Practical_5_solns

April 4, 2024

0.1 Multilevel Modelling of Longitudinal Data

This exercise uses the longitudinal.dta data file. The models presented in this session are taken from Twisk, J (2006), Applied Multilevel Analysis (pp91-101), if you want further information about these models, you might wish to consult the Twisk textbook (which is an excellent introductory textbook on multilevel modelling).

The key feature of this session is that syntax used to handle longitudinal models in R is essentially the same as for the cross-sectional models in previous session. This session will cover :-

```
1) How longitudinal data (when stored in long format) can be seen as identical to any othe
2) How to estimate explanatory relationships with longitudinal data
3) How to create growth curves (where an outcome measure is considered a function of time)
understanding change over time.
```

As such, this session provides a further chance for practicing the syntax required for random intercept, and random slope, models.

0.1.1 The Dataset, Variables and Research Questions

The file “longitudinal.dta” contains data of 147 patients who, while been treated by a doctor, were asked on four occasions about their lifestyle and their level of health at that time. As this is a Stata data file, it can be imported to R using the “read.dta” command in the “foreign” package. In addition, as in previous sessions, the “lme4” package is needed to estimate multilevel models.

[1]:
```
require(devtools)
install_version("foreign", version = "0.8-76")

library (foreign) ## a library only needs to be opened only once, typically at␣
 ↪the start of a syntax file
library (lme4)
```

```
Loading required package: devtools

Loading required package: usethis

Warning message:
"package ‘usethis’ was built under R version 4.2.3"
Downloading package from url:
https://cran.r-project.org/src/contrib/Archive/foreign/foreign_0.8-76.tar.gz
```

```
Loading required package: Matrix

Warning message:
"package ‘Matrix’ was built under R version 4.2.3"
```

[2]:

```
longdata <- read.dta("longitudinal.dta")
```

As with all analysis, it is important to get a feel for what variables are included in the dataset, their possible values, patterns of missing data etc. The "str" command gives us an overview of the variables in the dataset.

[3]:

```
str(longdata)
```

```
'data.frame':	588 obs. of  5 variables:
 $ id       : num  1 1 1 1 2 2 2 2 3 3 …
 $ health   : num  4.2 3.9 3.9 3.6 4.4 …
 $ lifestyle: num  2.51 2.1 2.16 2.26 2.48 …
 $ time     : num  1 2 3 4 1 2 3 4 1 2 …
 $ time2    : num  1 4 9 16 1 4 9 16 1 4 …
 - attr(*, "datalabel")= chr ""
 - attr(*, "time.stamp")= chr "10 Feb 2020 15:48"
 - attr(*, "formats")= chr [1:5] "%9.0g" "%9.0g" "%9.0g" "%9.0g" …
 - attr(*, "types")= int [1:5] 255 255 255 255 255
 - attr(*, "val.labels")= chr [1:5] "" "" "" "" …
 - attr(*, "var.labels")= chr [1:5] "" "" "" "" …
 - attr(*, "version")= int 12
```

147 patients measured on 4 occasions gives a total of 588 cases (remember, data for longitudinal analysis are stored in long format with one case per time point per person). As there are 588 cases in the dataset, we can see there is no missing data.

In addition, we can see the dataset contains 5 variables:-

1. id – an identification number for each patient. As we have time points clustered within patients this will be our Level 2 identifier)
2. time – a variable taking a value between 1 and 4 showing at which timepoint a particular measurement was taken i.e. the Level 1 identifier.
3. time2 – the value of the variable Time squared (i.e. it equals 1, 4, 9 or 16). This is needed by some software for creating quadratic growth curves.
4. health – the dependent variable. An indicator of the individual's health (for instance an index of several questions aimed at assessing someone's general health). This is a continuous measure with a range of 2.4 to 6.4. Higher scores are associated with more healthy individuals.
5. lifestyle – another continuous index variable (which we will use as an explanatory variable). This provides an indicator of a person's lifestyle and how healthy it is (i.e. do they exercise, smoke etc). Again, higher scores are associated with better lifestyles. The range is 1.57 to 9.05.

