Exercises from: Random Effects: Generalized Linear Mixed Models

Chapter 10 Agresti (2007)
Saúl Sotomayor Leytón
May 2012

Setup

```
> library(survival)
> library(lme4)
> library(gee)
> options(width=70)
```

Problem 1

First we set up the data. As it was stated in the notes the functions `glmmML` and `lmer` allow data to be grouped or ungrouped, however I'm not sure if the grouping variable was correctly set, therefore the analysis reported were done with the ungrouped data.

```
> tb8.10 <- read.table('supp_data/tb8-10')
> tb8.10long <- reshape(tb8.10, direction = "long", varying = c("believe.heav",
+     "believe.hell"), v.names = "response", timevar = "question")
> tb8.10long <- tb8.10long[rep(1:nrow(tb8.10long), tb8.10long$count),
+     ]
> tb8.10long$case <- 1:1120
> tb8.10long$question2 <- ifelse(tb8.10long$question == 2, "hell",
+     "heav")
> tb8.10long$question2 <- factor(tb8.10long$question2, levels = c("hell",
+     "heav"))
> row.names(tb8.10long) <- 1:nrow(tb8.10long)
> tb8.10long <- within(tb8.10long, {
+     rm(id, count)
+ })
```

Then we fit the model. For this we have two options to use the `glmer` function from the `lme4` package or to use the `glmmML` function with some particular arguments for the method and the number of quadrature points. General forms for both are indicated below,

```
glmmML(formula, family=binomial, data, cluster, weights, method=c("Laplace", "ghq"),
n.points=8, boot=0)
glmer(formula, data, family=family, nAGQ=1, weights, control=list(maxIter=300))
```

Important arguments are highlighted in red. In this sense, for the `glmmML` function we need to specify the method of numerical approximation, in this case we would choose `"ghq"` (which stands for Gauss-Hermite quadrature); also we would like to change the number of quadrature poins form its default 8 to a higher (or lower) value. Regarding the last point with the data for the present problem it seems to be a limit around 125 points; above that the function issues the message

```
Error en glmmML.fit(X, Y, weights, cluster.weights, start.coef, start.sigma,  :
  valor inicial en 'vmmin' no es finito
```

For the glmer function we specify the number of quadrature points with the nAGQ argument (which apparently stantds for, number of Adaptative Gauss-Hermite quadrature points). With the data for this problem the function accepts values as high as 1000. The manual indicates that if it's set to 1 the behaviour is the same as for the lmer function, *i.e.* it uses the Laplace approximation. An important option to avoid *false convergence* is the number of iterations, whose value is 300 by default but can accept values as high as 3000.

Now, regarding the fit of the data, it's interesting that with the Gauss-Hermite approximation the results are very different from those reported in Agresti (2007, p. 371), however with the Laplace approximation the results are very simmilar but a false–convergence message it's reported. Also interesting is that if the Laplace approximation is used with the grouped data the same values as those with the Gauss-Hermitte are reported.

```
> glmer(response ~ question2 + (1 | case), data = tb8.10long, family=binomial)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace
  Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: response ~ question2 + (1 | case)
   Data: tb8.10long
      AIC       BIC    logLik  deviance  df.resid
1063.4459 1080.5886 -528.7229 1057.4459      2237
Random effects:
 Groups Name        Std.Dev.
 case   (Intercept) 63.37
Number of obs: 2240, groups:  case, 1120
Fixed Effects:
  (Intercept)  question2heav
        12.36          13.68
```

Next, the commands used to fit the models with the Gauss-Hermite approximation, with the different quadrature points, are reported. It is important to note that except for the *q* points of 400 and 1000 the function returns a warning message about a false convergence, this despite changing the number of iterations to up to 10000. However it's interesting that for these q points values the AIC and BIC criteria are reported, thing that doesn't happen for q points of 400 or 1000

```
> fit.glmer1.n2 <- glmer(response ~ question2 + (1 | case), data = tb8.10long,
+     family = binomial, nAGQ = 2)
> fit.glmer1.n10 <- glmer(response ~ question2 + (1 | case), data = tb8.10long,
+     family = binomial, nAGQ = 10)
> fit.glmer1.n100 <- glmer(response ~ question2 + (1 | case), data = tb8.10long,
+     family = binomial, nAGQ = 100)
> ## fit.glmer1.n400 <- glmer(response ~ question2 + (1 | case), data = tb8.10long,
>     ## family = binomial, nAGQ = 400)
> ## fit.glmer1.n1000 <- glmer(response ~ question2 + (1 | case),
>     ## data = tb8.10long, family = binomial, nAGQ = 1000)
```

