Master Arbeit: GLMM und Threshold Modelle in der Tierzucht

Reto Zihlmann

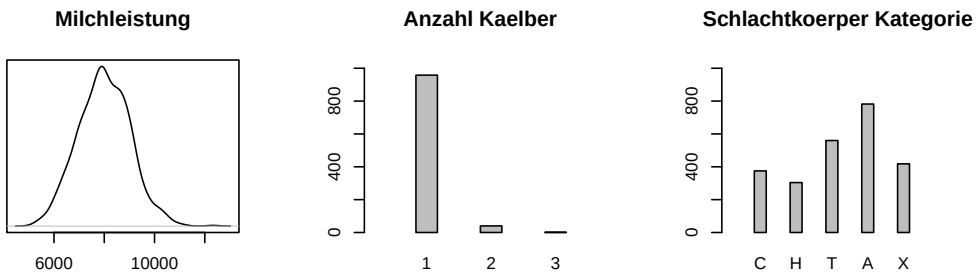
Seminar für Statistik, ETH Zürich

04 Nov 2020

Repetition

Repetition vom letzten Vortrag

- LMMs basieren auf der Annahme
  $(Y \mid \boldsymbol{B} = \boldsymbol{b}) \sim \mathcal{N}_{\mathrm{n}}\left(X\beta + Z\boldsymbol{b}, I\sigma^2\right)$
- Diese Annahme ist nicht erfüllt für viele zuchtrelevanten Merkmale wie
  - Zwillingsgeburten ($\approx$ binär)
  - Kälbersterblichkeit (binär)
  - Schlachtkörper Kategorie (ordinal)



Repetition vom letzten Vortrag

- Für nicht normale Merkmale brauchen wir ein allgemeineres Modell.
- Generalized linear mixed effect models (GLMMs) erlauben Zielvariablen aus einer viel breiteren Klasse von Verteilungen.

$$\begin{aligned} (\boldsymbol{Y} \mid \boldsymbol{B} = \boldsymbol{b}^*) &\sim \mathrm{EDM}_n(\boldsymbol{\mu}, I_n\phi) \\ \boldsymbol{g}(\boldsymbol{\mu}) &= X\boldsymbol{\beta} + Z\boldsymbol{b} \\ \boldsymbol{B} &\sim \mathcal{N}_q\left(\vec{0}, \Sigma\right). \end{aligned} \tag{1}$$

- In R gibt es das Paket `pedigreemm` welches solche Modelle fitten kann.

Pedigreemm

- Das Paket `pedigreemm`
  - berechnet L_A

$$A = L_A L_A^\intercal$$

  - transformiert Z zu Z^*

$$Z^* = Z L_A$$

  - schätzt Parameter mit `lme4`

Pedigreemm

- Probleme mit `pedigreemm`
  - Keine Möglichkeit Varianzkomponenten als gegeben anzunehmen
  - Langsam
  - Erlaubt keine random regression Modelle `(1 + lact | animal)`
  - Kann kein Tier Modell mit einzelnen Beobachtungen pro Tier schätzen
  - Fehler in der Berechnung der Zufallseffekte
- Die meisten Probleme konnten behoben werden und sind implementiert im Packet `cowfit`

Verbesserungen in pedigreemm

Gegebene Varianzkomponenten

- `lme4` hat keine Unterstützung für gegebene Varianzkomponenten
- Grund: Effiziente Parametrisierung um Varianzkomponenten zu finden
  - Likelihood funktion ist profiliert nach σ^2
  - Varianzkomponenten können desshalb nur relativ zu σ^2 definiert werden

Gegebene Varianzkomponenten

- Alle Varianzkomponenten und σ^2 sind bekannt
  - `lme4` ist schnell und exakt
- Nur Varianzkomponenten sind bekannt
  - σ^2 abschätzen
    $\Rightarrow$ schnell aber nicht exakt
  - Nummerische Optimierung
    $\Rightarrow$ langsam aber exakt
- Gar nichts bekannt
  - `lme4` ist schnell und exakt
  - Erlaubt singuläre $\Sigma = \mathrm{Var}(\boldsymbol{B})$)

