tcrossprod(L)
+      c(diag(V), cov2cor(V)[1,2])
+  })
```

The 95% confidence intervals

```
R> ### variance random intercept
R> quantile(ret[1,], c(.025, .5, .975))

  2.5%    50%  97.5%
0.7505 2.3150 4.7573

R> ### variance random slope
R> quantile(ret[2,], c(.025, .5, .975))

   2.5%     50%   97.5%
0.01799 0.04921 0.10334

R> ### correlation random intercept / random slope
R> quantile(ret[3,], c(.025, .5, .975))

   2.5%     50%   97.5%
-0.6485 -0.1799  0.4537
```

indicate rather strong unobserved heterogeneity affecting the intercept and less pronounced variability in the slope. There is only weak information about the correlation of the two random effects contained in the data.

The downside of this approach is that, although the model is nicely interpretable on the scale of marginal or conditional distribution functions, the direct interpretation of the fixed effect $\tilde{\beta}$ is not very straightforward because it corresponds to the conditional mean *after* transforming the outcome. This interpretability issue can be addressed by exchanging the probit link to a logit link

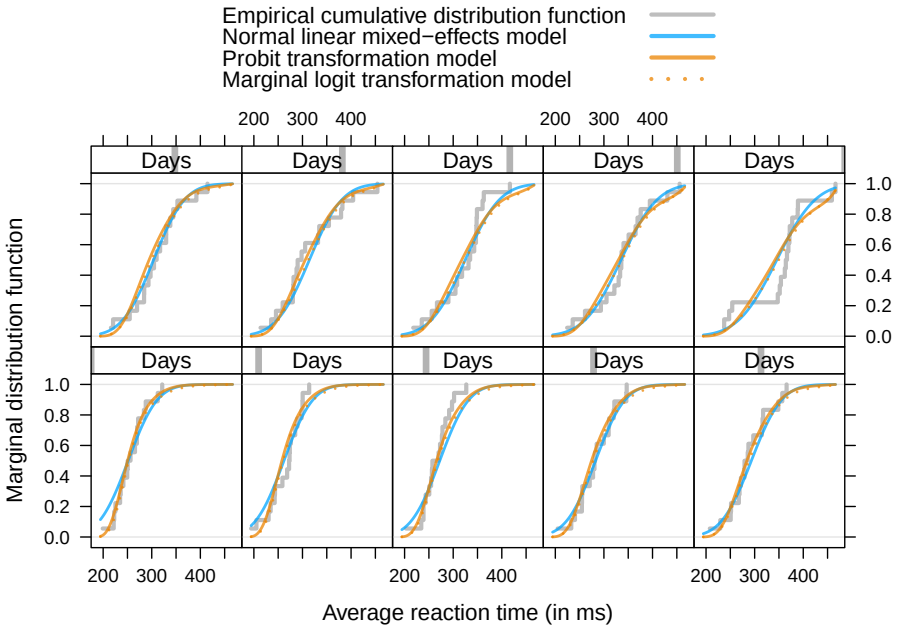


Figure 4: Sleep deprivation: Marginal distribution of reaction times, separately for each day of study participation. The grey step-function corresponds to the empirical cumulative distribution function, the blue line to the marginal cumulative distribution of a normal linear mixed-effects model, and the orange lines to a non-normal probit (solid) and marginal logit (dotted) transformation model.

```
R> sleep_C <- Colr(Reaction ~ Days, data = sleepstudy)
R> sleep_Cmer <- mtram(sleep_C, ~ (Days | Subject), data = sleepstudy)
R> logLik(sleep_Cmer)
```

```
'log Lik.' -858.6 (df=11)
```

Here, the in-sample log-likelihood increases compared to the probit model and the marginal distributions obtained from this model are shown in Figure 4. How to interpret models of this type is discussed in Section 4.

3. Models for Binary Outcomes

Here we compare different implementations of binary marginal and mixed models for the notoriously difficult toe nail data (Backer *et al.* 1998). The outcome was categorised to two levels (this being probably the root of all troubles, as quasi-separation issues have been reported by Sauter and Held 2016). A conditional density plot (Figure 5) suggests an improvement in both treatment groups over time, however with a more rapid advance in patients treated with terbinafine.

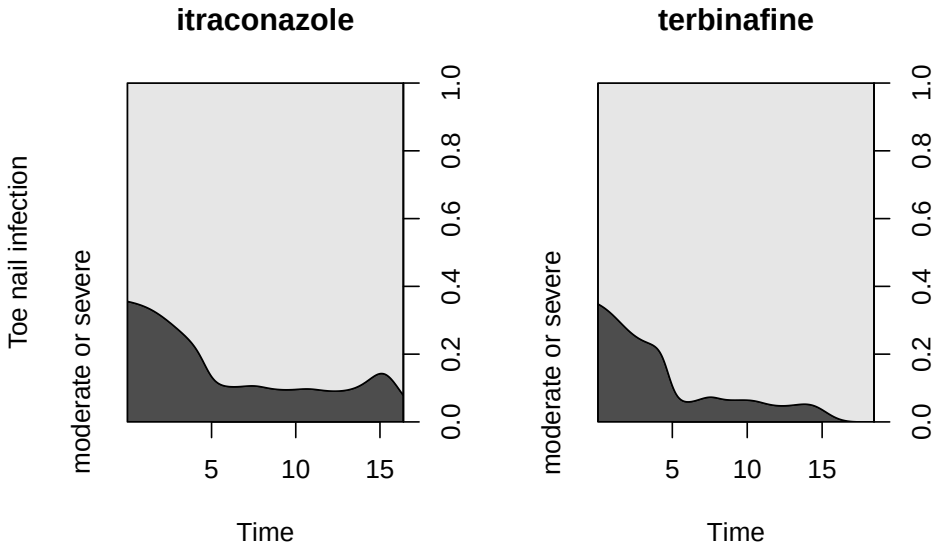


Figure 5: Toe nail data: Conditional density plot of two outcome classes (none or mild vs. moderate or severe) under two treatments.

3.1. Random Intercept Probit Models

We are first interested in binary probit models featuring fixed main and interaction effects β_1, β_2, and β_3 of treatment (itraconazole vs. terbinafine) and time. Subject-specific random intercept models were estimated by the `glmer` function from package **lme4** (Bates *et al.* 2015), by the `glmmTMB` function from package **glmmTMB** (Brooks *et al.* 2017), and by direct maximisation of the exact discrete log-likelihood given in Appendix B of Barbanti and Hothorn (2024).

The random intercept probit model fitted by Laplace and Adaptive Gauss-Hermite Quadrature (AGQ) approximations to the likelihood give quite different results:

```
R> ### Laplace
R> toenail_glmer_RI_1 <-
+      glmer(outcome ~ treatment * time + (1 | patientID),
+            data = toenail, family = binomial(link = "probit"),
+            nAGQ = 1)
R> summary(toenail_glmer_RI_1)

Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: binomial  ( probit )
Formula: outcome ~ treatment * time + (1 | patientID)
   Data: toenail

      AIC       BIC    logLik -2*log(L)  df.resid
   1286.1    1313.9    -638.1    1276.1      1903
```

```
Scaled residuals:
   Min     1Q Median     3Q    Max
 -3.52  -0.02   0.00   0.00  53.76

Random effects:
 Groups    Name        Variance Std.Dev.
 patientID (Intercept) 20.9     4.57
Number of obs: 1908, groups:  patientID, 294

Fixed effects:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)               -3.39426    0.21786  -15.58   <2e-16 ***
treatmentterbinafine      -0.02934    0.24949   -0.12     0.91
time                      -0.21795    0.00405  -53.79   <2e-16 ***
treatmentterbinafine:time -0.07080    0.00410  -17.28   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) trtmnt time
trtmnttrbnf -0.588
time         0.006  0.011
trtmnttrbn:  0.009 -0.017 -0.526
optimizer (Nelder_Mead) convergence code: 0 (OK)
Model failed to converge with max|grad| = 0.0188777 (tol = 0.002, component 1)
  See ?lme4::convergence and ?lme4::troubleshooting.

