Online Appendix:

Flexible Regression for Correlated and Censored Ecological Data with Mixed-Effects Additive Transformation Models

Bálint Tamási

Balint.Tamasi@uzh.ch

Torsten Hothorn

Torsten.Hothorn@R-project.org

16/3/2023

Abstract

This vignette serves as the online appendix for the manuscript *"Flexible Regression for Correlated and Censored Ecological Data with Mixed-Effects Additive Transformation Models"*. It presents details on the estimation and inference in the titular model class, four example analyses that use these models to reanalyze ecological phenomena from recently published studies, as well as a small simulation exercise to verify the correctness of the implementation.

S1   Estimation and inference

In this section, we give a brief summary on maximum likelihood estimation and inference in mixed-effects additive transformation models. The model can be written in its general form as

$$\mathbb{P}\left(Y_i \leq y \mid \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\gamma}\right) = F\left(h(y) + f_1(x_{1i}) + f_2(x_{2i}, x_{3i}) + \cdots + \boldsymbol{x}_{+i}^{\top}\boldsymbol{\beta} + \boldsymbol{z}_i^{\top}\boldsymbol{\gamma}\right) \qquad \boldsymbol{\gamma} \sim \mathcal{N}_q(\boldsymbol{0}, \boldsymbol{\Sigma}), \text{ (S1)}$$

with $F(\cdot)$ inverse link function and $h(y)$ baseline transformation function. Each smooth term $(f_j(\cdot))$ can be represented with the help of basis functions $f_j(x_{ji}) = \boldsymbol{b}_j(x_i)^{\top}\boldsymbol{\delta}_j$. Penalties on the $\boldsymbol{\delta}_j$ parameters can be added to the model to prevent overfitting in the general form of $\lambda_j \boldsymbol{\delta}_j^{\top} \boldsymbol{S}_j \boldsymbol{\delta}_j$. λ_j is the smoothing parameter that controls the amount of penalty imposed on the jth term and thus controls the smoothness of $f_j(\cdot)$, while the penalty matrix $\boldsymbol{S}_j$ defines how exactly the parameters $\boldsymbol{\delta}_j$ are penalized.

The log-likelihood of Model S1 can be formulated with the help of appropriate penalty terms for the smooth effects as well as the random effects

$$\ell(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\Sigma}, \boldsymbol{\lambda}) = \ell^{\star}(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) - \frac{1}{2}\sum_{j=1}^{J}\lambda_j \boldsymbol{\delta}_j^{\top} \boldsymbol{S}_j \boldsymbol{\delta}_j - \frac{1}{2}\boldsymbol{\gamma}^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{\gamma} + c, \qquad (\text{S2})$$

where $\ell^{\star}(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) = \sum_{i=1}^{N}\ell_i^{\star}(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta})$ is the unpenalized or conditional (on the vector of random effects and penalized coefficients) log-likelihood, which can be written as the sum of the log-likelihood contributions of the individual observations, assuming conditional independence. $\boldsymbol{\lambda}$ denotes the vector of smoothing parameters corresponding to $f_1(\cdot), f_2(\cdot), \ldots, f_J(\cdot)$ smooth terms, and $\boldsymbol{\delta} = \left(\boldsymbol{\delta}_1^{\top}, \boldsymbol{\delta}_2^{\top}, \ldots, \boldsymbol{\delta}_J^{\top}\right)^{\top}$. The covariance matrix of the stacked vector of random effects $(\boldsymbol{\Sigma})$ is positive definite and typically has a block structure, which can be leveraged in the calculations. The term c is a constant, which does not depend on the parameters (either fixed or random) of the model.

The conditional log-likelihood contributions of the observations, given that they are exactly observed can be written as

$$\ell_i^\star(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) = \log F' \left(h(y_i) + f_1(x_{1i}) + f_2(x_{2i}, x_{3i}) + \cdots + \boldsymbol{x}_{+i}^\top \boldsymbol{\beta} + \boldsymbol{z}_i^\top \boldsymbol{\gamma} \right) + \log h'(y_i)$$

where $F'(\cdot)$ and $h'(\cdot)$ are the derivative functions of $F(\cdot)$ and $h(\cdot)$, respectively. Because we directly parameterize the cumulative distribution function of the outcome in Equation S1, it is straightforward to express the log-likelihood contributions of randomly censored observations:

$$\ell_i^\star(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) = \log \left(\mathbb{P} \left(Y_i \leq \bar{y}_i \mid \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\gamma} \right) - \mathbb{P} \left(Y_i \leq \underline{y}_i \mid \boldsymbol{x}_i, \boldsymbol{z}_i, \boldsymbol{\gamma} \right) \right), \tag{S3}$$

for an interval-censored observation on $\underline{y}_i \leq y_i < \bar{y}_i$. The contribution of left $(-\infty < y_i < \bar{y}_i)$ and right-censored $(\underline{y}_i \leq y_i < \infty)$ observations can be expressed by setting the second part of this expression to 0 and the first part to 1, respectively.

Likelihood-based inference in the family of mixed-effects transformation models is presented by Tamási and Hothorn (2021). The estimation procedure described there is directly applicable in the setting with penalized smooth additive terms utilizing their mixed model representation (Ruppert et al., 2003, Wand, 2003). In the case of some smoothing bases, a reparameterization is necessary to fit into a mixed model-based estimation framework (Wood, 2004, Appendix and Fahrmeir et al., 2013, Chapter 8). The marginal log-likelihood function of the model can be efficiently evaluated by integrating over the vector of random effects using the Laplace approximation (see Section S2 for details). In the mixed-effects representation, the smoothing parameters can be estimated as variance components of the mixed model via maximum likelihood or restricted maximum likelihood (REML).

Frequentist inference for the penalized smooth terms can be based on the joint covariance matrix estimate of fixed and random parameters using the generalized delta method (Kristensen et al., 2016). This estimate approximates the mean squared error of the random effects predictions (Zheng and Cadigan, 2021), and it is also in agreement with results derived in a Bayesian setting for independent hierarchical models (Kass and Steffey, 1989).

The effective degrees of freedom (EDF) of the individual smooth terms give an approximation of the effective number of parameters used in modeling these penalized smooth functions. Following Wood et al. (2016) and Wood (2017, Section 6.11), the EDFs can be estimated by summing up the appropriate elements of the diagonal of the matrix $V\mathcal{I}$, where V is the joint variance-covariance matrix of all unpenalized (i.e., fixed-effects) and penalized (random) parameters. The matrix $\mathcal{I}$ is the negative Hessian of the *unpenalized* log-likelihood ($\ell^\star(\boldsymbol{\vartheta}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta})$ in Equation S2), evaluated at the vector of parameter estimates.

S2   Computational details

`tramME` offers several methods to approximate the baseline transformation function ($h(y)$ in Equation S1) of additive mixed-effects transformation models (see Tamási and Hothorn, 2021, for the available options). Similarly to Hothorn et al. (2018), the *Bernstein polynomial basis* (Farouki, 2012) is used for a general and flexible parameterization for continuous outcomes:

$$h(y) = \boldsymbol{a}_{\text{Bs},M}(y)^\top \boldsymbol{\vartheta} = \frac{1}{M+1} \sum_{m=0}^{M} \vartheta_m \binom{M}{m} \tilde{y}^m (1 - \tilde{y})^{M-m},$$

where $\boldsymbol{a}_{\text{Bs},M}(y)$ denotes the set of order M polynomials in Bernstein form, and $\tilde{y}$ equals to y scaled to the interval $[0, 1]$. In practice, the exact order of the polynomials in Bernstein form usually does not have a large effect on the results beyond a minimal value. Hothorn (2020) discusses the choice of

order in fixed effects-only transformation models and recommends choosing M between 5 and 10. In the application of the main article, we used the order of 6 as a default.

Likelihood-based estimation and inference in mixed-effects additive transformation models is implemented by the R add-on package `tramME` (Tamási and Hothorn, 2021). The package utilizes existing implementations of linear transformation models (`mlt` and `tram` packages by Hothorn, 2020) for the user interface and to set up inputs for the estimation procedure. The penalized smooth terms are defined with the help of the `mgcv` package (Wood, 2017), which provides functionality to reformulate these terms to their mixed-effects representations as described by Wood (2004, Appendix). `tramME` uses the Template Model Builder package (`TMB`, Kristensen et al., 2016) to evaluate the log-likelihood via Laplace approximation and to calculate the gradients of it with automatic differentiation (`glmmTMB`, by Brooks et al., 2017, uses the same underlying techniques). Consequently, model estimation and inference in `tramME` is fast and efficient.

The user interface of `tramME` follows standard model specification conventions of the most popular R packages (`lme4`, Bates et al., 2015, `mgcv`, Wood, 2017) to define mixed-effects additive transformation models. Thus, transition from a `gam`, `lmer`, or `glmer` model is very simple. For example, a mixed-effects additive parametric proportional hazards model can be defined with the function call

```
R> ## tramME is available from CRAN:
R> ## install.packages("tramME")
R> library("tramME")
R> mc1 <- CoxphME(                          ### conditional proportional hazards
+      Time ~                               ### response time intervals
+            Insects +                      ### fixed effects
+            Habitat +
+            Landscape +
+            s(Temperature, k = 20) +       ### non-linear terms, as in mgcv
+            s(Elevation100, k = 20) +
+            (1 | PlotID),                  ### random intercept, as in lme4
+      data = carrion,                      ### data
+      log_first = TRUE,                    ### log(Time) before modeling
+      order = 6                            ### order of Bernstein
+  )
```

