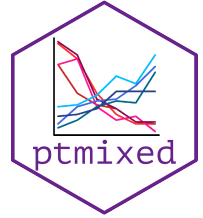
ptmixed: an R package for flexible modelling of overdispersed longitudinal counts



Mirko Signorelli[1]
Twitter: @signormirko

Joint work with Roula Tsonaka[1] and Pietro Spitali[2]

[1] Department of Biomedical Data Sciences, Leiden University Medical Center
[2] Department of Human Genetics, Leiden University Medical Center

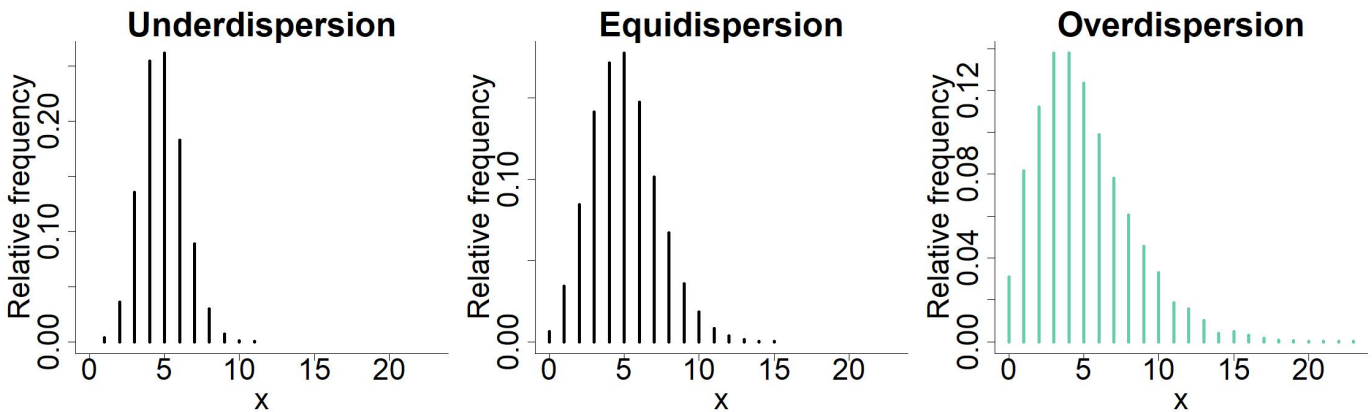
June 19, 2020
e-Rum2020

Leiden University Medical Center

Universiteit Leiden

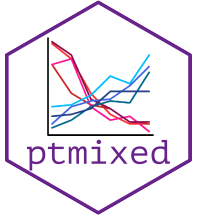
Count data come in different shapes

- Count data typically classified as
  - Underdispersed: $Var(X) < E(X)$
  - Equidispersed: $Var(X) = E(X)$
  - **Overdispersed**: $Var(X) > E(X)$



- Most common models for overdispersed counts: **negative binomial (NB)** GLM and GLMM

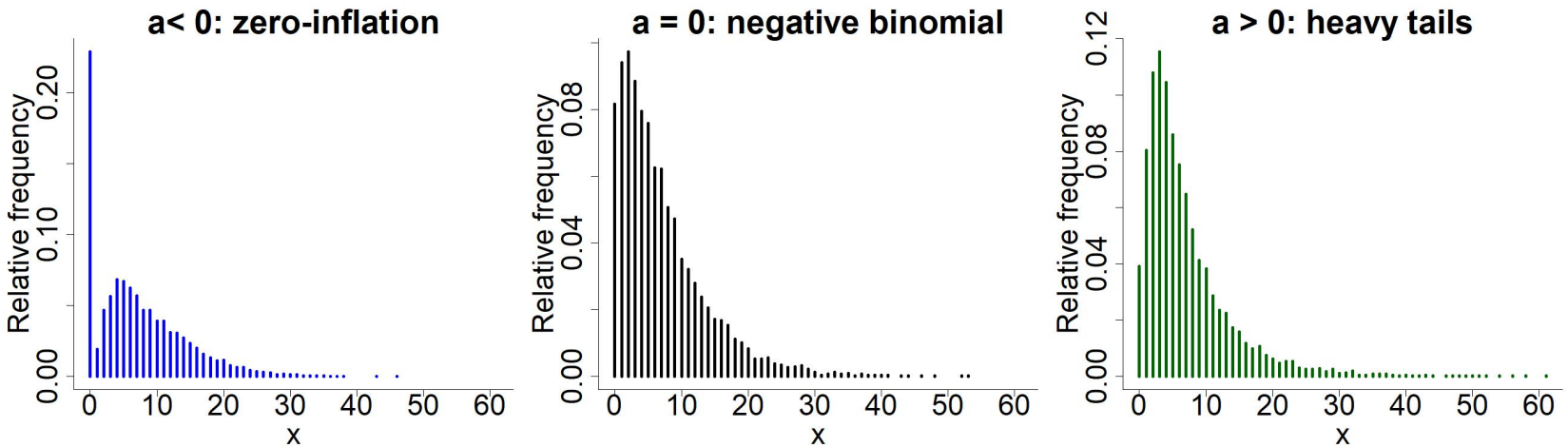
Poisson-Tweedie: fifty shades of overdispersion