R> toenail_glmer_RI_1@theta

[1] 4.573

R> ### Adaptive Gaussian Quadrature
R> toenail_glmer_RI_2 <-
+      glmer(outcome ~ treatment * time + (1 | patientID),
+            data = toenail, family = binomial(link = "probit"),
+            nAGQ = 20)
R> summary(toenail_glmer_RI_2)

Generalized linear mixed model fit by maximum likelihood (Adaptive
  Gauss-Hermite Quadrature, nAGQ = 20) [glmerMod]
 Family: binomial  ( probit )
Formula: outcome ~ treatment * time + (1 | patientID)
   Data: toenail

      AIC       BIC    logLik -2*log(L)  df.resid
   1284.6    1312.3    -637.3    1274.6      1903
```

```
Scaled residuals:
   Min     1Q Median     3Q    Max
 -2.86  -0.19  -0.08   0.00  33.86

Random effects:
 Groups    Name        Variance Std.Dev.
 patientID (Intercept) 4.49     2.12
Number of obs: 1908, groups:  patientID, 294

Fixed effects:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                -0.9105     0.2288   -3.98  6.9e-05 ***
treatmentterbinafine       -0.1073     0.3073   -0.35    0.727
time                       -0.1913     0.0206   -9.29  < 2e-16 ***
treatmentterbinafine:time  -0.0633     0.0310   -2.04    0.041 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) trtmnt time
trtmnttrbnf -0.650
time        -0.185  0.212
trtmnttrbn:  0.192 -0.285 -0.611
```

```
R> toenail_glmer_RI_2@theta
```

```
[1] 2.118
```

Package **glmmTMB** optimises the Laplace approximation utilising the Template Model Builder **TMB** package:

```
R> library("glmmTMB")
R> toenail_glmmTMB_RI_3 <-
+      glmmTMB(outcome ~ treatment * time + (1 | patientID),
+           data = toenail, family = binomial(link = "probit"))
R> summary(toenail_glmmTMB_RI_3)
```

```
 Family: binomial  ( probit )
Formula:          outcome ~ treatment * time + (1 | patientID)
Data: toenail

      AIC       BIC    logLik -2*log(L)  df.resid
     1298      1326      -644      1288      1903

Random effects:
```

```
Conditional model:
 Groups    Name        Variance Std.Dev.
 patientID (Intercept) 4.42     2.1
Number of obs: 1908, groups:  patientID, 294

Conditional model:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                -1.1007     0.3227   -3.41  0.00065 ***
treatmentterbinafine       -0.1739     0.3539   -0.49  0.62310
time                       -0.1893     0.0207   -9.13  < 2e-16 ***
treatmentterbinafine:time  -0.0611     0.0309   -1.97  0.04834 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Surprisingly, this model is very close to the one obtained by AGQ and quite off from the Laplace implementation in **lme4** (`nAGQ = 1` means Laplace).

Because of the probit link, this binary generalised linear model is equivalent to a linear transformation model and we can thus use the exact likelihood implemented for the latter model in `mtram()` for parameter estimation (it is still a bit nasty to set-up a constant transformation function $h(y) = \alpha$, we plan to add a more convenient interface later)

```
R> m <- ctm(as.basis(~ outcome, data = toenail),
+           shifting = ~ treatment * time,
+           data = toenail, todistr = "Normal", negative = TRUE)
R> toenail_probit <- mlt(m, data = toenail,
+                        fixed = c("outcomemoderate or severe" = 0))
R> toenail_mtram_RI <-
+      mtram(toenail_probit, ~ (1 | patientID), data = toenail)
R> coef(toenail_mtram_RI)

             (Intercept)      treatmentterbinafine
                 0.91009                  -0.10710
                    time treatmentterbinafine:time
                -0.19125                  -0.06332
                  gamma1
                 2.11702
```

For this random intercept model, the exact likelihood is defined as a one-dimensional integral over the unit interval. We use sparse grids (Heiss and Winschel 2008; Ypma 2013) to approximate this integral. The integrand is defined by products of normal probabilities, which are approximated as described by Matić *et al.* (2018). It is important to note that this likelihood can be computed as accurately as necessary whereas alternative implementations rely on approximations of limited accuracy (at least for non-probit links).

The results (model parameters and likelihoods) are very close to those obtained by AGQ (**lme4**) or **glmmTMB**, indicating a very good quality the various approximations used. We can also compare the corresponding covariances

```
R> vcov(toenail_glmer_RI_2)

4 x 4 Matrix of class "dpoMatrix"
                          (Intercept) treatmentterbinafine       time
(Intercept)                 0.0523479            -0.045691 -0.0008721
treatmentterbinafine       -0.0456910             0.094432  0.0013398
time                       -0.0008721             0.001340  0.0004237
treatmentterbinafine:time   0.0013601            -0.002716 -0.0003894
                          treatmentterbinafine:time
(Intercept)                               0.0013601
treatmentterbinafine                     -0.0027162
time                                     -0.0003894
treatmentterbinafine:time                 0.0009595

R> vcov(toenail_mtram_RI)[1:4, 1:4]

                          (Intercept) treatmentterbinafine       time
(Intercept)                  0.052267             0.045693  0.0008720
treatmentterbinafine         0.045693             0.094460  0.0013348
time                         0.000872             0.001335  0.0004236
treatmentterbinafine:time   -0.001356            -0.002709 -0.0003895
                          treatmentterbinafine:time
(Intercept)                              -0.0013559
treatmentterbinafine                     -0.0027090
time                                     -0.0003895
treatmentterbinafine:time                 0.0009598
```

which are also in good agreement.

The marginal effects, that is, a marginal binary probit model, are given by the scaled conditional coefficients

```
R> cf <- coef(toenail_mtram_RI)
R> cf[2:4] / sqrt(1 + cf["gamma1"]^2)

     treatmentterbinafine                      time
                 -0.04574                  -0.08169
treatmentterbinafine:time
                 -0.02704
```

Such marginal effects can be estimated directly by generalised estimation equations (GEE). For the probit model, three models corresponding to different working correlations can be estimated for example by package **geepack**:

```
R> library("geepack")
R> gin <- geeglm(I((0:1)[outcome]) ~ treatment * time,
+                id = patientID, data = toenail, corstr = "independence",
```

```
+                family = binomial(link = "probit"))
R> gex <- geeglm(I((0:1)[outcome]) ~ treatment * time,
+                id = patientID, data = toenail, cor = "exchangeable",
+                family = binomial(link = "probit"))
R> gun <- geeglm(I((0:1)[outcome]) ~ treatment * time,
+                id = patientID, data = toenail, cor = "unstructured",
+                family = binomial(link = "probit"))
```

The effects are not very close to what we obtained earlier, and it seems the choice of the working correlations matters here:

```
R> cbind(mtram = cf[2:4] / sqrt(1 + cf["gamma1"]^2),
+        indep = coef(gin)[-1],
+        excha = coef(gex)[-1],
+        unstr = coef(gun)[-1])
```

```
                            mtram    indep    excha    unstr
treatmentterbinafine      -0.04574 -0.01100 -0.01476  0.01635
time                      -0.08169 -0.09278 -0.09290 -0.06894
treatmentterbinafine:time -0.02704 -0.03199 -0.03718 -0.04468
```

At least in biostatistics, the probit model is less popular than the logit model owing to the better interpretability of the fixed effects as conditional log-odds ratios in the latter. Thus, we replicate the SAS analysis reported in Chapter 10 of Molenberghs and Verbeke (2005)

```
R> gin <- geeglm(I((0:1)[outcome]) ~ treatment * time,
+                id = patientID, data = toenail, corstr = "independence",
+                family = binomial())
R> gex <- geeglm(I((0:1)[outcome]) ~ treatment * time,
+                id = patientID, data = toenail, cor = "exchangeable",
+                family = binomial())
R> gun <- geeglm(I((0:1)[outcome]) ~ treatment * time,
+                id = patientID, data = toenail, cor = "unstructured",
+                family = binomial())
```

