Mixed Effects Models 4: logistic regression and more

Yury Zablotski

Last updated on Dec 28, 2019 · 16 min read · 0 Comments · R

Previous topics or when do we need it

To keep this post short, I'll skip lots of explanations which were made in the previous posts. Especially **Mixed Effects Model 1** below is recommended to improve a digestion of this post.

- Repeated Measures ANOVA
- Midex Effects Model 1: Random Intercept
- Midex Effects Model 2: Crossed vs. Nested Random Effects
- Midex Effects Model 3: Random Slopes

Why do we need it? What are the benefits?

Repeated measures are common in medical research, where experiments can not effort (even legally) to have to many patients. Such restrictions reduce suffering and save resources. Thus, taking repeated measures is often a sustainable way to do science. Moreover, especially in medical research we often study "**sick or healthy**" questions. Such **binomial outcomes** are classic for a logistic regression and allow to study factors of our interest in terms of probabilities, namely, whether any particular factor (predictor) increases or decreases the probability of becoming sick. However, repeated measures (**random effects**) can skew such probabilities (**fixed effects**), that is why it is important to remove the variance of the repeated measures from a model to "purify" fixed effects. Otherwise, we might make a conclusion which is opposite to the reality. Mixed effects models are **mixed** because we **mix a cocktail of fixed and random effects into one model**. In this way they describe more of the variation in the data and thus, are often more realistic models as compared to the usual models.

Effects soup: fixed, random, nested, crossed

Whether the effect is fixed or random heavily depends on the research question and modeler! (Schabenberger and Pierce 2001)[1]. The latter has created a lot of different, often non-agreeing, definitions of effects. I personally hate the fact that there are so many definitions of the same thing. It's like talking different statistical languages, confusing. There is however a "golden" rule for distinguishing fixed and random effects: random effect suppose to have at least five levels. Thus, variables with <5 categories might be more suitable for a fixed effect. However, making a very complex model by putting to many levels into it, quickly overfits the model (singularity). Whether the

model is singular or not can be checked with `isSingular(model)` function from the `lme4` package. Besides, if `VarCorr(model)` function shows a variances correlations close to 0 or exactly 1 (any perfect correlation is strange and rather not-real), you also might have an overfitted model.

How to compute Generalized Linear Mixed Effects Models in R

Load all needed packages to avoid interruptions.

```
library(tidyverse) # data wrangling and visualization
library(sjPlot)    # to visualizing mixed-effects models
library(effects)   # to visualizing mixed-effects models
library(lme4)      # "golden standard" for mixed-effects modelling in R (no p-values)
library(lmerTest)  # p-values for MEMs based on the Satterthwaite approximation
library(report)    # mainly for an "report" function
library(emmeans)   # post-hoc analysis
library(knitr)     # beautifying tables
library(sjstats)   # ICC - intraclass-correlation coefficient
library(caret)     # ML, model comparison & utility functions
```

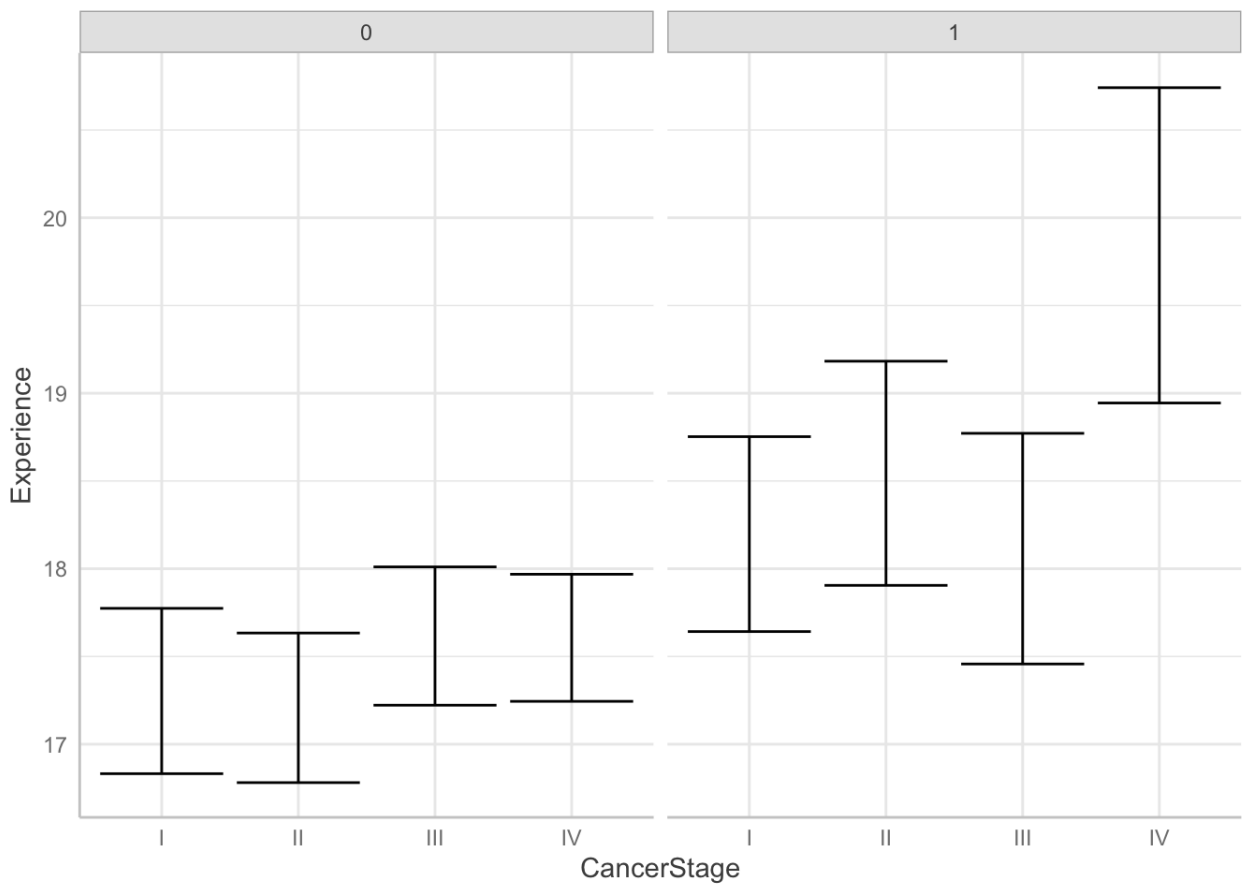
How to conduct mixed-effects logistic regression