S3 The rat carrion decomposition experiment

Our first example presents the reanalysis of the carrion decomposition data by Englmeier et al. (2022). In the original study, the authors analyzed the environmental factors that affect the decomposition process of small rodent carrion using data from an experiment, in which they placed rat carcasses in different environments and recorded the time until complete decomposition. As we describe it in the main article, the outcome variable (time until complete decomposition) is *interval censored* due to the discrete follow-up. Figure S1 presents some examples of the censored outcomes in the dataset. Segments with arrows denote right-censored observations, i.e., cases, where the total decomposition did not occur by the time of the last visit. As these examples demonstrate, the sites were visited at regular intervals, albeit we can see some variability in the lengths. The survival curves for specimen with and without the presence of insects are shown in Figure S2. Based on this marginal view of the data, i.e., without adjusting for other important factors and ignoring cluster-level heterogeneity, the presence of insects accelerated the decomposition process. The 25th, 50th and 75th percentiles in the
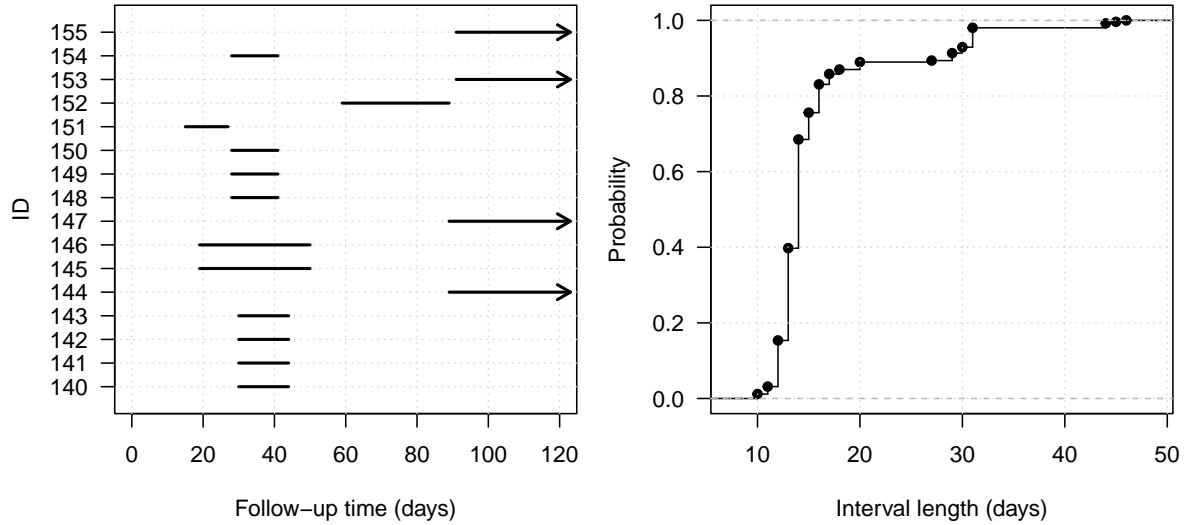
Figure S1: Censoring in the carrion decomposition dataset of Englmeier et al. (2022). *Left panel*: Examples of censoring types. Segments with arrows denote right-censored observations, other segments show interval-censored outcomes. *Right panel*: The distribution of the interval lengths of interval-censored observations.
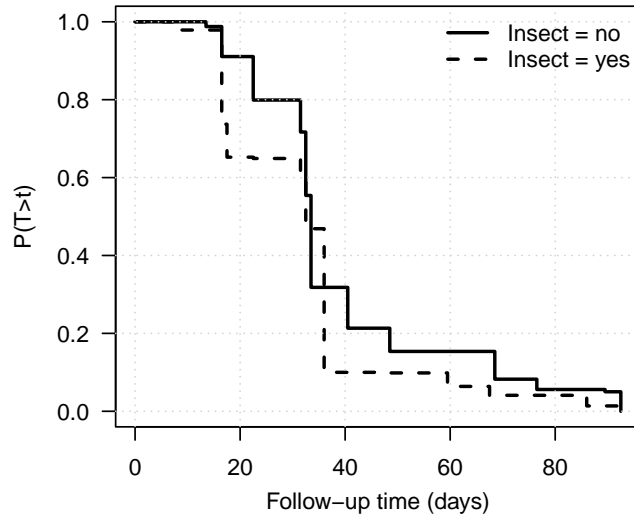

Figure S2: Non-parametric survival probability estimates associated with the decomposition times in groups where insect access was and was not allowed in the carrion decomposition experiment.

cases where insects were present were 16.5, 32.5, 36 days, respectively, while, without insects, the same quantiles amounted to 31.5, 33.5, 40.5 days.

The main environmental variables we include in our model are the indicator for the presence of insects (`Insects`), local (`Habitat`) and regional (`Landscape`) land use types, The average temperature on each experimental plot was measured with thermologgers (`Temperature`), and the elevational gradient was used as a surrogate for the long-term macroclimate (`Elevation100`). The plot-level unobserved sources of variability were modeled by including random intercepts (grouping variable: `PlotID`). The

outcome variable (`Time`) is a predefined `Surv` object that encodes the interval-censored decomposition times.

```
R> head(carrion$Time, 10)

 [1] [31, 51] [31, 51] [31, 51] [ 3, 30] [31, 51] [31, 51] [81, 95] [81, 95]
 [9] [81, 95] [31, 47]
```

We estimate the following proportional-hazards mixed-effect additive transformation model for the decomposition intervals

```
R> dcmp <- CoxphME(Time ~ Insects + Habitat + Landscape
+                   + s(Temperature, k = 20) + s(Elevation100, k = 20)
+                   + (1 | PlotID), data = carrion,
+                   log_first = TRUE, order = 6)
R> summary(dcmp)


Additive Mixed-Effects Parametric Cox Regression Model

Formula:
Time ~ Insects + Habitat + Landscape + s(Temperature, k = 20) +
    s(Elevation100, k = 20) + (1 | PlotID)


Fitted to dataset carrion


Fixed effects parameters:
==========================


                    Estimate Std. Error z value Pr(>|z|)
Insectsyes            1.0562     0.2493    4.24  2.3e-05 ***
Habitatarable field  -0.8986     0.4947   -1.82    0.069 .
Habitatmeadow        -0.1898     0.4665   -0.41    0.684
Habitatsettlement    -0.8492     0.5699   -1.49    0.136
Landscapeagriculture -0.3964     0.4332   -0.92    0.360
Landscapeurban        0.0337     0.4373    0.08    0.938
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Smooth shift terms:
===================


                 edf
s(Temperature)     1
s(Elevation100)    1


Random effects:
===============

```

```
Grouping factor: PlotID (144 levels)
Standard deviation:
(Intercept)
       1.73



Log-likelihood: -314 (npar = 18)
```

The smooth terms can be evaluated and plotted with

```
R> plot(smooth_terms(dcmp), panel.first = grid())
```

As the results in Figure S3 show, the effects of temperature and elevation look fairly linear.
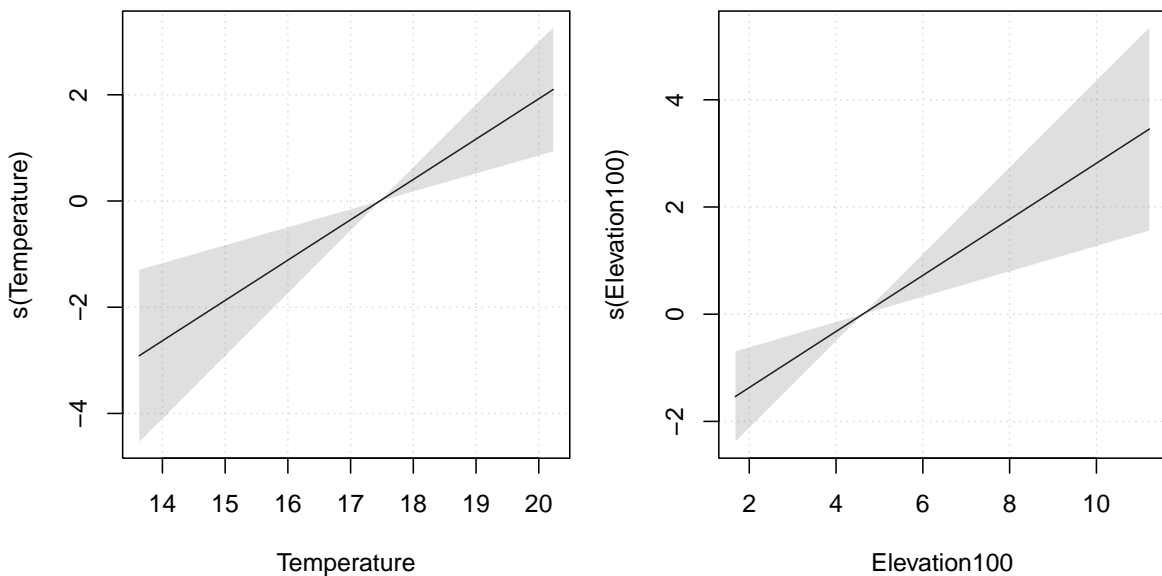

Figure S3: Smooth effects of the continuous variables in the model of the decomposition times.

As a general check of the appropriateness of the estimated model, we can evaluate the *marginal* distribution function or, as it is more commonly done in survival analysis, the marginal survivor function of the outcomes at the observations in our dataset by integrating over the random effects numerically.

$$\widehat{S}(t_i \mid \boldsymbol{x}_i) = \widehat{\mathbb{P}}(T_i > t_i \mid \boldsymbol{x}_i) = \int_{-\infty}^{+\infty} \widehat{\mathbb{P}}(T_i > t_i \mid \boldsymbol{x}_i, \gamma)\phi(\gamma)\, \mathsf{d}\gamma.$$

$\widehat{S}(t_i \mid \boldsymbol{x}_i)$ denotes the fitted survivor function, which is straightforward to calculate from the mixed-effects additive transformation model, due to its fully parametric approach to approximate the outcome distribution. By evaluating $-\log(\widehat{S}(t_i \mid \boldsymbol{x}_i))$ at the observations in our dataset, we get the Cox-Snell residuals (Klein and Moeschberger, 2003, Chapter 11), which are unit exponentially distributed under the correct model.

Interval-censored outcomes pose a technical difficulty in assessing the Cox-Snell residuals. In this situation we can either evaluate the marginal survivor function at the upper and lower bounds of the censoring intervals and assess the interval-censored version of the residuals, or we can apply the adjustment proposed by Farrington (2000), which replaces these intervals with expected values under unit

exponential distribution. To assess the distribution of the residuals, we estimate the cumulative hazard function of the Cox-Snell residuals, which should be close to a straight line with unit slope through the origin under the correct model. Figure S4 presents the distributions of the Cox-Snell residuals using the two approaches to interval censoring. Neither of these plots signals serious departures from the unit exponential distribution, which confirms the appropriateness of our regression model. Because the marginal Cox-Snell residuals are not independent in our case, these plots only provide a crude visual check of the model fits.
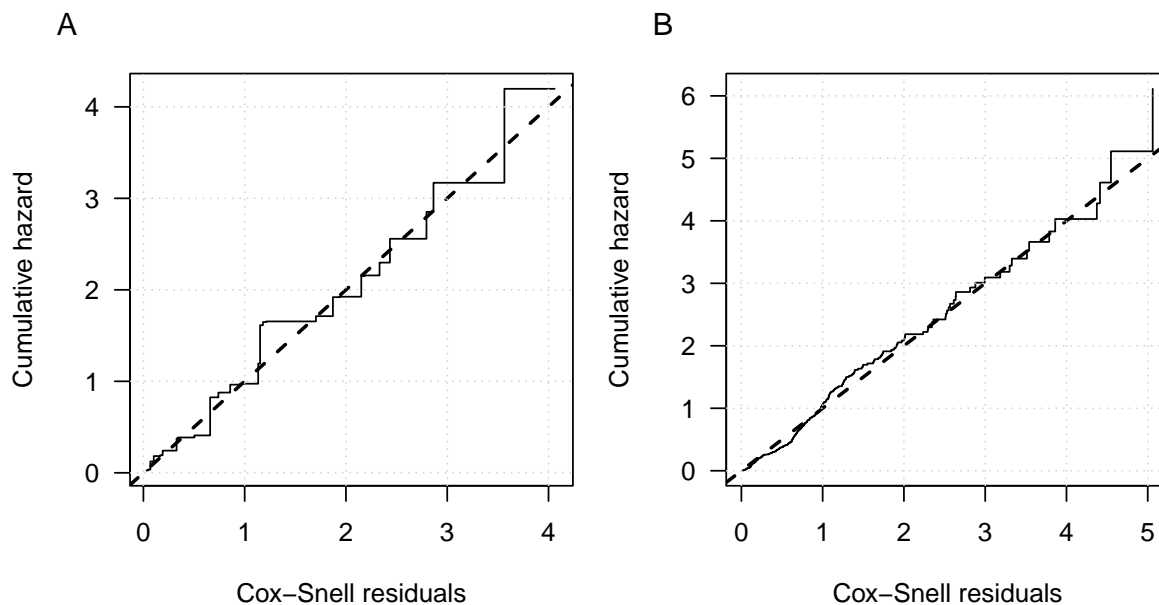
A

B

Figure S4: Cox-Snell residuals of the carrion decomposition model. *Panel A*: Treating the residuals as interval-censored and estimating the cumulative hazard function using the Turnbull nonparametric maximum likelihood estimator. *Panel B*: Using the adjustment proposed by Farrington (2000). The dashed lines correspond to the unit exponential distribution.