Again, results are dependent on hyperparameters and also not in very good agreement with SAS output reported by Molenberghs and Verbeke (2005)

```
R> coef(gin)
```

```
              (Intercept)      treatmentterbinafine
               -0.5566273                -0.0005817
                     time treatmentterbinafine:time
               -0.1703078                -0.0672216
```

```
R> coef(gex)
```

```
             (Intercept)      treatmentterbinafine
                -0.58192                   0.00718
                    time treatmentterbinafine:time
                -0.17128                  -0.07773
```

```
R> coef(gun)
```

```
             (Intercept)      treatmentterbinafine
                -0.73962                   0.03730
                    time treatmentterbinafine:time
                -0.13190                  -0.08961
```

Alternatively, we can use the transformation approach to compute marginally interpretable time-dependent log-odds ratios from random intercept transformation logit models:

```
R> m <- ctm(as.basis(~ outcome, data = toenail),
+           shifting = ~ treatment * time,
+           data = toenail, todistr = "Logistic", negative = TRUE)
R> toenail_logit <- mlt(m, data = toenail,
+                       fixed = c("outcomemoderate or severe" = 0))
R> toenail_mtram_logit <- mtram(toenail_logit, ~ (1 | patientID),
+                               data = toenail)
```

It is important to note that this model is *not* a logistic mixed-effects model and thus we can't expect to obtain identical results from `glmer()` as it was (partially) the case for the probit model. The marginal log-odds ratios are

```
R> cf <- coef(toenail_mtram_logit)
R> cf[2:4] / sqrt(1 + cf["gamma1"]^2)
```

```
     treatmentterbinafine                      time
                -0.05944                  -0.14916
treatmentterbinafine:time
                -0.05915
```

and an asymptotic confidence interval for the temporal treatment effect can be obtained from a small simulation

```
R> S <- rmvnorm(10000, mean = coef(toenail_mtram_logit),
+               sigma = vcov(toenail_mtram_logit))
R> (ci <- quantile(S[,"treatmentterbinafine:time"] / sqrt(1 + S[, "gamma1"]^2),
+                  prob = c(.025, .975)))
```

```
     2.5%     97.5%
-0.114722 -0.008042
```

The interval indicates a marginally significant treatment effect, that is, an odds ratio for none or mild symptoms of 0.94 per month, with 95% confidence interval $(0.89, 0.99)$.

A direct comparison of the marginal log-odds ratios with GEE results highlight the discrepancies

```
R> cbind(mtram = cf[2:4] / sqrt(1 + cf["gamma1"]^2),
+        indep = coef(gin)[-1],
+        excha = coef(gex)[-1],
+        unstr = coef(gun)[-1])
```

```
                             mtram      indep    excha    unstr
treatmentterbinafine      -0.05944 -0.0005817  0.00718  0.03730
time                      -0.14916 -0.1703078 -0.17128 -0.13190
treatmentterbinafine:time -0.05915 -0.0672216 -0.07773 -0.08961
```

Following Molenberghs and Verbeke (2005), we use the GEE with unstructured working correlation to compute a confidence interval for the temporal treatment effect on the odds ratio scale

```
R> exp(coef(gun)["treatmentterbinafine:time"] +
+      c(-1, 1) * qnorm(.975) * sqrt(diag(vcov(gun)))["treatmentterbinafine:time"])
```

```
[1] 0.8319 1.0049
```

In respect of this temporal treatment effect, GEE and marginal transformation models provide similar results, but the "significance" of the temporal treatment effect seems to be affected by numerical issues arising when fitting such models to this data.

From the marginal transformation model, we can compute and plot marginally interpretable probabilities and odds ratios over time

```
R> tmp <- toenail_logit
R> cf <- coef(tmp)
R> cf <- cf[names(cf) != "outcomemoderate or severe"]
R> sdrf <- rev(coef(toenail_mtram_logit))[1]
R> cf <- coef(toenail_mtram_logit)[names(cf)] / sqrt(sdrf^2 + 1)
R> cf <- c(cf[1], "outcomemoderate or severe" = 0, cf[-1])
R> coef(tmp) <- cf
R> time <- 0:180/10
R> treatment <- sort(unique(toenail$treatment))
R> nd <- expand.grid(time = time, treatment = treatment)
R> nd$prob_logit <- predict(tmp, newdata = nd, type = "distribution")[1,]
R> nd$odds <- exp(predict(tmp, newdata = nd, type = "trafo")[1,])
```

We can also sample from the distribution of the maximum likelihood estimators to obtain an idea about the uncertainty (Figure 6).

From the logit and probit models, we can also obtain marginally interpretable probabilities as (probit)

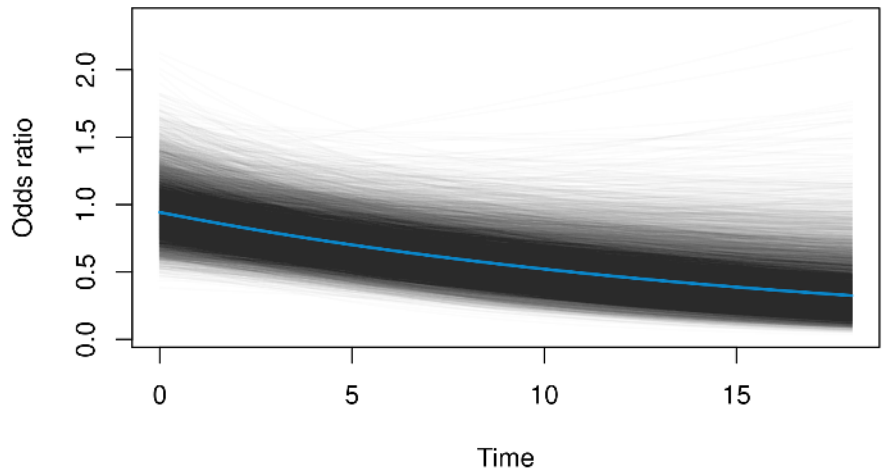


Figure 6: Toe nail data: Marginal odds ratio over time (from a logistic random intercept model). The blue line represents the maximum likelihood estimator, the grey lines are samples from the corresponding distribution.

```
R> tmp <- toenail_logit
R> cf <- coef(tmp)
R> cf <- cf[names(cf) != "outcomemoderate or severe"]
R> sdrf <- rev(coef(toenail_mtram_logit))[1]
R> cf <- coef(toenail_mtram_logit)[names(cf)]
R> cf <- c(cf[1], "outcomemoderate or severe" = 0, cf[-1])
R> coef(tmp) <- cf
R> pr <- predict(tmp, newdata = nd, type = "distribution")[1,]
R> nd$prob_logit <- pnorm(qnorm(pr) / sdrf)
```

and (logit)

```
R> tmp <- toenail_probit
R> cf <- coef(tmp)
R> cf <- cf[names(cf) != "outcomemoderate or severe"]
R> sdrf <- rev(coef(toenail_mtram_RI))[1]
R> cf <- coef(toenail_mtram_RI)[names(cf)] / sqrt(sdrf^2 + 1)
R> cf <- c(cf[1], "outcomemoderate or severe" = 0, cf[-1])
R> coef(tmp) <- cf
R> nd$prob_probit <- predict(tmp, newdata = nd, type = "distribution")[1,]
```

The marginal time-dependent probabilities obtained from all three models are very similar as shown in Figure 7.