The “summary” command provides descriptive statistics for each variable.

```
[4]: summary (longdata)
```

```
       id           health         lifestyle          time          time2      
 Min.   :  1   Min.   :2.400   Min.   :1.570   Min.   :1.00   Min.   : 1.00  
 1st Qu.: 37   1st Qu.:3.800   1st Qu.:2.390   1st Qu.:1.75   1st Qu.: 3.25  
 Median : 74   Median :4.200   Median :3.160   Median :2.50   Median : 6.50  
 Mean   : 74   Mean   :4.299   Mean   :3.488   Mean   :2.50   Mean   : 7.50  
 3rd Qu.:111   3rd Qu.:4.725   3rd Qu.:4.343   3rd Qu.:3.25   3rd Qu.:10.75  
 Max.   :147   Max.   :6.400   Max.   :9.050   Max.   :4.00   Max.   :16.00  
```

Finally, we can use the “head” command to look at the first few rows of the dataset.

```
[5]: head (longdata)
```

A data.frame: 6 × 5

	id	health	lifestyle	time	time2
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1	4.2	2.51	1	1
2	1	3.9	2.10	2	4
3	1	3.9	2.16	3	9
4	1	3.6	2.26	4	16
5	2	4.4	2.48	1	1
6	2	4.2	2.34	2	4

0.1.2 Constructing a Simple Regression Model of the Relationship between Lifestyle and Health

As a precursor to the using multilevel models, a naive single level model of the relationship between lifesyle and health can be constructed using the “lm” command. As a reminder, this model will not account for any clustering in the data and so the extent to which relationships are considered statistically significant is likely to be over estimated. This is particularly a problem with longitudinal where the correlation between cases in the same cluster (in this case the multiple time points clustered within paitents) is typically higher than we find within cross-sectional data.

```
[6]: singlemod <- lm (health~lifestyle, data = longdata)
     summary (singlemod)
```

```

Call:
lm(formula = health ~ lifestyle, data = longdata)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.85450 -0.48721 -0.04747  0.42121  2.06240 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.80806    0.07349  51.819  < 2e-16 ***
lifestyle    0.14083    0.01955   7.203 1.82e-12 ***
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6637 on 586 degrees of freedom
Multiple R-squared:  0.08134,	Adjusted R-squared:  0.07977
F-statistic: 51.89 on 1 and 586 DF,  p-value: 1.816e-12
```

The single level model suggests a positive and highly significant relationship between health and lifestyle. The coefficient is given as 0.141 (standard error = 0.020).

This relationship is shown to be highly significant.

0.1.3 Constructing a Multilevel Model of the Relationship between Lifestyle and Health - Random Intercept Model

Recreating the above model as a multilevel model (observations clustered within paitents) will take account of the likely lack of independence between the different observations provided by each patient.

Since the data are in long format (each row of the dataset represents one observation from one indivdual) the data are a akin to the hierarcial structure we have seen in previous weeks. They can be analysed using the “lmer” command in the “lme4” package.

[7]:

```
rimod <- lmer (health~lifestyle + (1|id), data = longdata, REML=FALSE)
```

Before analysising the results of the model in detail, it is important to establish if the multilevel model is a better fit to the data than the simple, single-level, model presented above.

Since both model have been estimated using identical cases, the “anova” command can be used to establish if the more complex model offers an improvment in terms of model fit - conducting a log-likelihood ratio test as in previous sessions.

[8]:

```
anova (rimod, singlemod)
```

A anova: 2 × 8

	npar <dbl>	AIC <dbl>	BIC <dbl>	logLik <dbl>	deviance <dbl>	Chisq <dbl>	Df <dbl>	Pr(>C <dbl
singlemod	3	1190.6649	1203.7951	-592.3325	1184.6649	NA	NA	NA
rimod	4	820.0064	837.5133	-406.0032	812.0064	372.6585	1	4.9357

The chi-squared value displayed in the above test is 372.66, while the multilevel model involves the estimation of one additional parameter compared to the single-level model. Assuming 95% confidence, the critical value for the chi-square test is 3.841. The output therefore indicates that the multilevel model is a much better fit to the data than the single level model was.

Having established that the multilevel model is most appropriate for the data, consideration can be given to the substantive findings of the model.

[9]:

```
summary (rimod)
```

```
Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ lifestyle + (1 | id)
```

```
   Data: longdata

     AIC      BIC   logLik deviance df.resid
   820.0    837.5   -406.0    812.0      584

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2458 -0.6220 -0.0538  0.5889  2.6716

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.3210   0.5666
 Residual             0.1278   0.3575
Number of obs: 588, groups:  id, 147

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.05366    0.09425  43.010
lifestyle    0.07042    0.02308   3.051

Correlation of Fixed Effects:
          (Intr)
lifestyle -0.854
```

The following points are worthy of note :-

1) The inclusion of a random intercept has reduced the strength of the relationship between lifestyle and health. Now given as 0.070 rather than the 0.141 in Figure 2. However, this relationship is still significant (standard error=0.023) and of the same direction as the previous model.

2) The relative importance of the patient level in explaining health can eb estimated. This is done by calculating a VPC as done in previous cross-sectional models (0.321/(0.321+0.128)).

