Paper AS03

Collinearity in Mixed Models

John Hendrickx, Danone Nutricia Utrecht, The Netherlands

ABSTRACT

Strong correlations among the independent variables in a model can lead to large standard errors for the estimates. There are well documented methods for evaluating the impact of collinearity in linear regression models but these methods are geared to models with continuous variables only. As an alternative, the R package “perturb” can be used to evaluate the impact of collinearity by adding random noise (perturbations) to selected variables. This method can deal with transformations, interaction effects, categorical variables. The perturb package has now been enhanced to work with the “`nlme`” and “`lme4`” packages for mixed models in R. This makes it suitable for collinearity in mixed models with a random slope variable or sparse categories in a levels variable.

INTRODUCTION

Collinearity occurs when two or more independent variables in a model are strongly correlated. This need not be a problem given sufficient data but it can result in large standard errors for the estimates. This makes sense intuitively, an attempt is made to estimate the separate effects of two variables that are nearly the same. This cannot be done with a high degree of certainty unless there is a good deal of information (i.e. a large dataset). Large standard errors can mean unstable results, where a small modification to the data would lead to substantially different inferences.

There are two main methods for evaluating collinearity in linear regression models. Most software packages can calculate VIF (Variance InFlation) statistics that indicate how much larger the standard error of an estimate would be due to collinearity. Conditioning indices are a second option and have the advantage that they can detect collinearity among a set of variables. Dorman et al. [2] give a more complete overview of methods for detecting and for dealing with collinearity.

These methods are geared to multiple linear regression with continuous independent variables. They are less suitable if a model includes transformed variables or interaction effects. These derived variables are treated in the model as regular independent variables. This does not do justice to the underlying problem, that a small change to the untransformed variable, or to the variables comprising the interaction effect, could affect the model results.

This is the rationale for the `perturb` package in R. This program evaluates collinearity by adding small random values (perturbations) to specified variables in a model, re-estimating the model, and repeating this process e.g. 100 times. The model estimates can then be evaluated to determine how much these random perturbations have affected the results. The perturbances are applied to the untransformed variables or to the variables included in an interaction effect, making it possible to evaluate how small changes are propagated to derived model terms. This principle can be extended to categorical variables where a probability of misclassification is specified. The `perturb` package can also be used in generalized linear models such as logistic regression.

The latest version of perturb can also be used in mixed models estimated by the `nlme` or `lme4` packages. This can take into account that a variable for a random slope is correlated with other variables in the mixed model. Another application that comes to mind is evaluating the impact of changes to a levels variable such as the site where data was collected. The site variable often contains small categories, applying random reclassifications can indicate how robust the results actually are.

ADDING PERTURBANCES

The perturb package evaluates collinearity by adding small, normally distributed random values to selected variables in a model. The SD of the normal distribution can be specified to control the magnitude of these perturbations. After adding the perturbances to the original variable, any derived variables are re-calculated. For example, the perturbance would be added to the linear term of a variable, then the squared term would be calculated from the modified variable. The perturbances thus indirectly affect model terms such as transformations or interaction effects that are derived from those variables. The model is re-estimated for a specified number of iterations (default 100), adding perturbations and deriving model terms at each iteration. Model coefficients are collected at each iteration, then summarised using descriptive statistics in order to evaluate the stability of the coefficients under the changes applied to the data.

This approach has the advantage that it provides concrete information on how strongly model results could be affected by changes of the magnitude selected. The basic principles apply to any regression-like model, including generalised linear models and mixed models. On the other hand, it can be difficult to choose an appropriate magnitude for the perturbations. For some variables, a certain amount of measurement error can be reasonable to expect. In other cases, increasing magnitudes for the perturbations could be used to show that model outcomes are stable unless strong changes to the data were to occur.

PERTUBANCES FOR CATEGORICAL VARIABLES

The perturb package extends this principle to categorical variables. Although it may not be self-evident, sparse categories or categories nearing 100% can be can be seen as a collinearity problem. Note that a categorical variable can be modelled by creating a set of dummy variables, one for each category except the reference category. Each dummy variable has the value of 1 for the category it belongs to and 0 otherwise. A dummy variable for a sparse category will consist almost entirely of zeros, which means a high negative correlation with the intercept. A dummy variable for a category near 100% will consist almost entirely of ones and will have a high positive correlation with the intercept.