Visualize the data

It's always a good idea to have a look at the data It allows to spot outliers or highly correlated predictors, to get some intuition about the distributions or levels of data and to make the expectations of the model more realistic. For instance the visualization below tells us that experience of the doctors is decisive for the remission of the cancer patients in every cancer-stage.

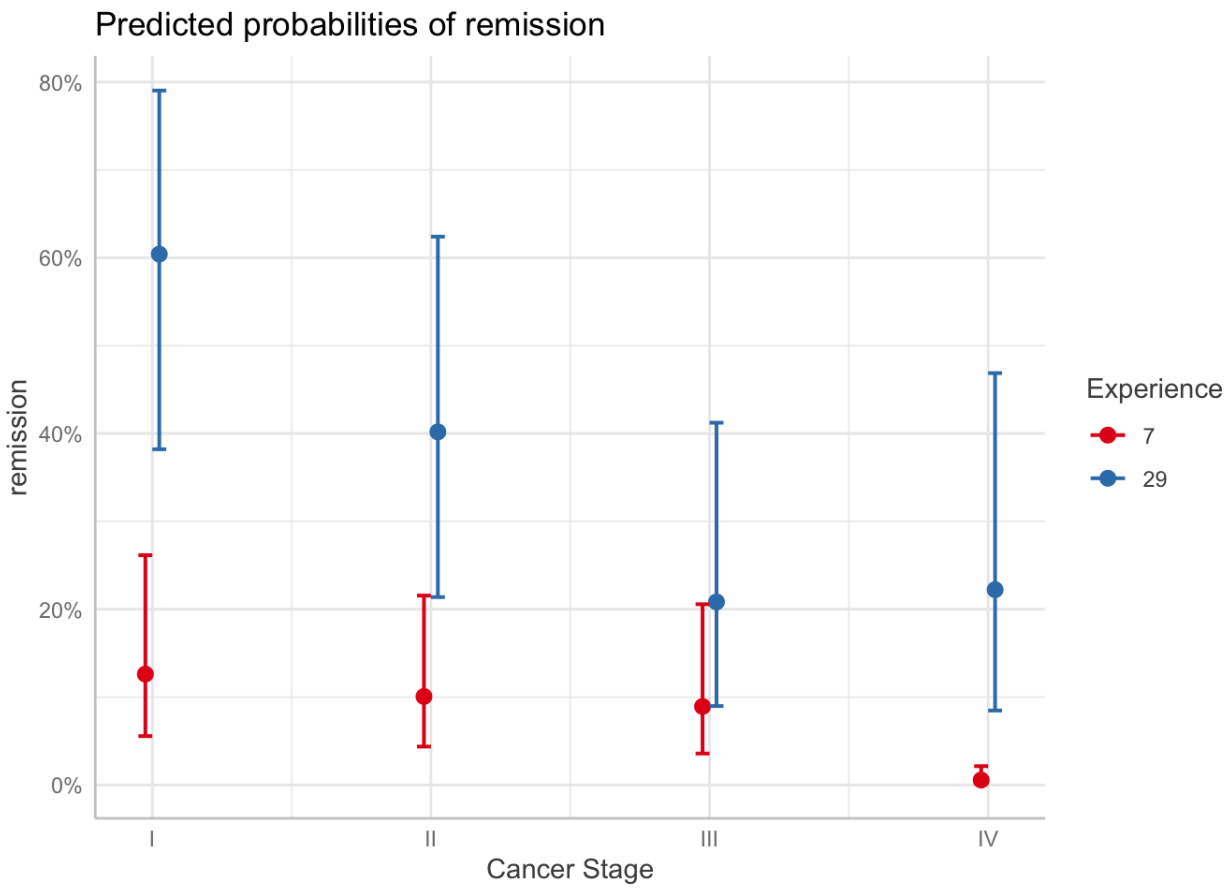
```
# sourse of data: https://stats.idre.ucla.edu/r/dae/mixed-effects-logistic-regression/
hdp <- read.csv("https://stats.idre.ucla.edu/stat/data/hdp.csv")
```

```
set.seed(3)
d <- hdp %>%
  select(DID, remission, Experience, CancerStage) %>%
  group_by(CancerStage) %>%
  sample_n(500)

ggplot(d, aes(CancerStage, Experience, fill = factor(remission)))+
  facet_wrap(~remission)+
  stat_summary(fun.data = mean_cl_boot, geom = "errorbar")+
  theme_ggeffects()
```