The results are summarized in the following table To sum up with more quadrature points the convergence is more likely.

Table 1: Estimates for the fixed and random effects, for the data in table 8.10, obtained with different quadrature points and compared with those obtained with Laplace approximation and those reported in Agresti (2007) with q=1000

	Agresti	**Laplace**	**q=2**	**q=10**	**q=100**	**q=400**	**q=1000**
$\hat{\beta}$	4.135	4.718	2.882	2.882	2.819	0.702	0.702
$\hat{\beta}$ SE	0.713	0.405	0.282	0.282	2.383	0.111	0.111
$\hat{\sigma}$	10.199	7.251	6.796	6.561	6.532	1.155	1.155
$\hat{\sigma}$ SE	1.792	–	–	–	–	–	–

b) According to the values reported for $q = 1000$ the odds of believing in heaven are $exp(0.702) = 2.02$ times that of believing in hell.

c) To fit the conditional logistic model, we use the `clogiy` function from the `survival` packages (see notes from chapter 8).

```
> clogit(response ~ question2 + strata(case), method = "exact",
+     data = tb8.10long)
```

```
Call:
clogit(response ~ question2 + strata(case), method = "exact",
    data = tb8.10long)


                  coef exp(coef) se(coef)     z        p
question2heav   4.1352   62.5000   0.7127 5.802 6.56e-09

Likelihood ratio test=155.5  on 1 df, p=< 2.2e-16
n= 2240, number of events= 1793
```

According to this, the odds of believing in heaven are $exp(\beta) = exp(4.14) = 62.5$ times that of believing in hell, a result similar to that reported in Agresti and that found with the Laplace approximation, see table 1.

Problem 2

The definition of a succesful outcome under the two scenarios is different, under the first one a success is when a person agrees with abortion, but under the second one a success is when a person opposes with abortion. Now considering what was said about the position of americans about abortion (Agresti 2007, p. 306) we would expect a negative odds ratio under the first scenario but a negative one under the second.

b) The second question should be, “whether the subject supports abortion if a woman wants it because she is unmarried (1=yes, 0=no).

Problem 3

Since we are being asked for the estimated probability we need to calculate $\hat{\pi} = exp(\mu_i + \alpha)/(1 + exp(\mu_i + \alpha)) = exp(\alpha)/(1 + exp(\alpha))$, where $\hat{\alpha} = -3.24$ and $\hat{\sigma} = 0.33$. For this we can define a function `antilogit`

```
> antilogit <- function(x) (exp(x)/(1 + exp(x)))
```

Then we can calculate the probabilities:

i $antilogit(-3.24) = 0.038$

ii $antilogit(-3.24 - 2 \times (0.33)) = 0.02$

iii $antilogit(-3.24 + 2 \times (0.33)) = 0.07$

b) A fixed effects model (*i.e.* $logit(\pi_i) = \beta_i$) has as many parameters as counties, thus is a saturated model, conversly the random effects model has only 2 parameters α and σ (see Agresti 2007, p. 302).