Perturb assess the stability of coefficients for categorical variables by randomly reclassifying a certain percentage of cases at each iteration. This is the equivalent to adding small random numbers to continuous variables. However, the reclassification procedure in perturb is not as straightforward as this sounds. The reason is that as such, reclassifying will cause small categories to increase in relative size while large categories will shrink. Table 1 provides an example of reclassification probabilities with a 90% chance for each case to remain in the same category and a 5% chance tor the case to be moved to one of the two other categories.

Table 1: Reclassification probabilities

	Reclassified		
Original	**a**	**b**	**c**
a	0.90	0.05	0.05
b	0.05	0.90	0.05
c	0.05	0.05	0.90

Table 2 shows the expected frequency distribution of a reclassified variable with frequencies 10, 70 and 20 for categories a, b, and c respectively. On the whole, reclassifying this variable would the largest category to shrink and the other two categories to increase.

Table 2: Example of a reclassified variable

	Reclassified			
Original	**a**	**b**	**c**	**total**
a	9	0.5	0.5	10
b	3.5	63	3.5	70
c	1	1	18	20
total	13.5	64.5	22	100

There is a method to derive reclassification probabilities that will not affect the relative size of categories. Table 3 contains reclassification probabilities that will produce the same expected frequencies for the original and reclassified categories. Table 4 shows the expected frequencies using these reclassification probabilities. The totals of the original and reclassified variable are now identical.

Table 3: Reclassification probabilities for the categories in Table 2

	Reclassified		
Original	**a**	**b**	**c**
a	0.854	0.096	0.050
b	0.014	0.966	0.020
c	0.025	0.070	0.905

Table 4: Expected frequencies using the reclassification probabilities from Table 3

	Reclassified			
Original	**a**	**b**	**c**	**total**
a	8.54	0.96	0.50	10
b	0.96	67.64	1.40	70
c	0.50	1.40	18.10	20
total	10	70	20	100

The reclassification probabilities are derived using loglinear models for square tables[1]. These models are based on patterns of association, e.g. Table 3 uses a model that allows a certain odds of staying in the same category versus moving to a different category (the diagonal of the table). For movers, the odds of moving are the same for all categories. But for a given frequency distribution, the expected frequencies of the reclassified variable are the same as for the original variable. The perturb program supports many loglinear models for square tables. Some of these models are suitable for ordered categories, for example to make small changes in category number more likely than large changes.

This reclassification method ensures that the expected frequency distribution of the reclassified variable is as close as possible to the original variable. This corresponds with the method used for continuous variables, where the random perturbations have a mean of 0. It would also be undesirable to have large categories shrink and small categories increase. That would increase the likelihood of stable results whereas the interest is whether results might become less stable.

PERTURBANCES IN MIXED MODELS

A mixed model expands on the linear regression model by allowing random intercepts, random slopes, a covariance structure among repeated measurements. A random intercept, possibly together with a random slope, can take the intraclass correlation into account within certain levels in which the population can be classified. For example, measurements within research sites could be relatively similar due to certain procedures or instruments. The resulting intraclass correlation can result in inefficient estimators (the standard errors are too large). The correlation among repeated measures can also lead to inefficient estimators.

Mixed models are more flexible but also more complex than linear regression. Still, the basic principles of a perturbation analysis can be extended to mixed models in a straightforward fashion. Random perturbations can be added to a (continuous) variable that has a random slope in the model. A levels variable such as site can be randomly reclassified just as any categorical variable. An interesting aspect of mixed models is that fixed effects and random effects are interdependent. Adding perturbances to one can have implications for the other.

Any problems in implementing perturb* for mixed models in R were of a technical nature. There are two main packages for mixed models in R. The `lme` program in the `nlme` package can estimate both random effects and covariance patterns for repeated measures. The `lmer` program in the `lme4` package is newer and faster but can only estimate random effects. Both `lme` and `lmer` will create an output object that contains the data used, the model formula, coefficients for fixed effects. One technical challenge was that the `coef` function in R returns the fixed effects only for the models, but both random and fixed effects were needed. A second challenge was that both `lme` and `lmer` report coefficients for random effects or repeated measures in the standard output but that the output object does not contain these values. Random coefficients needed to be reconstructed by the perturb program before they could be recorded. Fortunately, R provides an elegant solution to this in the use of S3 methods.