Now, let's compute the model and visualize model results (predictions). To avoid a warning of nonconvergence, we specify a different optimizer `control=glmerControl(optimizer="bobyqa")`.

```
m_ri <- glmer(remission ~ CancerStage * Experience  +
(1 | DID), data = d, family = binomial, control = glmerControl(optimizer = "bobyqa"))

plot_model(m_ri, type = "int")+theme_ggeffects()
```



Indeed growing experience leads to the higher probability of cancer remission, so, we better go to an older doctor ;)

But do we really need random effect of a doctor, or could a usual logistic model predict the same? Well, in order to see whether our mixed-effects model is better then a logistic regression without repeated measures, we can compare the models with two of the most used metrics: R^2 **an Akaike's Information Criteria (AIC):**

```
m2 <- glm(remission ~ CancerStage * Experience, data = d, family = binomial)

tab_model(m_ri, m2,  p.style = "both", show.aic = T)
```

	remission			remission		
Predictors	*Odds Ratios*	*CI*	*p*	*Odds Ratios*	*CI*	*p*
(Intercept)	0.07 ***	0.02 – 0.28	**<0.001**	0.23 ***	0.10 – 0.52	**<0.001**
CancerStage [II]	0.93	0.19 – 4.64	0.929	0.46	0.14 – 1.53	0.206
CancerStage [III]	1.05	0.19 – 5.68	0.955	0.78	0.22 – 2.75	0.696
CancerStage [IV]	0.02 ***	0.00 – 0.21	**0.001**	0.04 ***	0.01 – 0.19	**<0.001**
Experience	1.11 **	1.03 – 1.20	**0.006**	1.06 *	1.01 – 1.11	**0.015**
CancerStage [II] * Experience	0.97	0.89 – 1.06	0.565	1.02	0.96 – 1.09	0.505
CancerStage [III] * Experience	0.94	0.86 – 1.03	0.188	0.98	0.91 – 1.05	0.485
CancerStage [IV] * Experience	1.07	0.96 – 1.20	0.218	1.09 *	1.00 – 1.19	**0.045**
Random Effects						
σ^2	3.29					
τ_{00}	2.85 $_{DID}$					
ICC	0.46					
N	385 $_{DID}$					
Observations	2000			2000		
Marginal R^2 / Conditional R^2	0.145 / 0.542			0.066		
AIC	1886.286			2142.378		

- *p<0.05 ** p<0.01 *** p<0.001*

As we can see, the R^2 as a goodness-of-fit of our model to our data is very low in a model without repeated measures. In fact, the model's explanatory power is very weak (Tjur's R2 = 0.066 or <7%). In contrast, the total explanatory power of a mixed-effects model is substantial (conditional R2 = 0.54 or 54%%) and the part related to the fixed effects alone (marginal R2) is 15%, or twice as good as the usual logistic regression. Besides, the AIC of the mixed-effects model decreased by 12%, which is

significantly better as compared to the usual logistic regression. Therefore, the inclusion of random effects into the model often improves the performance of the model significantly. And there seem to be no overfitting (singularity) in our model either (see the output below) ;)

```
isSingular(m_ri)
```

```
## [1] FALSE
```

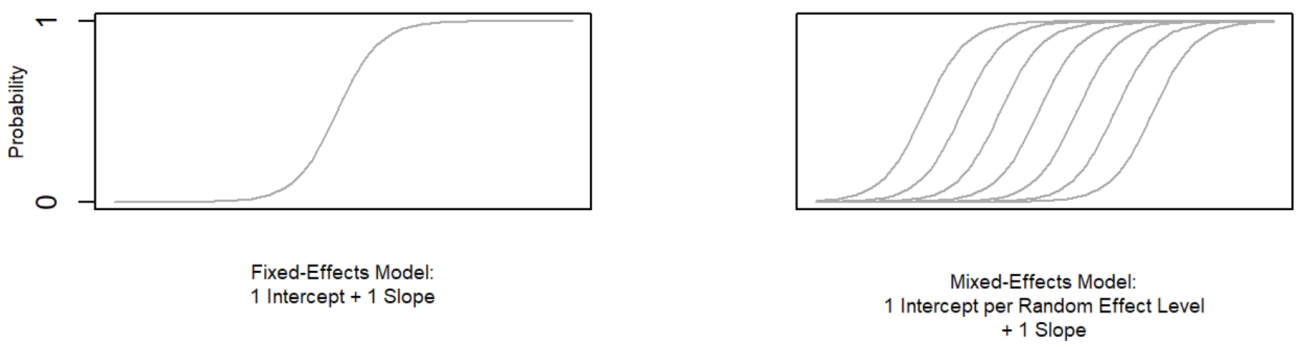
```
VarCorr(m_ri)
```

```
##  Groups Name        Std.Dev.
##  DID    (Intercept) 1.6868
```

How do we describe random effects in a model?