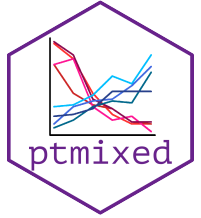
- El-Shaarawy et al. (2011) showed that NB can't capture well different levels of **zero-inflation (ZI)** and **heavy-tails (HT)** commonly observed with overdispersed counts
- Alternative: use the **Poisson-Tweedie (PT)** distribution

$$\mathbf{Y} \sim \mathbf{PT}(\mu, \mathbf{D}, \mathbf{a})$$

- Power $a \leq 1$ models extra ZI and HT for given dispersion D:



ptmixed: flexible modelling of longitudinal counts



Applicability of Poisson-Tweedie:

- cross-sectional data $\rightarrow$ PT GLM (Esnaola et al., 2013) ✓
- longitudinal data $\rightarrow$ no GLMM extension ✗

Our proposal: ptmixed

- We propose a **Poisson-Tweedie GLMM** to flexibly model **longitudinal counts** with different levels of **zero-inflation** and **heavy-tails**
- Implementation: `R` package `ptmixed` (published on `CRAN`)

Estimation of the PT mixed model

Poisson-Tweedie GLMM

$$Y_{ij} \mid v_i \sim \text{PT}(\mu_{ij},\ D,\ a)$$
$$\log(\mu_{ij}) = x_{ij}^T \beta + z_{ij}^T v_i$$
$$v_i \overset{\text{i.i.d.}}{\sim} N(0, \Sigma)$$

where $\beta \rightarrow$ fixed effects, $v_i \rightarrow$ random effects

How to estimate this model?

- Likelihood evaluation: approximation of PT pmf (Esnaola et al., 2013) + adaptive Gauss-Hermite quadrature
- Model estimation: maximum likelihood estimation

More details $\rightarrow$ Signorelli et al. (2020, in press), arXiv:2004.11193

Data preparation

1) Load package and data

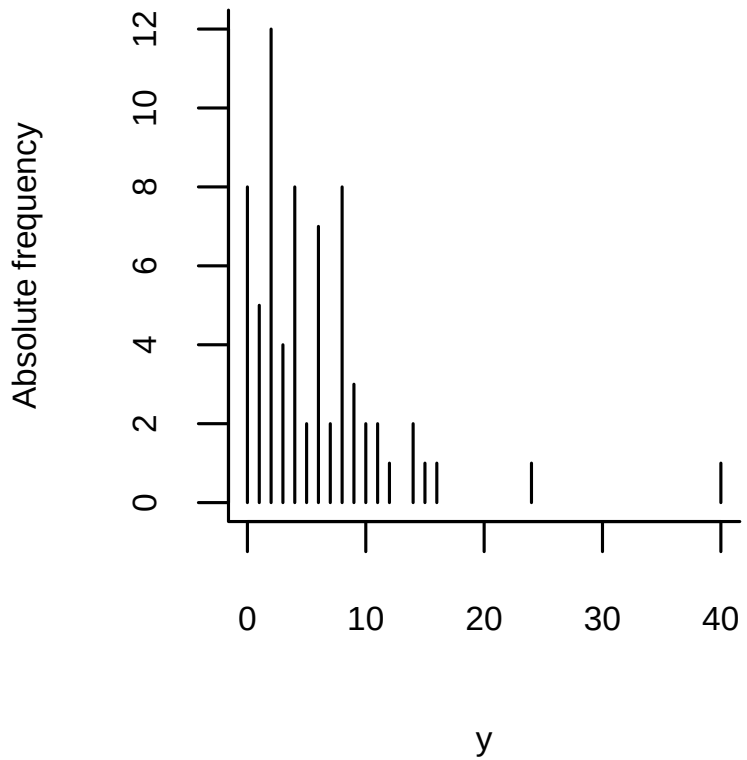
```
library(ptmixed)
head(data.long)
```

```
##    y id   group time
## 1  6  1 placebo    0
## 2  0  1 placebo    1
## 3  2  1 placebo    2
## 4  1  1 placebo    3
## 5  1  1 placebo    4
## 6 12  2 placebo    0
```