We can relax the assumption of proportional hazards by allowing for time-dependent covariate effects. A transformation model with time-dependent effects for the `Insects` indicator can be estimated as

```
R> dcmp2 <- CoxphME(Time | Insects ~ Habitat + Landscape
+                   + s(Temperature, k = 20) + s(Elevation100, k = 20)
+                   + (1 | PlotID), data = carrion,
+                   log_first = TRUE, order = 6)
```

In Figure S5, we compare the effect estimates of the presence of insects (on the log-hazard scale) from the proportional hazards and non-proportional hazards (time-varying effects) models. According to these results, the proportional hazards assumption seems plausible.

S4 *E. coli* concentrations in streams with different grazing periods

Hulvey et al. (2021) compare the concentration levels of *Escherichia coli* bacteria (most probable number, MPN) in streams under three different rotational grazing regimes. In the additive mixed
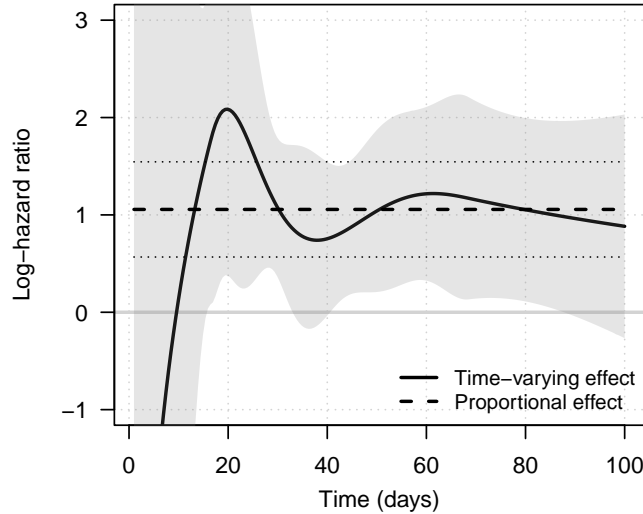
Figure S5: The effect of the presence of insects on the decomposition process from the proportional-hazards and non-proportional hazards models.

model specifications they estimated, within-year variability was modeled, as functions of the day of year (DOY), with cubic regression splines and between-year and location-level variability were captured by random intercepts of pasture-specific year effects and separate stream effects. Note that although the cyclic version of the cubic regression splines (`bs = "cc"` in `mgcv` and `tramME`) (Tamasi, 2024) would be more appropriate for modeling the within-year trend, the original article used `bs = 'cr'` and hence we also stick with this basis in our reanalysis.

As a first step, we replicate the results of all model variants that Hulvey et al. (2021) investigated in the original article with the R package `gamm4` (Wood and Scheipl, 2020). In a second step, we fit the *same* models using the software implementation of additive mixed transformation models in package `tramME`, that is, using a linear transformation function (function `tramME::LmME`). We do expect identical results in steps one and two, although the two implementations rely on two completely distinct code bases. Thus, these results are only interesting from a quality assurance point of view. In the last step, we relax the normal distributional assumption by allowing a nonlinear transformation function (`tramME::BoxCoxME` function) and evaluate how the model fits change. We are primarily interested in potential changes of the model interpretation induced by a shift from a normal to a distribution-free model.

As Table S1 shows, we managed to reproduce the `gamm4` results with `tramME`. Moreover, relaxing the distributional assumption of the normal linear model resulted in stronger model fits in terms of in-sample log-likelihood values.

```
R> ## specifications w/o random effects
R> mf <- c(log10(ecoli_MPN) ~ treatment + cattle +
+           s(DOY, bs = "cr", by = treatment),
+         log10(ecoli_MPN) ~ treatment + cattle + s(DOY, bs = "cr"),
+         log10(ecoli_MPN) ~ treatment + s(DOY, bs = "cr", by = treatment),
+         log10(ecoli_MPN) ~ cattle + s(DOY, bs = "cr"),
+         log10(ecoli_MPN) ~ treatment + s(DOY, bs = "cr"),
+         log10(ecoli_MPN) ~ s(DOY, bs = "cr"))
```

```
R> names(mf) <- paste("Model", c(1:5, "Null"))
R> ecoli_res <- data.frame(matrix(NA, nrow = length(mf), ncol = 3))
R> colnames(ecoli_res) <- c("gamm", "LmME", "BoxCoxME")
R> rownames(ecoli_res) <- names(mf)
R> for (i in seq_along(mf)) {
+     m_gamm <- gamm4(mf[[i]], data = ecoli,
+                     random = ~ (1 | year:stream:pasture) + (1 | stream),
+                     REML = FALSE)
+     ecoli_res$gamm[i] <- logLik(m_gamm$mer)
+     mf2 <- update(mf[[i]], . ~ . + (1 | year:stream:pasture) + (1 | stream))
+     m_LmME <- LmME(mf2, data = ecoli)
+     if (m_LmME$opt$convergence == 0) ecoli_res$LmME[i] <- logLik(m_LmME)
+     m_BCME <- BoxCoxME(mf2, data = ecoli)
+     if (m_BCME$opt$convergence == 0) ecoli_res$BoxCoxME[i] <- logLik(m_BCME)
+ }
```

Table S1: Log-likelihood values of the fitted models presented by Hulvey et al. (2021, *GAMM*), replicated as mixed-effects additive transformation models assuming conditional normality (*Additive normal transformation model*) and extended as flexible (non-normal) mixed-effects additive transformation models (*Additive non-normal transformation model*).

	GAMM	Additive normal transformation model	Additive non-normal transformation model
Model 1	-339.23	-339.23	-320.94
Model 2	-343.66	-343.66	-324.54
Model 3	-368.33	-368.33	-349.10
Model 4	-347.70	-347.70	-328.25
Model 5	-367.15	-367.15	-347.27
Model Null	-373.76	-373.76	-353.50

Let us focus on the most complicated specification, Model 1,

```
R> update(mf[[1]], . ~ . + (1 | year:stream:pasture) + (1 | stream))

log10(ecoli_MPN) ~ treatment + cattle + s(DOY, bs = "cr", by = treatment) +
    (1 | year:stream:pasture) + (1 | stream)
```

and compare the effect estimates from the normal model to its non-parametric counterpart. But first, notice that by changing the transformation from $h(y) = \vartheta_0 + \vartheta_1 y$ to $h(y) = \boldsymbol{a}(y)^\top \boldsymbol{\vartheta}$, we change the scale on which the coefficients and the smooth terms are interpreted. In the normal additive mixed model, the coefficient of a fixed effect captures the change in the expectation of the outcome when increasing the respective predictor by one unit (keeping everything else unchanged). In the non-normal transformation model with Φ as the inverse link, the coefficients capture similar effects but on a latent scale defined by the transformation $h(Y)$.

To cast the effect estimates from the two models to a common scale, we can calculate the *probabilistic indices* (PI, Thas et al., 2012). To simplify the notation, first, we will now focus on the simple, fixed

effects-only case with a single predictor:

$$\mathbb{P}(Y \leq y \mid \boldsymbol{X} = x) = \Phi\left(h(y) - \beta x\right)$$

The PI is the probability that one outcome $(Y^\star)$ is larger than the other (Y), given the same covariate values $(\boldsymbol{X})$ except for one, which is larger with one unit $(\boldsymbol{X}^\star)$. In our simplified example, this means

$$\mathbb{P}\left(Y < Y^\star \mid \boldsymbol{X} = x, \boldsymbol{X}^\star = x + 1\right) = \mathbb{P}\left(h(Y) < h(Y^\star) \mid \boldsymbol{X} = x, \boldsymbol{X}^\star = x + 1\right)$$
$$= \mathbb{P}\left(\frac{h(Y) - h(Y^\star) + \beta}{\sqrt{2}} < \frac{\beta}{\sqrt{2}}\right)$$
$$= \Phi\left(\frac{\beta}{\sqrt{2}}\right).$$

The third line follows from the fact that, in a transformation model with $\Phi(\cdot)$ as the inverse link, $h(Y)$ and $h(Y^\star)$ are independent, normally distributed random variables with unit variance and a mean difference of β. Notice that the PI does not depend on the transformation function. When random effects are present in the model, the PI is conditional on the cluster.

In the case of transformation models with non-linear additive terms the probabilistic index is a function of the covariate. In the simplest form of an additive transformation model with probit link, we have

$$\mathbb{P}(Y \leq y \mid \boldsymbol{X} = x) = \Phi\left(h(y) - f(x)\right)$$

and the PI is

$$PI(x) = \mathbb{P}\left(Y < Y^\star \mid \boldsymbol{X} = x, \boldsymbol{X}^\star = x + 1\right) = \Phi\left(\frac{f(x+1) - f(x)}{\sqrt{2}}\right).$$

By transforming the effect estimates to the probability scale, Figure S6 compares the smooth terms from the normal and non-normal versions of Model 1, while the first two blocks of Table S2 contrasts the fixed effects estimates. The results are very close to each other, which suggests that the original log-normal model is actually appropriate. As a built-in visual normality check, we can compare the fitted transformation functions of the normal and non-normal transformation models. The linear function corresponds to a conditional normal distribution in Figure S7. This result further confirms the appropriateness of the normal additive model in this specific example.
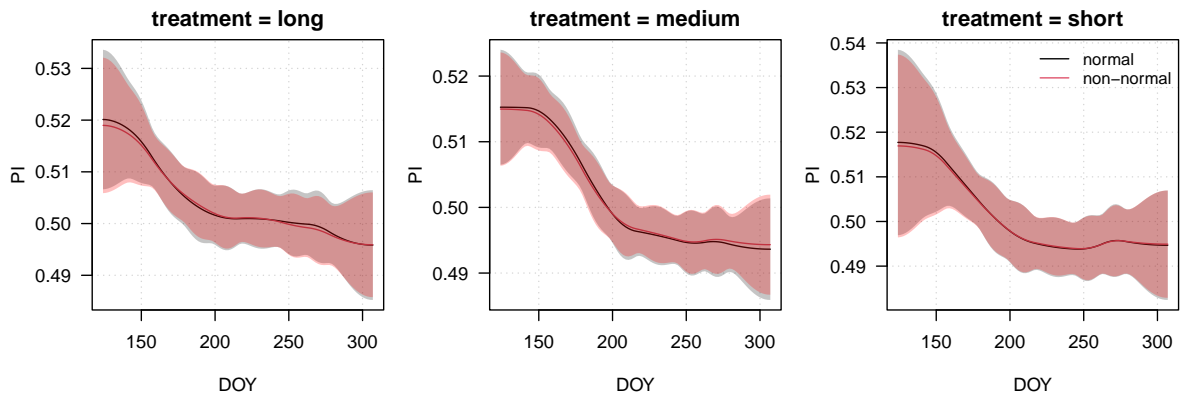

Figure S6: The comparison of the smooth terms from the normal and non-normal (probit link) mixed-effects additive transformation models (specification Model 1).