If you think about the simplest regression line possible, you have two numbers, intercept and slope, which can describe a lot of other numbers. A multiple linear regression has multiple intercepts and slopes. So, it's not surprising that **random effects** are also mainly characterized via **random intercept and random slope**. While using both, intercept and slope, in any mixed-effects model makes the most sense, in practice it requires huge amounts of data and is therefore often reduced to using **intercept only** models.

The picture and description below are borrowed from [this amazing giude on mixed-effects models](#) and does summarize possible effect of using random intercept and/or slopes very well:



"The upper left panel merely shows the logistic curve representing the predictions of a fixed-effects logistic regression with a **single intercept and slope**. The upper right panel shows the logistic curves representing the predictions of a of a mixed-effects logistic regression with **random intercepts** for each level of a grouping variable. The lower left panel shows the logistic curves representing the predictions of a mixed-effects logistic regression with **one intercept but random slopes** for each level of a grouping variable. The lower right panel shows the logistic curves representing the predictions of a mixed-effects logistic regression with **random intercepts and random slopes** for each level of a grouping variable."[2]

While usual regression has one intercept for all 385 doctors in our data set, the mixed effect model has **385 intercepts**. Thus, it's not surprising that if fits the data better 😉. So, would the addition of random slopes also improve the model? Let's find out:

```
m_rs <- glmer(remission ~ CancerStage * Experience  +
(CancerStage | DID), data = d, family = binomial, control = glmerControl(optimizer = "bobyqa"))

tab_model(m_ri, m_rs, show.aic = T)
```

		remission			remission	
Predictors	*Odds Ratios*	*CI*	*p*	*Odds Ratios*	*CI*	*p*

(Intercept)	0.07	0.02 – 0.28	**<0.001**	0.08	0.02 – 0.30	**<0.001**
CancerStage [II]	0.93	0.19 – 4.64	0.929	0.59	0.10 – 3.63	0.569
CancerStage [III]	1.05	0.19 – 5.68	0.955	0.79	0.12 – 5.39	0.810
CancerStage [IV]	0.02	0.00 – 0.21	**0.001**	0.01	0.00 – 0.19	**0.002**
Experience	1.11	1.03 – 1.20	**0.006**	1.11	1.03 – 1.19	**0.006**
CancerStage [II] * Experience	0.97	0.89 – 1.06	0.565	0.98	0.89 – 1.08	0.728
CancerStage [III] * Experience	0.94	0.86 – 1.03	0.188	0.95	0.86 – 1.04	0.269
CancerStage [IV] * Experience	1.07	0.96 – 1.20	0.218	1.09	0.96 – 1.24	0.169
Random Effects						
σ^2	3.29			3.29		
τ_{00}	2.85 $_{DID}$			2.11 $_{DID}$		
τ_{11}				0.60 $_{DID.CancerStageII}$		
				1.02 $_{DID.CancerStageIII}$		
				1.13 $_{DID.CancerStageIV}$		
ρ_{01}				1.00		
				0.08		
				0.30		
ICC	0.46			0.53		
N	385 $_{DID}$			385 $_{DID}$		
Observations	2000			2000		
Marginal R^2 / Conditional R^2	0.145 / 0.542			0.158 / 0.601		
AIC	1886.286			1895.713		

```
anova(m_ri, m_rs)
```

```
## Data: d
## Models:
## m_ri: remission ~ CancerStage * Experience + (1 | DID)
## m_rs: remission ~ CancerStage * Experience + (CancerStage | DID)
##      npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m_ri    9 1886.3 1936.7 -934.14   1868.3
## m_rs   18 1895.7 1996.5 -929.86   1859.7 8.5726  9     0.4776
```

Yes, it does improve the model by explaining 6% more of the data (see Conditional R^2 of both models). However, the Likelihood ratio test (done by the `anova()`) shows that such improvement is not significant and we would tend towards a simpler model, namely **intercept only** model.

