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 2022) to conventional conditional formulations of mixed-effects models where such an overlap exists. In addition, novel transformation models going beyond the capabilities of conventional mixed-effects models are estimated and interpreted. A proof-of-concept implementation available in package tram (Hothorn et al. 2022) is applied. The results presented in this document can be reproduced from the mtram demo

\[
\text{R> install.packages("tram")}
\text{R> demo("mtram", package = "tram")}
\]

2. Normal and Non-normal Mixed-effects Models

First we consider mixed-effects models for reaction times in the sleep deprivation study (Belenky et al. 2003). The average reaction times to a specific task over several days of sleep

---

Please cite this document as: Luisa Barbanti and Torsten Hothorn (2022) Some Applications of Marginally Interpretable Linear Transformation Models for Clustered Observations. R package vignette version 0.8-0, URL https://CRAN.R-project.org/package=tram.
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 \texttt{lmer()} function from the \texttt{lme4} add-on package \citep{Bates2015}:

\begin{verbatim}
R> sleep_lmer <- lmer(Reaction ~ Days + (Days | Subject), + data = sleepstudy, REML = FALSE)
\end{verbatim}

The corresponding conditional model for subject \( i \) reads

\[
P(\text{Reaction} \leq y \mid \text{day}, i) = \Phi \left( \frac{y - \alpha - \beta \text{day} - \alpha_i - \beta_i \text{day}}{\sigma} \right), \quad (\alpha_i, \beta_i) \sim N_2(0, G(\gamma))
\]

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

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

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

\begin{verbatim}
R> library("tram")
\end{verbatim}
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 \( \gamma \) in the model (here also looking at the conditional model for subject \( i \))

\[
\Pr(\text{Reaction} \leq y \mid \text{day}, i) = \Phi \left( \vartheta_1 y + \tilde{\alpha} - \tilde{\beta}_i \text{day} - \tilde{\alpha}_i - \tilde{\beta}_i \text{day} \right), \quad (\tilde{\alpha}_i, \tilde{\beta}_i) \sim N_2(0, \Lambda(\gamma)\Lambda(\gamma))
\]

that is, all fixed and random effect parameters are divided by the residual standard deviation \( \sigma \) (this is the reparameterisation applied by `lm()`). Of course, the parameter \( \vartheta_1 \), 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

\[
\text{R> logLik(sleep_lmer)}
\]

'log Lik.' -875.9697 (df=6)

\[
\text{R> logLik(sleep_LMmer)}
\]

'log Lik.' -875.9697 (df=6)

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

\[
\text{R> (sdinv} <- 1 / \text{summary(sleep_lmer)}\$\text{sigma)}
\]

[1] 0.03907485

\[
\text{R> coef(sleep_LMmer)["Reaction"]}
\]

Reaction
0.03907741

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

\[
\text{R> fixef(sleep_lmer) \* c(-1, 1) \* sdinv}
\]

(Intercept) Days
-9.8236175 0.4090077

\[
\text{R> coef(sleep_LMmer)[c("(Intercept)", "Days")]
\]
Marginally Interpretable Transformation Models

(Intercept)  Days
-9.8243917  0.4089265

The random-effect parameters \( \gamma \) are also reasonably close

\[
R> sleep_lmer@theta
\]

[1] 0.92919061 0.01816575 0.22264321

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

             gamma1     gamma2     gamma3
0.92901066  0.01843056  0.22280431

Consequently, the variance-covariance and correlation matrices

\[
R> sleep_LMmer$G * (1 / sdinv)^2
\]

2 x 2 sparse Matrix of class "dsCMatrix"

[1,]  565.2580  11.21410
[2,]  11.2141  32.73513

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

2 x 2 sparse Matrix of class "dsCMatrix"

[1,] 1.0000000  0.08243925
[2,]  0.08243925 1.0000000

\[
R> unclass(VarCorr(sleep_lmer))$Subject
\]

(Intercept)  Days
(Intercept)  565.47697  11.05512
Days  11.05512  32.68179
attr("stddev")
(Intercept)  Days
  23.779760  5.716799
attr("correlation")
(Intercept)  Days
(Intercept)  1.0000000  0.08132109
Days  0.08132109 1.0000000