Gegebene Varianzkomponenten

- Welche Varianzkomponentnen braucht es

```
library(cowfit)
(myvar <- cowfit_var_comp(formula = y ~ (1|herd) + (protein|sire),
                          data = sim_milk))

##        grp        var1    var2 vcov
## 1     herd (Intercept)    <NA>   NA
## 2     sire (Intercept)    <NA>   NA
## 3     sire     protein    <NA>   NA
## 4     sire (Intercept) protein   NA
## 5 Residual        <NA>    <NA>   NA

myvar$vcov <- c(500, 400, 300, -200, 50)
myvar

##        grp        var1    var2 vcov
## 1     herd (Intercept)    <NA>  500
## 2     sire (Intercept)    <NA>  400
## 3     sire     protein    <NA>  300
## 4     sire (Intercept) protein -200
## 5 Residual        <NA>    <NA>   50
```

Gegebene Varianzkomponenten

- LMM fitten mit geschätztem σ^2

```
system.time({
  fit <- cowfit_lmer(formula = y ~ (1|herd) + (protein|sire),
                     data = sim_milk,
                     pedigree = list(sire = pedSires),
                     var_comp = myvar$vcov,
                     cowfit_verbose = FALSE)
})

##    user  system elapsed
##   0.420   0.012   0.433

as.data.frame(VarCorr(fit))

##        grp        var1    var2       vcov      sdcor
## 1     herd (Intercept)    <NA>  642.89733 25.3554202
## 2     sire (Intercept)    <NA>  514.31787 22.6785773
## 3     sire     protein    <NA>  385.73840 19.6402240
## 4     sire (Intercept) protein -257.15893 -0.5773503
## 5 Residual        <NA>    <NA>   64.28973  8.0180879
```

Gegebene Varianzkomponenten

- LMM fitten mit exakten Varianzkomponenten

```
system.time({
  fit <- cowfit_lmer(formula = y ~ (1|herd) + (protein|sire),
                     data = sim_milk,
                     pedigree = list(sire = pedSires),
                     var_comp = myvar$vcov,
                     exact_var_comp = TRUE,
                     cowfit_verbose = FALSE)
})
```

```
##    user  system elapsed
##  11.055   0.028  11.094
```

```
as.data.frame(VarCorr(fit))
```

```
##        grp        var1    var2       vcov      sdcor
## 1     herd (Intercept)    <NA>  499.99951 22.3606688
## 2     sire (Intercept)    <NA>  399.99961 19.9999902
## 3     sire     protein    <NA>  299.99971 17.3204996
## 4     sire (Intercept) protein -199.99980 -0.5773503
## 5 Residual        <NA>    <NA>   69.28546  8.3237889
```

Gegebene Varianzkomponenten

- LMM Zuchtwerte mit gegebenem Varianzkomponenten zu schätzen ist relativ kompliziert in `lme4`
- Aber wir sind ja haptsächlich an GLMMs interessiert.
- Für viele Verteilungen der exponential distribution family gibt es keinen Skalierungsparameter
  - Binomial (p)
  - Poisson (λ)
- Für solche Verteilungen keine zusätzlicher numerische Optimierung notwendig.

Random Regression Modelle

- Im vorherigen Beispiel wurde bereits ein Random Regression Modell verwendet
- Diese Modelle wurden in `pedigreemm` nicht unterstützt.

```
pedigreemm(formula = y ~ (1|herd) + (protein|sire),
           data = sim_milk, pedigree = list(sire = pedSires))

  ## Error in `levels<-`(`*tmp*`, value = as.character(levels)):
                  factor level [2] is duplicated
```

Random Regression Modelle

- Transformation in `pedigreemm`

$$Z^* = ZL_A \qquad \text{with} \qquad A = L_A L_A^\intercal$$

- Verallgemeinerte Tranformation

$$Z^* = ZT_A \qquad \text{with} \qquad T_A = \begin{pmatrix} L_{A_1} \otimes I_{p_1} & & \\ & \ddots & \\ & & L_{A_k} \otimes I_{p_k} \end{pmatrix}$$

Weitere Verbesserungen

- Kann nun Tier Modell mit einzelnen Beobachtungen pro Tier schätzen

```
control$checkControl$check.nobs.vs.nlev <- "ignore"
control$checkControl$check.nobs.vs.nRE <- "ignore"
```

- Fehler in der Berechnung der Zufallseffekte behoben
  - Auch mit den Autoren von `pedigreemm` Kontakt aufgenommen aber noch keine Antwort erhalten.