The outcome variable (MPN per 100 ml) was measured with the Quanti-Tray System, which can detect *E. coli* concentrations up to a maximum of 2,419.6 MPN without dilution. This means that
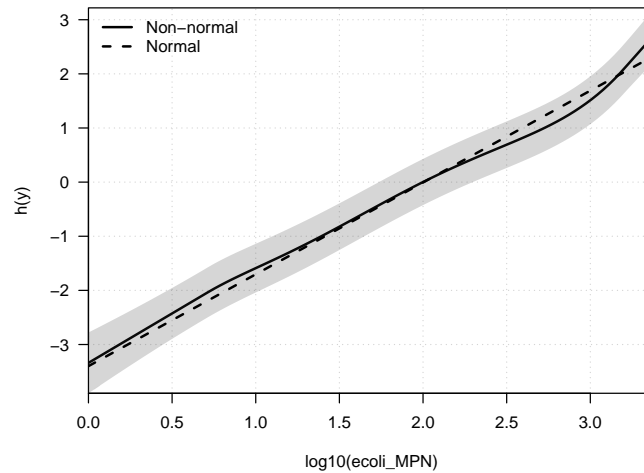
Figure S7: Baseline transformation functions from the normal and non-normal mixed-effects additive transformation models.

there is an effective upper detection limit on the outcome, i.e., the 25 observations with the value of 2,419.6 are *right censored*. The authors of the original article mention this fact, but they do not take into account in the subsequent analyses. Because random censoring can be easily handled in `tramME`, we will rerun the model taking the upper limit into account.

```
R> fm1c <- update(fm1, Surv(log10(ecoli_MPN), event = ecoli_MPN < 2419.6) ~ .)
R> ecoli_m1_cens <- BoxCoxME(fm1c, data = ecoli)
R> summary(ecoli_m1_cens)


Non-Normal (Box-Cox-Type) Linear Additive Mixed-Effects Regression Model

Formula:
Surv(log10(ecoli_MPN), event = ecoli_MPN < 2419.6) ~ treatment +
    cattle + s(DOY, bs = "cr", by = treatment) + (1 | year:stream:pasture) +
    (1 | stream)


Fitted to dataset ecoli

Fixed effects parameters:
==========================


              Estimate Std. Error z value Pr(>|z|)
treatmentmedium  -0.680      0.230   -2.95   0.0032 **
treatmentshort   -0.772      0.317   -2.44   0.0148 *
cattlePresent     1.108      0.149    7.42  1.2e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Smooth shift terms:
===================
```

```
                         edf
s(DOY):treatmentlong    4.38
s(DOY):treatmentmedium  4.55
s(DOY):treatmentshort   4.30


Random effects:
===============


Grouping factor: year:stream:pasture (32 levels)
Standard deviation:
(Intercept)
      0.431

Grouping factor: stream (12 levels)
Standard deviation:
(Intercept)
   0.000154



Log-likelihood: -358 (npar = 18)
```

The fitted non-linear terms are compared to the original (normal linear) estimates in Figure S8 and the fixed effects are presented in the third block of Table S2.

Table S2: Estimates of the parametric fixed-effects terms on the *probability scale* (PI: probabilistic index) from the normal, non-normal and non-normal (with censoring taken into account) models, respectively.

	Normal		Non-normal		Non-normal, censored	
	PI	95% CI	PI	95% CI	PI	95% CI
treatment = medium	0.32	0.22—0.44	0.33	0.22—0.45	0.32	0.21—0.44
treatment = short	0.29	0.16—0.45	0.30	0.17—0.46	0.29	0.16—0.46
cattle = present	0.79	0.72—0.84	0.78	0.72—0.84	0.78	0.72—0.84

Because the transformation model approximates the conditional distribution of the outcome, in theory, we do not even have to take the base 10 logarithm of the *E. coli* most probable numbers (MPN) on the left-hand side of the model formula. `tramME` should be able to approximate the *most likely transformation*.

```
R> f_nontr <- update(fm1, Surv(ecoli_MPN, event = ecoli_MPN < 2419.6) ~ .)
R> ecoli_nontr <- BoxCoxME(f_nontr, data = ecoli, log_first = TRUE)
R> summary(ecoli_nontr)


Non-Normal (Box-Cox-Type) Linear Additive Mixed-Effects Regression Model
```
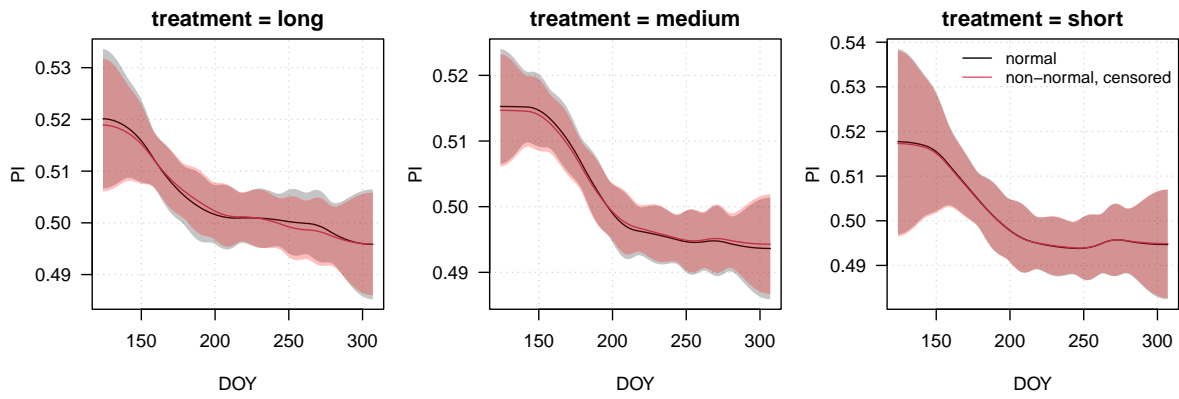
Figure S8: The comparison of the smooth terms from the original model (normal linear) and the non-normal (probit link) extension where censoring is also taken into account.

```
Formula:
Surv(ecoli_MPN, event = ecoli_MPN < 2419.6) ~ treatment + cattle +
    s(DOY, bs = "cr", by = treatment) + (1 | year:stream:pasture) +
    (1 | stream)


Fitted to dataset ecoli


Fixed effects parameters:
=========================


               Estimate Std. Error z value Pr(>|z|)
treatmentmedium    -0.680      0.230   -2.95   0.0032 **
treatmentshort     -0.772      0.317   -2.44   0.0148 *
cattlePresent       1.108      0.149    7.42  1.2e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Smooth shift terms:
===================


                       edf
s(DOY):treatmentlong    4.38
s(DOY):treatmentmedium 4.55
s(DOY):treatmentshort   4.30


Random effects:
===============


Grouping factor: year:stream:pasture (32 levels)
Standard deviation:
(Intercept)
      0.431
```

13

```
Grouping factor: stream (12 levels)
Standard deviation:
(Intercept)
   0.000439


Log-likelihood: -2027 (npar = 18)
```

Notice that we set `log_first = TRUE` in the function call, to take the natural logarithm of the outcome before setting up the Bernstein bases. This usually helps the approximation in the case of positive right-skewed outcomes. With this, we basically estimate the same model as the original, but with the natural logarithm instead of base-ten. Because of this difference, the log-likelihood values are also different, but the fixed effects and variance components parameter estimates, as well as the smooth terms are essentially the same as in the case of the model `ecoli_m1_cens`.

In summary, after bringing the estimates to the same scale, the results of the additive mixed effects model did not change much in this specific example by switching to the transformation model approach. The originally applied base 10 logarithm falls very close to the fitted "most likely transformation", i.e., taking the logarithm of the outcome was sufficient to achieve (close) conditional normality. This could be verified through comparing the baseline transformation functions of the normal and non-normal models, which can also serve as a visual check on conditional normality. Moreover, the number of censored outcomes was relatively small in the sample, so taking the censoring properly into account did not result in large differences, either. However, as the example demonstrated, transformation models are flexible enough to accommodate these properties of the response of interest (non-normality and censoring) automatically, without the need to apply ad hoc transformations or to implement new estimation procedures. In this sense, `tramME::BoxCoxME` provides a simple way of checking the impact of the more restrictive assumptions hard-wired in `gamm4::gamm4` on model interpretation and of handling censoring properly in the estimation procedure.

S5   Sea urchin removal experiment

Andrew and Underwood (1993) analyzed the percentage cover of filamentous algae under four sea urchin removal treatments (Control/33%/66%/Removal). The algae colonization was measured on five quadrants located on several larger patches, so there is a clear grouped structure in the data. Douma and Weedon (2019) reanalyzed the data as a demonstration for the usage of mixed-effects models for zero-inflated beta regression. Here, we fit mixed-effects transformation models to the data and compare the results to zero-inflated mixed-model estimates obtained from `glmmTMB` (Brooks et al., 2017, 2024). Figure S9 presents the empirical cumulative distribution functions of the outcome under the four treatments. Note the large number of zeros, especially in the control group.

First, we fit a zero-inflated beta regression model with random intercepts for the patches. The probability of observing zero values is allowed to vary with the treatment.

```
R> urchin_zib <- glmmTMB(pALGAE ~ TREAT + (1 | PATCH), ziformula = ~ TREAT,
+                        data = andrew, family = beta_family())
```

As an alternative to the traditional beta regression approach, we estimate a mixed-effects continuous outcome logistic regression.
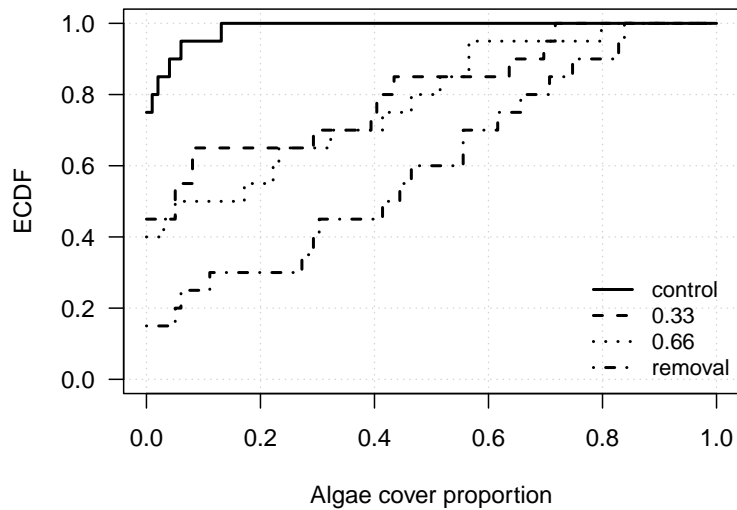
Figure S9: Empirical CDFs of the algae cover proportions under the four treatments.

```
R> urchin_tram <- ColrME(
+    Surv(pALGAE, pALGAE > 0, type = "left") ~ TREAT + (1 | PATCH),
+    bounds = c(-0.1, 1), support = c(-0.1, 1), data = andrew,
+    order = 6)
R> summary(urchin_tram)



Mixed-Effects Continuous Outcome Logistic Regression Model


Formula:
Surv(pALGAE, pALGAE > 0, type = "left") ~ TREAT + (1 | PATCH)


Fitted to dataset andrew


Fixed effects parameters:
=========================


           Estimate Std. Error z value Pr(>|z|)
TREAT0.33     -2.04       1.31   -1.56   0.1178
TREAT0.66     -2.49       1.31   -1.90   0.0571 .
TREATremoval  -4.10       1.34   -3.06   0.0022 **

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


Random effects:
===============
```

15

```
Grouping factor: PATCH (16 levels)
Standard deviation:
(Intercept)
       1.48


Log-likelihood: -26.3 (npar = 11)
```