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

Cholesky factors, but the likelihood and corresponding optimisation relies on an independent implementation. So why are we doing this? Because \texttt{mtram()} is able to deal with models or likelihoods not available in \texttt{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] [229.5600, 269.5600] [238.7047, 278.7047] [230.8006,270.8006]
[4] [301.4398, 341.4398] [336.8519, 376.8519]

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:

R> sleep_LM_I <- Lm(Reaction_I ~ Days, data = sleepstudy)
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.' -214.9675 (df=6)
R> coef(sleep_LMmer_I)

(Intercept) Reaction_I Days gamma1 gamma2 gamma3
-9.78770607 0.03900116 0.41633415 0.83398952 0.07584130 0.19038611
R> coef(sleep_LMmer)

(Intercept) Reaction Days gamma1 gamma2 gamma3
-9.82439168 0.03907741 0.40892652 0.92901066 0.01843056 0.22280431

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

\[ \mathbb{P}(\text{Reaction} \leq y \mid \text{day}, i) = \Phi \left( h(y) - \tilde{\beta}_{\text{day}} - \tilde{\alpha}_i - \tilde{\beta}_i \text{day} \right), \quad (\tilde{\alpha}_i, \tilde{\beta}_i) \sim N_2(0, \Lambda(\gamma) \Lambda(\gamma)) \]

where \( h(y) = a(y)^T \theta \) 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 \( h(y) = a(y)^T \theta \) under suitable constraints (Hothorn et al. 2018) are applied instead by
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.

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

'log Lik.' -859.5455 (df=11)
```

The increase in the log-likelihood compared to the normal model is not a big surprise. Plotting the transformation function $h(y) = a(y)^T \vartheta$ 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), 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 2022) with respect to the parameters $\vartheta$, $\beta$, and $\gamma$. 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 $\beta$).

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
R> cov2cor(sleep_BCmer$G)
2 x 2 sparse Matrix of class "dsCMatrix"

[1,] 1.0000000  -0.1946629
[2,] -0.1946629  1.0000000
```
What is the uncertainty associated with this parameter? The correlation is a non-linear function of $\gamma$ 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
R> library("mvtnorm")
R> VC <- solve(sleep_BCmer$Hessian)
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
R> ### variance random intercept
R> quantile(ret[1,], c(.025, .5, .975))