Bayesian Implementierung

Bayesian Implementierung

- `lme4` schätzt alle parameter mit Maximum Likelihood oder REML.
- In der angewanten Statistik werden GLMMs häufiger mit Bayes Methoden geschätzt.
- Grund:
  - Bessere Schätztung der standard errors von random effects möglich.
  - Flexibler
- Interessante Pakete: `rstanarm`, `brms`, `MCMCglmm`

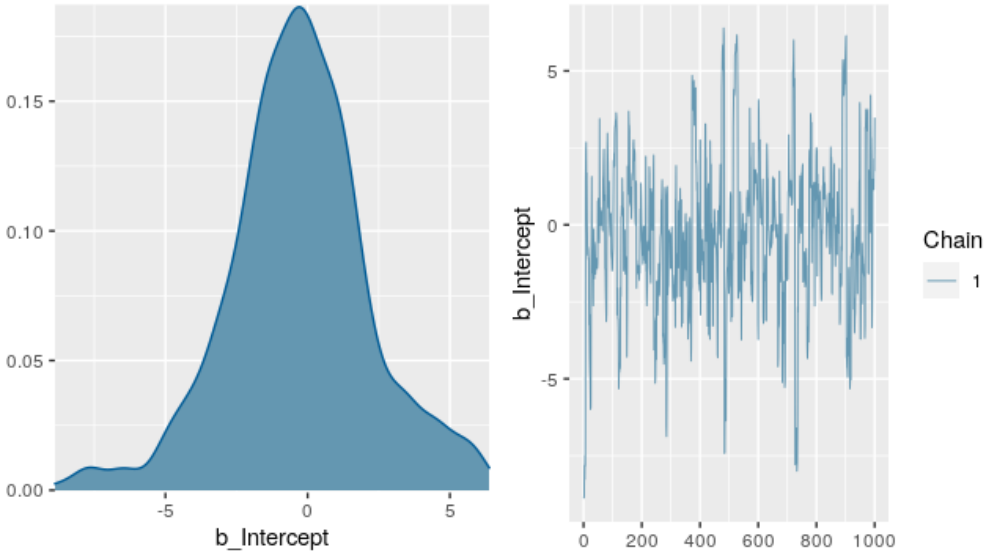
brms

- Erlaubt Korrelationsstrukturen aufgrund von pedigrees.
- Wir implementierten kleine Änderungen
  - Gleicher Syntax wie `pedigreemm`
  - Gegebene Varianzkomponenten
  - Spezifizieren der Korrelation mit L_A anstatt von A

Beispiel Bayesian Estimation

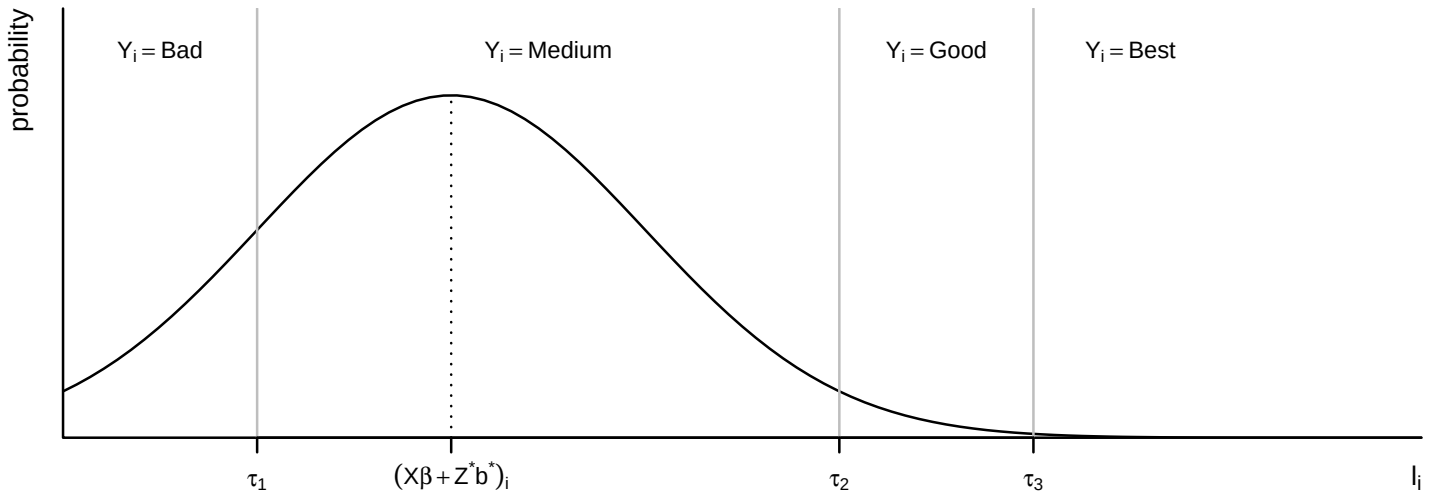
```
library(cowfit)
fit <- cowfit_brm(formula = y ~ (1|sire),
                  data = sim_twin,
                  family = "bernoulli",
                  pedigree = list(sire = pedSires),
                  var_comp = 400,
                  chains = 1)
```