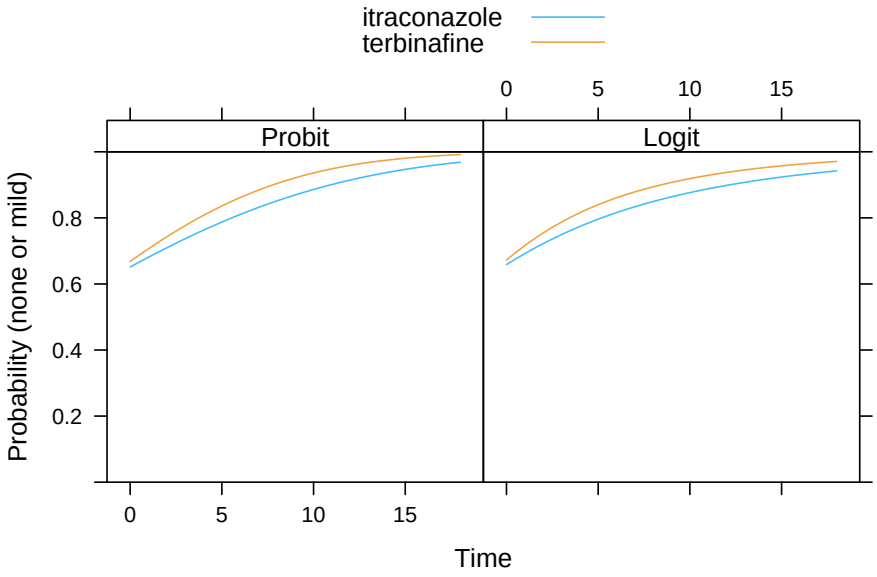


Figure 7: Toe nail data: Comparison of marginal probabilities obtained from a probit linear mixed-effects model and a logistic transformation model with marginal log-odds ratio treatment effect.

3.2. Random Intercept / Random Slope Models

Things get a bit less straightforward when a random slope is added to the model. We switch back to the probit link allowing comparison of our implementation with other packages. Some implementations do not allow clusters consisting of a single observation, so we remove patients without follow-up

```
R> (rlev <- levels(toenail$patientID)[xtabs(~ patientID,
+                                           data = toenail) == 1])

[1] "45"  "48"  "63"  "99"  "377"

R> toenail_gr1 <- subset(toenail, !patientID %in% rlev)
R> toenail_gr1$patientID <- toenail_gr1$patientID[, drop = TRUE]
```

The two implementations of the Laplace approximation in packages **lme4**

```
R> toenail_glmer_RS <-
+      glmer(outcome ~ treatment * time + (1 + time | patientID),
+            data = toenail_gr1, family = binomial(link = "probit"))
R> summary(toenail_glmer_RS)

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

```
Formula: outcome ~ treatment * time + (1 + time | patientID)
   Data: toenail_gr1

     AIC      BIC   logLik deviance df.resid
   985.8   1024.7   -485.9    971.8     1896

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.85421 -0.00210 -0.00037  0.00000  2.35828

Random effects:
 Groups    Name        Variance Std.Dev. Corr
 patientID (Intercept) 118.433  10.883
           time          3.305   1.818   -0.90
Number of obs: 1903, groups:  patientID, 289

Fixed effects:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)               -4.30119    0.26361 -16.317   <2e-16 ***
treatmentterbinafine       0.05419    0.34652   0.156   0.8757
time                      -0.06791    0.07846  -0.866   0.3867
treatmentterbinafine:time -0.23478    0.13885  -1.691   0.0909 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
            (Intr) trtmnt time
trtmnttrbnf -0.662
time        -0.453  0.342
trtmnttrbn:  0.270 -0.438 -0.335
```

```
R> toenail_glmer_RS@theta

[1] 10.882696 -1.635963  0.793088
```

and **glmmTMB**

```
R> toenail_glmmTMB_RS_1 <-
+      glmmTMB(outcome ~ treatment * time + (1 + time | patientID),
+           data = toenail_gr1, family = binomial(link = "probit"))
R> summary(toenail_glmmTMB_RS_1)

 Family: binomial  ( probit )
Formula:          outcome ~ treatment * time + (1 + time | patientID)
Data: toenail_gr1

     AIC      BIC   logLik deviance df.resid
```

```
   962.0   1000.8   -474.0    948.0     1896

Random effects:

Conditional model:
 Groups    Name        Variance Std.Dev. Corr
 patientID (Intercept) 121.185  11.008
           time          3.512   1.874   -0.90
Number of obs: 1903, groups:  patientID, 289

Conditional model:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)               -4.29367    0.26699 -16.082   <2e-16 ***
treatmentterbinafine       0.05612    0.35074   0.160   0.8729
time                      -0.07152    0.08140  -0.879   0.3796
treatmentterbinafine:time -0.24147    0.14454  -1.671   0.0948 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

are in good agreement.

The optimisation of the exact discrete likelihood in the transformation framework gives

```
R> m <- ctm(as.basis(~ outcome, data = toenail_gr1),
+           shifting = ~ treatment * time,
+           data = toenail, todistr = "Normal", negative = TRUE)
R> toenail_probit <- mlt(m, data = toenail_gr1,
+                        fixed = c("outcomemoderate or severe" = 0))
R> toenail_mtram_RS <-
+      mtram(toenail_probit, ~ (1 + time | patientID),
+            data = toenail_gr1)
R> logLik(toenail_mtram_RS)

'log Lik.' -545.1162 (df=7)

R> coef(toenail_mtram_RS)

              (Intercept)      treatmentterbinafine
                1.5782656                 0.2685859
                     time treatmentterbinafine:time
               -0.5338050                -0.1845690
                   gamma1                    gamma2
                5.2238567                -0.3726468
                   gamma3
                0.5297758
```

Here, substantial differences for all parameters can be observed. Because the parameters have the same meaning in all three implementations, we can compare the three models in light

	RI				RI + RS		
	`glmer` L	`glmer` AGQ	`glmmTMB` L	(7)	`glmer` L	`glmmTMB` L	(7)
α	-3.39	-0.91	-1.10	0.91	-4.30	-4.30	1.58
β_1	-0.03	-0.11	-0.17	-0.11	0.05	0.05	0.27
β_2	-0.22	-0.19	-0.19	-0.19	-0.07	-0.07	-0.53
β_3	-0.07	-0.06	-0.06	-0.06	-0.23	-0.23	-0.18
γ_1	4.57	2.12	2.10	2.12	10.88	11.01	5.22
γ_2	0.00	0.00	0.00	0.00	-1.64	-1.68	-0.37
γ_3	0.00	0.00	0.00	0.00	0.79	0.83	0.53
LogLik	-675.12	-637.28	-638.49	-637.28	-628.11	-630.65	-545.12
Time (sec)	3.21	1.98	0.88	1.68	6.16	1.26	7.67

Table 1: Toe nail data. Binary probit models featuring fixed intercepts α, treatment effects β_1, time effects β_2, and time-treatment interactions β_3 are compared. Random intercept (RI) and random intercept/random slope (RI + RS) models were estimated by the Laplace (L) and Adaptive Gauss-Hermite Quadrature (AGQ) approximations to the likelihood (implemented in packages **lme4** and **glmmTMB**). In addition, the exact discrete log-likelihood (7) was used for model fitting and evaluation (the in-sample log-likelihood (7) for all models and timings of all procedures are given in the last two lines).

of the exact discrete log-likelihood (Equation 6 in Barbanti and Hothorn 2024) evaluated at these parameters. The results are given in Table 1. For the random intercept models, AGQ, Laplace, and the discrete log-likelihood give the same results, the Laplace approximation seemed to fail. It was not possible to apply the AGQ approach to the random intercept / random slope model. The two implementations of the Laplace approximation in packages **lme4** and **glmmTMB** differed for the random intercept model but agreed for the random intercept / random slope model. The log-likelihood obtained by direct maximisation of (7) resulted in the best fitting model with the least extreme parameter estimates. Computing times for all procedures were comparable.

4. Proportional Odds Models for Bounded Responses

Manuguerra and Heller (2010) proposed a mixed-effects model for bounded responses whose fixed effects can be interpreted as log-odds ratios. We fit a transformation model to data from a randomised controlled trial on chronic neck pain treatment (Chow *et al.* 2006). The data are visualised in Figure 8. Subjective neck pain levels were assessed on a visual analog scale, that is, on a bounded interval.