EXAMPLES

EXAMPLE 1: ADDING PERTURBANCES TO A RANDOM SLOPE VARIABLE

This first example is taken from chapter 8 of “Applied Longitudinal Analysis” [3]. This is an interesting example because

- The data are publicly available
- The model contains a random slope variable that is a transformation of another variable in the model
- The `lmer` output reports “Model is nearly unidentifiable: very large eigenvalue - Rescale variables?”. (Interestingly, PROC MIXED in SAS does not report any issues).

If the model is nearly unidentifiable, then how might the results be affected if there were minor changes to the data?

* The current version of perturb at CRAN has not been updated yet to work with mixed models. An updated version is being prepared for submission to CRAN.

The data* from chapter 8 are from the "ACTG Study 193A". "Applied Longitudinal Analysis" reports this was a randomised, double-blind study of 1309 AIDS patients where subjects with dual combinations of HIV-1 inhibitors were compared to subjects with triple combinations. The dependent variable in the dataset is `logcd4`, calculated as the log of the CD4 counts plus 1. The CD4 counts were assessed at baseline and between 1 and 9 times during the 40 weeks follow-up. The timing of assessments relative to baseline is measured in weeks.

On page 226 of [3], lowess smoothed curves of logcd4 indicate a peak at 16 weeks. Based on this, the example in that book fits a piecewise linear spline model with a knot at week 16.

```
m1<-lmer(logcd4 ~ week+week_16+trt:week+trt:week_16 + (1+week+week_16 | id),data=cd4)
```

Variable `week_16` is calculated as the maximum of 0 or `week - 16`. Variable `trt` is 1 for subjects with triple therapy (`group = 4` in the data) and 0 otherwise. Variable `id` is the subject identifier.

Table 5: Fixed effects of model for ACTG study 193A

	Estimate	Std. Error	t value
(Intercept)	2.941	0.026	114.808
week	-0.007	0.002	-3.696
week_16	-0.012	0.003	-3.793
week:trt	0.027	0.004	6.980
week_16:trt	-0.028	0.006	-4.475

Table 5 shows the fixed effects of this model. The fixed effects estimates and their S.E.'s match those on page 227 of [3] to 3 decimal places (t-values match with the Z-values to 2 decimal places). The negative effect of `week` indicates that CD4 counts decreased during the first 16 weeks for the reference group (dual therapy, `trt`=0). The positive value of `week + week:trt` indicates an increase of CD4 counts in the first 16 weeks in the triple therapy group. After week 16, CD4 counts decline somewhat more quickly for subjects with dual therapy and decline at almost the same rate for subjects with triple therapy.

Table 6: Random effects of model for ACTG study 193A

Group	Variable	With:	Variance/ Covariance	S.D./ Correlation
id	(Intercept)		0.586	0.765
id	week		0.001	0.030
id	week_16		0.001	0.035
id	(Intercept)	week	0.007	0.312
id	(Intercept)	week_16	-0.012	-0.458
id	week	week_16	-0.001	-0.858
Residual			0.306	0.553

Table 6 contains the random effects from the model. The model has a random intercept at the subject level and random slopes for both `week` and `week_16`, with an unconstrained covariance structure. The variances/covariances correspond with the values on [3: 228]. The correlations show a strong negative value of -0.858 between the random slopes for `week` and `week_16`.

To test the stability of the results, the model was re-estimated 100 times, adding a random number from normal distribution with mean of 0 and an SD of 0.5 to variable `week` at each iteration. An SD of 0.5 will result in 95.4% of the perturbations being between -1 and 1 week. The variable `week_16` was derived from week at each iteration prior to re-estimating the model. At each iteration, the random and fixed effects of the model were stored, then presented at the end of the procedure using summary statistics.

* The ACTG study 193A data are available at https://content.sph.harvard.edu/fitzmaur/ala2e/.

