influence.ME: Tools for Detecting Influential Data in Mixed Effects Models

by Rense Nieuwenhuis, Manfred te Grotenhuis, and Ben Pelzer

Abstract   **influence.ME** provides tools for detecting influential data in mixed effects models. The application of these models has become common practice, but the development of diagnostic tools has lagged behind. **influence.ME** calculates standardized measures of influential data for the point estimates of generalized mixed effects models, such as DFBETAS, Cook's distance, as well as percentile change and a test for changing levels of significance. **influence.ME** calculates these measures of influence while accounting for the nesting structure of the data. The package and measures of influential data are introduced, a practical example is given, and strategies for dealing with influential data are suggested.

The application of mixed effects regression models has become common practice in the field of social sciences. As used in the social sciences, mixed effects regression models take into account that observations on individual respondents are nested within higher-level groups such as schools, classrooms, states, and countries (Snijders and Bosker, 1999), and are often referred to as multilevel regression models. Despite these models' increasing popularity, diagnostic tools to evaluate fitted models lag behind.

We introduce **influence.ME** (Nieuwenhuis, Pelzer, and te Grotenhuis, 2012), an R-package that provides tools for detecting influential cases in mixed effects regression models estimated with **lme4** (Bates and Maechler, 2010). It is commonly accepted that tests for influential data should be performed on regression models, especially when estimates are based on a relatively small number of cases. However, most existing procedures do not account for the nesting structure of the data. As a result, these existing procedures fail to detect that higher-level cases may be influential on estimates of variables measured at specifically that level.

In this paper, we outline the basic rationale on detecting influential data, describe standardized measures of influence, provide a practical example of the analysis of students in 23 schools, and discuss strategies for dealing with influential cases. Testing for influential cases in mixed effects regression models is important, because influential data negatively influence the statistical fit and generalizability of the model. In social science applications of mixed models the testing for influential data is especially important, since these models are frequently based on

large numbers of observations at the individual level while the number of higher level groups is relatively small. For instance, Van der Meer, te Grotenhuis, and Pelzer (2010) were unable to find any country-level comparative studies involving more than 54 countries. With such a relatively low number of countries, a single country can easily be overly influential on the parameter estimates of one or more of the country-level variables.

Detecting Influential Data

All cases used to estimate a regression model exert some level of influence on the regression parameters. However, if a single case has extremely high or low scores on the dependent variable relative to its expected value — given other variables in the model, one or more of the independent variables, or both — this case may overly influence the regression parameters by 'pulling' the estimated regression line towards itself. The simple inclusion or exclusion of such a single case may then lead to substantially different regression estimates. This runs against distributional assumptions associated with regression models, and as a result limits the validity and generalizability of regression models in which influential cases are present.

The analysis of residuals cannot be used for the detection of influential cases (Crawley, 2007). Cases with high *residuals* (defined as the difference between the observed and the predicted scores on the dependent variable) or with high standardized residuals (defined as the residual divided by the standard deviation of the residuals) are indicated as outliers. However, an influential case is not always an outlier. On the contrary: a strongly influential case dominates the regression model in such a way, that the estimated regression line lies closely to this case. Although influential cases thus have extreme values on one or more of the variables, they can be *onliers* rather than *outliers*. To account for this, the *(standardized) deleted residual* is defined as the difference between the observed score of a case on the dependent variable, and the predicted score from the regression model that is based on data from which that case was removed.

Just as influential cases are not necessarily outliers, outliers are not necessarily influential cases. This also holds for deleted residuals. The reason for this is that the amount of influence a case exerts on the regression slope is not only determined by how well its (observed) score is fitted by the specified regression model, but also by its score(s) on the

independent variable(s). The degree to which the scores of a case on the independent variable(s) are extreme is indicated by the *leverage* of this case. A higher leverage means more extreme scores on the independent variable(s), and a greater potential of overly influencing the regression outcomes. However, if a case has very extreme scores on the independent variable(s) but is fitted very well by a regression model, and if this case has a low deleted (standardized) residual, this case is not necessarily overly influencing the outcomes of the regression model.

Since neither outliers, nor cases with a high leverage, are necessarily influential, a different procedure is required for detecting influential cases. The basic rationale behind measuring influential cases is based on the principle that when single cases are iteratively omitted from the data, models based on these data should not produce substantially different estimates. If the model parameters change substantially after a single case is excluded, this case may be regarded as too influential. However, how much change in the model parameters is acceptable? To standardize the assessment of how influential a single case is, several measures of influence are commonly used. First, DFBETAS is a standardized measure of the absolute difference between the estimate with a particular case included and the estimate without that particular case (Belsley, Kuh, and Welsch, 1980). Second, Cook's distance provides an overall measurement of the change in all parameter estimates, or a selection thereof (Cook, 1977). In addition, we introduce the measure of percentile change and a test for changing levels of significance of the fixed parameters.