- `y` → response variable
- `id` → subject id $(1, \ldots, n)$
- `group` → 2 groups (treated / placebo)
- `time` → 5 time points $(0 \rightarrow 4)$

2) Visualize distribution of the response

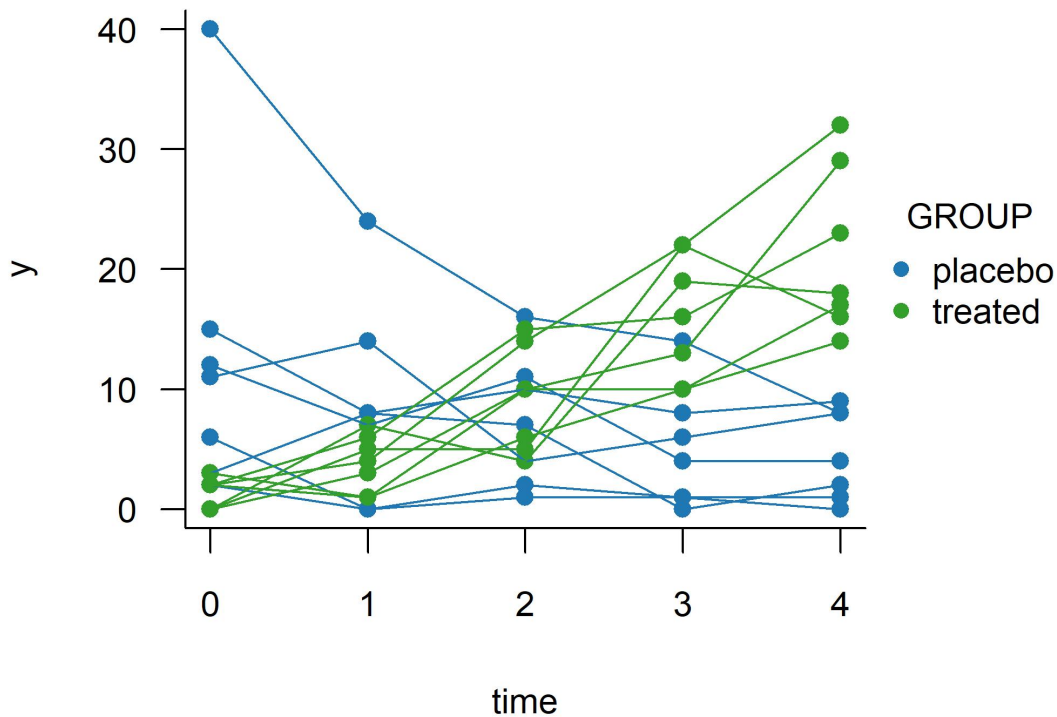
```
pmf(data.long$y)
```



make.spaghetti()

3) Visualize longitudinal trajectories

```
make.spaghetti(x = time, y = y, id = id,
        group = group, data = data.long)
```

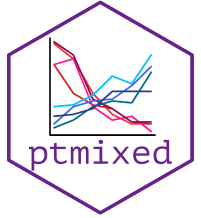


make.spaghetti()

```
make.spaghetti(x, y, id, group, data)
```

. . . even easier than making spaghetti!

ptmixed(): estimate the Poisson-Tweedie GLMM!



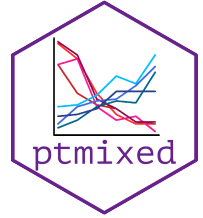
4) Fit the Poisson-Tweedie GLMM

```
fit <- ptmixed(fixef.formula = y ~ group*time,
                    id = id, data = data.long)
```

The package also allows to estimate a few simpler models:

Function	Model
nbglm	Negative binomial GLM
nbmixed	Negative binomial GLMM
ptglm	Poisson-Tweedie GLM
ptmixed	Poisson-Tweedie GLMM

summary()