Table 7: Summary statistics of model coefficients after 100 perturbations (SD = 0.5)

	mean	s.d.	min	max
(Intercept)	2.941	0.001	2.940	2.944
week	-0.007	0.000	-0.008	-0.007
week_16	-0.012	0.000	-0.013	-0.011
week:trt	0.027	0.000	0.026	0.028
week_16:trt	-0.028	0.000	-0.029	-0.027
id|(Intercept)	0.760	0.014	0.714	0.768
id|week	0.029	0.003	0.019	0.031
id|week_16	0.035	0.002	0.024	0.037
id|(Intercept)|week	0.380	0.196	0.291	1.000
id|(Intercept)|week_16	-0.500	0.106	-0.838	-0.419
id|week|week_16	-0.853	0.020	-0.872	-0.787
Residual	0.556	0.008	0.552	0.582

Table 7 shows the results of the perturbation analysis. The SD of the coefficients over the iterations shows which coefficients were most affected by the perturbations. This is the case for the correlations of the intercept with variables `week` and `week_16`. The correlations between the intercept and `week` had an SD of 0.196 over the iterations and ranged from 0.291 to 1.000. The correlation between the intercept and `week_16` also has a moderately high SD of 0.106 and ranges from -0.838 to -0.419. The fixed effects, on which the substantive conclusions are based, remain quite stable.

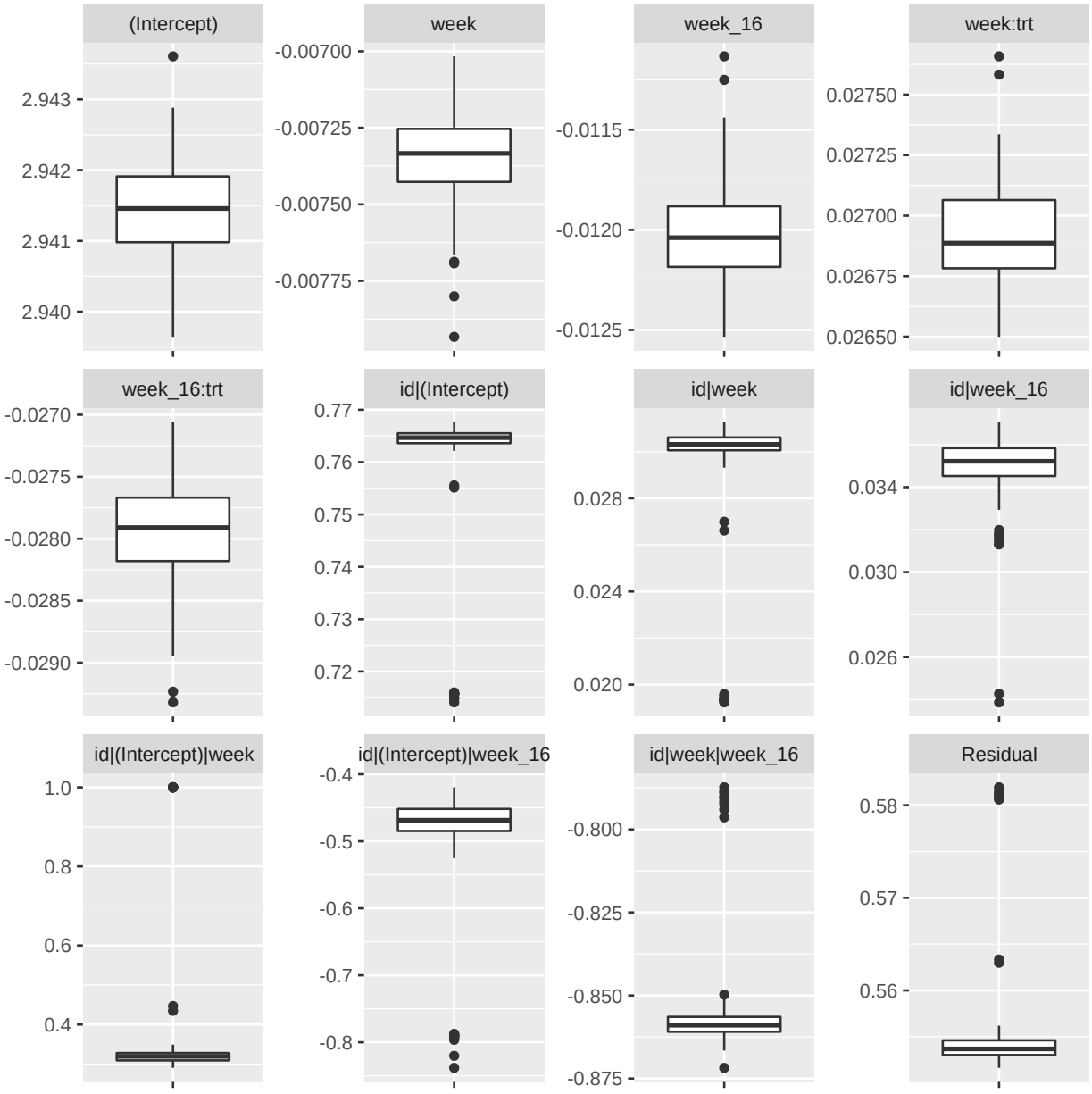
The `perturb` program captures warning messages from `lmer`. Of the 100 perturbed models, 11 had convergence problems. Of these, 9 models produced a warning “unable to evaluate scaled gradient; Model failed to converge: degenerate Hessian with 1 negative eigenvalues”, 2 iterations had 2 negative eigenvalues. The remaining 89 models reported that the model was nearly unidentifiable due to a very large eigenvalue, as had the original, unperturbed model.