Up to this point, this discussion on influential data was limited to how single cases can overly influence the point estimates (or BETAS) of a regression model. Single cases, however, can also bias the confidence intervals of these estimates. As indicated above, cases with high leverage can be influential because of their extreme values on the independent variables, but not necessarily are. Cases with high leverage but a low deleted residual compress standard errors, while cases with low leverage and a high deleted residual inflate standard errors. Inferences made to the population from models in which such cases are present may be incorrect.

Detecting Influential Data in Mixed Effects Models

Other options are available in R that help evaluating the fit of regression models, including the detection of influential data. The base R installation provides various plots for regression models, including but not limited to plots showing residuals versus the fitted scores, Cook's distances, and the leverage versus the deleted residuals. The latter plot can be used to detect cases that affect the inferential properties of the model, as discussed above. These plots,

however, are not available for mixed effects models. The **LMERConvenienceFunctions** package provides model criticism plots, including the density of the model residuals and the fitted values versus the standardized residuals (Tremblay, 2012). However, while this package works with the **lme4** package, it only is applicable for *linear* mixed effects models.

The **influence.ME** package introduced here contributes to these existing options, by providing several measures of influential data for *generalized* mixed effects models. The limitation is that, unfortunately, as far as we are aware, the measure of leverage was not developed for generalized mixed effects models. Consequently, the current installment of **influence.ME** emphasizes detecting the influence of cases on the point estimates of generalized mixed effect models. It does, however, provide a basic test for detecting whether single cases change the level of significance of an estimate, and therefore the ability to make inferences from the estimated model.

To apply the logic of detecting influential data to generalized mixed effects models, one has to measure the influence of a particular higher level group on the estimates of a predictor measured at that level. The straightforward way is to delete all observations from the data that are nested within a single higher level group, then re-estimate the regression model, and finally evaluate the change in the estimated regression parameters. This procedure is then repeated for each higher-level group separately.

The `influence` function in the **influence.ME** package performs this procedure automatically, and returns an object containing information on the parameter estimates excluding the influence of each higher level group separately. The returned object of class `"estex"` (ESTimates EXcluding the influence of a group) can then be passed on to one of the functions calculating standardized measures of influence (such as DFBETAS and Cook's Distance, discussed in more detail in the next section). Since the procedure of the `influence` function entails re-estimating mixed effects models several times, this can be computationally intensive. Unlike the standard approach in R, we separated the estimation procedure from calculating the measures of influence themselves. This allows the user to process a single model once using the `influence` function, and then to evaluate it using various measures and plots of influence.

In detecting influential data in mixed effects models, the key focus is on changes in the estimates of variables measured at the group-level. However, most mixed effects regression models estimate the effects of both lower-level and higher-level variables simultaneously. Langford and Lewis (1998) developed a procedure in which the mixed effects model is modified to neutralize the group's influence on the higher-level estimate, while at the same time allowing the lower-level observations nested within

that group to help estimate the effects of the lower-level predictors in the model. For each higher-level unit evaluated based on this method, the intercept-vector of the model is set to 0, and an (additional) dummy variable is added to the model, with score 1 for the respective higher level case. This way, the case under investigation does not contribute to the variance of the random intercept, nor to the effects of variables measured at the group-level. **influence.ME** provides this functionality, which is accessed by specifying `delete=FALSE` as an option to the `influence` function. As a result of the specific modification of the model-specification, this specific procedure suggested by Langford and Lewis (1998) does not work when factor-variables are used in the regression model.

Finally, **influence.ME** also allows for detecting the influence of lower-level cases in the mixed effects model. In social science applications of mixed effects models, with a great number of lower-level observations nested in a limited number of groups, this will not always be feasible. Detecting influence of lower-level observations is supported for applications in various disciplines where mixed effects models are typically applied to only a limited number of observations per group. This procedure is accessed by specifying `obs=TRUE` as an option to the `influence` function. The `influence` function can either determine the influence of higher-level cases, or of single observations, but not both at the same time.

The Outcome Measures

The `influence` function described above returns an object with information on how much the parameter estimates in a mixed effects model change, after the (influence of) observations of higher-level groups and their individual-level observations were removed from it iteratively. This returned object can then be passed on to functions that calculate standardized measures of influence. **influence.ME** offers four such measures, which are detailed in this section.

DFBETAS

DFBETAS is a standardized measure that indicates the level of influence observations have on single parameter estimates (Fox, 2002). Regarding mixed models, this relates to the influence a higher-level unit has on the parameter estimate. DFBETAS is calculated as the difference in the magnitude of the parameter estimate between the model including and the model excluding the higher level case. This absolute difference is divided by the standard error of the parameter estimate excluding the higher level unit under investigation:

$$DFBETAS_{ij} = \frac{\hat{\gamma}_i - \hat{\gamma}_{i(-j)}}{se\left(\hat{\gamma}_{i(-j)}\right)}$$

in which i refers to the parameter estimate, and j the higher-level group, so that $\hat{\gamma}_i$ represents the original estimate of parameter i, and $\hat{\gamma}_{i(-j)}$ represents the estimate of parameter i, after the higher-level group j has been excluded from the data.