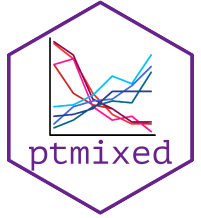


5) View estimated $\hat{\beta}$, $\hat{D}$, $\hat{\alpha}$, $\hat{\sigma}_0^2$, standard errors and univariate Wald tests

```
summary(fit)
```

```
## Loglikelihood: -140.539
## Parameter estimates:
##                     Estimate Std. error       z p.value
## (Intercept)           2.0012     0.2888  6.9286  0.0000
## grouptreated         -1.4546     0.4677 -3.1102  0.0019
## time                 -0.1360     0.0765 -1.7784  0.0753
## grouptreated:time     0.5115     0.1465  3.4908  0.0005
##
## Dispersion = 1.64
## Power = -0.14
## Variance = 0.42
```

Predicted random effects



6) Compute the best linear unbiased predictor of the random effects

```
ranef(fit)
```

```
##       1       2       3       4       5       6
## -0.7467  0.3204  1.1391 -1.1709  0.4988  0.4097
##       7       8       9      10      11      12
## -0.2220  0.3224 -0.3528 -0.0221  0.1815 -0.2705
##      13      14
## -0.1814  0.6586
```

Multivariate Wald test

7) Test more complex hypotheses, e.g. $H_0 : \beta_1 = \beta_3 = 0$[1]:

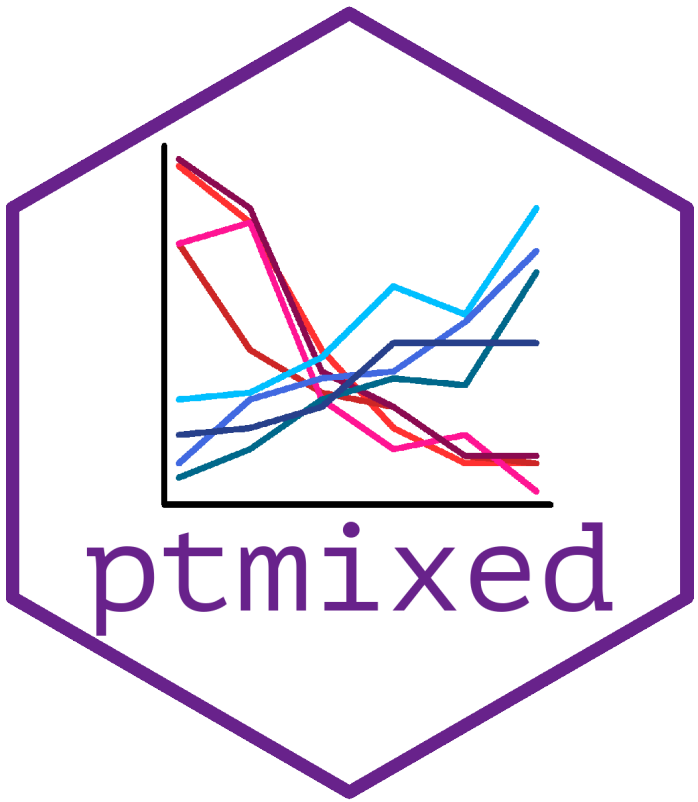
```
L = matrix(0, nrow = 2, ncol = 4)
L[1, 2] = L[2, 4] = 1
ptmixed::wald.test(fit, L = L, k = c(0, 0))
```

```
##       chi2 df             P
## 1 14.1051  2 0.0008651998
```

- Likelihood ratio test also possible, see sec. 2.4 of Signorelli et al. (2020)

[1] NB: the hypothesis is coded in the form $L\beta = k$

Useful resources

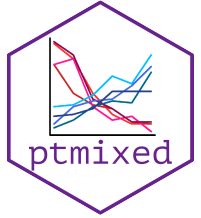


More info about ptmixed here:

1) arXiv:2004.11193 preprint of Signorelli et al. (2020, in press)
2) CRAN package page
3) Vignette of the R package:

```
browseVignettes("ptmixed")
```

References



El-Shaarawi, A. H., Zhu, R., & Joe, H. (2011). Modelling species abundance using the poisson–Tweedie family. *Environmetrics*, *22*(2), 152–164.

Esnaola, M., Puig, P., Gonzalez, D., Castelo, R., & Gonzalez, J. R. (2013). A flexible count data model to fit the wide diversity of expression profiles arising from extensively replicated RNA-seq experiments. *BMC Bioinformatics*, *14*(1), 254.

Signorelli, M., Spitali, P., & Tsonaka, R. (2020). Poisson-Tweedie mixed-effects model: a flexible approach for the analysis of longitudinal RNA-seq data. *To Appear in Statistical Modelling*. https://arxiv.org/abs/2004.11193