Testing significance of random and fixed effects

The likelihood ratio test can also help to sort out random effects. You just compare a model with and a model without a particular random effect to each other:

```
m2 <- lmer(Reaction~Days+(1|Subject)+(0+Days|Subject),sleepstudy,REML=FALSE)
m1 <- update(m2,.~Days+(1|Subject))
m0 <- lm(Reaction~Days,sleepstudy)
anova(m2,m1,m0)
```

```
## Data: sleepstudy
## Models:
## m0: Reaction ~ Days
## m1: Reaction ~ Days + (1 | Subject)
## m2: Reaction ~ Days + (1 | Subject) + (0 + Days | Subject)
##    npar    AIC    BIC  logLik deviance   Chisq Df Pr(>Chisq)
## m0    3 1906.3 1915.9 -950.15   1900.3
## m1    4 1802.1 1814.8 -897.04   1794.1 106.214  1  < 2.2e-16 ***
## m2    5 1762.0 1778.0 -876.00   1752.0  42.075  1  8.782e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

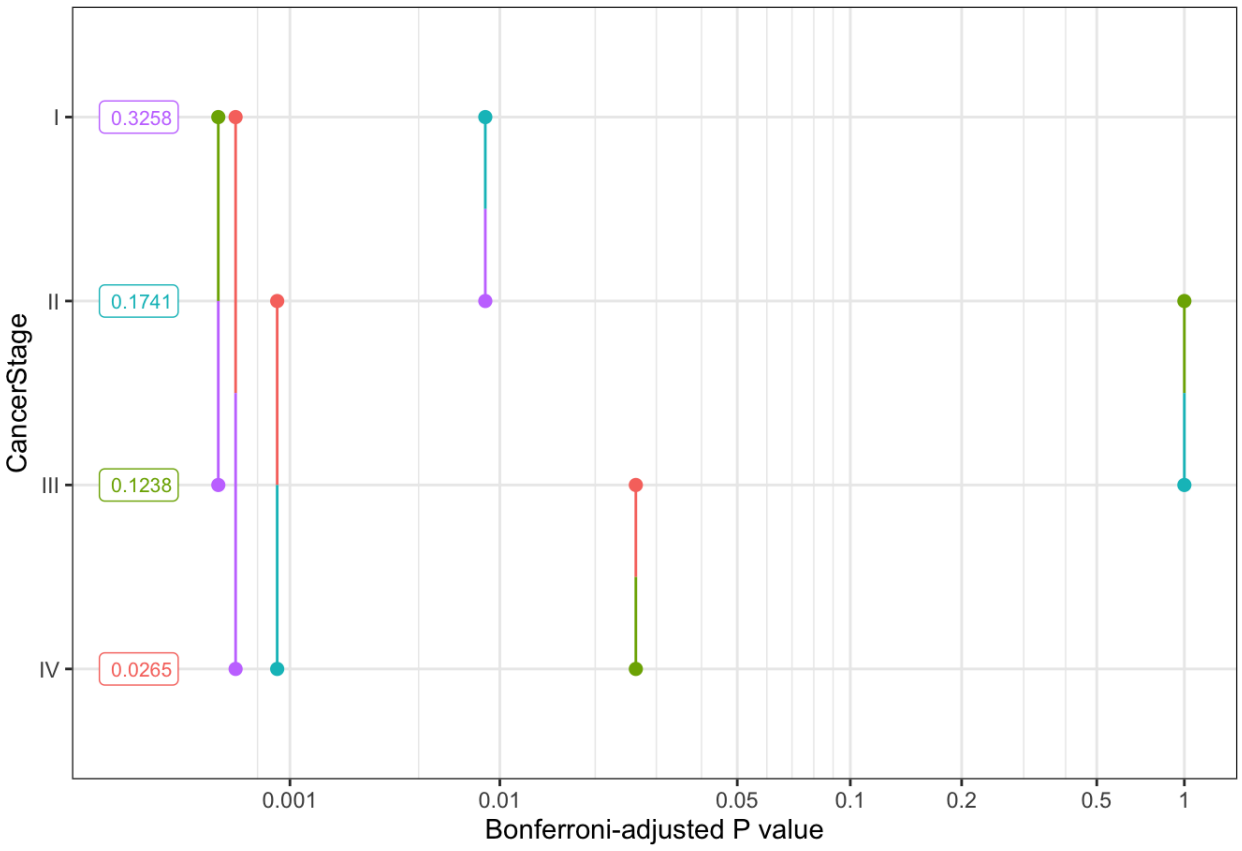
The mixed-effect models in `anova()` **must be listed first**. However, try not to test the significance of random effects, but simply include them into the model if they make sense. “Using stepwise approaches to eliminate non-significant terms in order to squeeze more significance out of the remaining terms is dangerous in any case”, - Ben Bolker. Thus, think whether the variables to be included first.

Post-hoc with `emmeans` package

```
emmeans(m_rs, pairwise ~ CancerStage | Experience, adjust = "bonferroni")$contrasts %>%
  tidy() %>%
  mutate_if(is.numeric, ~round(., 4)) %>%
  kable()
```

level1	level2	Experience	estimate	std.error	df	z.ratio	p.value
I	II	17.7085	0.8292	0.2610	Inf	3.1775	0.0089
I	III	17.7085	1.2294	0.3006	Inf	4.0898	0.0003
I	IV	17.7085	2.8753	0.5026	Inf	5.7208	0.0000
II	III	17.7085	0.4002	0.3494	Inf	1.1454	1.0000
II	IV	17.7085	2.0461	0.5349	Inf	3.8253	0.0008
III	IV	17.7085	1.6458	0.5775	Inf	2.8498	0.0262