   2.5%      50%     97.5%
0.9127821 2.5713595 5.2493469

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

   2.5%      50%     97.5%
0.01890987 0.05348231 0.10594879

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

   2.5%      50%     97.5%
-0.6193527 -0.1883314  0.4689778
```

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.' -860.6377 (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 toenail 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.
3.1. Random Intercept Probit Models

We are first interested in binary probit models featuring fixed main and interaction effects $\beta_1$, $\beta_2$, and $\beta_3$ of treatment (itraconazole vs. terbinafine) and time. Subject-specific random intercept models were estimated by the \texttt{glmer} function from package \texttt{lme4} \citep{Bates2015}, by the \texttt{glmmTMB} function from package \texttt{glmmTMB} \citep{Brooks2017}, and by direct maximisation of the exact discrete log-likelihood given in Appendix B of Barbanti and Hothorn \citeyear{Barbanti2022}.

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

\begin{verbatim}
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 deviance df.resid
1286.1   1313.9  -638.1  1276.1     1903
\end{verbatim}
Scaled residuals:
  Min  1Q Median  3Q Max
-3.519 -0.017 -0.004  0.000 54.237

Random effects:
  Groups     Name   Variance   Std.Dev.
  patientID (Intercept)  20.93   4.575
Number of obs: 1908, groups: patientID, 294

Fixed effects:
  Estimate   Std. Error   z value  Pr(>|z|)
(Intercept)  -3.39483    0.21921  -15.487 <2e-16 ***
treatmentterbinafine  -0.02875    0.25202   -0.114  0.9092
  time          -0.21797    0.02257   -9.657 <2e-16 ***
treatmentterbinafine:time  -0.07135    0.03425  -2.083  0.0372 *

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

Correlation of Fixed Effects:
   (Intr) treatment time
treatmentterbinafine  -0.591
  time          -0.008   0.099
  treatmentterbinafine  0.093  -0.141  -0.630

R> toenail_glmer_RI_1@theta

[1] 4.574859

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 deviance df.resid
1284.6 1312.3 -637.3 1274.6   1903

Scaled residuals:
  Min  1Q Median  3Q Max
-2.857 -0.191 -0.078 -0.001 33.862

Random effects:

<table>
<thead>
<tr>
<th>Groups</th>
<th>Name</th>
<th>Variance</th>
<th>Std.Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>patientID</td>
<td>(Intercept)</td>
<td>4.486</td>
<td>2.118</td>
</tr>
</tbody>
</table>

Number of obs: 1908, groups: patientID, 294

Fixed effects:

|                         | Estimate | Std. Error | z value | Pr(>|z|) |
|-------------------------|----------|------------|---------|---------|
| (Intercept)             | -0.91050 | 0.22880    | -3.980  | 6.9e-05 *** |
| treatmentterbinafine    | -0.10726 | 0.30730    | -0.349  | 0.727   |
| time                    | -0.19128 | 0.02058    | -9.293  | < 2e-16 *** |
| treatmentterbinafine:time | -0.06331 | 0.03098    | -2.044  | 0.041 *  |

---

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

Correlation of Fixed Effects:

<table>
<thead>
<tr>
<th></th>
<th>(Intr)</th>
<th>trtmnt</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>trtmnttrbnf</td>
<td>-0.650</td>
<td></td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>-0.185</td>
<td>0.212</td>
<td></td>
</tr>
<tr>
<td>trtmnttrbnf</td>
<td>0.192</td>
<td>-0.285</td>
<td>-0.611</td>
</tr>
</tbody>
</table>

R> toenail_glmer_RI_2@theta

[1] 2.117954

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  deviance  df.resid
1298.1 1325.9  -644.0   1288.1   1903

Random effects:

<table>
<thead>
<tr>
<th>Groups</th>
<th>Name</th>
<th>Variance</th>
<th>Std.Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>patientID</td>
<td>(Intercept)</td>
<td>4.417</td>
<td>2.102</td>
</tr>
</tbody>
</table>
Number of obs: 1908, groups: patientID, 294

Conditional model:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| (Intercept) | -1.10073 | 0.32274 | -3.411 | 0.000648 *** |
| treatmentterbinafine | -0.17391 | 0.35387 | -0.491 | 0.623101 |
| time | -0.18933 | 0.02073 | -9.134 | < 2e-16 *** |
| treatmentterbinafine:time | -0.06106 | 0.03093 | -1.974 | 0.048340 * |

---

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 \texttt{lme4} (\texttt{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 \texttt{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
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, Hessian = TRUE)
R> coef(toenail_mtram_RI)
```

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 (\texttt{lme4}) or \texttt{glmmTMB}, indicating a very good quality the various approximations used. We can also compare the corresponding covariances

```r
R> vcov(toenail_glmer_RI_2)
```
4 x 4 Matrix of class "dpoMatrix"

(Intercept) treatmentterbinafine time
(Intercept) 0.052347885 -0.045691072 -0.000872134

treatmentterbinafine -0.045691072 0.094431279 0.0013398065

time -0.000872134 0.001339806 0.0004236656

treatmentterbinafine:time 0.001360056 -0.002716156 -0.0003893870

which are also in good agreement.

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

R> solve(toenail_mtram_RI$Hessian)[1:4, 1:4]

[1,] 0.0521524646 0.045580017 0.0008711729 -0.001346117
[2,] 0.0455800172 0.094251843 0.001333365  -0.0026823091
[3,] 0.0008711729 0.001333365 0.0004220717 -0.0003886723
[4,] -0.0013461179 -0.002682309 -0.0003886723 0.0009473105

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

<table>
<thead>
<tr>
<th></th>
<th>mtram</th>
<th>indep</th>
<th>excha</th>
<th>unstr</th>
</tr>
</thead>
<tbody>
<tr>
<td>treat</td>
<td>-0.04633378</td>
<td>-0.01100164</td>
<td>-0.01486371</td>
<td>0.01635082</td>
</tr>
<tr>
<td>time</td>
<td>-0.08170616</td>
<td>-0.09278168</td>
<td>-0.09289552</td>
<td>-0.06893793</td>
</tr>
<tr>
<td>treat</td>
<td>-0.02679084</td>
<td>-0.03198835</td>
<td>-0.03717801</td>
<td>-0.04468491</td>
</tr>
</tbody>
</table>

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
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
R> coef(gin)

(Intercept) treatmentterbinafine
    -0.5566272539       -0.0005816551
     time treatmentterbinafine:time
     -0.1703077912       -0.0672216238

R> coef(gex)