In **influence.ME**, values for DFBETAS in mixed effects models can be calculated using the function `dfbetas`, which takes the object returned from `influence` as input. Further options include `parameters` to provide a vector of index numbers or names of the selection of parameters for which DFBETAS is to be calculated. The default option of `dfbetas` is to calculate DFBETAS for estimates of all fixed effects in the model.

As a rule of thumb, a cut-off value is given for DFBETAS (Belsley et al., 1980):

$$2/\sqrt{n}$$

in which n, the number of observations, refers to the number of groups in the grouping factor under evaluation (and not to the number of observations nested within the group under investigation). Values exceeding this cut-off value are regarded as overly influencing the regression outcomes for that specific estimate.

Cook's Distance

Since DFBETAS provides a value for each parameter and for each higher-level unit that is evaluated, this often results in quite a large number of values to evaluate (Fox, 2002). An alternative is provided by Cook's distance, a commonly used measure of influence. Cook's distance provides a summary measure for the influence a higher level unit exerts on *all* parameter estimates simultaneously, or a selection thereof. A formula for Cook's Distance is provided (Snijders and Bosker, 1999; Snijders and Berkhof, 2008):

$$C_j^{0F} = \frac{1}{r+1}\left(\hat{\gamma} - \hat{\gamma}_{(-j)}\right)'\widehat{\Sigma}_F^{-1}\left(\hat{\gamma} - \hat{\gamma}_{(-j)}\right)$$

in which $\hat{\gamma}$ represents the vector of original parameter estimates, $\hat{\gamma}_{(-j)}$ the parameter estimates of the model excluding higher-level unit j, and $\widehat{\Sigma}_F$ represents the covariance matrix. In **influence.ME**, the covariance matrix of the model *excluding* the higher-level unit under investigation j is used. Finally, r is the number of parameters that are evaluated, excluding the intercept vector.

As a rule of thumb, cases are regarded as too influential if the associated value for Cook's Distance exceeds the cut-off value of (Van der Meer et al., 2010):

$$4/n$$

in which n refers to the number of groups in the grouping factor under evaluation.

In **influence.ME**, values for Cook's distance in mixed effects models are calculated using the function `cooks.distance`, which takes the object returned from `influence` as input. Further options include `parameters` to provide a vector of index numbers or names of the parameters for which Cook's Distance is to be calculated. In addition, the user can specify `sort=TRUE` to have the values for Cook's distance returned in descending order.

As a final note, it is pointed out that if Cook's distance is calculated based on a single parameter, the Cook's distance equals the squared value of DFBETAS for that parameter. This is also reflected in their respective cut-off values:

$$\sqrt{\frac{4}{n}} = \frac{2}{\sqrt{n}}$$

Percentile Change

Depending upon the goal for which the mixed model is estimated (prediction vs. hypothesis testing), the use of formal measures of influence as DFBETAS and Cook's distance may be less desirable. The reason for this is that based on these measures it is not immediately clear to what extent parameter estimates change. For substantive interpretation of the model outcomes, the relative degree to which a parameter estimate changes may provide more meaningful information. A simple alternative is therefore offered by the function `pchange`, which takes the same input-options as the `dfbetas` function. For each higher-level group, the percentage of change is calculated as the absolute difference between the parameter estimate both including and excluding the higher-level unit, divided by the parameter estimate of the complete model and multiplied by 100%. A percentage of change is returned for each parameter separately, for each of the higher-level units under investigation. In the form of a formula:

$$\left(\hat{\gamma} - \hat{\gamma}_{(-j)}\right) \frac{1}{\hat{\gamma}} * 100\%$$

No cut-off value is provided, for determining what percent change in parameter estimate is considered too large will primarily depend on the goal for which the model was estimated and, more specifically, the nature of the hypotheses that are tested.

Test for changes in significance

As discussed above, even when cases are not influential on the point estimates (BETAS) of the regression model, cases can still influence the standard errors of these estimates. Although **influence.ME** cannot provide the leverage measure to detect this, it provides a test for changes in the statistical significance of the fixed parameters in the mixed effects model.

The `sigtest` function tests whether excluding the influence of a single case changes the statistical significance of any of the variables in the model. This test of significance is based on the test statistic provided by the **lme4** package. The nature of this statistic varies between different distributional families in the generalized mixed effects models. For instance, the t-statistic is related to a normal distribution while the z-statistic is related to binomial distributions.

For each of the cases that are evaluated, the test statistic of each variable is compared to a test-value specified by the user. For the purpose of this test, the parameter is regarded as statistically significant if the test statistic of the model exceeds the specified value. The `sigtest` function reports for each variable the estimated test statistic after deletion of each evaluated case, whether or not this updated test statistic results in statistical significance based on the user-specified value, and whether or not this new statistical significance differs from the significance in the original model. So, in other words, if a parameter was statistically significant in the original model, but is no longer significant after the deletion of a specific case from the model, this is indicated by the output of the `sigtest` function. It is also indicated when an estimate was not significant originally, but reached statistical significance after deletion of a specific case.