The descriptive statistics of coefficients over the iterations can also be represented by a set of box-plots. In these boxplots, the lower and upper hinges correspond with the first and third quartiles. The upper whisker extends from the hinge to the highest value under 1.5 * IQR (Inter-Quartile Range), the lower whisker extends to the lowest value under 1.5 * IQR. Values outside the upper and lower whiskers can be considered outliers.

The boxplot for the correlation between the random intercept and the random slope for `week` is quite interesting. There is an outlier for the value 1, followed by a gap down to under 0.5. An examination of the perturbed parameters shows 9 values of exactly 1. If these values are excluded, the maximum correlation between the intercept and week is 0.45. These 9 models with a perfect correlation all had convergence problems, 2 models with convergence problems had more moderate correlations between the random intercept and random slope for week.

The conclusion must be that this model is indeed unstable, in the sense that changes to the data could lead to convergence problems. What is very interesting is that the fixed effects remain stable, despite the fact that random effects show strong shifts or perfect correlations among random effects.

Figure 1: Box plots of coefficients over 100 perturbations



EXAMPLE 2: ADDING PERTURBANCES TO A LEVELS VARIABLE

One of the uses of mixed models is to take homogeneity into account within strata of the research sample. In the context of clinical research, this might be appropriate for research sites where the trials take place. Mixed models can take the intraclass correlation within sites into account, resulting in more efficient estimators.

However, the number of subjects per site can cause problems. If a site is small, the column in the model matrix for that site will consist mainly of zeros. That column will almost be a linear transformation of the intercept, which consists of ones. A similar issue can exist for large sites. Extremely large or extremely small sites can also be seen as a collinearity problem.

To illustrate this, one of the analyses from Soininen et al. [9] will be used in a perturbation analysis. Soininen et al. report the results of the LipiDiDiet trial, a 24-month randomised, controlled, double-blind, parallel-group, multicentre trial that analysed the effects of Fortaysyn Connect (Souvenaid)*, a medical food, on subjects with prodromal Alzheimer's disease. There were 11 sites in the LipiDiDiet trial but certain sites were so small that they were pooled with other sites in the mixed model analyses. All mixed models in the LipiDiDiet trial included a random intercept for the pooled site variable.

* Fortasyn Connect is a product of Danone Nutricia.

Table 8: Pooled sites in the LipiDiDiet study

Pooled Site	Count	Percent
01	99	35.0%
02	45	15.9%
03	38	13.4%
04	35	12.4%
08	32	11.3%
10	18	6.4%
11	16	5.7%

The first step was to replicate the model used in [9] for a sensitivity analysis of CDR-SB (Clinical Dementia Rating Sum of Boxes). This was a mixed model for repeated measures with change from baseline as outcome. Planned visit (baseline, 12 months, 24 months) was included as a categorical variable, baseline score for CDR-SB and baseline MMSE were included as covariates. A compound symmetry covariance structure was used for the repeated measures and a random intercept was included for pooled sites.