Problem 4

This problem is similar to the example of section 10.2.1 thus we use the same commands as those in the notes (pp. 5-6), but first we construct the data

```
> tb10.9 <- read.table("supp_data/tb10-9", header = TRUE)
> tb10.9 <- within(tb10.9, {
+     prop <- n.made/n.attempts
+ })
```

Then we fit the model

```
> fit.glmer4 <- glmer(prop ~ 1 | game, weights = n.attempts, data = tb10.9,
+     family = binomial, nAGQ = 100) # nAGQ =1000
> fit.lmer4 <- glmer(prop ~ 1 | game, weights = n.attempts, data = tb10.9,
+     family = binomial)
```

Note the similar values obtained with the Gauss-Hermite and with the Laplace approximation, respectively. According to the former, the intercept ($\hat{\alpha}$) estimate is -0.176 with a standard error ($\hat{\sigma}$) of 0.369.

b) The probability for O'Neal of scoring on an average game is $exp(-0.176)/(1+exp(-0.176)) = 0.456$.

c) This question ask for the estimated proportions of success throughout the games, thus we also use the commands in the notes (p. 6)

```
> tb10.9.res <- cbind(tb10.9[, 1:3], Obs = round(tb10.9[, 2]/tb10.9[,
+     3], 3), Fit.1 = round(exp(fitted(fit.glmer4))/(1 + exp(fitted(fit.glmer4))),
+     3), Fit.2 = round(exp(fitted(fit.lmer4))/(1 + exp(fitted(fit.lmer4))),
+     3))
> summary(tb10.9.res[, 4:6])
```

```
      Obs              Fit.1            Fit.2
 Min.   :0.1670   Min.   :0.5940   Min.   :0.5950
 1st Qu.:0.2900   1st Qu.:0.6015   1st Qu.:0.6025
 Median :0.4290   Median :0.6090   Median :0.6090
 Mean   :0.4486   Mean   :0.6114   Mean   :0.6115
 3rd Qu.:0.5855   3rd Qu.:0.6225   3rd Qu.:0.6220
 Max.   :1.0000   Max.   :0.6440   Max.   :0.6430
```

Based on the estimated values of α and σ we can say that the proportion of successful free-throws vary between 0.56 to a maximum of 0.71 throughout the games, with a mean of 0.61.

Problem 5

First we construct the data

```
> tb.ex5 <- data.frame(C = 1:10, N = rep(5, 10), H = c(2, 4, 1,
+     3, 3, 5, 4, 2, 3, 1))
> tb.ex5$obs <- tb.ex5$H/tb.ex5$N
> tb.ex5
```

```
    C N H obs
1   1 5 2 0.4
2   2 5 4 0.8
3   3 5 1 0.2
4   4 5 3 0.6
5   5 5 3 0.6
6   6 5 5 1.0
7   7 5 4 0.8
8   8 5 2 0.4
9   9 5 3 0.6
10 10 5 1 0.2
```

a) The observed proportions of heads corresponds to the model $logit(\pi_i) = \beta_i$.

b) The random effects model for this data would be, $logit(\pi_i) = \mu_i + \alpha$, where μ_i comes from a normal distribution with a mean of 0 and a standard deviation of (σ). To fit this model we use the following commands,

```
> fit.glmer5 <- glmer(obs ~ 1 | C, data = tb.ex5, weights = N,
+     family = binomial, nAGQ = 100) # nAGQ = 1000, options = list(maxIter = 1000)
```

```
Random effects:
 Groups Name        Variance Std.Dev.
 C      (Intercept) 2.6667   1.633
Number of obs: 10, groups: C, 10

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.2412     0.6035     0.4    0.689
```

According to the estimates, the probability of throwing a coin and obtaining a head is $exp(0.24)/(1 + exp(0.24)) = 0.56$. Agresti reports $\hat{\alpha} = 0.259$ and $\hat{\sigma} = 0.557$. Note that the estimate is similar to that obtained with the Laplace approximation but the standard error is similar to that obtained with the Gauss-Hermite approximation.