Plots

All four measures of influence discussed above, can also be plotted. The `plot` function takes the output of the `influence` function to create a dotplot of a selected measure of influence (cf. Sarkar, 2008). The user can specify which measure of influence is to be plotted using the `which=` option. The `which=` option defaults to `"dfbetas"`. Other options are to select `"cook"` to plot the Cook's distances, `"pchange"` to plot the percentage change, and `"sigtest"` to plot the test statistic of a parameter estimate after deletion of specific cases.

All plots allow the output to be sorted (by specifying `sort=TRUE` and the variable to sort on using `to.sort=` (not required for plotting Cook's distances). In addition, a cut-off value can be specified using (`cutoff=`). Values that exceed this cut-off value will be plotted visually differently, to facilitate the identification of influential cases. By default, the results for all cases and all variables are plotted, but a selection of these can be made by specifying `parameters=` and / or `groups=`. Finally, by specifying `abs=TRUE` the absolute values of the measure of influence are plotted.

Example: students in 23 schools

In our example, we are interested in the relationship between the degree of structure that schools attempt to enforce in their classrooms and students' performance on a math test. Could it be that a highly structured class affects their performance?

The **influence.ME** package contains the `school23` data.frame, that provides information on the performance of 519 students in 23 schools. Measurements include individual students' score on a math test, school-level measurements of class structure, and several additional independent variables. Student's class and school are equivalent in this data, since only one class per school is available. These data are a subset of the NELS-88 data (National Education Longitudinal Study of 1988). The data are publicly available from the internet: http://www.ats.ucla.edu/stat/examples/imm/, and are reproduced with kind permission of Ita Kreft and Jan de Leeuw (1998).

First, using the **lme4** package, we estimate a multivariate mixed effects model with students nested in schools, a random intercept, a measurement of individual students' time spent on math homework, and a measurement of class structure at the school level. For the purpose of our example, we assume here that the math, homework, and structure variables were correctly measured at the interval level.

```
library(influence.ME)
data(school23)

school23 <- within(school23,
   homework <- unclass(homework))

m23 <- lmer(math ~ homework + structure
   + (1 | school.ID),
   data=school23)

print(m23, cor=FALSE)
```

This results in the summary of the model based on 23 schools (assigned to object `m23`), as shown below.

```
Linear mixed model fit by REML
Formula: math ~ homework +
   structure + (1 | school.ID)
   Data: school23
 AIC  BIC logLik deviance REMLdev
3734 3756  -1862     3728    3724
Random effects:
 Groups    Name        Variance Std.Dev.
 school.ID (Intercept) 19.543   4.4208
 Residual              71.311   8.4446
Number of obs: 519, groups: school.ID, 23
```

```
Fixed effects:
            Estimate Std. Error t value
(Intercept)  52.2356     5.3940   9.684
homework      2.3938     0.2771   8.640
structure    -2.0950     1.3237  -1.583
```

Based on these estimates, we may conclude that students spending more time on their math homework score better on a math test. Regarding class structure, however, we do not find a statistically significant association with math test scores. But, can we now validly conclude that class structure does not influence students' math performance, based on the outcomes of this model?

Visual Examination

Since the analysis in the previous section has been based on the limited number of 23 schools, it is, of course, possible that observations on single schools have overly influenced these findings. Before using the tools provided in the **influence.ME** package to formally evaluate this, a visual examination of the relationship between class structure and math test performance, aggregated to the school level, will be performed.

```
struct <- unique(subset(school23,
   select=c(school.ID, structure)))

struct$mathAvg <- with(school23,
   tapply(math, school.ID, mean))

dotplot(mathAvg ~ factor(structure),
   struct,
   type=c("p","a"),
   xlab="Class structure level",
   ylab="Average Math Test Score")
```
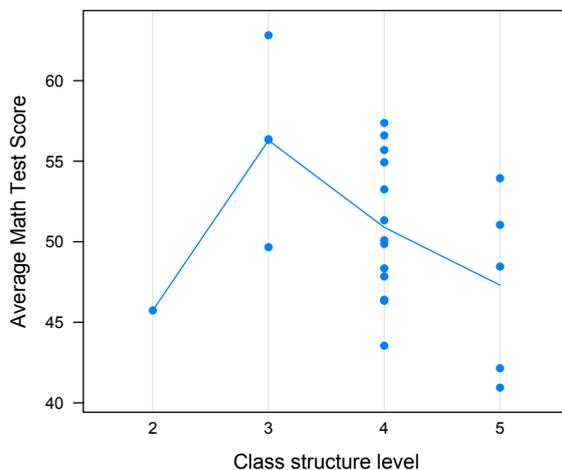

Figure 1: Visual examination of class structure and math performance

In the syntax above, a bivariate plot of the aggregated math scores and class structure is created, which is shown in Figure 1. In this plot, it is clear that one single school represented in the lower-left corner of the graph seems to be an outlier, and - more importantly - the non-linear curve shown in this graph clearly indicates this single school with class structure level of 2 may overly influence a linear regression line estimated based on these data.