Table 9: Fixed effects for CDR-SB (LipiDiDiet Study)

	Value	Std.Error	DF	t-value	p-value
(Intercept)	3.664	1.286	216	2.849	0.005
Baseline	-0.118	0.083	216	-1.420	0.157
Treatment(Active)	-0.604	0.209	216	-2.887	0.004
Visit	-0.689	0.148	147	-4.655	0.000
Baseline MMSE	-0.079	0.045	216	-1.763	0.079
Treatment(Active):Visit	0.352	0.214	147	1.646	0.102

Table 9 shows the fixed effects of this model using `lme` in R from the `nlme` package. The estimates and S.E.'s correspond with values found using SAS to at least 3 decimal places. Although `lme` uses a different method for degrees of freedom, the p-values are the same to 3 decimal places as found in SAS, with the exception of the p-value for the intercept. Table 10 shows the random effects. The standard `lme` output reports the SD of for the covariance structure whereas SAS reports the variance. The `lme` results are consistent with those in SAS.

Table 10: Random effects for CDR-SB (LipiDiDiet Study)

	StdDev
Intercept(poolsite)	0.369
CS	0.927
Residual	0.968

Table 11 shows the reclassification probabilities to be used in the perturbation analysis. Overall, there is a 5% probability for a subject to be reclassified. However smaller sites have a higher probability of subjects staying and larger sites have a lower probability. This will prevent the expected size of small sites from growing and large sites from shrinking systematically during the iterations of the perturbance analysis.

Table 11: Reclassification probabilities for pooled sites

Original	Recoded						
	01	**02**	**03**	**04**	**08**	**10**	**11**
01	97.50%	0.55%	0.47%	0.49%	0.39%	0.31%	0.30%
02	1.31%	95.70%	0.71%	0.74%	0.60%	0.48%	0.45%
03	1.53%	0.98%	94.83%	0.87%	0.70%	0.56%	0.53%
04	1.47%	0.94%	0.80%	95.08%	0.67%	0.54%	0.51%
08	1.80%	1.16%	0.98%	1.02%	93.75%	0.66%	0.63%
10	2.21%	1.42%	1.20%	1.25%	1.01%	92.15%	0.77%

Original	Recoded						
	01	**02**	**03**	**04**	**08**	**10**	**11**
11	2.32%	1.48%	1.26%	1.32%	1.06%	0.85%	91.72%