c) The predicted values are obtained with

```
> (tb.ex5.res <- cbind(tb.ex5[, 1:3], Obs = round(tb.ex5[, 3]/tb.ex5[,
+     2], 2), Fit.1 = round(exp(fitted(fit.glmer5))/(1 + exp(fitted(fit.glmer5))),
+     2)))
```

```
    C N H Obs Fit.1
1   1 5 2 0.4  0.63
2   2 5 4 0.8  0.65
3   3 5 1 0.2  0.61
4   4 5 3 0.6  0.64
5   5 5 3 0.6  0.64
6   6 5 5 1.0  0.66
7   7 5 4 0.8  0.65
8   8 5 2 0.4  0.63
9   9 5 3 0.6  0.64
10 10 5 1 0.2  0.61
```

d) As it's stated in Agresti, pp. 302–303, if we are confident that the logits of probabilities vary according to a normal distribution, then the random effects model borrows from the whole to estimate the probabilities of a particular cluster. Contrary the model where the observed proportion corresponds to the ML estimate only use the data in a particular cluster to estimate the probability; when the sample in the cluster is small the estimates might not be the correct and have larger standard errors. Therefore the estimates obtained with the random effects model are preferred.

e) If we assume that all the toss were done with a fair coin (*i.e.* π_i=0.5) then we see that the difference between that and the estimates obtained with the random effects model are smaller than the observed proportions (see Agresti 2007, p. 303).

```
> cbind(tb.ex5.res, Diff = abs(0.5 - tb.ex5.res$Fit.1), Diff2 = abs(0.5 -
+     tb.ex5.res$Obs))
```

```
    C N H Obs Fit.1 Diff Diff2
1   1 5 2 0.4  0.63 0.13   0.1
2   2 5 4 0.8  0.65 0.15   0.3
3   3 5 1 0.2  0.61 0.11   0.3
4   4 5 3 0.6  0.64 0.14   0.1
5   5 5 3 0.6  0.64 0.14   0.1
6   6 5 5 1.0  0.66 0.16   0.5
7   7 5 4 0.8  0.65 0.15   0.3
8   8 5 2 0.4  0.63 0.13   0.1
9   9 5 3 0.6  0.64 0.14   0.1
10 10 5 1 0.2  0.61 0.11   0.3
```

Problem 6

Although Agresti provides the results of the random intercept model, it's better to fit the data with the functions provided in R since, as it was noted they result in different estimates. So first we construct the data.

```
> tb7.3 <- read.table("supp_data/tb7-3", header = TRUE)
> tb7.3long <- reshape(tb7.3, direction = "long", varying = c("cigarrette",
+     "alcohol", "marijuana"), v.names = "response", timevar = "question",
+     times = c("cigarrette", "alcohol", "marijuana"))
> tb7.3long <- tb7.3long[rep(1:nrow(tb7.3long), tb7.3long$count),
+     ]
> row.names(tb7.3long) <- 1:nrow(tb7.3long)
> tb7.3long <- within(tb7.3long, {
+     rm(id, count)
+ })
> tb7.3long$response <- ifelse(tb7.3long$response == "yes", 1,
+     0)
> tb7.3long$question <- factor(tb7.3long$question, levels = c("cigarrette",
+     "alcohol", "marijuana"))
> tb7.3long$case <- 1:nrow(tb7.3long)
> tb7.3long <- tb7.3long[order(tb7.3long$case), ]
```

Now, note that the model reported in the book is $logit[P(Y_{it} = 1)] = \mu_i + \beta_t$, this is equivalent to a model with no intercept, thus to fit that we must specify -1 in the model formula.

```
> fit.glmer6 <- glmer(response ~ -1 + question + (1|case), data = tb7.3long,
+                     family = binomial, nAGQ = 60)
> fit.lmer6 <- glmer(response ~ -1 + question + (1|case), data = tb7.3long,
+                   family = binomial)
```

Note that according to the results obtained with the Laplace approximation, it seems that the estimates reported in Agresti are not correct, namely β_1 corresponds to the estimate for alcohol and β_2 to cigarrete.

a) In this model, β_t represents the odds of consuming substance t. For example the odds of consuming cigarrete are $exp(1.621) = 5.058$ while the odds of consuming marijuana are $exp(-0.775) = 0.461$.

b) A large value of σ represents greater heterogeneity among the subjects, caused by not including certain explanatory variables that are associated with the response (*e.g.* . gender, age) (Agresti 2007, p. 299).

c) A large value of μ_i imply that the odds of consuming a given substance is higher despite the substance type.