(Intercept) treatmentterbinafine
    -0.581922602       0.007180366
     time treatmentterbinafine:time
     -0.171280029       -0.077733152

R> coef(gun)
```
Marginally Interpretable Transformation Models

(Intercept) treatmentterbinafine
-0.73961933 0.03730057
time treatmentterbinafine:time
-0.13189562 -0.08960660

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

```r
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, Hessian = TRUE)
```

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

<table>
<thead>
<tr>
<th>treatmentterbinafine</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.06026026</td>
<td>-0.14915910</td>
</tr>
</tbody>
</table>

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

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

<table>
<thead>
<tr>
<th>2.5%</th>
<th>97.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.114030986</td>
<td>-0.007916931</td>
</tr>
</tbody>
</table>

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

\[
R > \exp(\text{coef(gun)}["treatmentterbinafine:time"] +
+ c(-1, 1) \times qnorm(.975) \times \sqrt{\text{diag(vcov(gun))}}["treatmentterbinafine:time"])
\]

[1]  0.8318745  1.0048723

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:

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

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):

\[
R > \text{tmp <- toenail_logit}
R > cf <- \text{coef(tmp)}
R > cf <- cf[\text{names(cf) != "outcomemoderate or severe"}]
R > \text{sdrf <- rev(coef(toenail_mtram_logit))[1]}
R > \text{cf <- coef(toenail_mtram_logit)[names(cf)]}
\]
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> 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.

### 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
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.

```r
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
899.807 971.71 -459.912 919.823     1896
```
Scaled residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-1.85421</td>
<td>-0.00210</td>
<td>-0.00037</td>
<td>0.00000</td>
<td>2.35828</td>
</tr>
</tbody>
</table>

Random effects:

<table>
<thead>
<tr>
<th>Groups</th>
<th>Name</th>
<th>Variance</th>
<th>Std.Dev.</th>
<th>Corr</th>
</tr>
</thead>
<tbody>
<tr>
<td>patientID</td>
<td>(Intercept)</td>
<td>118.433</td>
<td>10.883</td>
<td></td>
</tr>
<tr>
<td></td>
<td>time</td>
<td>3.305</td>
<td>1.818</td>
<td>-0.90</td>
</tr>
</tbody>
</table>

Number of obs: 1903, groups: patientID, 289

Fixed effects:

|                         | Estimate | Std. Error | z value | Pr(>|z|) |
|-------------------------|----------|------------|---------|---------|
| (Intercept)             | -4.30120 | 0.26361    | -16.316 | <2e-16 *** |
| treatmentterbinafine    | 0.05419  | 0.34652    | 0.156   | 0.8757  |
| time                    | -0.06792 | 0.07847    | -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:

(Intercept)  treatmentterbinafine  time  treatmentterbinafine:time
(Intercept) -0.662

treatmentterbinafine -0.453  0.342

time -0.270 -0.438 -0.335

R> toenail_glmer_RS@theta

[1] 10.8826790 -1.6359589 0.7930842

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:

<table>
<thead>
<tr>
<th>Groups</th>
<th>Name</th>
<th>Variance</th>
<th>Std.Dev.</th>
<th>Corr</th>
</tr>
</thead>
</table>

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
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.' -545.1163 (df=7)
R> coef(toenail_mtram_RS)

(Intercept) treatmentterbinafine
1.5773582 0.2680624
time treatmentterbinafine:time
-0.5336223 -0.1845193

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 of the exact discrete log-likelihood (Equation 6 in Barbanti and Hothorn 2022) 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
RI

RI + RS

<table>
<thead>
<tr>
<th></th>
<th>glmer</th>
<th>glmer</th>
<th>glmmTMB</th>
<th></th>
<th>glmer</th>
<th>glmmTMB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L</td>
<td>L</td>
<td>(7)</td>
<td></td>
<td>L</td>
<td>L</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>-3.39</td>
<td>-0.91</td>
<td>-1.10</td>
<td>0.91</td>
<td>-4.30</td>
<td>-4.30</td>
</tr>
<tr>
<td>$\beta_1$</td>
<td>-0.03</td>
<td>-0.11</td>
<td>-0.17</td>
<td>-0.11</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>-0.22</td>
<td>-0.19</td>
<td>-0.19</td>
<td>-0.19</td>
<td>-0.07</td>
<td>-0.07</td>
</tr>
<tr>
<td>$\beta_3$</td>
<td>-0.07</td>
<td>-0.06</td>
<td>-0.06</td>
<td>-0.06</td>
<td>-0.23</td>
<td>-0.23</td>
</tr>
<tr>
<td>$\gamma_1$</td>
<td>4.57</td>
<td>2.12</td>
<td>2.10</td>
<td>2.11</td>
<td>10.88</td>
<td>11.01</td>
</tr>
<tr>
<td>$\gamma_2$</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-1.64</td>
<td>-1.68</td>
</tr>
<tr>
<td>$\gamma_3$</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.79</td>
<td>0.83</td>
</tr>
<tr>
<td>LogLik</td>
<td>-675.22</td>
<td>-637.34</td>
<td>-638.54</td>
<td>-637.34</td>
<td>-628.12</td>
<td>-630.65</td>
</tr>
<tr>
<td>Time (sec)</td>
<td>3.49</td>
<td>2.18</td>
<td>1.89</td>
<td>2.16</td>
<td>7.50</td>
<td>3.87</td>
</tr>
</tbody>
</table>

Table 1: Toe nail data. Binary probit models featuring fixed intercepts $\alpha$, treatment effects $\beta_1$, time effects $\beta_2$, and time-treatment interactions $\beta_3$ 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).

The 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

$$
\text{logit}(P(\text{pain } \leq y \mid \text{treatment, time, } i)) = h(y) + \beta_{\text{Active}} + \beta_{7 \text{ weeks}} + \beta_{12 \text{ weeks}} + \beta_{7 \text{ weeks, Active}} + \beta_{12 \text{ weeks, Active}} + \alpha_i
$$

with random intercepts $\alpha_i$ such that the odds at baseline, for example, are given by

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

R> library("ordinalCont")
Figure 8: Neck pain: Trajectories of neck pain assessed on a visual analog scale with and without low-level laser therapy.

\begin{verbatim}
R> neck_ocm <- ocm(vas ~ laser * time + (1 | id), data = pain_df, + scale = c(0, 1))
\end{verbatim}

The results

\begin{verbatim}
R> summary(neck_ocm)
\end{verbatim}

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 Std.Err  t.value  p.value
laserActive        -2.07922  0.65055 -3.1961   0.001918 **
time7 weeks        -0.60366  0.35744 -1.6889   0.094689 .
time12 weeks       -0.23804  0.36365 -0.6546   0.514395
laserActive:time7 weeks  4.40817  0.56073  7.8615  7.604e-12 ***
laserActive:time12 weeks 3.38593  0.53925  6.2790  1.159e-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:
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 (Támasi et al. 2022):

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

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
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,
  + Hessian = TRUE)
```