3) The value of interclass correlations in longitudinal studies are generally higher than in cross-sectional models. This reflects how repeated measures taken from a single individual are generally highly correlated. This example fits that pattern with 71% of the variance being attributed to the patient level.

0.1.4 Constructing a Multilevel Model of the Relationship between Lifestyle and Health - Random Slope Model

To complete the process of studying how lifestyle influences health, consideration might be given to if the strength of the relationship varies between individuals. This can be tested by allowing the coefficient associated with “Lifestyle” to vary at the individual (ID) level (a random slope model).

Again, the syntax follows the pattern seen in previous weeks when considering cross-sectional data, i.e. the variable “lifestyle” needs to be added to the random part of the model.

[10]:

```
rsmod <- lmer (health~lifestyle + (1+lifestyle|id), data = longdata, REML=FALSE)
summary (rsmod)
```

```
Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ lifestyle + (1 + lifestyle | id)
   Data: longdata

     AIC      BIC   logLik deviance df.resid
   822.2    848.5   -405.1    810.2      582

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.3103 -0.6160 -0.0396  0.5679  2.4774

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 id       (Intercept) 0.51105  0.7149
          lifestyle   0.01011  0.1006   -0.64
 Residual             0.12349  0.3514
Number of obs: 588, groups:  id, 147

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.04619    0.10251  39.470
lifestyle    0.07226    0.02536   2.849

Correlation of Fixed Effects:
          (Intr)
lifestyle -0.876
```

As above, the "Anova" command can be used to compare the fit of the two models; in this case the random slope model compared to the random intercept model.

[11]:

```
anova (rsmod, rimod)
```

A anova: 2 × 8

	npar <dbl>	AIC <dbl>	BIC <dbl>	logLik <dbl>	deviance <dbl>	Chisq <dbl>	Df <dbl>	Pr(>Chisq) <dbl>
rimod	4	820.0064	837.5133	-406.0032	812.0064	NA	NA	NA
rsmod	6	822.2372	848.4975	-405.1186	810.2372	1.769244	2	0.4128702

This test suggests that the random slope model is not a significantly better fit than the random intercept model. That is to say, that allowing the impact of lifestyle on health to vary across patients does not improve our model; it appears that the impact of lifestyle on health is therefore consistent across patients.

Had the random slope model been found to be a better fit for the data then substantive interpretation would have followed teh same steps as for random slope models with cross-sectional data, considering :-

1) The extent to which the strenght of the relationship between lifestyle and health varies be-

tween indivduals (i.e. the amount of variance in the random slope).

2) Whether the introduction of a random slope had changed the effect of the fixed effect (compare the fixed effect for lifestyle in this model with the one in the previous random intercept model)
3) If there is a relationship between an indivdual's random intercept and their random slope (i.e. is the relationship between lifestyle and health stronger for those who have, on average, higher levels of health)

0.1.5 Construction Growth Curves of Health Over Time

Recall that the creation of a simple growth curve analysis (which aim to show the pattern of development in a single measure over time) requires the construction of a multilevel model which includes "time" as the only explanatory variable.

Including a constant as the first explanatory variable (associated with B0) and allowing this to vary between patients gives a random intercept model, essentially capturing the (highly highly) possibility that paitents report different levels of health at the opening time point.

Begin by treating time as a fixed effect.

[12]:

```
gc1 <- lmer (health~time + (1|id), data = longdata, REML=FALSE)
summary (gc1)
```

```
Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ time + (1 | id)
   Data: longdata

     AIC      BIC   logLik deviance df.resid
   785.8    803.3   -388.9    777.8      584

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.4754 -0.6063 -0.0302  0.5689  2.9619

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.3540   0.5950
 Residual             0.1151   0.3392
Number of obs: 588, groups:  id, 147

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.50952    0.05985  75.342
time        -0.08408    0.01251  -6.719

Correlation of Fixed Effects:
     (Intr)
time -0.523
```

The coefficent associated with time, provides an indication of the mean (across paitents) linear impact of time on self-reported heath. In this case, the coefficent is -0.084 (with a stanadrad error equal to 0.01251, suggesting a very significant relationship). This suggest that (self-reported) health falls as time passes.

Allowing the effect of "time" to vary between paitents (through a random intercept model) would help to establish if the way health changes over time is the same for all paitents. The syntax for this model is shown below, and follows the established format for a random intercept model introduced in previous sessions, i.e. the variable "time" now appears in both the fixed, and random, parts of the equation.