Problem 7

Because the model wasn't fitted in chapter 9, we fit it now,

```
> gee(response ~ -1 + question, id = case, family = binomial, corstr = "exchangeable",
+     data = tb7.3long)
```

```
questioncigarrette    questionalcohol  questionmarijuana
         0.6493063          1.7851115         -0.3154188
```

```
 GEE:  GENERALIZED LINEAR MODELS FOR DEPENDENT DATA
 gee S-function, version 4.13 modified 98/01/27 (1998)

Model:
 Link:                      Logit
 Variance to Mean Relation: Binomial
 Correlation Structure:     Exchangeable

Call:
gee(formula = response ~ -1 + question, id = case, data = tb7.3long,
    family = binomial, corstr = "exchangeable")

Number of observations :  6828

Maximum cluster size   :  1


Coefficients:
questioncigarrette    questionalcohol  questionmarijuana
         0.6493063          1.7851115         -0.3154188

Estimated Scale Parameter:  1.00044
Number of Iterations:  1

Working Correlation[1:4,1:4]
[1] 1


Returned Error Value:
[1] 0
```

Then we can compare the estimates with those obtained in the previous problem; they are summarized in the followin table In section 2.2 of Agresti the author mentions that "when the link function is not linear, like the logit, the population averaged effects of the marginal model (*i.e.* GEE) is tupically smaller in magnitude than the cluster–specific model (*i.e.* GLMM)". The more heterogenic are the observations the greater the difference and as it was seen the estimate of σ in the GLMM is a big one.

Table 2: Estimates for the data in table 7.3, about the consumption of alcohol, cigarrette and marijuana, fitted with the GEE (with exchangeable correlation structure) and GLMM model with Gauss-Hermite approximation (q=60)

Estimate	GEE	GLMM
Cigarrette	0.649	1.485
Alcohol	1.785	3.864
Marijuana	-0.315	-0.687

b) Loglinear models try to find an association between two or more response variables, in this case among the consumption of three substances, for example the odds of consuming cigarrettes for those who have consumed alcohol and those who haven't. Also, just like the GEE these estimates are population averaged. On the other hand, GLMM models estimate the odds of consuming or not a particular substance, it doesn't take into account whether the subject has consumed another substance or not, but more importantly, GLMM are subject specific.

The response that Agresti gives is: "Loglinear model focuses on strength of association between use of one sub- stance and use of another, given whether or not one used remaining substance. The focus is not on the odds of having used one substance compared with the odds of using another".

c) As it's stated in Agresti (2007, p. 302), when $\sigma = 0$, all the probabilities (π_i) are equal. This means that no matter how much change the explanatory variables the response variable is the same, in other words they are independent. Regarding the loglinear models the equivalent to this would be the *mutual independence model* (X, Y, Z) where all pairs of variables are treated as independent, both conditionally and marginally (see Agresti 2007, p. 208).
Agresti states that, "If $\hat{\sigma} = 0$, GLMM has the same fit as loglinear model (A, C, M), since conditional independence of responses given random effect translates to conditional independence marginally also.

Problem 8

First we construct the data, being careful to code the factors in the same order as it was done in problem 9.2

```
> tb7.13 <- read.table("supp_data/tb7-13", header = TRUE)
> tb7.13long <- reshape(tb7.13, varying = c("C", "A", "M"), direction = "long",
+     v.names = "response", times = c("C", "A", "M"), timevar = "question")
> tb7.13long <- tb7.13long[rep(1:nrow(tb7.13long), tb7.13long$Count),
+     ]
> tb7.13long <- within(tb7.13long, {
+     rm(id, Count)
+ })
> tb7.13long$case <- 1:2276
> tb7.13long <- tb7.13long[order(tb7.13long$case), ]
> row.names(tb7.13long) <- 1:nrow(tb7.13long)
> tb7.13long$R <- factor(tb7.13long$R, levels = c("O", "W"))
> tb7.13long$G <- factor(tb7.13long$G, levels = c("M", "F"))
> tb7.13long$question <- factor(tb7.13long$question, levels = c("M",
+     "A", "C"))
```

a)Then we fit the model, first with the Gauss-Hermite approximation and then with the Laplace approximation[1].

```
> fit.glmer8 <- glmer(response ~ question + R + G + (1|case), data = tb7.13long,
+                     family = binomial, nAGQ = 60) #  control = list(maxIter = 500)
> fit.lmer8 <- glmer(response ~ question + R + G + (1|case), data = tb7.13long,
+                   family = binomial) #  control = list(maxIter = 500)
```