Calculating measures of influence

In the previous section, based on Figure 1 we suspected that the combination in one specific school of the low average math test results of students, and the low level of class structure in that school, may have overly influenced the original analysis of our data. However, this is only a bivariate examination of the data, and therefore does not take into account other variables in the model. Hence, in our example, our preliminary conclusion that this may be an influential case is not controlled for possible effects of the homework variable. A better test is provided by standardized measures of influence, as calculated from the regression model rather than from the raw data.

The first step in detecting influential data is to determine the extent to which the parameter estimates in model `m23` change, when iteratively each of the schools is deleted from the data. This is done with the `influence` function:

```
estex.m23  <- influence(m23, "school.ID")
```

The `influence` function takes a mixed effects regression model as input (here: `m23`), and the grouping factor needs to be specified, which in our case is `school.ID`. We assign the output of the `influence` function to an object named `estex.m23`. Below, we use this object as input to the `dfbetas` function, to calculate DFBETAS.

```
dfbetas(estex.m23,
    parameters=c(2,3))
```

This results in a substantial amount of output, a portion of which is shown below. Only the DFBE-TAS for the `homework` and `structure` variables were returned, since `parameters=c(2,3)` was specified.

```
        homework     structure
6053   -0.13353732  -0.168139487
6327   -0.44770666   0.020481057
6467    0.21090081   0.015320965
7194   -0.44641247   0.036756281
7472   -0.55836772   1.254990963
...
72292   0.62278508   0.003905031
72991   0.52021424   0.021630219
```

The numerical output given above by the `dfbetas` function provides a detailed report of the values of DFBETAS in the model. For each variable, as well as for each nesting group (in this example: each school), a value for DFBETAS is computed and reported upon. The cut-off value of DFBETAS equals $2/\sqrt{n}$ (Belsley et al., 1980), which in this case equals $2/\sqrt{23} = .41$. The estimate for class structure in this model seems to be influenced most strongly by observations in school number 7472. The DFBETAS for the `structure` variable clearly exceeds the cut-off value of .41. Also, the estimates of the `homework` variable changes substantially with the deletion of several schools, as indicated by the high values of DF-BETAS.

A plot (shown in Figure 2) of the DFBETAS is created using:

```
plot(estex.m23,
    which="dfbetas",
    parameters=c(2,3),
    xlab="DFbetaS",
    ylab="School ID")
```

Based on Figure 2, it is clear that both the `structure` and the `homework` variables are highly susceptible to the influence of single schools. For the `structure` variable this is not all that surprising, since class structure was measured at the school level and shown in Figure 1 to be very likely to be influenced by a single case: school number 7472. The observation that high values of DFBETAS were found for the `homework` variable, suggests that substantial differences between these schools exist in terms of how much time students spend on average on their homework. Therefore, we suggest that in mixed effects regression models, both the estimates of individual-level and group-level variables are evaluated for influential data.

The measure of Cook's distance allows to determine the influence a single higher-level group has on the estimates of multiple variables simultaneously. So, since the `cooks.distance` function allows to specify a selection of variables on which the values for Cook's distance are to be calculated, this can be used to limit the evaluation to the measurements at the group-level exclusively. Note, that whereas DFBE-TAS always relates to single variables, Cook's distance is a summary measure of changes on all parameter estimates it is based on. Reports on Cook's distance thus should always specify on which variables these values are based.

To continue our example, we illustrate the `cooks.distance` function on a single variable, since class structure is the only variable measured at the school-level. In the example below, we use the same object that was returned from the `influence` function. The specification of this function is similar to `dfbetas`, and to create a plot of the Cook's distances we again use the `plot` function with the spec-
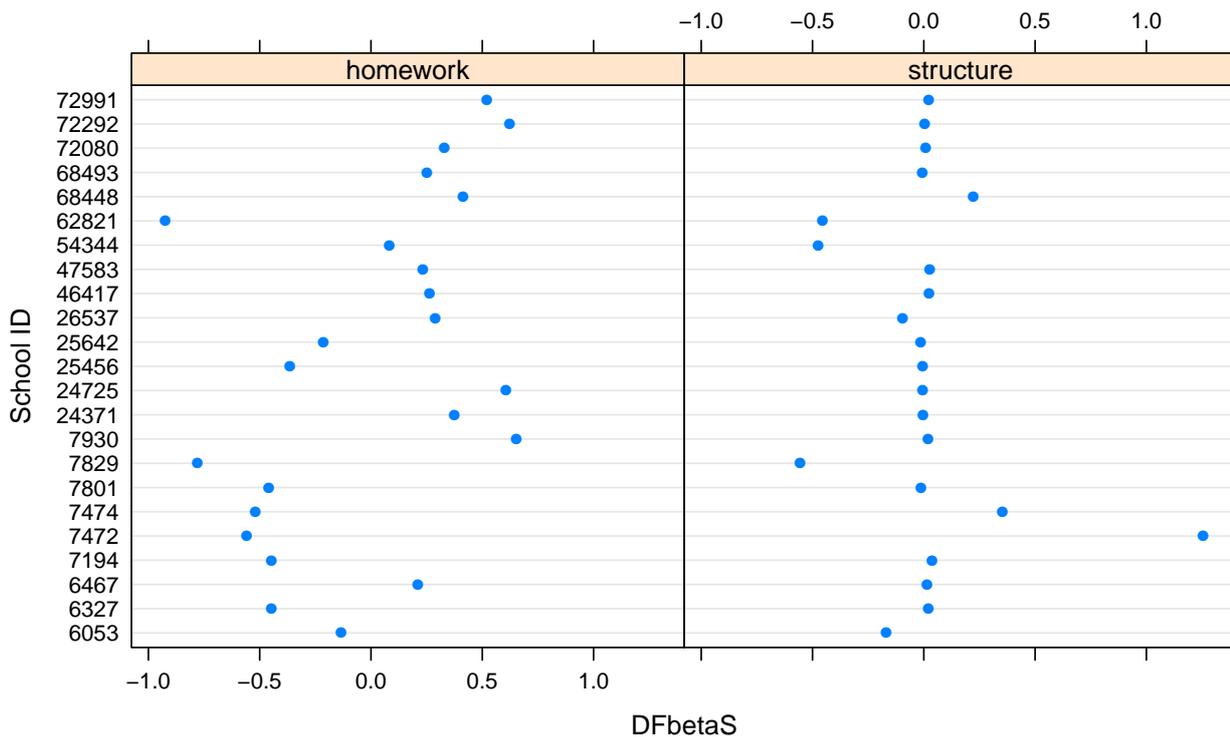
Figure 2: DFBETAS of class structure and homework