- Resultat ist eine posteriori Verteilung der unbekannten Variablen



Threshold Model

- Bayes Methoden können auch zum Schätzen von sogenanten Threshold Modellen genutzt werden

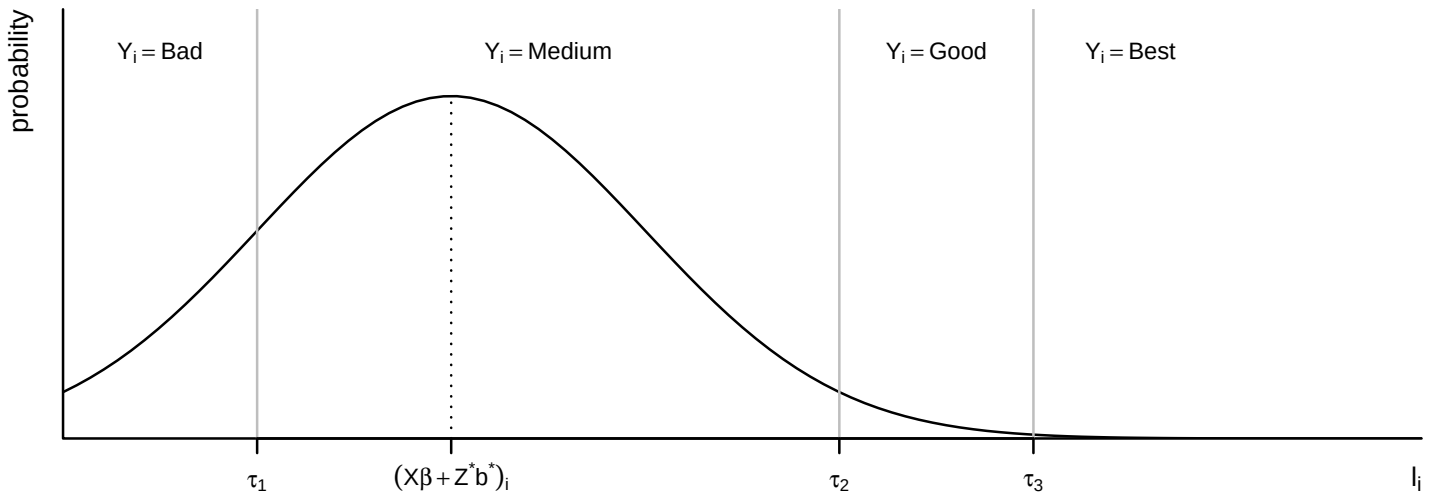


Threshold Model

$$(Y_i \mid l_i, \tau) = \begin{cases} 1 & \text{for} \quad -\infty \leq l_i \leq \tau_1 \\ 2 & \text{for} \quad \tau_1 < l_i \leq \tau_2 \\ \vdots & \vdots \\ k+1 & \text{for} \quad \tau_k < l_i \leq \infty \end{cases} \tag{2}$$

$$(\boldsymbol{l} \mid \boldsymbol{B}^* = \boldsymbol{b}^*) \sim \mathcal{N}_{\mathrm{n}}\left(X\beta + Z^*\boldsymbol{b}^*, I_n\right)$$

$$\boldsymbol{B}^* \sim \mathcal{N}_{\mathrm{q}}\left(\vec{0}, \Sigma^*\right).$$