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 $\hat{\beta}_1, \ldots, \hat{\beta}_7$, $\hat{\beta}_{Active}, \hat{\beta}_7$ weeks, $\hat{\beta}_{12}$ weeks, Active, $\hat{\beta}_{12}$ weeks, Active, $\hat{\alpha}_i$ and compute confidence intervals for the marginal treatment effect after 7 and 12 weeks.
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.

R> S <- solve(neck_Colrmer$Hessian)
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))}))

<table>
<thead>
<tr>
<th></th>
<th>2.5%</th>
<th>50%</th>
<th>97.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>laserActive</td>
<td>0.1126589</td>
<td>0.2468169</td>
<td>0.5037629</td>
</tr>
<tr>
<td>time7 weeks</td>
<td>0.4490773</td>
<td>0.6895826</td>
<td>1.0542314</td>
</tr>
<tr>
<td>time12 weeks</td>
<td>0.5545526</td>
<td>0.8495724</td>
<td>1.2969043</td>
</tr>
<tr>
<td>laserActive:time7 weeks</td>
<td>7.9513592</td>
<td>15.7011167</td>
<td>33.5790835</td>
</tr>
<tr>
<td>laserActive:time12 weeks</td>
<td>4.4653904</td>
<td>8.4837448</td>
<td>17.0816646</td>
</tr>
</tbody>
</table>

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)))