[13]:

```
gc2 <- lmer (health~time + (1+time|id), data = longdata, REML=FALSE)
summary (gc2)
```

```
Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ time + (1 + time | id)
   Data: longdata

     AIC      BIC   logLik deviance df.resid
   766.4    792.6   -377.2    754.4      582

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.9326 -0.5297 -0.0457  0.4888  3.2405

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 id       (Intercept) 0.39147  0.6257
          time        0.01591  0.1261   -0.33
 Residual             0.08857  0.2976
Number of obs: 588, groups:  id, 147

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.50952    0.05972  75.507
time        -0.08408    0.01512  -5.559

Correlation of Fixed Effects:
     (Intr)
time -0.530
```

Notable in the above output is the warning message ""Model failed to converge with max|grad| = 0.00281508". This message indicates that the chosen extimation method has not derived a reliable,robust, estimate for all the parameters included in the model. This might be due to a shortage of degres of freedom, or because the model is simply very poorly defined, notably having random efects for all the fixd effects included.

This difficulty neds to be addressed before the model can be interpreted. A range of options exist.

1. It might be that setting the REML argument to TRUE (i.e. "REML=TRUE") might help address the issue. Changing this setting means the model will be estimated by Restrictied

Maximum Likelihood rater than Maximum Likelihood. REML tries to “factor out” the influence of the fixed effects X before moving into finding the optimal random-effect variance structure.

2. An alternative specification of the model could be tried, which removes the correlation between the random intercept and rndom slope for time. This is achieve through the following syntax,

gc2x <- lmer (health~time + (1|id)+(0+time|id), data = longdata, REML=FALSE)

In this case, both the intercept (1) and the effect of “time” are allowd to vary between paitents, yet because they are now included in seperate random effects statments “(1|id)+(0+time|id)” the covariance of teh two is no longer estimated, simplifying the model. In this case, the model does now seem to estimate correctly (see below). However, you will note that no correlation between the random effect of the intercept and the random effect for tim is now provided. Such covariances are often of substntive interest in growth curve models since they can indicate how change over time is related to the initial starting level (for instance, do patients with initially high levels of self-reported health experience more, or less, decline over time?).

3. A final alternative is to switch the estimation method used. By default, the “lme4” package currently uses the BOBYQA optimiser (https://en.wikipedia.org/wiki/BOBYQA) but instead you could opt to use the Nelder-Mead optimisation routine (https://en.wikipedia.org/wiki/Nelder%E2%80%93Mead_method). The syntax for this is shown eblow as model “gc3x”.

Different estimation approximations use different forms of constraints which can means that one approach can provide estimats for one type of models when another can’t. It is useful to understand the strenghts and weaknesses of different approaches if you are going to use them regularly (the R documentation is usful for references). As a minimum, if you wish to compare model fit between different models it is imporant to make sure you use the same estimation routines in each case, i.e. in this case to compare model to compare models “gc1” to “gc3x” you would want to rerun model “gc1” using the Nelder Mead optimizer.

[14]:
```
gc2x <- lmer (health~time + (1|id)+(0+time|id), data = longdata, REML=FALSE)
summary (gc2x)
```

```
Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ time + (1 | id) + (0 + time | id)
   Data: longdata

     AIC      BIC   logLik deviance df.resid
   769.4    791.3   -379.7    759.4      583

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.8425 -0.5522 -0.0403  0.5397  3.1794

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept) 0.31888  0.5647
 id.1     time        0.01058  0.1029
 Residual             0.09505  0.3083
Number of obs: 588, groups:  id, 147

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.50952    0.05603  80.488
time        -0.08408    0.01419  -5.926

Correlation of Fixed Effects:
     (Intr)
time -0.407
```

[15]:

```
gc3x <- lmer (health~time + (1+time|id), data = longdata, REML=FALSE,control = lmerControl(optimizer ="Nelder_Mead"))
summary (gc3x)
```

```
Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ time + (1 + time | id)
   Data: longdata
Control: lmerControl(optimizer = "Nelder_Mead")

     AIC      BIC   logLik deviance df.resid
   766.4    792.6   -377.2    754.4      582

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.9326 -0.5297 -0.0457  0.4888  3.2405

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 id       (Intercept) 0.39147  0.6257
          time        0.01591  0.1261   -0.33
 Residual             0.08857  0.2976
Number of obs: 588, groups:  id, 147

Fixed effects:
            Estimate Std. Error t value
(Intercept)  4.50952    0.05972  75.507
time        -0.08408    0.01512  -5.559

Correlation of Fixed Effects:
     (Intr)
time -0.530
```

How might you interpret the above results?