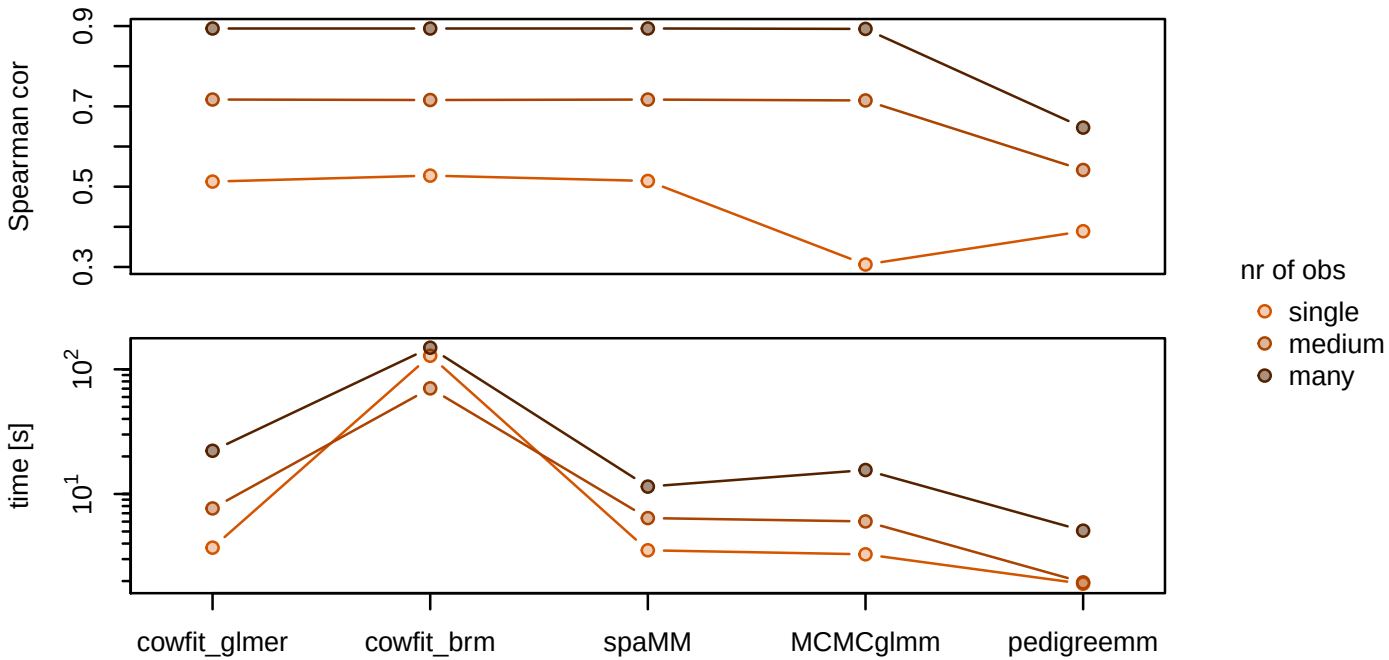


Threshold Model

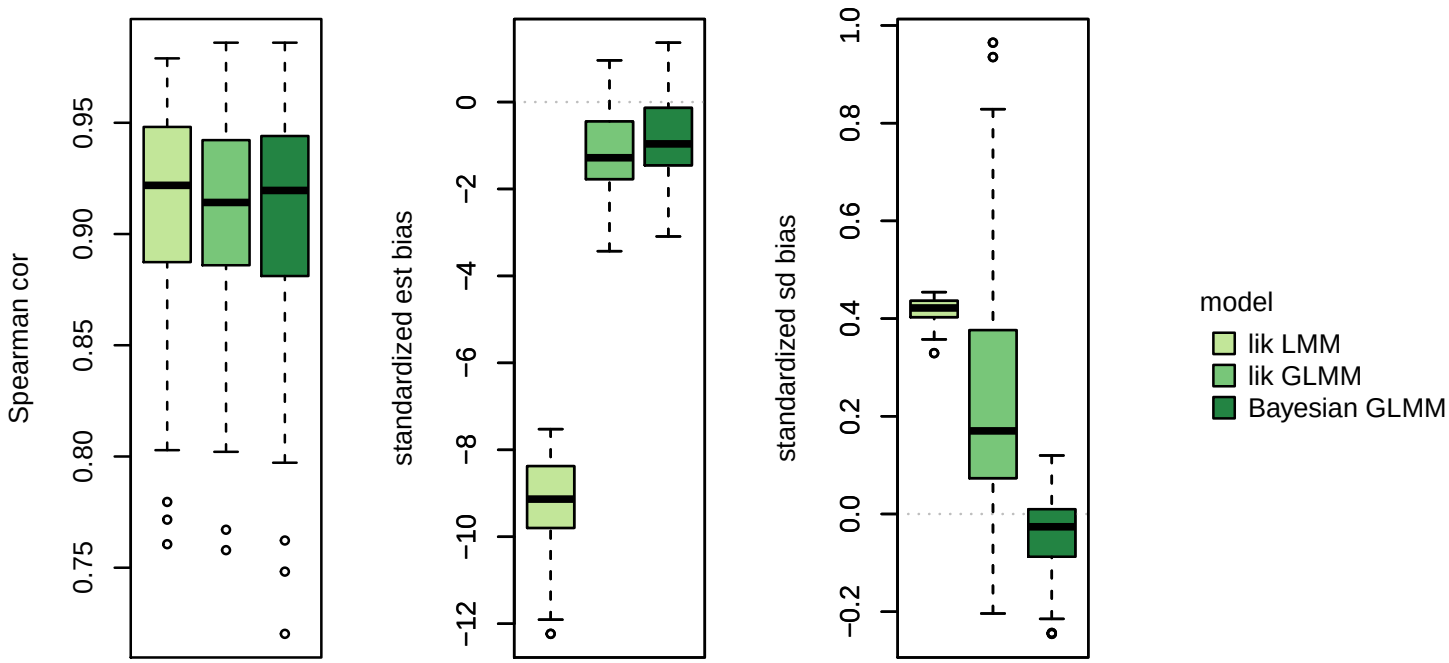
- Threshold Modelle sind sehr weit verbreited in der Tierzucht.
- Wahrscheinlich gibt es weitere gute implementierungen.

Simulation results

Vergleich mit bestehenden Paketen



Vergleich LMM vs GLMM



Diskussion

Diskussion: LMMs vs GLMMs/Threshold Modelle

- GLMM und Threshold modelle verbessern die Schätzugen von Zuchtwerten deutlich
- Das verwenden von LMMs für nicht normale Zielvariablen kann zwar zu einem guten Ranking der Tiere führen, aber die absoluten Zuchtwerte sowie deren Standardabweichung sind nicht vertrauenswürdig.
- Eine weitere Schwierigkeit besteht im Schätzen von Heritabilitäten.
    - Golan et al. (2014) liefert eine Formel um Heritabilitäten von LMMs zu korrigieren.

Diskussion: Software

- Bayes Threshold Modelle scheinen mir sehr weit verbreitet zu sein in der Tierzucht (Gianola und Foulley 1983)
- Threshold Modell praktisch, da es auch gleich Bernoulli Modelle beinhaltet.
- Evt. gibt es eine spezialisierte Tierzuchtsoftware für Bayes Threshold Modelle
- Innerhalb von R
  - Bayes: `MCMCglmm` sehr vielversprechend
  - Likelihood: Verbesserte Versionen von `cowfit`
- Bayes vs. Likelihood
  - Bayes war relativ schnell für grössere Datensätze und erlaubt Threshold Modelle

Paket `cowfit`

- Zu finden auf GitHub (retodomax/cowfit)
- Mit kleinem Tutorial
- Wird wahrscheinlich noch weiter ausgebaut
- Gerne auch Verbesserungen vorschlagen