To allow for a jump in the conditional CDF of the outcome, we expand its bound and treat the zero observations as left-censored. This way, we can place a point mass on zero, i.e., introduce a jump at 0 (see Figure S10).
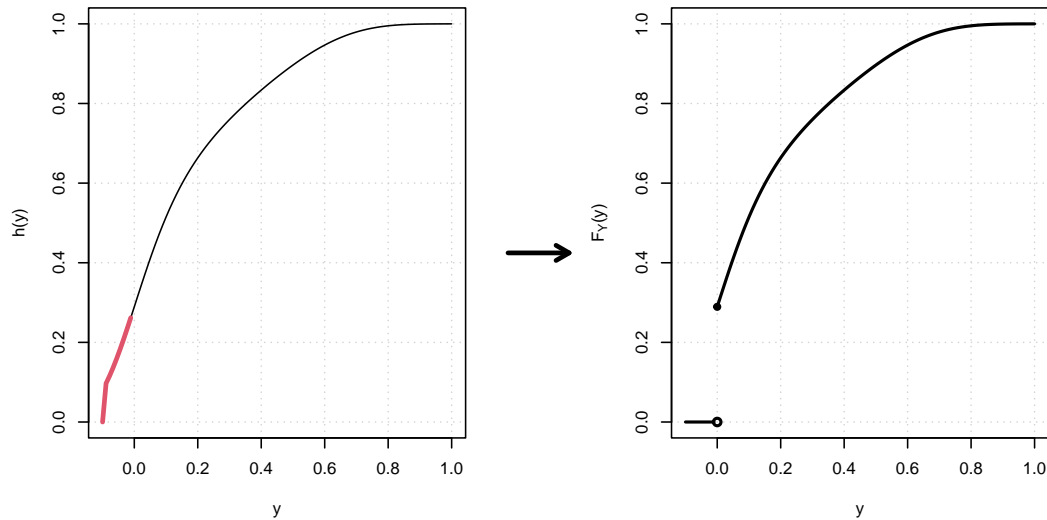

Figure S10: Visual demonstration of how a discrete jump is introduced in the CDF by extending the support and treating the edge cases as censored.

Because the zero-inflated beta model is a mixture of two models, the interpretation of its results is cumbersome. It is not clear which parameters, or combinations of parameters, one needs to inspect to contrast the effects of the various treatments. Moreover, extra steps are needed to calculate the marginal effects of the covariates. In contrast, the mixed-effects transformation model only contains a single set of fixed effects parameters and their interpretation is straightforward: For example, the odds of observing higher proportions of algae cover under the 33% removal treatment is about $\exp(-\widehat{\beta}_{0.33}) = 7.71$ times higher compared to the control group.

To assess the fits of the two models we can marginalize the conditional distributions by integrating over the random effects numerically, and compare against the ECDFs. As Figure S11 shows, both model overestimate the dispersion in the control group.

Systematic differences in the outcome variability in the treatment groups occur in many situations (Douma and Weedon, 2019). By modeling the dispersion separately, we can incorporate such differences in the beta regression model.

```
R> urchin_zib_disp <- glmmTMB(pALGAE ~ TREAT + (1 | PATCH),
+                             ziformula = ~ TREAT, dispformula = ~ TREAT,
+                             data = andrew, family = beta_family())
```
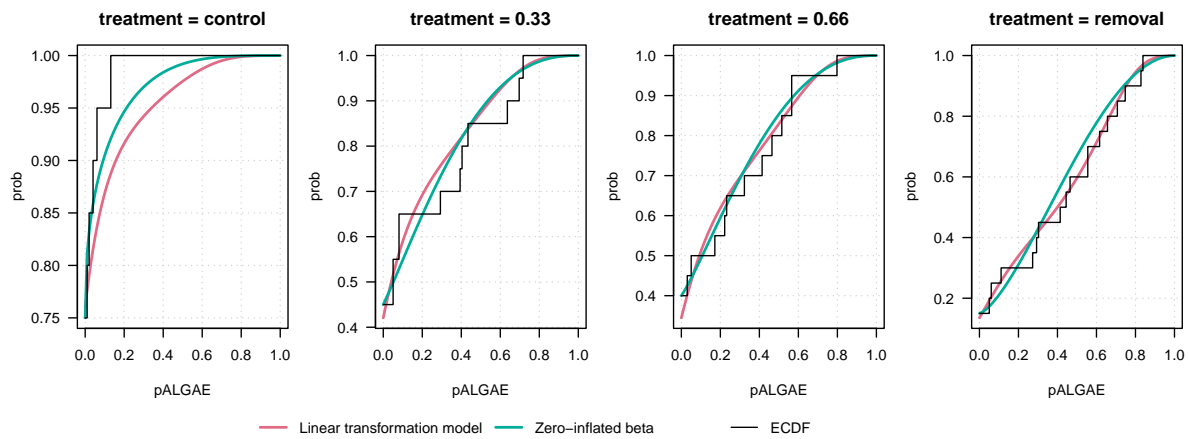
Figure S11: Fitted marginal distributions of algae cover proportion from the zero-inflated beta regression and the mixed-effects transformation model, respectively. The step functions show the empirical cumulative distribution functions in the four treatment groups.

In the mixed-effects linear transformation model, we stratify to the treatment group to allow for separate transformation functions.

```
R> urchin_tram_strat <- ColrME(
+     Surv(pALGAE, pALGAE > 0, type = "left") | 0 + TREAT ~ 1 + (1 | PATCH),
+     bounds = c(-0.1, 1), support = c(-0.1, 1), data = andrew,
+     order = 6, control = optim_control(iter.max = 1e3, eval.max = 1e3,
+                                        rel.tol = 1e-9))
R> summary(urchin_tram_strat)


Stratified Mixed-Effects Continuous Outcome Logistic Regression Model

Formula:
Surv(pALGAE, pALGAE > 0, type = "left") | 0 + TREAT ~ 1 + (1 |
    PATCH)

Fitted to dataset andrew

Fixed effects parameters:
==========================

No estimated shift coefficients.

Random effects:
===============

Grouping factor: PATCH (16 levels)
Standard deviation:
(Intercept)
      1.51
```

```
Log-likelihood: -22.9 (npar = 29)
```

As Figure S12 illustrates, the two models fit the data much better. However, the cost of this flexibility is that we cannot reduce the group comparisons to inference on a small set of parameters anymore.
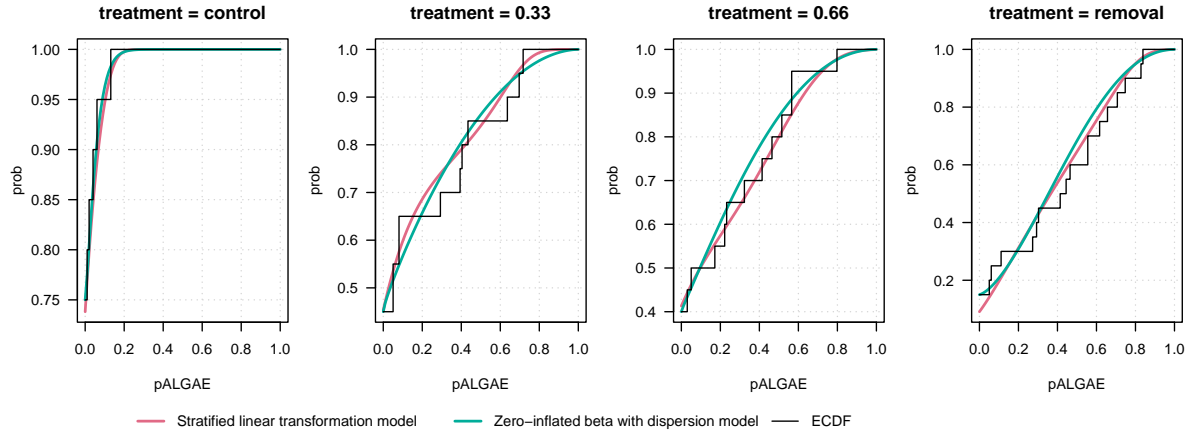

Figure S12: Fitted marginal distributions of algae cover proportion from the zero-inflated beta regression with dispersion model and the stratified mixed-effects transformation model, respectively. The step functions show the empirical cumulative distribution functions in the four treatment groups.

Figures S11 and S12 demonstrate the flexibility of the distribution-free approach of transformation models compared to the parametric alternative. This is also reflected in the log-likelihood values (Table S3).

Table S3: Log-likelihood values of the four model specifications for the sea urchin removal experiment.

	$\log \mathcal{L}$	Number of parameters
Zero-inflated beta without dispersion model	-33.60	10
Zero-inflated beta with dispersion model	-30.93	13
Linear transformation model	-26.27	11
Stratified linear transformation model	-22.86	29

In summary, although the shift-scale beta regression model is not a special case of a transformation model and one thus cannot expect identical results with a specific parameterisation of `tramME::ColrME`, the simpler transformation model (with one instead of two linear predictors) produced a better model fit (when comparing the in-sample log-likelihoods).

S6 Mosquito control trial

Juarez et al. (2021) presented the results of a cluster randomized crossover trial that assessed the efficacy of Autocidal Gravid Ovitrap (AGO) as a tool for against the mosquito species *Aedes aegypti*.

The outcome of interest was the number of female mosquitoes collected on glue boards that were placed either inside or outside of the selected houses in various neighborhoods. Within-year patterns in mosquito counts as well as the coverage of the treatment in different areas were modeled with non-linear smooths, while unobserved household and community level effects were captured by nested random effects. The original article presented the results of a conditional Poisson and a negative binomial model. We reproduce these results with `gamm4`, and also estimate a mixed-effects additive transformation model for count data with "expit" inverse link function. Detailed exposition of count transformation models is given by Siegfried and Hothorn (2020). For fitting such a model, we will use the following custom-made 'CotramME' model class implementing the likelihood for count data via interval censoring (Siegfried and Hothorn, 2020), which is currently not part of the `tramME` package.

```
R> ## Additive count transformation model
R> ## See ?cotram::cotram for the documentation
R> CotramME <- function(formula, data,
+                       method = c("logit", "cloglog", "loglog", "probit"),
+                       log_first = TRUE, plus_one = log_first, prob = 0.9,
+                       ...) {
+     method <- match.arg(method)
+     rv <- all.vars(formula)[1]
+     stopifnot(is.integer(data[[rv]]), all(data[[rv]] >= 0))
+     data[[rv]] <- data[[rv]] + as.integer(plus_one)
+     sup <- c(-0.5 + log_first, quantile(data[[rv]], prob = prob))
+     bou <- c(-0.9 + log_first, Inf)
+     data[[rv]] <- as.Surv(R(data[[rv]], bounds = bou))
+     fc <- match.call()
+     fc[[1L]] <- switch(method, logit = quote(ColrME), cloglog = quote(CoxphME),
+                        loglog = quote(LehmannME), probit = quote(BoxCoxME))
+     fc$method <- NULL
+     fc$plus_one <- NULL
+     fc$prob <- NULL
+     fc$log_first <- log_first
+     fc$bounds <- bou
+     fc$support <- sup
+     fc$data <- data
+     out <- eval(fc, parent.frame())
+     out$call$data <- match.call()$data
+     class(out) <- c("CotramME", class(out))
+     out
+ }
R> mosquito_tram <- CotramME(AEAfemale ~ Year + Income*Placement
+     + s(Week) + s(CovRate200) + (1|HouseID)
+     + (1|Community), offset = -log(daystrapping), data = AGO,
+     method = "logit", order = 5, log_first = TRUE, prob = 0.9)
```

Table S4 compares the log-likelihood values of the three model versions. In terms of in-sample model fit, as measured by the log-likelihood value, both the negative binomial and the transformation model perform much better than the Poisson GAMM. The results suggest slight improvement in the model fit when we relax the conditional distribution assumption of the negative binomial GAMM and

follow the distribution-free transformation model approach.