```
pwpp(emmeans(m_rs, ~ CancerStage), type = "response", adjust = "bonferroni") + theme_bw()
```



Methods to fit (estimate) GLMMs

Method	Advantages	Disadvantages	Packages
Penalized quasi-likelihood	Flexible, widely implemented	Likelihood inference may be inappropriate; biased for large variance or small means	PROC GLIMMIX (SAS), GLMM (GenStat), glmmPQL (R:MASS), ASREML-R
Laplace approximation	More accurate than PQL	Slower and less flexible than PQL	glmer (R:lme4,lme4a), glmm.admb (R:glmmADMB), INLA, glmmTMB, AD Model Builder, HLM
Gauss-Hermite quadrature	More accurate than Laplace	Slower than Laplace; limited to 2-3 random effects	PROC NLMIXED (SAS), glmer (R:lme4, lme4a), glmmML (R:glmmML), xtlogit (Stata)
Markov chain Monte Carlo	Highly flexible, arbitrary number of random effects; accurate	Slow, technically challenging, Bayesian framework	MCMCglmm (R:MCMCglmm), rstanarm (R), brms (R), MCMCpack (R), WinBUGS/OpenBUGS (R interface: BRugs/R2WinBUGS), JAGS (R interface: rjags/R2jags), AD Model Builder (R interface: R2admb), glmm.admb (post hoc MCMC after Laplace fit) (R:glmmADMB)

formula	meaning
`(1|group)`	random group intercept
`(x|group)` = `(1+x|group)`	random slope of x within group with correlated intercept
`(0+x|group)` = `(-1+x|group)`	random slope of x within group: no variation in intercept
`(1|group) + (0+x|group)`	uncorrelated random intercept and random slope within group
`(1|site/block)` = `(1|site)+(1|site:block)`	intercept varying among sites and among blocks within sites (nested random effects)
`site+(1|site:block)`	*fixed* effect of sites plus random variation in intercept among blocks within sites
`(x|site/block)` = `(x|site)+(x|site:block)` = `(1 + x|site)+(1+x|site:block)`	slope and intercept varying among sites and among blocks within sites
`(x1|site)+(x2|block)`	two different effects, varying at different levels
`x*site+(x|site:block)`	fixed effect variation of slope and intercept varying among sites and random variation of slope and intercept among blocks within sites
`(1|group1)+(1|group2)`	intercept varying among crossed random effects (e.g. site, year)

Or in a little more detail:

equation	formula
$\beta_0 + \beta_1 X_i + e_{si}$	n/a (Not a mixed-effects model)
$(\beta_0 + b_{S,0s}) + \beta_1 X_i + e_{si}$	`~ X + (1|Subject)`
$(\beta_0 + b_{S,0s}) + (\beta_1 + b_{S,1s})X_i + e_{si}$	`~ X + (1 + X|Subject)`
$(\beta_0 + b_{S,0s} + b_{I,0i}) + (\beta_1 + b_{S,1s})X_i + e_{si}$	`~ X + (1 + X|Subject) + (1|Item)`
As above, but S_{0s}, S_{1s} independent	`~ X + (1|Subject) + (0 + X| Subject) + (1|Item)`
$(\beta_0 + b_{S,0s} + b_{I,0i}) + \beta_1 X_i + e_{si}$	`~ X + (1|Subject) + (1|Item)`
$(\beta_0 + b_{I,0i}) + (\beta_1 + b_{S,1s})X_i + e_{si}$	`~ X + (0 + X|Subject) + (1|Item)`

Source of above pictures

When NOT to use Mixed Effects Logistic regression

If random effect are not significant, use the usual logistic regression.

The GLMMa are **linear** and thus, are not able to address non-linear relationships in data. The generalized additive mixed models (GAMMs) can, but this would be outside of the scope of this article.

Despite the fact that mixed-effect logistic regression is so cool, it has some limitations. Particularly, it is quite prone to producing singular fits or other convergence problems due to the limited amount of information provided by each data point (i.e., 0 or 1). Thus, if the model produces nasty warnings, or simplify the model.

- double-check the model specification and the data for mistakes
- center and scale continuous predictor variables (e.g. with `scale()`)
- try to either understand warnings and fix them or
- simplify the model
- try all available optimizers (e.g. BOBYQA, Nelder-Mead, L-BFGS-B from optim, nlminb() etc.).