ification `which="cook"`. We specify two additional arguments to augment the figure. First, we specify `sort=TRUE` to have the resulting Cook's distances sorted in a descending order in the figure. The appropriate cut-off value for Cook's distance with 23 nesting groups equals to $4/23 = .17$. By specifying the cut-off value with `cutoff=.17`, Cook's distances exceeding the specified value are easily identified in the resulting figure. Thus, to receive both numeric output and a graphical representation (Figure 3), the following specification of `cooks.distance` and `plot` is given:

```
cooks.distance(estex.m23,
    parameter=3, sort=TRUE)

plot(estex.m23, which="cook",
    cutoff=.17, sort=TRUE,
    xlab="Cook´s Distance",
    ylab="School ID")
```

The output below shows one value of Cook's distance for each nesting group, in this case for each school.

```
                [,1]
24371  6.825871e-06
72292  1.524927e-05
...
54344  2.256612e-01
7829   3.081222e-01
7472   1.575002e+00
```
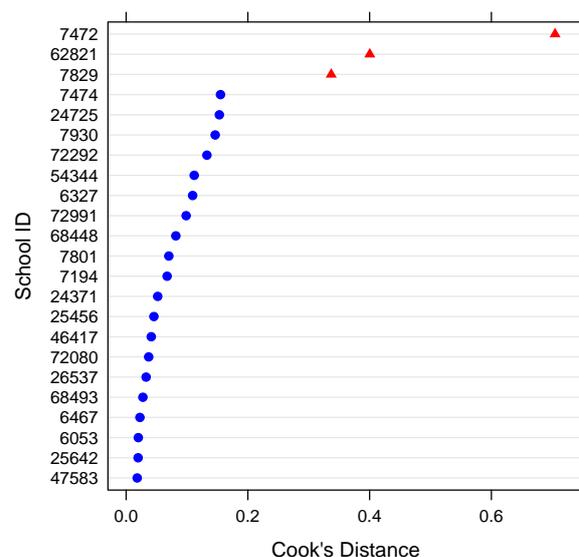

Figure 3: Cook's Distance based on class structure

Only a selection of the output is shown here. A

few schools exceed the cut-off value (in Figure 3 these are indicated with red triangles), but one school stands out: 7472. Clearly, this school strongly influences the outcomes regarding the `structure` variable, as we already suspected based on our bivariate visual examination in Figure 1.

Testing for Changes in Statistical Significance (sigtest)

In the example below, the `sigtest` function is used to test for changing levels of significance after deletion of each of the 23 schools from our example model. We are specifically interested in the level of significance of the `structure` variable, for which it was already established above that school with number 7472 is very influential. Since we observed a negative effect in the original model, we specify `test=-1.96` to test for significance at a commonly used value (-1.96) of the test statistic. Note that since we estimated a normally distributed model, the test statistic here is the t-value.

```
sigtest(estex.m23, test=-1.96)$structure[1:10,]
```

In the example above, we only request the results for the `structure` variable and for the first 10 schools. In the results presented below, three columns are shown. The first column (`Altered.Teststat`) shows the value of the test statistic (here for the `structure` variable) after the deletion of the respective schools (indicated in the row labels). Especially school number 7472 stands out. In the original model, the test statistic for the structure variable was -1.583, which was not significant. When the influence of school number 7472 is excluded from the model, the test statistic now is -2.72, which exceeds the selected value of -1.96 selected by us. That the structure variable would be significant by deletion of school 7472 is indicated in the second column (`Altered.Sig`). The `Changed.Sig` column finally confirms whether the level of significance of the `structure` variable (which was not significant in the original model) changed to significant after each of the schools was deleted.