The estimates are the following

[1] The former was done with 60 quadrature points and up to 500 iterations, while the latter was done with up to 500 iterations. Also another fit was done with 600 q points, but it yield considerabily different estimates, and also no AIC value.

```
Random effects:
 Groups Name        Variance Std.Dev.
 case   (Intercept) 8.9206   2.9867  
Number of obs: 6828, groups: case, 2276

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.46163    0.27806   -5.26 1.47e-07
questionA    4.55384    0.12289   37.05  < 2e-16
questionC    2.17346    0.09408   23.10  < 2e-16
RW           0.90615    0.27503    3.29 0.000985
GF          -0.12420    0.14819   -0.84 0.401968
```

Because we have set marijuana as a base category, the estimates for the remaining substances (*i.e.* Alcohol and Cigarretes) are interpreted as the log odds of comsuming one of the remaining substances instead of marijuana, exponentiating this results in the odds ratio, for example the odds of consuming alcohol (conditioning on race and gender) instead of marijuana is $exp(4.554) = 95.012$ times that of marijuana. Similarly the odds of a white person to consume one of the substances, conditioning on the gender are $exp(0.906) = 2.474$ times that for other races. Finally the odds for a female (conditioning on the race) to consume a given substance is $exp(-0.124) = 0.883$ times that of males, in other words the odds for females to consume are 12% lower than males. However note that the p-value for the gender effect is non-significant.

b) The estimates just calculated and those of problem 9.2 are summarized in the following table. As it was stated for the previous problem (a) when the link function is non-linear and the observations show heterogeneity (as it's shown in the ρ and $\hat{\sigma}$ values) the conditional model's estimates are greater in magnitude than the marginal ones.

Table 3: Estimates for the data in table 7.13, about substance consumption based on gender and race, calculated with GEE and GLMM

	GEE Estimate (SE)	GLMM Estimate (SE)
(Intercept)	-0.627 (0.137)	-1.462 (0.278)
questionA	2.106 (0.061)	4.554 (0.123)
questionC	0.967 (0.042)	2.173 (0.094)
RW	0.378 (0.135)	0.906 (0.275)
GF	-0.077 (0.074)	-0.124 (0.148)
ρ	0.437	–
$\hat{\sigma}$	–	2.987

Problem 9

Looking at the answers given in Agresti, it seems like the model used in problem 9.6 was correct, however it's interesting that in order to match the values reported the number of quadrature points must be around 600. In previous problems q values as high as 600 give completely different results.

```
> tb9.6long <- read.table("supp_data/tb9-6b", header = TRUE)
> fit.glmer9 <- glmer(outcome ~ seq + treat + (1 | case), data = tb9.6long,
+     family = binomial, nAGQ = 100) # control = list(maxIter = 500)
```

In this model, treatment A (placebo) was set to zero, thus the estimates represent the log odds of relief when taking one of the doses (low or high) compared to taking the placebo; for example the odds of relief when taking the low dose is $exp(1.992) = 7.33$ times that obtained with the placebo. The estimate for σ is 0.41. Agresti (2007, p.371) reports values of $\hat{\beta}_A = 0$, $\hat{\beta}_B = 1.99$ (SE $= 0.35$), $\hat{\beta}_C = 2.51$ (SE $=$ 0.37), with $\sigma = 0$. Note that except for σ the estimates values are the same as those reported in the book.

Problem 10

References

Agresti, Alan (2007). *An Introduction to Categorical Data Analysis.* 2nd ed. Hoboken NJ: Wiley-Interscience. ISBN: 978-0-471-22618-5.