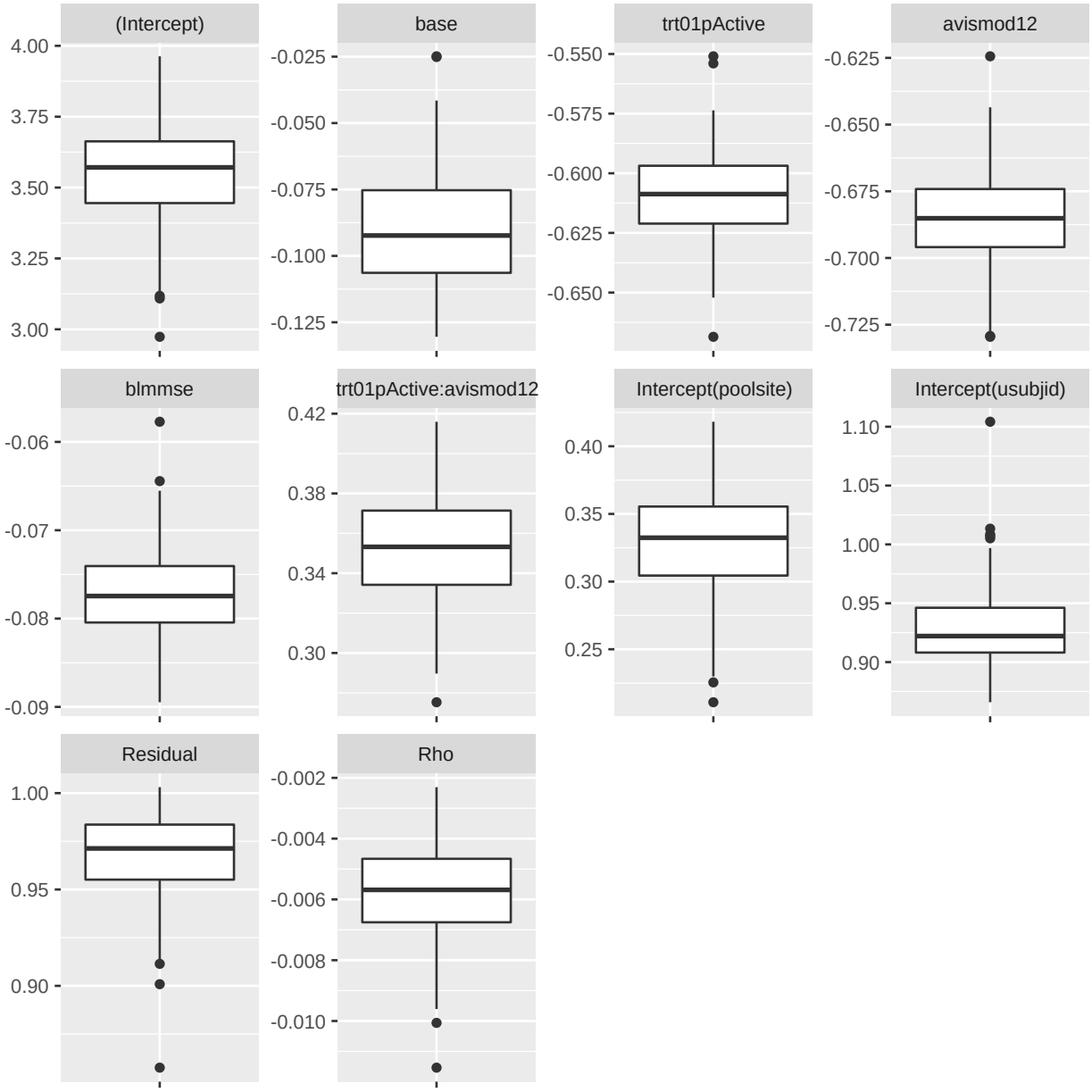
Table 12 contains the results of the perturbation analysis. The model was re-estimated 100 times, with a 5% probability of a subject being reclassified to a different site at each iteration. The SD values show that the intercept is affected most strongly and ranges from 3.0 to 4.0. The SD of the intercept at pooled site level is also affected but to a much smaller degree, as is the CS parameter. The fixed effects are least affected in this perturbation analysis.

Table 12: Summary statistics of the coefficients after 100 random perturbations with a 5% probability of reclassification

	mean	**s.d.**	**min**	**max**
(Intercept)	3.544	0.186	2.974	3.963
Baseline	-0.089	0.024	-0.130	-0.025
Treatment(Active)	-0.608	0.020	-0.669	-0.551
Visit	-0.686	0.019	-0.730	-0.624
Baseline MMSE	-0.077	0.006	-0.089	-0.058
Treatment(Active):Visit	0.352	0.028	0.275	0.416
Intercept(poolsite)	0.326	0.043	0.211	0.418
CS	0.932	0.036	0.866	1.104
Residual	0.967	0.025	0.858	1.003
Rho	-0.006	0.002	-0.012	-0.002

Figure 2 present the results in graphical manner using box plots. These can be useful for detecting outliers, if coefficients are not strongly affected on the whole but that there is a strong impact for some iterations. For this model, this is not the case.

Figure 2: Box plots of the coefficients over 100 perturbations



CONCLUSION

Perturbation analysis can be a useful tool for evaluating the stability of model results when the data contains strong correlations. This can be particularly useful when models contain derived terms, such as transformations or interaction effects. Re-estimating the model a suitable number of times while adding random perturbations to selected variables can show how stable the results would be under small changes to the data. Any derived terms are re-calculated at each iteration, allowing the perturbations to propagate to the derived terms.

Very small or very large categories can be seen as a form of collinearity. The column of the model matrix for a small category will consist almost entirely of zeros, the column for a large category will consist mainly of ones. In either case, the column is highly correlated with the intercept which constitutes a collinearity problem. The impact can be assessed by randomly reclassifying cases. This is done in such a way as to maintain the relative sizes of the categories. Reclassifying can take ordered categories in to account by making small changes in category number more likely than large changes.