Table S4: Log-likelihood values of the fitted Poisson and negative binomial GAMMs reproduced from Juarez et al. (2021) along with the log-likelihood of an additive transformation model for count data.

	Log-likelihood
Poisson GAMM	-6875.73
Negative binomial GAMM	-4883.26
Additive count transformation model	-4873.07

We will now concentrate on comparing the estimates from the negative binomial and the count transformation models. Note that the scales on which the parameters are interpreted are different in the two models: While in the negative binomial model, the parametric and smooth terms affect the log of the conditional mean of the outcome, in the transformation model with "logit" link (i.e., "expit" inverse link), they are interpreted on the log-odds scale. Unlike in the example application of Section S4, we cannot easily transform the negative binomial parameters to the probability scale. Although the magnitudes of the effect estimates of the two models are not directly comparable, their directions, significance and the general shapes of the smooths can be contrasted.

Figure S13 compares the smooth estimates of the GAMM from `gamm4` and the transformation model from `tramME`. Although the within-year time patterns (`s(Week)`) from the two models are almost identical (on different scales), the difference of the smooth estimates of the coverage rate (`s(CovRate200)`) is marked. The general shapes of the smooths are similar, but the negative binomial GAMM penalizes it more, which is also reflected in the EDFs: 2.96 and 17.49 for the negative binomial and count transformation models, respectively.

Because the parametric and smooth terms of the two models are defined on different scales, the magnitudes of the effect estimates are not directly comparable. As Table S5 shows, the directions of the effects match and neither model finds evidence that the main effect of middle income is different from zero.

Again, the models compared for this example are not nested and it is therefore hard to compare them directly. The transformation model leads to a similar model interpretation as the model based on the negative-binomial distribution. Model uncertainty was larger in the transformation model, at

Table S5: Point estimates and 95% confidence intervals of the parametric fixed effects terms from the negative binomial and count transformation models of the mosquito control data by Juarez et al. (2021). Note that the scale of the parameters are different and the effect sizes are not directly comparable.

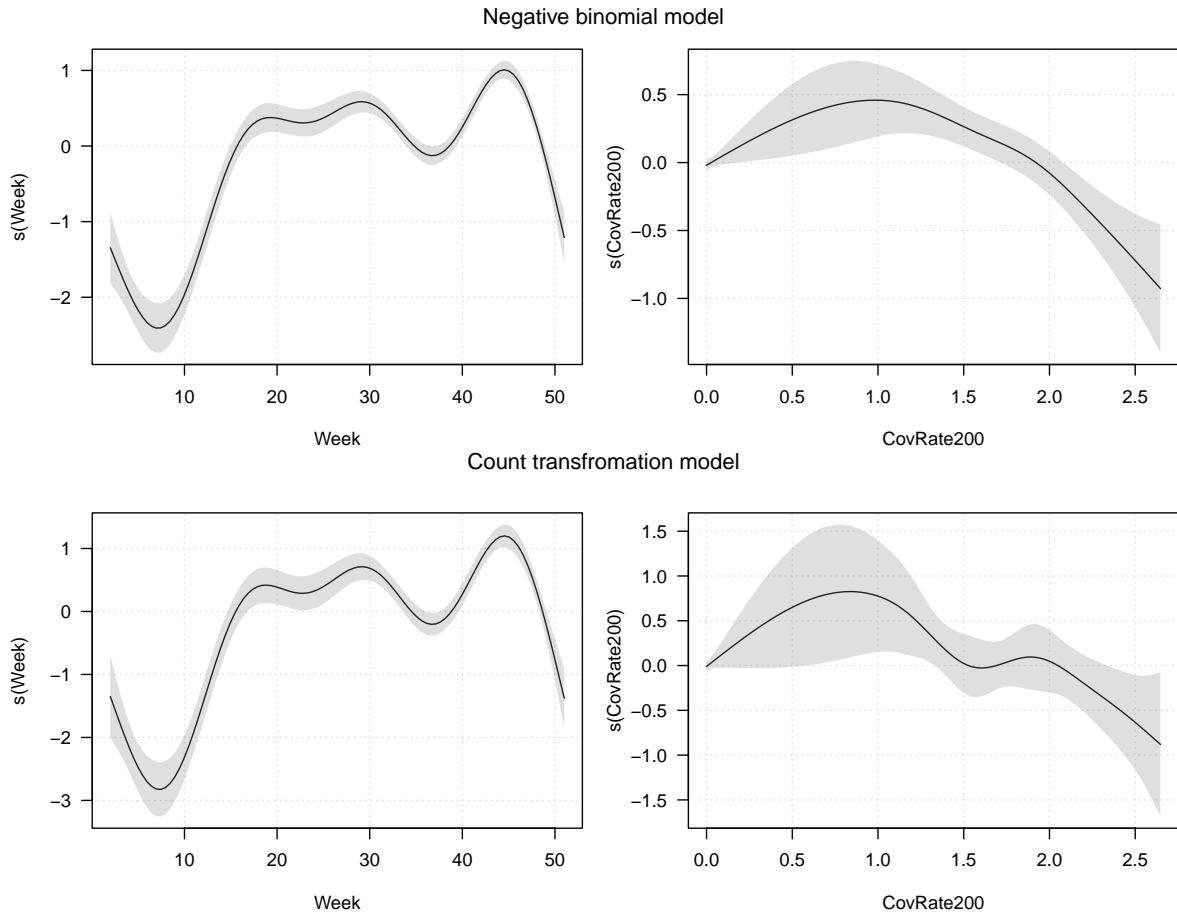|	Negative binomial		Count transformation	
	$\widehat{\beta}$	95% CI	$\widehat{\beta}$	95% CI
Year = 2018	-0.20	-0.34 — -0.06	-0.35	-0.55 — -0.15
Income = middle	-0.78	-1.69 — 0.13	-0.83	-2.02 — 0.36
Placement = out	2.37	2.22 — 2.52	3.01	2.79 — 3.24
Income = middle & Placement = out	0.38	0.13 — 0.64	0.51	0.16 — 0.86

Figure S13: Smooth terms from the negative binomial and transformation models of the *A. aegypti* counts. The dashed lines and the grey areas denote the 95% confidence intervals

least for the nonlinear effect of `CovRate200`, and thus one might wonder if the stricter distributional assumption lead to overconfident model interpretation.

S7   Simulation experiments

The goals of the following simulation experiments are threefold. 1. First, we want to evaluate the loss of model performance when using the flexible, distribution-free mixed-effects additive transformation model approach in a scenario when a simple normal linear additive mixed model is appropriate. 2. We are interested in comparing the performances of the normal and non-normal model under a highly non-normal data-generating process. 3. Finally, we investigate the performance of the non-normal transformation model in the case of interval-censored data.

We consider a normal and a non-normal data-generating process for simulating grouped data with non-linear covariate effects. The conditional distributions corresponding to both of these can be written in the general transformation model form:

$$\mathbb{P}(Y \le y \mid x_1, x_2, \gamma_j) = \Phi\left(h(y) - \beta x_1 - f(x_2) - \gamma_j\right) \qquad \gamma_{\{1:J\}} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma),$$

where Φ denotes the CDF of the standard Gaussian distribution, βx_1 is the parametric fixed-effect term, $f(x_2)$ is a non-linear term and γ_j is the random intercept of the group $j = 1 \dots J$ with σ standard deviation. The data are simulated by numerically inverting the conditional distribution function.

```
R> ##' @param n Numeric vector, number of observations in each group
R> ##' @param beta Numeric, effect size
R> ##' @param sfun Function with a numeric argument, non-linear smooth shift term
R> ##' @param sigma Numeric, SD of random intercepts
R> ##' @param hinv Function with a numeric argument, inverse transformation function
R> ##' @param link Function with a single numeric argument on [0, 1] link function
R> ##' @param scale Numeric, optional scaling constant on the transformation scale
R> ##' @param seed Seed for the random number generator
R> ##' @param two_sets Logical; generate both an estimation and a test sample
R> gen_smpl <- function(n, beta, sfun, sigma, hinv, link, scale = 1,
+                       seed = NULL, two_sets = TRUE) {
+     ## -- setting up the seed
+     if (!exists(".Random.seed", envir = .GlobalEnv, inherits = FALSE))
+       runif(1)
+     if (is.null(seed))
+       RNGstate <- get(".Random.seed", envir = .GlobalEnv)
+     else {
+       R.seed <- get(".Random.seed", envir = .GlobalEnv)
+       set.seed(seed)
+       RNGstate <- structure(seed, kind = as.list(RNGkind()))
+       on.exit(assign(".Random.seed", R.seed, envir = .GlobalEnv))
+     }
+     if (two_sets) n <- rep(n, 2)
+     x1 <- runif(sum(n))
+     x2 <- runif(sum(n))
+     gr <- factor(rep(seq_along(n), n))
+     re <- rep(rnorm(length(n), mean = 0, sd = sigma), n)
+     lp <- x1 * beta + sfun(x2) + re
+     y  <- hinv((link(runif(sum(n))) + lp) * scale)
+     if (two_sets) {
+       n <- sum(n) / 2
+       out <- list(est  = data.frame(x1 = x1[1:n], x2 = x2[1:n], gr = gr[1:n],
+                                     y = y[1:n]),
+                   test =  data.frame(x1 = tail(x1, n), x2 = tail(x2, n),
+                                      gr = tail(gr, n), y = tail(y, n)))
+     } else {
+       out <- data.frame(x1 = x1, x2 = x2, gr = gr, y = y)
+     }
+     out
+ }
```

For the inverse baseline transformations and the non-linear shift term we set the following functional forms:

$$h_1(y) = y, \qquad h_2(y) = \Phi^{-1}\left(F_{\chi_3^2}(y)\right), \qquad f(x) = \xi \sin(\pi x),$$

with $\Phi^{-1}(\cdot)$ as the quantile function of the standard normal distribution and $F_{\chi_3^2}(\cdot)$ is the distribution function of the χ^2-distribution with 3 degrees of freedom. By using a linear function, $h_1(y)$ as the trans-

formation function, we assume a normal additive mixed-model. In contrast, setting $h_2(t)$ introduces non-normality. In this case, the conditional distribution implied by the data-generating mechanism is χ^2 (df = 3) when the linear predictor is zero. In the simulations, we use $\beta = \sqrt{2}\Phi^{-1}(0.7)$ so that the effect size of the parametric fixed-effect term is 0.7 on the probabilistic index scale in both the normal and the non-normal case (see Section S4 for details). Values of x_1 and x_2 are drawn from the uniform distribution on the interval between 0 and 1 for each observation separately. The the parameters σ and ξ are set to values that result in equal variances for the parametric (βx_1), the non-linear $(f(x_2))$ and the random-effects terms of the linear predictor on the transformation scale.