In the case of our example, the results for Cook's Distance and the results of this test for changing levels of significance both indicate that school number 7472 overly influences the regression outcomes regarding the school-level `structure` variable. Referring to the discussion on influential data above, however, we emphasize that this is not necessarily always the case. Cases can influence the point estimates without affecting their level of significance, or affect the level of significance without overly affecting the point estimate itself. Therefore, both tests should be performed.

	Altered.Teststat	Altered.Sig	Changed.Sig
6053	-1.326409	FALSE	FALSE
6327	-1.688663	FALSE	FALSE
6467	-1.589960	FALSE	FALSE
7194	-1.512686	FALSE	FALSE
7472	-2.715805	TRUE	TRUE
7474	-1.895138	FALSE	FALSE
7801	-1.534023	FALSE	FALSE
7829	-1.045866	FALSE	FALSE
7930	-1.566117	FALSE	FALSE
24371	-1.546838	FALSE	FALSE

Before, using DFBETAS, we identified several schools that overly influence the estimate of the `homework` variable. We have there performed `sigtest` test to evaluate whether deletion of any of the schools changes the level of significance of the `homework` variable. These results are not shown here, but indicated that the deletion of none of the schools changed the level of significance of the `homework` variable.

Measuring the influence of lower-level observations

Finally, it is possible that a single lower-level observation affects the results of the mixed effects model, especially for data with a limited number of lower-level observations per group. In our example, this would refer to a single student affecting the estimates of either the individual-level variables, the school-level variables, or both. Here, we test whether one or more individual students affect the estimate of the school-level `structure` variable.

To perform this test, the `influence` function is used, and `obs=TRUE` is specified to indicate that single observations (rather than groups) should be evaluated. The user is warned that this procedure often will be computationally intensive when the number of lower-level observations is large.

Next, we request Cook's Distances specifically for the `structure` variable. Since the number of student-level observations in this model is 519, and cut-off value for Cook's Distance is defined as $4/n$, the cut-off value is $4/519 = .0077$. The resulting output is extensive, since a Cook's Distance is calculated for any of the 519 students. Therefore, in the example below, we directly test which of the resulting Cook's Distances exceeds the cut-off value.

```
estex.obs  <- influence(m23, obs=TRUE)
cks.d <- cooks.distance(estex.obs, parameter=3)
which(cks.d > 4/519)
```

The output is not shown here, but the reader can verify that students with numbers 88 and 89 exert too much influence on the estimate of the structure variable. Using the `sigtest` function, however, showed that the deletion of none of the students from the

data affected the level of significance of the structure variable, nor of any of the other variables in the model.

Dealing with Influential Data

Now that overly influential cases have been identified in our model, we have to decide how to deal with them. Generally, there are several strategies, including getting more data, checking data consistency, adapting model specification, deleting the influential cases from the model, and obtaining additional measurements on existing cases to account for the overly influential cases (Van der Meer et al., 2010; Harrell, Jr., 2001).

Since overly influential data are a problem especially encountered in models based on a limited number of cases, a straightforward remedy would be to observe more cases in the population of interest. In our example, if we would be able to sample more schools, it may very well turn out that we observe several additional schools with a low score on the `structure` variable, so that school number 7472 is no longer influential. Secondly, there may have been measurement, coding, or transcription errors in the data, that have lead to extreme scores on one or more of the variables (i.e. it may be worthwhile, if possible, to check whether class structure and / or students' math performance in school 7472 really is that low). Thirdly, the model specification may be improved. If the data are used to estimate too complex models, or if parameterization is incorrect, influential cases are more likely to occur. Perhaps the `structure` variable should have been treated as categorical.

These are all general strategies, but cannot always be applied. Depending on the research setting, it is not always feasible to obtain more observations, to return to the raw data to check consistency, or to reduce model complexity or change parameterization.

The fourth strategy, deleting influential cases from the model, can often be applied. In general, we suggest deleting influential cases one at the time and then to re-evaluating the model. Deleting one or more influential cases from a mixed effects model is done with the `exclude.influence` function. The input of this function is a mixed effects model object, and it returns an updated mixed effects model from which a specified group was deleted. To illustrate, we delete school number 7472 (which was identified as being overly influential) and its individual-level observations, using the example code below:

```
m22 <- exclude.influence(m23,
    "school.ID", "7472")

print(m22, cor=FALSE)
```

The `exclude.influence` function takes a mixed effects model as input, and requires the specification of the grouping factor (`school.ID`) and the group to be deleted (`7472`). It returns a re-estimated mixed effects model, that we assign to the object `m22`. The summary of that model is shown below:

```
Linear mixed model fit by REML
Formula: math ~ homework + structure
    + (1 | school.ID)
    Data: ..1
  AIC  BIC logLik deviance REMLdev
 3560 3581  -1775     3554     3550
Random effects:
 Groups     Name          Variance Std.Dev.
 school.ID (Intercept) 15.333   3.9157
 Residual                70.672   8.4067
Number of obs: 496, groups: school.ID, 22

Fixed effects:
            Estimate Std. Error t value
(Intercept) 59.4146     5.9547   9.978
homework     2.5499     0.2796   9.121
structure   -3.8949     1.4342  -2.716
```

Two things stand out when this model summary is compared to our original analysis. First, the number of observations is lower (496 versus 519), as well as the number of groups (22 versus 23). More importantly, though, the negative effect of the `structure` variable now is statistically significant, whereas it was not in the original model. So, now these model outcomes indicate that higher levels of class structure indeed are associated with lower math test scores, even when controlled for the students' homework efforts.