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>laserActive</td>
<td>-1.4103130</td>
</tr>
</tbody>
</table>
Then we compute the marginal treatment effect for weeks 0, 7, 12 by multiplying the shift vector with the following contrast matrix:

```r
R> ctr_mat <- matrix(c(1, 0, 0, 0,
+ 1, 0, 0, 1,
+ 1, 0, 0, 0, 1), nrow = 3, byrow = TRUE)
R> ctr_mat %*% std_beta
[,1]       [,2]       [,3]
[1,]  -1.4103130  1.3453573  0.7307912
[2,]   0.5925189  1.3474191  2.1561143
[3,]  -0.0160528  0.7350224  1.5193881
```

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
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,] -2.1833902 -1.3991085 -0.6856496
[2,]  0.5925189  1.3474191  2.1561143
[3,] -0.0160528  0.7350224  1.5193881
R> PI(-ci_emp, link = "logistic")
     2.5%      50%      97.5%
[1,]  0.8145589  0.7189678  0.6125138
[2,]  0.4023882  0.2881899  0.1883363
[3,]  0.5026754  0.3796606  0.2647637
```

Alternatively, we can compute the probabilistic index by passing a CoMr 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
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]
```
Barbanti and Hothorn

<table>
<thead>
<tr>
<th></th>
<th>lwr</th>
<th>upr</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-1</td>
<td>0.7205063</td>
<td>0.8277764</td>
</tr>
<tr>
<td>5-2</td>
<td>0.2884774</td>
<td>0.4327285</td>
</tr>
<tr>
<td>6-3</td>
<td>0.3803291</td>
<td>0.5354269</td>
</tr>
</tbody>
</table>

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$iDFS
R> exact <- tmp[,3] == 1
R> tmp[exact,2] <- tmp[exact,1] + 2
R> tmp[exact,1] <- pmax(tmp[exact,1] - 2, 0)
R> tmp[exact,3] <- 3
R> CAOsurv$iDFS2 <- tmp

We start with the random intercept model

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

assuming a marginal Weibull model whose effects are scaled depending on the variance \( \gamma_1^2 \) of a block-specific (interaction of lymph node involvement, tumor grading, and study center) random intercept:
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).

\begin{verbatim}
R> CAO_SR <- Survreg(iDFS2 ~ randarm, data = CAOsurv)
R> CAO_SR_mtram <- mtram(CAO_SR, ~ (1 | Block), data = CAOsurv,
+ Hessian = TRUE)
R> logLik(CAO_SR_mtram)
'log Lik.' -2081.542 (df=4)
R> (cf <- coef(CAO_SR_mtram))
         (Intercept) log(iDFS2) randarm5-FU + Oxaliplatin
gamma1     -6.2990054 0.7412855 0.2328600 0.1683613
R> (OR <- exp(-cf["randarm5-FU + Oxaliplatin"] / sqrt(cf["gamma1"]^2 + 1)))
randarm5-FU + Oxaliplatin
  0.794829
\end{verbatim}

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

\[ \exp \left( -\beta_{5-FU} + \alpha \gamma_1 \right) \sqrt{\gamma_1^2 + 1} \]

We simply sample from the joint normal distribution of the maximum likelihood estimators \( \hat{\beta}_1, \hat{\beta}_2, \beta_{5-FU} + \hat{\alpha}, \hat{\gamma}_1 \) and compute confidence intervals for the marginal treatment effect 0.79 as
In a next step, we stratify with respect to lymph node involvement and tumor grading: For each of the four strata, the parameters $\vartheta_1$ and $\vartheta_2$ are estimated separately:

\[
R> CAO_SR_2 <- \text{Survreg}(iDFS2 | 0 + strat_n:strat_t ~ \text{randarm, data = CAOsurv})
\]
\[
R> CAO_SR_2_mtram <- \text{mtram}(CAO_SR_2, ~ (1 | \text{Block}), \text{data = CAOsurv,} + \text{Hessian = TRUE})
\]
\[
R> \text{logLik(CAO_SR_2_mtram)}
\]

'log Lik.' -2067.797 (df=10)

\[
R> (cf <- \text{coef(CAO_SR_2_mtram)})
\]

(Intercept):strat_ncN0:strat_tcT1-3 log(iDFS2):strat_ncN0:strat_tcT1-3
-7.8833653 0.9584499
(Intercept):strat_ncN+:strat_tcT1-3 log(iDFS2):strat_ncN+:strat_tcT1-3
-6.2225174 0.7198965
(Intercept):strat_ncN0:strat_tcT4 log(iDFS2):strat_n cN0:strat_tcT4
-3.0467542 0.3711277
(Intercept):strat_ncN+:strat_tcT4 log(iDFS2):strat_n c+:strat_tcT4
-4.8207089 0.6214653
randarm5-FU + Oxaliplatin
0.2240023 0.1474685