For the third question, to generate censored outcomes on an interval of a given length from continuous data, we define the following function:

```
R> make_intcens <- function(x, length = 1) {
+    Surv(floor(x / length) * length, ceiling(x / length) * length,
+        type = "interval2")
+ }
```

In all scenarios, we simulate moderate-sized datasets with 100 subjects/groups ($J = 100$) and four observations per each group. We repeat all simulations 500 times.

Our primary focus in these simulation experiments is on inference about the fixed-effects parameter β and the general predictive performances of the model variants evaluated through the out-of-sample log-likelihood. For the out-of-sample log-likelihood calculations, we integrate over the random intercepts in the log-likelihood (i.e., the exponential of Equation S2) of a new set of observations and evaluate it at the fitted values of the remaining parameters (coefficients corresponding to the baseline transformation, the fixed effect and the smooth term as well as the variance component and smoothing parameters).

S7.1   Normal data-generating process

To investigate our first question, we draw samples from the (conditionally) normal additive mixed model (i.e., $h(y) = h_1(y) = y$) and fit a mixed-effects additive transformation model (with standard normal inverse link), as well as the correctly specified normal additive mixed model, to it.

According to Figure S14, both the correctly specified additive model and the flexible transformation model results in unbiased estimates for the β parameter, with no visible additional variability in the case of the more general, distribution-free approach. The coverage rates of the 95% Wald confidence intervals from the two models are very close to their nominal levels.

```
R> cvi <- get_res(sims1, what = function(x) {
+    ci <- x$beta + x$beta_se * qnorm(0.975) * c(-1, 1)
+    (ci[1] <= b) && (b < ci[2])
+ })
R> (cvr <- colMeans(cvi, na.rm = TRUE)) ## Coverage rate

   LmME BoxCoxME
  0.928    0.930

R> sqrt(cvr * (1 - cvr) / nrow(cvi)) ## Monte Carlo SE

   LmME BoxCoxME
 0.0116   0.0114
```
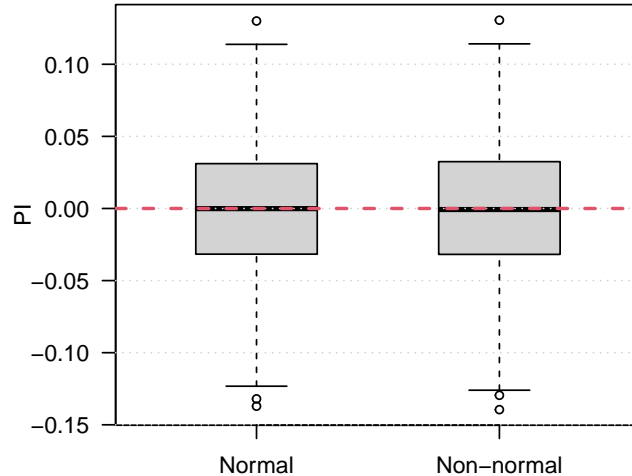
Figure S14: Bias of the coefficient of the parametric fixed term in the case of the conditionally normal data-generating process. The results are presented on the probabilistic index scale. "Normal" corresponds to the results from the normal additive mixed model, while "Non-normal" denotes the distribution-free mixed-effects additive transformation model.

The Monte Carlo distributions of out-of-sample log-likelihood are presented in Figure S15. For each simulated dataset for fitting the two models, we generated an independent evaluation dataset of the same size and structure. We calculated the log-likelihood of this new data given the parameters fitted to the estimation sample. The predictive performance of the two models are very close to each other, with just slight reduction in the log-likelihood values for the more complex transformation model.

To visually assess, how well the mixed-effects additive transformation model approximates the conditionally normal response distribution, we plot the fitted baseline transformation functions $(\widehat{h}(y))$, and compare to the true function $h(y) = y$ (Figure S16).

S7.2 Non-normal data-generating process

In many applied regression problems, the assumption that the outcome is normally distributed, conditionally on the predictors, is violated. Normal additive mixed models are still often used in these cases, mainly due to the fact that the normal linear model can be remarkably robust to some violations of the model assumptions (Schielzeth et al., 2020). In the next simulation experiment, we compare the normal additive mixed model and non-normal mixed-effects transformation model based on the inference on the parametric fixed term and out-of-sample predictive performance under a non-normal, skewed data-generating process.

As an additional goal, we also investigate the performance of the transformation approach when the outcome variable is interval-censored, i.e., none of the values are exactly observed, only up to an interval. For each iteration in the Monte Carlo simulations, we generate a dataset with exactly observed outcomes (from the conditionally non-normal data-generating process) and fit the normal additive model and the non-normal transformation model to it. We then create the censored version of the same variable by binning its values into intervals (of length two in our simulations) and define the corresponding `Surv` object.
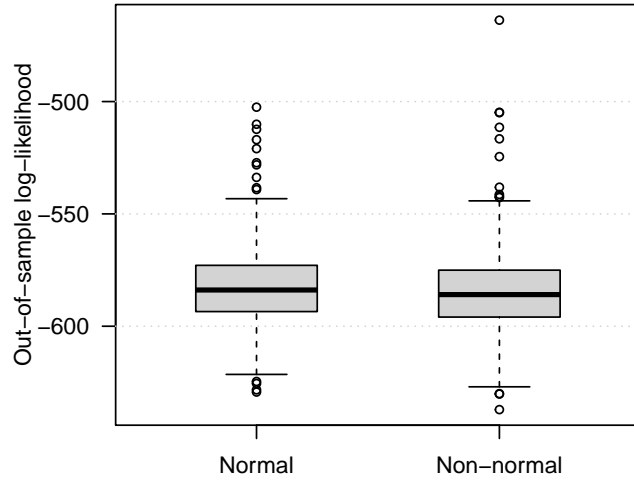
Figure S15: Predictive performance of the normal additive mixed model and the mixed-effects additive transformation model evaluated by the out-of-sample log-likelihood of an independently generated dataset.
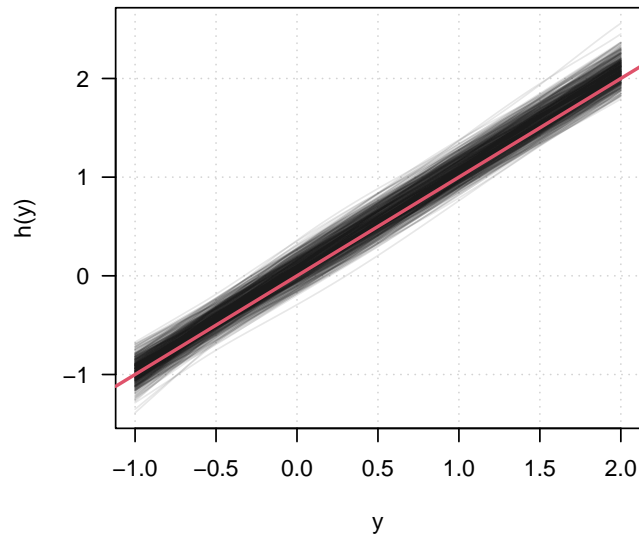

Figure S16: Fitted baseline transformation functions from the mixed-effects additive transformation model under normal data-generating mechanism. The red line shows the true value.

```
R> dat <- gen_smpl(rep(4, 100), beta = b, sfun = sf, sigma = sig,
+                   hinv = hi2, link = lnk, seed = 1, two_sets = FALSE)
R> head(dat$y) ## exact values

[1]  5.61  6.61  3.48 14.39  1.45  2.05

R> head(make_intcens(dat$y, length = 2)) ## interval-censored version

[1] [ 4,  6] [ 6,  8] [ 2,  4] [14, 16] [ 0,  2] [ 2,  4]
```

In the next step, we refit the transformation model using the interval-censored data, and compare its results to the additive mixed model and the transformation model with exact responses. To ensure the comparability of the out-of-sample predictions, we only transform the response in the estimation sample, but not the evaluation sample, to interval-censored.

Figure S17 show the bias in the estimates of the coefficient β in the case of the two models and the transformation model under interval-censored data. In the case of the (misspecified) normal additive mixed-model, there is a sign of slight downward bias, while the transformation model gives unbiased estimates for the parametric fixed-effects coefficient, irrespective of whether it uses the exact observations or their interval-censored version. The coverages of the 95% Wald confidence intervals are close to their nominal levels in all cases.

```
R> cvi <- get_res(sims2, what = function(x) {
+    ci <- x$beta + x$beta_se * qnorm(0.975) * c(-1, 1)
+    (ci[1] <= b) && (b < ci[2])
+  })
R> (cvr <- colMeans(cvi, na.rm = TRUE)) ## Coverage rate

     LmME    BoxCoxME BoxCoxME_ic
    0.950       0.950       0.938

R> sqrt(cvr * (1 - cvr) / nrow(cvi)) ## Monte Carlo SE

      LmME    BoxCoxME BoxCoxME_ic
   0.00975     0.00975     0.01078
```

The differences between the normal and the distribution-free approaches are more marked in their out-of-sample predictive performances. As Figure S18 shows, the normal additive mixed model performs poorly compared to the transformation model, even in the interval-censored case, when the observations in the estimation sample are not exactly observed.

Figure S19 presents the fitted smooth non-linear terms from the non-normal transformation model in the case of the exactly observed and interval-censored scenarios. As the results show, the model is able to recover the non-linear term even when the information on the outcome is limited to intervals.

In summary, the results of the simulation experiments presented above suggest that, at least in the scenarios considered, the cost of using the distribution-free transformation model framework in mixed-effects additive modeling is low, while it can improve the inference and predictive performance when the distributional assumptions of the available alternatives are not met. Moreover, our results confirm that the model provides reliable estimates when the outcome variable is censored.
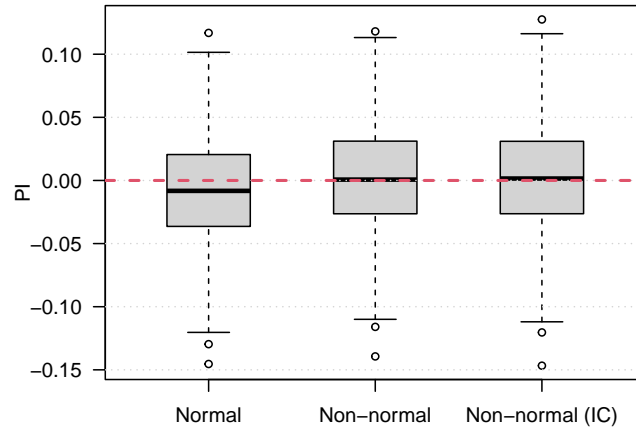
Figure S17: Bias of the coefficient of the parametric fixed term in the case of the conditionally non-normal data-generating process. The results are presented on the probabilistic index scale. "Normal" corresponds to the results from the normal additive mixed model, "Non-normal" denotes the distribution-free mixed-effects additive transformation model and "Non-normal (IC)" show the results of the transformation model under interval-censoring.
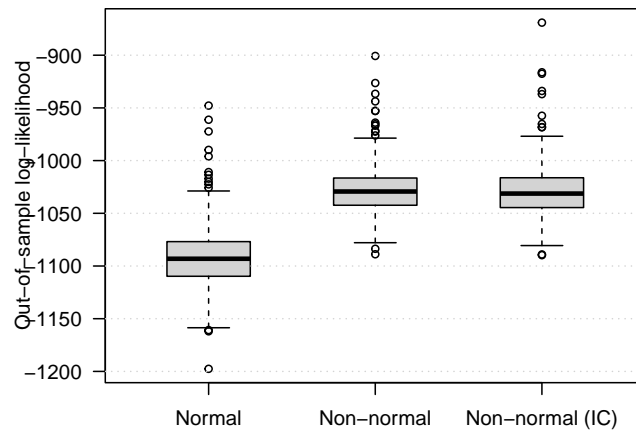