Further analyses should repeat the analysis for influential data, for other schools may turn out to be overly influential as well. These repetitive steps are not presented here, but as it turned out, three other schools were overly influential. However, the substantive conclusions drawn based on model `m22` did not change after their deletion.

Finally, we suggest an approach for dealing with influential data, based on Lieberman (2005). He argues that the presence of outliers may indicate that one or more important variables were omitted from the model. Adding additional variables to the model may then account for the outliers, and improve the model fit. We discussed above that an influential case is not necessarily an outlier in a regression model. Nevertheless, if additional variables in the model can account for the fact that an observation has extreme scores on one or more variables, the case may no longer be an influential one.

Thus, adding important variables to the model may solve the problem of influential data. When the

observations in a regression model are, for instance, randomly sampled respondents in a large-scale survey, it often is impossible to return to these respondents for additional measurements. However, in social science applications of mixed effects models, the higher-level groups are often readily accessible cases such as schools and countries. It may very well be possible to obtain additional measurements on these schools or countries, and use these to remedy the presence of influential data.

Summary

influence.ME provides tools for detecting influential data in mixed effects models. The application of these models has become common practice, but the development of diagnostic tools lag behind. **influence.ME** calculates standardized measures of influential data such as DFBETAS and Cook's distance, as well as percentile change and a test for changing in statistical significance of fixed parameter estimates. The package and measures of influential data were introduced, a practical example was given, and strategies for dealing with influential data were suggested.

Bibliography

D. Bates and M. Maechler. *lme4: Linear mixed-effects models using S4 classes*, 2010. URL http://CRAN.R-project.org/package=lme4. R package version 0.999375-35. [p38]

D. Belsley, E. Kuh, and R. Welsch. *Regression Diagnostics. Identifying Influential Data and Sources of Collinearity*. Wiley, 1980. [p39, 40, 43]

R. Cook. Detection of influential observations in linear regression. *Technometrics*, 19(1):15–18, 1977. [p39]

M. J. Crawley. *The R Book*. Wiley, 2007. [p38]

J. Fox. *An R and S-Plus Companion to Applied Regression*. Sage, 2002. [p40]

F. E. Harrell, Jr. *Regression Modeling Strategies. With Applications to Linear Models, Logistic Regression, and Survival Analysis*. Springer, 2001. [p46]

I. Kreft and J. De Leeuw. *Introducing Multilevel Modeling*. Sage Publications, 1998. [p42]

I. Langford and T. Lewis. Outliers in multilevel data. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 161:121–160, 1998. [p39, 40]

E. S. Lieberman. Nested analysis as a mixed-method strategy for comparative research. *American Political Science Review*, 99:435–452, 2005. [p46]

R. Nieuwenhuis, B. Pelzer, and M. te Grotenhuis. *influence.ME: Tools for detecting influential data in mixed effects models*, 2012. URL http://CRAN.R-project.org/package=influence.ME. R package version 0.9. [p38]

D. Sarkar. *Lattice. Multivariate Data Visualization with R*. Springer, 2008. [p41]

T. Snijders and J. Berkhof. Diagnostic checks for multilevel models. In J. De Leeuw and E. Meijer, editors, *Handbook of Multilevel Analysis*, chapter Diagnostic checks for multilevel models, pages 141–175. Springer, 2008. [p40]

T. Snijders and R. Bosker. *Multilevel analysis, an introduction to basic and advanced multilevel modelling*. Sage, 1999. [p38, 40]

A. Tremblay. *LMERConvenienceFunctions: A suite of functions to back-fit fixed effects and forward-fit random effects, as well as other miscellaneous functions.*, 2012. URL http://CRAN.R-project.org/package=LMERConvenienceFunctions. R package version 1.6.8.2. [p39]

T. Van der Meer, M. te Grotenhuis, and B. Pelzer. Influential cases in multilevel modeling. A methodological comment. *American Sociological Review*, 75: 173–178, 2010. [p38, 40, 46]

Rense Nieuwenhuis
Institute for Innovation and Governance Studies (IGS),
University of Twente
P.O. Box 217, 7500 AE, Enschede
The Netherlands
r.nieuwenhuis@utwente.nl

Manfred te Grotenhuis
Radboud University
Nijmegen
The Netherlands
m.tegrotenhuis@maw.ru.nl

Ben Pelzer
Radboud University
Nijmegen
The Netherlands
b.pelzer@maw.ru.nl