Manuguerra and Heller (2010) suggested the conditional model

$$\begin{aligned}&\text{logit}(\mathbb{P}(\text{pain} \le y \mid \text{treatment}, \text{time}, i)) = \\ &\quad h(y) + \beta_{\text{Active}} + \beta_{\text{7 weeks}} + \beta_{\text{12 weeks}} + \beta_{\text{7 weeks, Active}} + \beta_{\text{12 weeks, Active}} + \alpha_i\end{aligned}$$

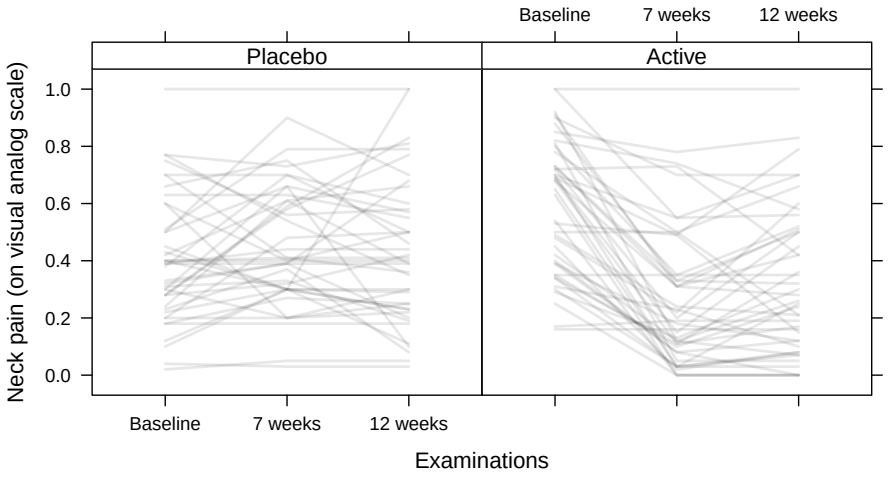


Figure 8: Neck pain: Trajectories of neck pain assessed on a visual analog scale with and without low-level laser therapy.

with random intercepts $\tilde{\alpha}_i$ such that the odds at baseline, for example, are given by

$$\frac{\mathbb{P}(\text{pain} \leq y \mid \text{Active}, \text{baseline}, i)}{\mathbb{P}(\text{pain} > y \mid \text{Active}, \text{baseline}, i)} = \exp(\beta_{\text{Active}}) \frac{\mathbb{P}(\text{pain} \leq y \mid \text{Placebo}, \text{baseline}, i)}{\mathbb{P}(\text{pain} > y \mid \text{Placebo}, \text{baseline}, i)}$$

```
R> library("ordinalCont")

R> neck_ocm <- ocm(vas ~ laser * time + (1 | id), data = pain_df,
+                  scale = c(0, 1))
```

The results

```
R> summary(neck_ocm)

Call:
ocm(formula = vas ~ laser * time + (1 | id), data = pain_df,
    scale = c(0, 1))

Random effects:
         Name Variance Std.Dev.
 Intercept|id    5.755    2.399

Coefficients:
                         Estimate StdErr t.value p.value
laserActive                -2.079  0.651   -3.20  0.0019 **
time7 weeks                -0.604  0.357   -1.69  0.0947 .
time12 weeks               -0.238  0.364   -0.65  0.5144
laserActive:time7 weeks     4.408  0.561    7.86 7.6e-12 ***
```

```
laserActive:time12 weeks    3.386  0.539    6.28 1.2e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

suggest that there is a difference at baseline; the pain distribution of subjects in the placebo group on the odds scale is only 12.5% of the odds in the active group for any cut-off y:

```
R> exp(cbind(coef(neck_ocm)[2:6], confint(neck_ocm)[2:6,]))
```

```
                                    2.5 %   97.5 %
laserActive                0.1250  0.03493   0.4475
time7 weeks                0.5468  0.27138   1.1018
time12 weeks               0.7882  0.38644   1.6075
laserActive:time7 weeks   82.1194 27.36208 246.4577
laserActive:time12 weeks  29.5455 10.26786  85.0162
```

In contrast, there seems to be a very large treatment effect (at week 7, the odds in the placebo group is 0.55 times larger than in the active group. This levels off after 12 weeks, but the effect is still significant at the 5% level.

For comparison, we can fit a conditional mixed-effects transformation model with a different parametrisation of the transformation function h using a Laplace approximation of the likelihood (Tamási *et al.* 2022):

```
R> library("tramME")
R> neck_ColrME <- ColrME(vas ~ laser * time + (1 | id), data = pain_df,
+                        bounds = c(0, 1), support = c(0, 1))
```

and coefficients

```
R> exp(coef(neck_ColrME))
```

```
             laserActive               time7 weeks              time12 weeks
                  0.1040                    0.5185                    0.7806
 laserActive:time7 weeks laserActive:time12 weeks
                130.6998                  41.9852
```

The model is the same as `neck_ocm`, but the parameter estimates for log-odds ratios differ quite substantially due to an alternative parameterisation of h and due to different estimation procedures being applied.

Our marginally interpretable transformation model with the same transformation function as the model `neck_ColrME` but with a completely different model formulation and optimisation procedure for maximising the log-likelihood, can be estimated by

```
R> neck_Colr <- Colr(vas ~ laser * time, data = pain_df,
+                    bounds = c(0, 1), support = c(0, 1),
+                    extrapolate = TRUE)
R> neck_Colrmer <- mtram(neck_Colr, ~ (1 | id), data = pain_df)
```

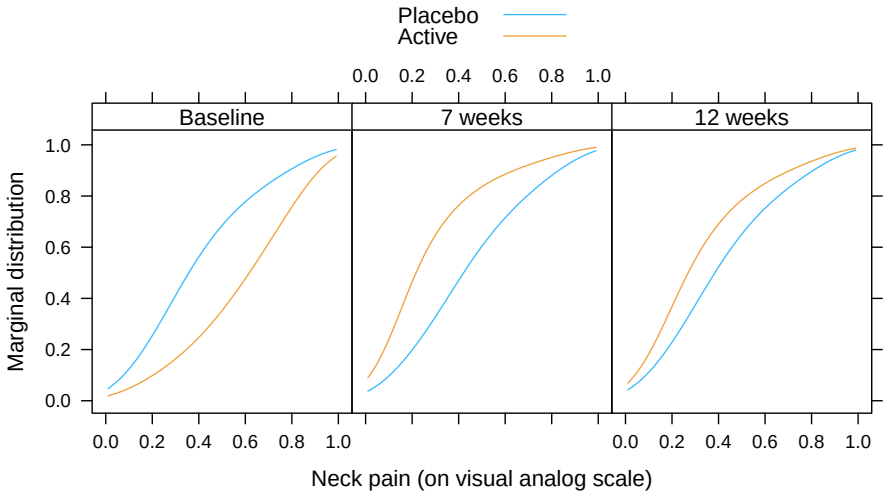


Figure 9: Neck pain: Marginal distribution functions of chronic neck pain evaluated at three different time points under placebo or active low-level laser therapy.

Based on this model, it is possible to derive the marginal distribution functions in the two groups, see Figure 9.

We sample from the joint normal distribution of the maximum likelihood estimators $\hat{\vartheta}_1, \ldots, \hat{\vartheta}_7$, $\hat{\beta}_{\text{Active}}, \hat{\beta}_{7 \text{ weeks}}, \hat{\beta}_{12 \text{ weeks}}, \hat{\beta}_{7 \text{ weeks, Active}}, \hat{\beta}_{12 \text{ weeks, Active}}, \hat{\alpha}_i$ and compute confidence intervals for the marginal treatment effect after 7 and 12 weeks

```
R> S <- vcov(neck_Colrmer)
R> rbeta <- rmvnorm(10000, mean = coef(neck_Colrmer), sigma = S)
R> s <- rbeta[, ncol(rbeta)]
R> rbeta <- rbeta[,-ncol(rbeta)] / sqrt(s^2 + 1)
R> t(apply(rbeta[, 8:12], 2, function(x) {
+    quantile(exp(x),prob = c(.025, .5, .975))}))
```