```
modelfit.all <- lme4::allFit(m_ri)
```

```
## bobyqa : [OK]
## Nelder_Mead : [OK]
## nlminbwrap : [OK]
## nmkbw : [OK]
## optimx.L-BFGS-B : [OK]
## nloptwrap.NLOPT_LN_NELDERMEAD : [OK]
## nloptwrap.NLOPT_LN_BOBYQA : [OK]
```

```
ss <- summary(modelfit.all)
```

Which R packages (functions) fit GLMMs?

Two of the first packages `lmer` and `nlme` would cover the most needs of a frequentist statistician.

- `lme4`, glmer (Laplace approximation and adaptive Gauss-Hermite quadrature [AGHQ]). **Advantages** over `lmer`: (1) faster and thus suitable for a big data, (2) under active development, especially for GLMMs, (3) handles crossed effects easily. **Disadvantages**: (1) no complex variance structures.

- `nlme` - one of the first widely-used mixed-models packages. Nested random effects easily modeled. Crossed random effects difficult. Stable (maintenance-mode). Multiple functions (lme for linear, nlme for nonlinear, gls for no random terms). **Advantages** over `lmer`: (1) well documented (Pinheiro and Bates 2000), (2) utility/plotting methods (ACF and plot.ACF), (3) complex (and custom) variance structures possible: autoregressive, exponential, spatial, spherical, autoregressive heterogeneous! **Disadvantages**: lme is (1) slower than lme4, (2) doesn't handle crossed random effects as easily and (3) can't handle repeated samples at the same location.

- MASS::glmmPQL (penalized quasi-likelihood)
- MCMCglmm (Markov chain Monte Carlo)
- brms, built on Stan; has autocorrelation capabilities (AR, MA, ARMA) via an autocorr argument.
- glmmML (AGHQ)
- glmmAK (AGHQ?)
- glmmADMB (Laplace): flexible, but slower than other R packages.
- glmm (from Jim Lindsey's repeated package: AGHQ)
- gamlss.mx
- amer – Additive mixed models with lme4
- gamm4 – Generalized additive mixed models using mgcv and lme4
- mgcv (gamm function, via glmmPQL in MASS package)
- ASREML-R
- sabreR

Diagnostics

Mixed-effects logistic regression

Booth GD (1994) suggests that VIFs should ideally be <1.5, or <20 for interactions. Higher values would indicate multicollinearity and thus that the model is unstable.

```
car::vif(m_ri)
```

```
##                               GVIF Df GVIF^(1/(2*Df))
## CancerStage             10687.935226  3        4.693343
## Experience                  1.824074  1        1.350583
## CancerStage:Experience 11211.822944  3        4.730925
```

```
probs = 1/(1+exp(-fitted(m_ri)))
probs = binomial()$linkinv(fitted(m_ri))
Hmisc::somers2(probs, as.numeric(d$remission))
```

```
##           C         Dxy           n     Missing
##   0.9198960   0.8397921 2000.0000000   0.0000000
```

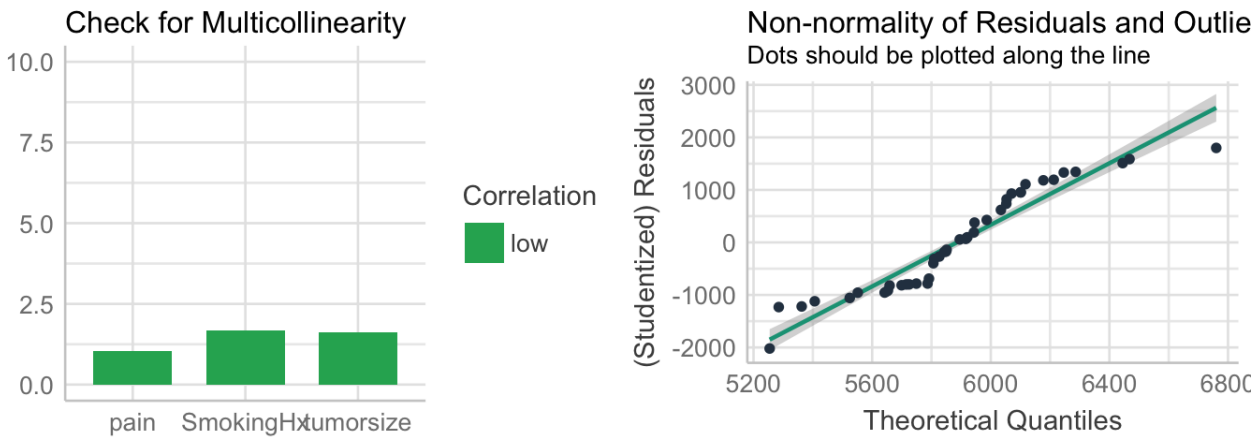
Both the C-value and Somers's Dxy show the quality of predictions. If C-value = 0.5, the predictions are random, if C = 1, the predictions are perfect, the C-values above 0.8 indicate very good predictive capability of the model. Somers' Dxy represents a rank correlation between predicted probabilities and observed responses. Somers' Dxy values range between 0, which indicates complete randomness, and 1, which indicates perfect prediction.

Mixed-effects linear regression

For a linear mixed-effects model (not logistic), we also need to check for the normality of the residuals and for the heterogeneity of variance. Below we use the `nlme` package for creating a mixed-effects model in order to apply its advantages (as compared to `lme4`):

```
library(nlme)
set.seed(1)
m <- lme(WBC ~  pain +  SmokingHx + tumorsize, random = ~1|DID, data = hdp %>% sample_frac(.005))
performance::check_model(m)
```

```
## Could not compute standard errors from random effects for diagnostic plot.
## Homogeneity of variance could not be computed. Cannot extract residual variance from objects of class '
```



The model performs ok! There is no multicollinearity in the model, the residuals are distributed approximately normal and there is no heteroskedasticity in the residuals. Such perfect case is very rare in the real world. The real data are messy and often we have to fix stuff. For instance, in order to fix the heteroscedasticity, we create a new model which uses weights to compensate for heterogeneity of variance and/or the influence of outliers.