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

randarm5-FU + Oxaliplatin
0.8012313

The corresponding confidence interval for the marginal treatment effect is then

\[
2.5\% 50\% 97.5\%
0.6539043 0.8040660 0.9921574
\]

We now relax the Weibull assumption in the Cox model

\[
P(Y > y | \text{treatment}) = \exp \left( - \exp \left( \frac{a(\log(y))^\top \theta + \beta_5-FU + Ox}{\sqrt{\vartheta_1 + 1}} \right) \right)
\]

(note the positive sign of the treatment effect).
```
R> CAO_Cox_2 <- Coxph(iDFS2 | 0 + strat_n:strat_t ~ randarm, data = CAOsurv,
+     support = c(1, 1700), log_first = TRUE, order = 4)
R> logLik(CAO_Cox_2)
'log Lik.' -2021.878 (df=21)
R> CAO_Cox_2_mtram <- mtram(CAO_Cox_2, ~ (1 | Block), data = CAOsurv,
+     Hessian = TRUE)
R> logLik(CAO_Cox_2_mtram)
'log Lik.' -2029.496 (df=22)
R> coef(CAO_Cox_2_mtram)

Bs1(iDFS2):strat_ncN0:strat_tcT1-3 Bs2(iDFS2):strat_ncN0:strat_tcT1-3
  -6.261908e+01  -2.770310e+00
Bs3(iDFS2):strat_ncN0:strat_tcT1-3 Bs4(iDFS2):strat_ncN0:strat_tcT1-3
  -2.723851e+00  -2.439057e+00
Bs5(iDFS2):strat_ncN0:strat_tcT1-3 Bs1(iDFS2):strat_ncN+:strat_tcT1-3
  -7.315446e-01  -2.943890e+01
Bs2(iDFS2):strat_ncN+:strat_tcT1-3 Bs3(iDFS2):strat_ncN+:strat_tcT1-3
  -4.965570e+00  -2.118701e+00
Bs4(iDFS2):strat_ncN+:strat_tcT1-3 Bs5(iDFS2):strat_ncN+:strat_tcT1-3
  -1.923702e+00  -9.153759e-01
Bs1(iDFS2):strat_ncN0:strat_tcT4 Bs2(iDFS2):strat_ncN0:strat_tcT4
  -4.247591e+00  -1.691163e+00
Bs3(iDFS2):strat_ncN0:strat_tcT4 Bs4(iDFS2):strat_ncN0:strat_tcT4
  -1.638797e+00  -3.916641e-01
Bs5(iDFS2):strat_ncN0:strat_tcT4 Bs1(iDFS2):strat_ncN+:strat_tcT4
   2.554375e-10  -3.900165e+01
Bs2(iDFS2):strat_ncN+:strat_tcT4 Bs3(iDFS2):strat_ncN+:strat_tcT4
  -1.336609e+00  -1.301176e+00
Bs4(iDFS2):strat_ncN+:strat_tcT4 Bs5(iDFS2):strat_ncN+:strat_tcT4
  -7.004120e-01  -2.361698e-01
randarm5-FU + Oxaliplatin gamma1
  -2.784567e-01   4.926647e-02
```

with confidence interval

<table>
<thead>
<tr>
<th></th>
<th>2.5%</th>
<th>50%</th>
<th>97.5%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.6532571</td>
<td>0.8076616</td>
<td>0.9554790</td>
</tr>
</tbody>
</table>

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.
but we can compute the same manually as follows:

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

```
  2.5%     50%    97.5%
0.5113836 0.5532009 0.6048666
```

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

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

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

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

```
randarm5-FU + Oxaliplatin
0.7906073
```

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

```
lwr     upr      est
randarm5-FU + Oxaliplatin 0.6406382 0.9756832 0.7906073
```

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

### References


A. Simulations

Empirical results presented in Section 4 of Barbanti and Hothorn (2022) can be reproduced using

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

(this takes quite some time).
Affiliation:
Luisa Barbanti, Torsten Hothorn
Institut für Epidemiologie, Biostatistik und Prävention
Universität Zürich
Hirschengraben 84, CH-8001 Zürich, Switzerland
Torsten.Hothorn@R-project.org