```
                            2.5%     50%   97.5%
laserActive               0.1159  0.2453  0.5119
time7 weeks               0.4465  0.6920  1.0518
time12 weeks              0.5517  0.8555  1.3230
laserActive:time7 weeks   7.9384 15.6122 33.7558
laserActive:time12 weeks  4.3424  8.4563 17.5138
```

Because the model `neck_Colrmer` has a marginal interpretation, we can derive the marginal probabilistic index and corresponding confidence intervals for the three time points as follows. In this case, the marginal probabilistic index obtained from model `neck_Colrmer` is the probability that, for a randomly selected patient in the treatment group, the neck pain score at time t is higher than the score for a subject in the placebo group randomly selected at the same time point.

There are two possible ways to compute the marginal probabilistic index. First, we consider the standardised version of the marginal treatment effects, that is:

```
R> beta <- coef(neck_Colrmer)[8:12]
R> alpha <- coef(neck_Colrmer)[13]
R> (std_beta <- cbind(beta / sqrt(1 + alpha^2)))
```

```
                            [,1]
laserActive              -1.4103
time7 weeks              -0.3701
time12 weeks             -0.1603
laserActive:time7 weeks   2.7557
laserActive:time12 weeks  2.1411
```

Then we compute the marginal treatment effect for weeks $0, 7, 12$ by multiplying the shift vector with the following contrast matrix

```
R> ctr_mat <- matrix(c(1, 0, 0, 0, 0,
+                      1, 0, 0, 1, 0,
+                      1, 0, 0, 0, 1), nrow = 3, byrow = TRUE)
R> ctr_mat %*% std_beta
```

```
        [,1]
[1,] -1.4103
[2,]  1.3453
[3,]  0.7308
```

We simulate from the asymptotic distribution of the parameters to obtain an empirical 95% confidence interval and pass it to the `PI` function by specifying the correct link function

```
R> (ci_emp <- t(apply(ctr_mat %*% t(rbeta[, 8:12]), 1, function(x) {
+    quantile(x, prob = c(.025, .5, .975))})))
```

```
          2.5%     50%   97.5%
[1,] -2.154929 -1.4053 -0.6697
[2,]  0.604925  1.3399  2.1367
[3,] -0.002496  0.7303  1.5116
```

```
R> PI(-ci_emp, link = "logistic")
```

```
       2.5%    50%  97.5%
[1,] 0.8115 0.7198 0.6100
[2,] 0.4004 0.2892 0.1904
[3,] 0.5004 0.3804 0.2658
```

Alternatively, we can compute the probabilistic index by passing a `Colr` model to the `PI` function. However, we have to make sure that the marginal model has the correct coefficients as obtained by standardising the coefficients from the `mtram` model:

```
R> nd <- expand.grid(time = unique(pain_df$time),
+                    laser = unique(pain_df$laser))
R> neck_Colr_marg <- neck_Colr
R> neck_Colr_marg$coef <- coef(neck_Colrmer)[1:12] /
+                         sqrt(coef(neck_Colrmer)[13]^2 + 1)
R> (neck_Colr_PI <- PI(neck_Colr_marg, newdata = nd[1:3, ],
+                       reference = nd[4:6, ],
+                       one2one = TRUE, conf.level = .95))[1:3, 1:3]

    Estimate    lwr    upr
4-1   0.7205 0.5841 0.8278
5-2   0.2885 0.1749 0.4327
6-3   0.3803 0.2446 0.5354
```

At baseline, we obtain a probabilistic index of 0.72. After 7 weeks, its value is 0.29 and after 12 weeks 0.38. These values reflect the effect of the low-level laser therapy for patients in the treatment group.

Of course, the confidence intervals for the estimates of the probabilistic index differ slightly across the two methods, but the point estimates coincide.

5. Marginally Interpretable Weibull and Cox Models

The CAO/ARO/AIO-04 randomised clinical trial (Rödel *et al.* 2015) compared Oxaliplatin added to fluorouracil-based preoperative chemoradiotherapy and postoperative chemotherapy to the same therapy using fluorouracil only for rectal cancer patients. Patients were randomised in the two treatment arms by block randomisation taking the study center, the lymph node involvement (negative, positive), and tumour grading (T1-3 vs. T4) into account. The primary endpoint was disease-free survival, defined as the time between randomisation and non-radical surgery of the primary tumour (R2 resection), locoregional recurrence after R0/1 resection, metastatic disease or progression, or death from any cause, whichever occurred first. The observed outcomes are a mix of exact dates (time to death or incomplete removal of the primary tumour), right-censoring (end of follow-up or drop-out), and interval-censoring (local or distant metastases). We are interested in a clustered Cox or Weibull model for interval-censored survival times. The survivor functions, estimated separately for each of the four strata defined by lymph node involvement and tumour grading, are given in Figure 10.

The implementation of marginally interpretable linear transformation models is currently not able to deal with mixed exact and censored outcomes in the same cluster. We therefore recode exact event times as being interval-censored by adding a 4-day window to each exact event time (variable `iDFS2`).

```
R> ### convert "exact" event dates to interval-censoring (+/- one day)
R> tmp <- CAOsurv$iDFS
R> exact <- tmp[,3] == 1
R> tmp[exact,2] <- tmp[exact,1] + 2
R> tmp[exact,1] <- pmax(tmp[exact,1] - 2, 0)
```

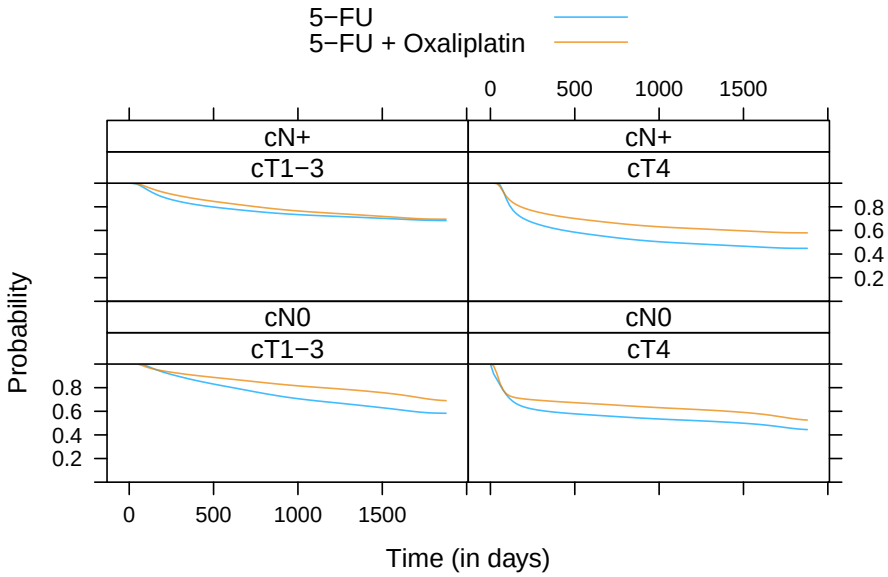


Figure 10: Rectal cancer: Distribution of disease-free survival times for two treatments in the four strata defined by lymph node involvement (negative or positive) and tumor grading (T1-3 or T4).