Consider,

```
    1) What the fixed effect of the intercept implies.
    2) What the fixed effect of time implies.
    3) The existance of random variation around the intercept and effect of time
    4) The correlation of those random effects, and what they mean in substantive terms.
```

[16]:

```
gc4x <- lmer (health~time+time2 + (1+time+time2|id), data = longdata,␣
 ↪REML=FALSE,control = lmerControl(optimizer ="Nelder_Mead"))
summary (gc4x)
```

```
boundary (singular) fit: see help('isSingular')


Linear mixed model fit by maximum likelihood  ['lmerMod']
Formula: health ~ time + time2 + (1 + time + time2 | id)
   Data: longdata
Control: lmerControl(optimizer = "Nelder_Mead")

     AIC      BIC   logLik deviance df.resid
   768.7    812.5   -374.3    748.7      578

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.9799 -0.5259 -0.0279  0.5042  3.4378

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 id       (Intercept) 0.501104 0.70789
          time        0.102622 0.32035  -0.54
          time2       0.001466 0.03828   0.52 -1.00
 Residual             0.085323 0.29210
Number of obs: 588, groups:  id, 147

Fixed effects:
             Estimate Std. Error t value
(Intercept)  4.533333   0.088923  50.981
time        -0.107891   0.066647  -1.619
time2        0.004762   0.012453   0.382

Correlation of Fixed Effects:
      (Intr) time
time  -0.802
time2  0.742 -0.975
optimizer (Nelder_Mead) convergence code: 0 (OK)
boundary (singular) fit: see help('isSingular')
```

Once again R displays an error message, in this case “boundary (singular) fit: see ?isSingular”, despite the command already employing “optimizer =”Nelder_Mead” ” Complex mixed-effect models (i.e., those with a large number of variance-covariance parameters) frequently result in singular fits. This is typically shown through random-effect variance estimates of (nearly) zero, or estimates of correlations that are (almost) exactly -1 or 1.

While singular models are statistically well defined (it is theoretically sensible for the true maximum likelihood estimate to correspond to a singular fit), there are real concerns that (1) singular fits correspond to overfitted models that may have poor power; (2) chances of numerical problems and mis-convergence are higher for singular models (e.g. it may be computationally difficult to compute profile confidence intervals for such models); (3) standard inferential procedures such as Wald statistics and likelihood ratio tests may be inappropriate.

There is not yet consensus about how to deal with singularity, or more generally to choose which random-effects specification (from a range of choices of varying complexity) to use. Some proposals include:

1) avoid fitting overly complex models in the first place, i.e. design experiments/restrict models a priori such that the variance-covariance matrices can be estimated precisely enough to avoid singularity (Matuschek et al 2017)
2) use some form of model selection to choose a model that balances predictive accuracy and overfitting/type I error (Bates et al 2015, Matuschek et al 2017)
3) "keep it maximal", i.e. fit the most complex model consistent with the experimental design, removing only terms required to allow a non-singular fit (Barr et al. 2013), or removing further terms based on p-values or AIC
4) use a partially Bayesian method that produces maximum a posteriori (MAP) estimates using regularizing priors to force the estimated random-effects variance-covariance matrices away from singularity (Chung et al 2013, blme package)
5) use a fully Bayesian method that both regularizes the model via informative priors and gives estimates and credible intervals for all parameters that average over the uncertainty in the random effects parameters (Gelman and Hill 2006, McElreath 2015; MCMCglmm, rstanarm and brms packages)

In short, the model involving time, the quadratic of time, allowing the impact of both of these measures to vary across patients, and allowing covariance between these variations, is too complex to estimate with our data through the "lmer" command.

As indicated above, one remedy might be to estimate our model using Bayesian/MCMC approaches - we will consider these methods later in the course.

References

Douglas Bates, Reinhold Kliegl, Shravan Vasishth, and Harald Baayen. Parsimonious Mixed Models. arXiv:1506.04967 [stat], June 2015. arXiv: 1506.04967.

Yeojin Chung, Sophia Rabe-Hesketh, Vincent Dorie, Andrew Gelman, and Jingchen Liu. A nondegenerate penalized likelihood estimator for variance parameters in multilevel models. Psychometrika, pages 1–25, 2013.

Andrew Gelman and Jennifer Hill. Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge University Press, Cambridge, England, 2006.

Hannes Matuschek, Reinhold Kliegl, Shravan Vasishth, Harald Baayen, and Douglas Bates. Balancing type I error and power in linear mixed models. Journal of Memory and Language, 94:305–315, 2017.

Richard McElreath. Statistical Rethinking: A Bayesian Course with Examples in R and Stan. Chapman and Hall/CRC, Boca Raton, December 2015

[ ]:

```
```

[ ]:

```
```