This approach can be applied to any regression-like statistical model and the `perturb` program is compatible with the `glm` program in R. Perturbation analysis can be extended to mixed models in a straightforward fashion. Perturbations can be added to a random slope variable, or a levels variable such as site can be randomly

reclassified. This can be a particularly enlightening exercise because fixed and random effects in a mixed model are not separate spaces but model the data jointly. Adding perturbations to a mixed model has the potential to cause shifts in both fixed and random effects. An interesting aspect of a perturbation analysis in mixed models is that it can cause model issues. Example 1 showed that a correlation of 1 can occur among random effects, together with a warning message concerning a negative Hessian. The perturb program should work with generalised mixed models using `lme4` or `nlme` packages (this has not been tested at this time).

An issue in perturbation analyses is how to determine a reasonable range for the perturbations. Too large a range will always cause erratic swings in the results, a small range could result in meaninglessly stable results. In some cases, a reasonable evaluation of measurement error could be likely. A number of ranges will have to be used in most cases. One approach could be to increase the range of perturbations until issues occur and then to evaluate whether changes to the data of this magnitude could be reasonably expected.

ACKNOWLEDGEMENTS

I would like to thank the LipiDiDiet clinical study group for permission to use data from that study.

NOTE

[1] The reclassification probabilities are created using a special application of loglinear models. The goal is to create a square table with the frequency distribution of the original variable as its row and column totals and for which the body of the table has a suitable association pattern. A suitable pattern of association can be selected from a wide range of loglinear models for square tables [4]. The most obvious choice is the “quasi-independence (constrained)” (QI-C) model. This model fits a single coefficient for the diagonal of the table for the odds of staying versus changing. Off-diagonal cells are the reference category. This means that if a change of category takes place, the probability of one category over another is the same.

A table can be created with pre-specified row and column totals and any pre-specified association pattern [7]. First, a table is created using the frequency distribution of the categorical variable and basic reclassification as was done in Table 2. The QI-C model is fit to this table and used to calculate expected frequencies with this pattern of association. The required table can now be generated using a second loglinear model. This model fits an independence model with equal main effects together with an offset variable based on the expected frequencies with the required pattern found in the previous step. The expected frequencies from this model meet the requirements.

REFERENCES

[1] Belsley DA. (1991). *Conditioning diagnostics, collinearity and weak data in regression.* New York: John Wiley & Sons.

[2] Dormann CF, Elith J, Bacher S, Buchmann C, Carl G, Carré G, García Marquéz JR, Gruber B, Brun Lafourcade B, Leitão PJ, Münkemüller T, McClean C, Osborne PE, Reineking B, Schröder B, Skidmore AK, Zurell D, Lautenbach S. (2013). Collinearity: a review of methods to deal with it and a simulation study evaluating their performance. *Ecography* 36: 027–046

[3] Fitzmaurice G., Laird N, Ware J.. (2004). *Applied Longitudinal Analysis.* New York: John Wiley & Sons.

[4] Goodman, LA. (1984). *The analysis of cross-classified data having ordered categories.* Cambridge, Mass: Harvard University Press.

[5] Hendrickx J., Pelzer B., Grotenhuis M., Lammers J.. (2004). *Collinearity involving ordered and unordered categorical variables*. Paper presented at the RC33 conference* in Amsterdam, August 17-20 2004.

[6] Hendrickx J. (2004). "PERTURB: Stata module to evaluate collinearity and ill-conditioning". *Statistical Software Components* S445201, Boston College Department of Economics,.

[7] Hendrickx J. (2005). Using standardised tables for interpreting loglinear models. *Quality and Quantity* 38:603-620

[8] Hendrickx, John. (2012). Perturb: Tools for evaluating collinearity. https://CRAN.R-project.org/package=perturb

[9] Soininen H, Solomon A, Visser PJ, Hendrix SB, Blennow K, Kivipelto M, Hartmann T. 24-month intervention with a specific multinutrient in people with prodromal Alzheimer's disease (LipiDiDiet): a randomised, double-blind, controlled trial. *Lancet Neurology* 2017; 16: 965–75

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

John Hendrickx
Danone Nutricia Research
Uppsalalaan 12
Utrecht, The Netherlands NL-3584CT
John.Hendrickx@Danone.com

* Available at: https://www.researchgate.net/publication/235994590_Collinearity_involving_ordered_and_unordered_categorical_variables