```
R> tmp[exact,3] <- 3
R> CAOsurv$iDFS2 <- tmp
```

We start with the random intercept model

$$\mathbb{P}(Y > y \mid \text{treatment}) = \exp\left(-\exp\left(\frac{\vartheta_1 + \vartheta_2 \log(y) - \beta_{\text{5-FU + Ox}}}{\sqrt{\gamma_1^2 + 1}}\right)\right)$$

assuming a marginal Weibull model whose effects are scaled depending on the variance γ_1^2 of a block-specific (interaction of lymph node involvement, tumor grading, and study center) random intercept:

```
R> CAO_SR <- Survreg(iDFS2 ~ randarm, data = CAOsurv)
R> CAO_SR_mtram <- mtram(CAO_SR, ~ (1 | Block), data = CAOsurv)
R> logLik(CAO_SR_mtram)

'log Lik.' -2082 (df=4)

R> (cf <- coef(CAO_SR_mtram))

               (Intercept)                log(iDFS2)
                   -6.3072                    0.7424
randarm5-FU + Oxaliplatin                    gamma1
                    0.2331                    0.1696
```

```
R> (OR <- exp(-cf["randarm5-FU + Oxaliplatin"] / sqrt(cf["gamma1"]^2 + 1)))

randarm5-FU + Oxaliplatin
                   0.7947
```

We are, of course, interested in the marginal treatment effect, that is, the hazards ratio

$$\exp\left(-\beta_{\text{5-FU + Ox}}/\sqrt{\gamma_1^2 + 1}\right).$$

We simply sample from the joint normal distribution of the maximum likelihood estimators $\hat{\vartheta}_1, \hat{\vartheta}_2, \hat{\beta}_{\text{5-FU + Ox}}, \hat{\gamma}_1$ and compute confidence intervals for the marginal treatment effect 0.79 as

```
R> S <- vcov(CAO_SR_mtram)
R> # sqrt(diag(S))
R> rbeta <- rmvnorm(10000, mean = coef(CAO_SR_mtram),
+                   sigma = S)
R> s <- rbeta[, ncol(rbeta)]
R> rbeta <- rbeta[, -ncol(rbeta)] / sqrt(s^2 + 1)
R> quantile(exp(-rbeta[, ncol(rbeta)]), prob = c(.025, .5, .975))

  2.5%    50%  97.5%
0.6449 0.7971 0.9820
```

In a next step, we stratify with respect to lymph node involvement and tumor grading: For each of the four strata, the parameters ϑ_1 and ϑ_2 are estimated separately:

```
R> CAO_SR_2 <- Survreg(iDFS2 | 0 + strat_n:strat_t ~ randarm, data = CAOsurv)
R> CAO_SR_2_mtram <- mtram(CAO_SR_2, ~ (1 | Block), data = CAOsurv)
R> logLik(CAO_SR_2_mtram)

'log Lik.' -2068 (df=10)

R> (cf <- coef(CAO_SR_2_mtram))

(Intercept):strat_ncN0:strat_tcT1-3  log(iDFS2):strat_ncN0:strat_tcT1-3
                            -7.8972                               0.9604
(Intercept):strat_ncN+:strat_tcT1-3  log(iDFS2):strat_ncN+:strat_tcT1-3
                            -6.2306                               0.7210
  (Intercept):strat_ncN0:strat_tcT4    log(iDFS2):strat_ncN0:strat_tcT4
                            -3.0493                               0.3715
  (Intercept):strat_ncN+:strat_tcT4    log(iDFS2):strat_ncN+:strat_tcT4
                            -4.8260                               0.6222
          randarm5-FU + Oxaliplatin                               gamma1
                             0.2243                               0.1491

R> (OR_2 <- exp(-cf["randarm5-FU + Oxaliplatin"] / sqrt(cf["gamma1"]^2 + 1)))
```

```
randarm5-FU + Oxaliplatin
                  0.8011
```

The corresponding confidence interval for the marginal treatment effect is then

```
  2.5%    50%  97.5%
0.6530 0.8039 0.9834
```

We now relax the Weibull assumption in the Cox model

$$\mathbb{P}(Y > y \mid \text{treatment}) = \exp\left(-\exp\left(\frac{\boldsymbol{a}(\log(y))^\top \boldsymbol{\vartheta} + \beta_{\text{5-FU + Ox}}}{\sqrt{\gamma_1^2 + 1}}\right)\right)$$

(note the positive sign of the treatment effect).

```
R> CAO_Cox_2 <- Coxph(iDFS2 | 0 + strat_n:strat_t ~ randarm, data = CAOsurv)
R> logLik(CAO_Cox_2)

'log Lik.' -2020 (df=29)

R> CAO_Cox_2_mtram <- mtram(CAO_Cox_2, ~ (1 | Block), data = CAOsurv)
R> logLik(CAO_Cox_2_mtram)

'log Lik.' -2020 (df=30)

R> coef(CAO_Cox_2_mtram)

Bs1(iDFS2):strat_ncN0:strat_tcT1-3 Bs2(iDFS2):strat_ncN0:strat_tcT1-3
                          -26.5160                           -4.0225
Bs3(iDFS2):strat_ncN0:strat_tcT1-3 Bs4(iDFS2):strat_ncN0:strat_tcT1-3
                           -4.0225                           -4.0225
Bs5(iDFS2):strat_ncN0:strat_tcT1-3 Bs6(iDFS2):strat_ncN0:strat_tcT1-3
                           -2.0622                           -1.2510
Bs7(iDFS2):strat_ncN0:strat_tcT1-3 Bs1(iDFS2):strat_ncN+:strat_tcT1-3
                           -0.6983                           -8.7081
Bs2(iDFS2):strat_ncN+:strat_tcT1-3 Bs3(iDFS2):strat_ncN+:strat_tcT1-3
                           -6.0906                           -2.8519
Bs4(iDFS2):strat_ncN+:strat_tcT1-3 Bs5(iDFS2):strat_ncN+:strat_tcT1-3
                           -2.8519                           -2.1197
Bs6(iDFS2):strat_ncN+:strat_tcT1-3 Bs7(iDFS2):strat_ncN+:strat_tcT1-3
                           -0.9350                           -0.9057
  Bs1(iDFS2):strat_ncN0:strat_tcT4   Bs2(iDFS2):strat_ncN0:strat_tcT4
                           -3.8577                           -2.0170
  Bs3(iDFS2):strat_ncN0:strat_tcT4   Bs4(iDFS2):strat_ncN0:strat_tcT4
                           -0.8188                           -0.8188
  Bs5(iDFS2):strat_ncN0:strat_tcT4   Bs6(iDFS2):strat_ncN0:strat_tcT4
                           -0.8188                           -0.7989
```

```
  Bs7(iDFS2):strat_ncN0:strat_tcT4   Bs1(iDFS2):strat_ncN+:strat_tcT4
                           -0.2126                           -22.3604
  Bs2(iDFS2):strat_ncN+:strat_tcT4   Bs3(iDFS2):strat_ncN+:strat_tcT4
                           -1.3490                           -1.3490
  Bs4(iDFS2):strat_ncN+:strat_tcT4   Bs5(iDFS2):strat_ncN+:strat_tcT4
                           -1.3490                           -1.3490
  Bs6(iDFS2):strat_ncN+:strat_tcT4   Bs7(iDFS2):strat_ncN+:strat_tcT4
                           -0.2706                           -0.2706
        randarm5-FU + Oxaliplatin                            gamma1
                           -0.2305                            0.1711
```

with confidence interval

```
  2.5%    50%  97.5%
0.6484 0.7964 0.9799
```

```
R> nd <- CAOsurv[1:2, ]
R> tmp <- CAO_Cox_2
R> cf <- coef(CAO_Cox_2_mtram)
R> tmp$coef <- cf[names(cf) != "gamma1"] / sqrt(cf["gamma1"]^2 + 1)
R> (CAO_Cox_PI <- PI(tmp, newdata = nd[2, ], reference = nd[1, ],
+                    one2one = TRUE, conf.level = .95))[1, ]
```

```
Estimate      lwr      upr
  0.5566   0.5045   0.6074
```

For the marginally interpretable models that can be derived from model `CAO_Cox_2_mtram` we can compute the probabilistic index. This value is the meaning that over all study centers, a randomly selected patient receiving Oxaliplatin has a 56% probability of staying disease-free longer than a randomly selected patient receiving the standard treatment only, given that they both have the same lymph node state and tumor grading:

```
R> nd <- CAOsurv[1:2, ]
R> tmp <- CAO_Cox_2
R> cf <- coef(CAO_Cox_2_mtram)
R> tmp$coef <- cf[names(cf) != "gamma1"] / sqrt(cf["gamma1"]^2 + 1)
R> (CAO_Cox_PI <- PI(tmp, newdata = nd[2, ], reference = nd[1, ],
+                    one2one = TRUE, conf.level = .95))[1, ]
```

```
Estimate      lwr      upr
  0.5566   0.5045   0.6074
```

but we can compute the same manually as follows:

```
R> ci_man <- quantile(-rbeta[, ncol(rbeta)], prob = c(.025, .5, .975))
R> PI(ci_man, link = "minimum extreme value")
```

We can fit mixed-effects transformation models (Tamási and Hothorn 2021; Tamási *et al.* 2022) as follows:

```
R> CAO_Cox_2_tramME <- CoxphME(iDFS2 | 0 + strat_n:strat_t ~ randarm + (1 | Block),
+                              data = CAOsurv)
```

From this conditional model, we can obtain the conditional hazard ratio with confidence interval:

```
R> exp(coef(CAO_Cox_2_tramME))

randarm5-FU + Oxaliplatin
                    0.796

R> exp(confint(CAO_Cox_2_tramME, parm = "randarm5-FU + Oxaliplatin",
+              estimate = TRUE))

                              lwr   upr   est
randarm5-FU + Oxaliplatin 0.6451 0.982 0.796
```

which is similar to the one of the marginally interpretable model.

References

Backer MD, Vroey CD, Lesaffre E, Scheys I, Keyser PD (1998). "Twelve Weeks of Continuous Oral Therapy for Toenail Onychomycosis Caused by Dermatophytes: A Double-blind Comparative Trial of Terbinafine 250 mg/day versus Itraconazole 200 mg/day." *Journal of the American Academy of Dermatology*, **38**(5(2)), S57–S63. doi:10.1016/S0190-9622(98)70486-4.

Barbanti L, Hothorn T (2024). "A Transformation Perspective on Marginal and Conditional Models." *Biostatistics*, **25**(2), 402–428. doi:10.1093/biostatistics/kxac048.

Bates D, Mächler M, Bolker B, Walker S (2015). "Fitting Linear Mixed-Effects Models Using lme4." *Journal of Statistical Software*, **67**(1), 1–48. doi:10.18637/jss.v067.i01.

Belenky G, Wesensten NJ, Thorne DR, Thomas ML, Sing HC, Redmond DP, Russo MB, Balkin TJ (2003). "Patterns of Performance Degradation and Restoration During Sleep Restriction and Subsequent Recovery: A Sleep Dose-response Study." *Journal of Sleep Research*, **12**, 1–12. doi:10.1046/j.1365-2869.2003.00337.x.

Box GEP, Cox DR (1964). "An Analysis of Transformations." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **26**(2), 211–252.

Brooks ME, Kristensen K, van Benthem KJ, Magnusson A, Berg CW, Nielsen A, Skaug HJ, Mächler M, Bolker BM (2017). "glmmTMB Balances Speed and Flexibility Among Packages for Zero-inflated Generalized Linear Mixed Modeling." *The R Journal*, **9**(2), 378–400. doi:10.32614/RJ-2017-066.

Chow RT, Heller GZ, Barnsley L (2006). "The Effect of 300 mW, 830 nm Laser on Chronic Neck Pain: A Double-blind, Randomized, Placebo-controlled Study." *Pain*, **124**(1–2), 201–210. doi:10.1016/j.pain.2006.05.018.

Gurka MJ, Edwards LJ, Muller KE, Kupper LL (2006). "Extending the Box-Cox Transformation to the Linear Mixed Model." *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, **169**(2), 273–288. doi:10.1111/j.1467-985X.2005.00391.x.

Heiss F, Winschel V (2008). "Likelihood Approximation by Numerical Integration on Sparse Grids." *Journal of Econometrics*, **144**(1), 62–80. doi:10.1016/j.jeconom.2007.12.004.

Hothorn T, Möst L, Bühlmann P (2018). "Most Likely Transformations." *Scandinavian Journal of Statistics*, **45**(1), 110–134. doi:10.1111/sjos.12291.

Hothorn T, Siegfried S, Kook L (????). *tram: Transformation Models*. R package version 1.4-8, URL https://codeberg.org/thothorn/tram.

Hutmacher MM, French JL, Krishnaswami S, Menon S (2011). "Estimating Transformations for Repeated Measures Modeling of Continuous Bounded Outcome Data." *Statistics in Medicine*, **30**(9), 935–949. doi:10.1002/sim.4155.

Manuguerra M, Heller GZ (2010). "Ordinal Regression Models for Continuous Scales." *International Journal of Biostatistics*, **6**(1), 14. doi:10.2202/1557-4679.1230.

Maruo K, Yamaguchi Y, Noma H, Gosho M (2017). "Interpretable Inference on the Mixed Effect Model with the Box-Cox Transformation." *Statistics in Medicine*, **36**(15), 2420–2434. doi:10.1002/sim.7279.

Matić I, Radoiĉić R, Stefanica D (2018). "A Sharp Pólya-Based Approximation to the Normal CDF." *Applied Mathematics and Computation*, **322**, 111–122. doi:10.2139/ssrn.2842681.

Molenberghs G, Verbeke G (2005). *Models for Discrete Longitudinal Data*. Springer, New York, U.S.A.

Rödel C, Graeven U, Fietkau R, Hohenberger W, Hothorn T, Arnold D, Hofheinz RD, Ghadimi M, Wolff HA, Lang-Welzenbach M, Raab HR, Wittekind C, Ströbel P, Staib L, Wilhelm M, Grabenbauer GG, Hoffmanns H, Lindemann F, Schlenska-Lange A, Folprecht G, Sauer R, Torsten Liersch on behalf of the German Rectal Cancer Study Group (2015). "Oxaliplatin Added to Fluorouracil-based Preoperative Chemoradiotherapy and Postoperative Chemotherapy of Locally Advanced Rectal Cancer (the German CAO/ARO/AIO-04 study): Final Results of the Multicentre, Open-label, Randomised, Phase 3 Trial." *The Lancet Oncology*, **16**(8), 979–989. doi:10.1016/S1470-2045(15)00159-X.

Sauter R, Held L (2016). "Quasi-complete Separation in Random Effects of Binary Response Mixed Models." *Journal of Statistical Computation and Simulation*, **86**(14), 2781–2796. doi:10.1080/00949655.2015.1129539.

Tamási B, Crowther M, Puhan MA, Steyerberg E, Hothorn T (2022). "Individual Participant Data Meta-Analysis with Mixed-Effects Transformation Models." *Biostatistics*, **23**(4), 1083–1098. doi:10.1093/biostatistics/kxab045.

Tamási B, Hothorn T (2021). "tramME: Mixed-Effects Transformation Models Using Template Model Builder." *The R Journal*, **13**(2), 398–418. doi:10.32614/RJ-2021-075.

Tang N, Wu Y, Chen D (2018). "Semiparametric Bayesian Analysis of Transformation Linear Mixed Models." *Journal of Multivariate Analysis*, **166**, 225–240. doi:10.1016/j.jmva.2018.03.007.

Wu H, Wang L (2019). "Normal Frailty Probit Model for Clustered Interval-censored Failure Time Data." *Biometrical Journal*, **61**(4), 827–840. doi:10.1002/bimj.201800114.

Ypma J (2013). *SparseGrid: Sparse grid integration in R*. doi:10.32614/CRAN.package.SparseGrid. R package version 0.8.2, URL https://CRAN.R-project.org/package=SparseGrid.

A. Simulations

Empirical results presented in Section 4 of Barbanti and Hothorn (2024) can be reproduced using

```
R> source(system.file("simulations", "mtram_sim.R", package = "tram"), echo = TRUE)
```

(this takes quite some time).

B. Change Log

1.2-0 **tram** utilises novel log-likelihood (`mvtnorm::lpRR()`) and score functions (`mvtnorm::slpRR()`) for reduced rank problems introduced in **mvtnorm** 1.3-2. The default optimiser is now `auglag::auglag()` with numerically approximated Hessians. Score functions are now available for both continuous and discrete response (but there is little hope that score functions for a mix of continuous and discrete responses will become available). Model fit is now faster and more accurate for models with discrete responses. Some parameter estimates and standard errors changed slightly, however, model interpretation was not affected.

Affiliation:

Luisa Barbanti, Torsten Hothorn
Institut für Epidemiologie, Biostatistik und Prävention
Universität Zürich
Hirschengraben 84, CH-8001 Zürich, Switzerland
`Torsten.Hothorn@R-project.org`