Figure S18: Predictive performance of the normal additive mixed model and the mixed-effects additive transformation model with exact ("Non-normal") and interval-censored ("Non-normal (IC)") observations evaluated by the out-of-sample log-likelihood of an independently generated dataset.
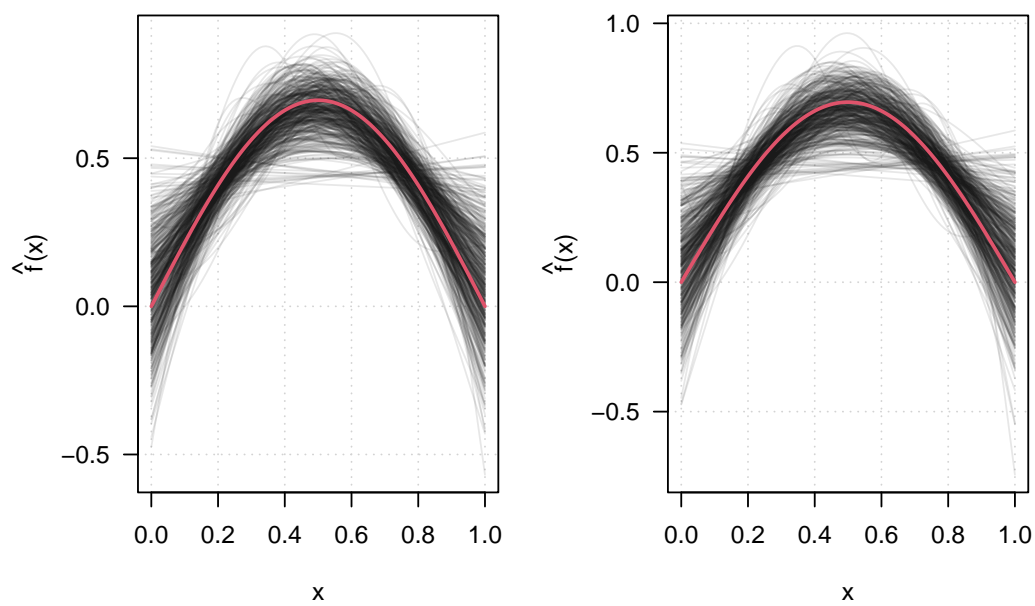
27

Figure S19: Fitted non-linear effects from the mixed-effects additive transformation model with exact observations (*left panel*) and interval-censored observations (*right panel*) under the non-normal data-generating mechanism. The red curves show the true function.

References

Neil L. Andrew and Antony J. Underwood. Density-dependent foraging in the sea urchin Centrostephanus rodgersii on shallow subtidal reefs in New South Wales, Australia. *Marine Ecology Progress Series*, 99:89–98, 1993. doi:10.3354/meps099089.

Douglas Bates, Martin Mächler, Benjamin M. Bolker, and Steve Walker. Fitting linear mixed-effects models using `lme4`. *Journal of Statistical Software*, 67(1):1–48, 2015. doi:10.18637/jss.v067.i01.

Mollie Brooks, Ben Bolker, Kasper Kristensen, Martin Maechler, Arni Magnusson, Hans Skaug, Anders Nielsen, Casper Berg, and Koen van Bentham. *glmmTMB: Generalized Linear Mixed Models using Template Model Builder*, 2024. URL `https://CRAN.R-project.org/package=glmmTMB`. R package version 1.1.10.

Mollie E. Brooks, Kasper Kristensen, Koen J. van Benthem, Arni Magnusson, Casper W. Berg, Anders Nielsen, Hans J. Skaug, Martin Mächler, and Benjamin M. Bolker. glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling. *The R Journal*, 9(2): 378–400, 2017. doi:10.32614/RJ-2017-066.

Jacob C. Douma and James T. Weedon. Analysing continuous proportions in ecology and evolution: A practical introduction to beta and Dirichlet regression. *Methods in Ecology and Evolution*, 10(9): 1412–1430, 2019. doi:10.1111/2041-210X.13234.

Jana Englmeier, Oliver Mitesser, M. Eric Benbow, Torsten Hothorn, Christian von Hoermann, Caryl Benjamin, Ute Fricke, Cristina Ganuza, Maria Haensel, Sarah Redlich, Rebekka Riebl, Sandra Rojas Botero, Thomas Rummler, Ingolf Steffan-Dewenter, Elisa Stengel, Cynthia Tobisch, Johannes Uhler, Lars Uphus, Jie Zhang, and Jörg Müller. Diverse effects of climate, land use, and insects on dung and carrion decomposition. *Ecosystems*, 2022. doi:10.1007/s10021-022-00764-7.

Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, and Brian Marx. *Regression: Models, Methods and Applications*. Springer, New York, 2013. doi:10.1007/978-3-642-34333-9.

Rida T. Farouki. The Bernstein polynomial basis: A centennial retrospective. *Computer Aided Geometric Design*, 29(6):379–419, 2012. doi:10.1016/j.cagd.2012.03.001.

C. Paddy Farrington. Residuals for proportional hazards models with interval-censored survival data. *Biometrics*, 56(2):473–482, 2000. doi:10.1111/j.0006-341X.2000.00473.x.

Torsten Hothorn. Most likely transformations: The mlt package. *Journal of Statistical Software*, 92 (1):1–68, 2020. doi:10.18637/jss.v092.i01.

Torsten Hothorn, Lisa Möst, and Peter Bühlmann. Most likely transformations. *Scandinavian Journal of Statistics*, 45(1):110–134, 2018. doi:10.1111/sjos.12291.

Kristin B. Hulvey, Cassie D. Mellon, and Andrew R. Kleinhesselink. Rotational grazing can mitigate ecosystem service trade-offs between livestock production and water quality in semi-arid rangelands. *Journal of Applied Ecology*, 58(10):2113–2123, 2021. doi:10.1111/1365-2664.13954.

Jose G. Juarez, Luis F. Chaves, Selene M. Garcia-Luna, Estelle Martin, Ismael Badillo-Vargas, Matthew C. I. Medeiros, and Gabriel L. Hamer. Variable coverage in an Autocidal Gravid Ovitrap intervention impacts efficacy of Aedes aegypti control. *Journal of Applied Ecology*, 58(10): 2075–2086, 2021. doi:10.1111/1365-2664.13951.

Robert E. Kass and Duane Steffey. Approximate Bayesian inference in conditionally independent hierarchical models (parametric empirical Bayes models). *Journal of the American Statistical Association*, 84(407):717–726, 1989. doi:10.1080/01621459.1989.10478825.

John P. Klein and Melvin L. Moeschberger. *Survival Analysis: Techniques for Censored and Truncated Data*. Statistics for biology and health. Springer, New York, 2nd edition, 2003.

Kasper Kristensen, Anders Nielsen, Casper W. Berg, Hans J. Skaug, and Bradley M. Bell. TMB: Automatic differentiation and Laplace approximation. *Journal of Statistical Software*, 70(5):1–21, 2016. doi:10.18637/jss.v070.i05.

David Ruppert, M. P. Wand, and Raymond J. Carroll. *Semiparametric Regression*. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, Cambridge; New York, 2003.

Holger Schielzeth, Niels J. Dingemanse, Shinichi Nakagawa, David F. Westneat, Hassen Allegue, Céline Teplitsky, Denis Réale, Ned A. Dochtermann, László Zsolt Garamszegi, and Yimen G. Araya-Ajoy. Robustness of linear mixed-effects models to violations of distributional assumptions. *Methods in Ecology and Evolution*, 11(9):1141–1152, 2020. doi:10.1111/2041-210X.13434.

Sandra Siegfried and Torsten Hothorn. Count transformation models. *Methods in Ecology and Evolution*, 11(7):818–827, 2020. doi:10.1111/2041-210X.13383.

Balint Tamasi. *tramME: Transformation Models with Mixed Effects*, 2024. URL http://ctm.R-forge.R-project.org. R package version 1.0.7.

Bálint Tamási and Torsten Hothorn. **tramME**: Mixed-effects transformation models using Template Model Builder. *The R Journal*, 13(1):581–594, 2021. doi:10.32614/RJ-2021-075.

Olivier Thas, Jan De Neve, Lieven Clement, and Jean-Pierre Ottoy. Probabilistic index models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(4):623–671, 2012. doi:10.1111/j.1467-9868.2011.01020.x.

Matt P. Wand. Smoothing and mixed models. *Computational Statistics*, 18(2):223–249, 2003. doi:10.1007/s001800300142.

Simon Wood and Fabian Scheipl. *gamm4: Generalized Additive Mixed Models using 'mgcv' and 'lme4'*, 2020. URL https://CRAN.R-project.org/package=gamm4. R package version 0.2-6.

Simon N Wood. Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association*, 99(467):673–686, 2004. doi:10.1198/016214504000000980.

Simon N. Wood. *Generalized Additive Models: An Introduction with R*. CRC Press/Taylor & Francis Group, Boca Raton, FL, U.S.A., second edition, 2017.

Simon N. Wood, Natalya Pya, and Benjamin Säfken. Smoothing parameter and model selection for general smooth models. *Journal of the American Statistical Association*, 111(516):1548–1563, 2016. doi:10.1080/01621459.2016.1180986.

Nan Zheng and Noel Cadigan. Frequentist delta-variance approximations with mixed-effects models and TMB. *Computational Statistics & Data Analysis*, 160:107227, 2021. doi:10.1016/j.csda.2021.107227.

```
R> sessionInfo()

R version 4.4.2 (2024-10-31)
Platform: x86_64-redhat-linux-gnu
Running under: Fedora Linux 41 (Workstation Edition)

Matrix products: default
BLAS/LAPACK: FlexiBLAS OPENBLAS-OPENMP;  LAPACK version 3.12.0

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

time zone: Europe/Zurich
tzcode source: system (glibc)

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
 [1] gamm4_0.2-6    lme4_1.1-35.5  Matrix_1.7-1   xtable_1.8-4
 [5] glmmTMB_1.1.10 mgcv_1.9-1     nlme_3.1-166   survival_3.7-0
 [9] tramME_1.0.7   tram_1.2-0     mvtnorm_1.3-2  mlt_1.6-1
[13] basefun_1.1-4  variables_1.1-1

loaded via a namespace (and not attached):
 [1] coneproj_1.19      compiler_4.4.2       highr_0.11
 [4] Rcpp_1.0.13-1      orthopolynom_1.0-6.1 splines_4.4.2
 [7] boot_1.3-31        lattice_0.22-6       alabama_2023.1.0
[10] TH.data_1.1-2      Formula_1.2-5        knitr_1.49
[13] rbibutils_2.3      MASS_7.3-61          nloptr_2.1.1
[16] minqa_1.2.8        TMB_1.9.15           multcomp_1.4-26
[19] BB_2019.10-1       xfun_0.49            quadprog_1.5-8
[22] polynom_1.4-1      Rdpack_2.6.2         grid_4.4.2
[25] sandwich_3.1-0     reformulas_0.4.0     evaluate_1.0.1
[28] numDeriv_2016.8-1.1 codetools_0.2-20    zoo_1.8-12
[31] tools_4.4.2
```