```
# does not work for logistic regression
m_weighted <- update(m, weights = varIdent())
# compare models
anova(m, m_weighted)
```

```
##            Model df      AIC      BIC    logLik
## m              1  7 667.3872 678.8503 -326.6936
## m_weighted     2  7 654.6581 666.1212 -320.3291
```

In this case there is only a small improvement, mainly because there was no heteroscedasticity in the model. The improvement is usually higher. Another way to improve the model is to define a new correlation structure: autoregressive (corAR1), autoregressive heterogeneous (corAR1 with weights) which is good for repeated measures which are not equally spaced in time, exponential spatial correlation (corExp), spherical spatial correlation (corSpher) etc.

```
lme_exp   <- update(m, ~., correlation = corExp())
lme_spher <- update(m, ~., correlation = corSpher())
lme_AR1   <- update(m, ~., correlation = corAR1())
lme_ARH1  <- update(m, ~., correlation = corAR1(),
                    weights = varIdent())

anova(m, lme_exp, lme_spher, lme_AR1, lme_ARH1)
```

```
##           Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## m             1  7 667.3872 678.8503 -326.6936
## lme_exp       2  8 666.7460 679.8467 -325.3730 1 vs 2 2.641268  0.1041
## lme_spher     3  8 668.9283 682.0290 -326.4642
## lme_AR1       4  8 674.5228 687.6235 -329.2614
## lme_ARH1      5  8 672.8043 685.9050 -328.4022
```

```
rbind(
  cbind(model = "lme_basic", rsquared(m) ),
  cbind(model = "lme_exp",   rsquared(lme_exp) ),
  cbind(model = "lme_spher", rsquared(lme_spher) ),
  cbind(model = "lme_AR1",   rsquared(lme_AR1) ),
  cbind(model = "lme_ARH1",  rsquared(lme_ARH1) )
) %>%
  dplyr::select(-link, -method) %>%
  kable()
```

model	Response	family	Marginal	Conditional
lme_basic	WBC	gaussian	0.0878074	0.0878074
lme_exp	WBC	gaussian	0.1588566	0.7598333

model	Response	family	Marginal	Conditional
lme_spher	WBC	gaussian	0.0756146	0.0756146
lme_AR1	WBC	gaussian	0.0592117	0.0592117
lme_ARH1	WBC	gaussian	0.1832789	0.1832790

What's next

The next chapter provides some resources on the mixed-effects models to deepen your knowledge.

Further readings and references

- very advanced and very useful collections of information on the topic of GLMM by Ben Bolker: http://bbolker.github.io/mixedmodels-misc/glmmFAQ.html and http://bbolker.github.io/mixedmodels-misc/

- Ben Bolker himself recommends following literature: Littell et al. (2006) and Pinheiro and Bates (2000) are two places to start, although Pinheiro and Bates is probably more useful if you want to use R. Other useful references include Gelman and Hill (2006) (focused on Bayesian methods) and Zuur et al. (2009b). If you are going to use generalized linear mixed models, you should understand generalized linear models (Dobson and Barnett (2008), Faraway (2006), and McCullagh and Nelder (1989) are standard references; the last is the canonical reference, but also the most challenging).

- very good summary on R (and other) packages and their capabilities: http://glmm.wikidot.com/pkg-comparison.

- hacking covariance structures in lme4 Ben Bolker 09 Oct 2019 http://bbolker.github.io/mixedmodels-misc/notes/varmats.html. Amazing, but way to complex for non-mathematicians. It's much easier and more secure (because you have to know what you are doing in lme4) to use `nlme` package for modelling complex covariance structure and the `emmeans` for the post-hoc analysis

- covariance structure examples with `gls` https://stats.idre.ucla.edu/r/seminars/repeated-measures-analysis-with-r/

- CAREFUL! very math-heavy and SAS-examply paper on covariance structure: https://www.researchgate.net/publication/272579574_Heterogeneous_Variance_Covariance_Structures_for_Repeated_Meas

---

1. Schabenberger, Oliver, and Francis J. Pierce. 2001. Contemporary Statistical Models for the Plant and Soil Sciences. Boca Raton, FL: CRC Press.↩

2. Schabenberger, Oliver, and Francis J. Pierce. 2001. Contemporary Statistical Models for the Plant and Soil Sciences. Boca Raton, FL: CRC Press.↩

linear logistic repeated measures regression statistics

Yury Zablotski

Data Scientist at LMU Munich, Faculty of Veterinary Medicine

Passion for applying Biostatistics and Machine Learning to Life Science Data

Related

- Mixed Effects Models 1: Random Intercept
- Mixed Effects Models 3: Random Slopes
- Mixed Effects Models 2: Crossed vs. Nested Random Effects
- Statistical tests vs. linear regression
- Model diagnostics