# A simple solution for missing observations based on random effects models

<table>
<thead>
<tr>
<th>Journal:</th>
<th>Behavior Research Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manuscript ID:</td>
<td>Draft</td>
</tr>
<tr>
<td>Manuscript Type:</td>
<td>Original Manuscript</td>
</tr>
<tr>
<td>Date Submitted by the Author:</td>
<td>n/a</td>
</tr>
</tbody>
</table>
| Complete List of Authors: | Stevens, Michael; Ghent University, Department of Experimental Psychology  
Brysbaert, Marc; Ghent University, Department of Experimental Psychology |
A simple solution for missing observations based on random effects models

Michaël Stevens¹
Marc Brysbaert¹

¹Department of Experimental Psychology, Ghent University, Ghent, Belgium

Corresponding author: Michaël Stevens, Department of Experimental Psychology, Ghent University, Henri Dunantlaan 2, B-9000 Gent, Belgium. Tel: +32 (0)9 264 94 30, Fax: +32 (0)9 264 64 96. Electronic mail can be sent to michael.stevens@ugent.be
Abstract

This article presents a simple imputation method for handling missing observations in large-scale lexical decision databases. The method is based on the grand mean, participant effect, stimulus effect and error variance estimated by a random effects model. This model also yields estimates of the participant and stimulus variance that are used to compute several Intraclass Correlation Coefficients (ICCs, reliability indices) of the data. The algorithm is fast. This makes it ideal for large datasets. It also does not change the expected value for the participants or the stimuli. Most importantly, it does not artificially inflate the reliability of the data as indexed by the ICC. Because the algorithm is based on estimates of the variance components in the data, we call it the VarIaNce Components Imputation (VINCI) algorithm. We present two simulation studies with the algorithm and provide an application. The first study indicates that ICC estimates based on random effects models are not biased because of missing values, whereas ICC estimates based on ANOVA or split-half correlations are. The second simulation shows that our imputation algorithm does not bias the ICC estimates. We then apply our algorithm to the data of the British, Dutch, English and French Lexicon Projects. The implementation of the algorithm in R and the imputed lexicon projects are available at http://crr.ugent.be/vinci.
A simple solution for missing observations based on random effects models

Missing observations are numerous in empirical research. Some of these could have been avoided (e.g., technical malfunctioning or problems with data storage), but others are inherent to the task. For instance, in many psychological experiments participants have to make binary decisions (“Is this a word not?”, “Does this picture refer to a tool or not?”, “Is the stimulus symmetric or not?”). In this type of tasks, errors are unavoidable, partly because the participants misjudge the stimulus, but also because their responses are noisy (i.e., inconsistent from trial to trial; Diependaele, Neri & Brysbaert, 2012; Neri, 2010). It is customary in latency analysis only to include the reaction times (RTs) to the correct responses, leaving out some 5-20% of the responses (and including an unknown percentage of RTs to trials with “correct” responses based not on veridical judgment but on lucky guessing).

Missing values in experimental psychology have traditionally been dealt with by using averages across multiple trials per participant. If an experiment has two conditions with 50 trials each per participant, then all existing parametric and non-parametric techniques requiring one value per participant per condition can easily handle the situation of missing values by calculating the mean values per condition on the remaining trials. So, RT data in binary decision tasks typically consist of mean RTs per participant of the correct trials in the “yes” condition and in the “no” condition. In addition, a variable of percentage error (or accuracy) is calculated to indicate how many data are missed because of erroneous responding.
The situation is different in research areas where all observations matter for the analysis. In factor analysis, for instance, many algorithms discard all the data from a participant as soon as there is one missing observation. Then, a 5% rate of missing observations can already have detrimental effects on the number of participants left in the analysis. As a result, researchers working in these areas have developed algorithms to impute missing values (Schafer & Graham, 2002). Data imputation involves the replacement of a missing value with a likely value based on the remaining data.

For two reasons, data imputation is an interesting option for experimental psychologists as well. First, in many experiments the missing data are not a random sample. Take an RT experiment for instance. Chances of a missing RT are higher for difficult trials than for easy trials, meaning that missing observations tend to decrease the difference between two conditions if one is easier than the other. This is partly solved by the fact that the error rate is higher for the difficult condition but still it means that the effect is spread over two different variables.¹ The second reason why data imputation is becoming more interesting for experimental psychologists is that more refined statistical techniques are becoming available than the traditional analyses of variance (ANOVA). Although some of these techniques work fine with missing data, this is not true for all. In addition, there are indications that the degree of systematic variance in the data decreases with missing values (Courrieu & Rey, 2011).

¹ Researchers have tried to combine RTs and errors in a so-called Inverse Efficiency Score, but analysis of datasets suggests that this does not work well because it considerably increases the noise in the data (Bruyer & Brysbaert, 2011).
We were confronted with the problem of what to do with missing values in the megastudy approach to word processing. In this approach, naming latencies or lexical decision latencies are collected for a large number of words, to be used for regression analyses or virtual experiments (Balota et al., 2007; Keuleers, Diependaele & Brysbaert, 2010; Keuleers, Lacey, Rastle & Brysbaert, 2012). Thus far the tradition has been to provide users with mean RT per word based on the correct trials (together with the percentage accuracy for the word). A better approach would be to have a full matrix of RTs. Then all analyses one can think of become feasible and we can be surer about the quality of the mean RTs of the words (as they are no longer distorted by missing observations). Below we describe the reasoning we followed to find an acceptable imputation algorithm. As will become clear, implementing such an algorithm has become easier (and also better) with the recent introduction of mixed effects models (which are rapidly replacing the traditional ANOVAs).

The Philosophy Behind Data Imputation

In most studies the data of an experiment can be represented in a matrix of participants and stimuli. Table 1, for instance, shows the matrix of 6 participants responding to 5 stimuli.

Table 1: Data from a toy study in which 6 participants (P1-P6) respond to 5 stimuli (S1-S5). Also shown are the means of the participants (last column) and the stimuli (bottom line). The value 22.4 in the right bottom corner is the grand mean (i.e., the mean of all the values)
<table>
<thead>
<tr>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>22</td>
<td>20</td>
<td>12</td>
<td>23</td>
<td>13</td>
<td>18</td>
</tr>
<tr>
<td>S2</td>
<td>23</td>
<td>15</td>
<td>26</td>
<td>22</td>
<td>22</td>
<td>30</td>
</tr>
<tr>
<td>S3</td>
<td>27</td>
<td>22</td>
<td>18</td>
<td>29</td>
<td>32</td>
<td>34</td>
</tr>
<tr>
<td>S4</td>
<td>25</td>
<td>9</td>
<td>17</td>
<td>30</td>
<td>27</td>
<td>24</td>
</tr>
<tr>
<td>S5</td>
<td>30</td>
<td>16</td>
<td>19</td>
<td>22</td>
<td>19</td>
<td>26</td>
</tr>
<tr>
<td><strong>Mean</strong></td>
<td>25.4</td>
<td>16.4</td>
<td>18.4</td>
<td>25.2</td>
<td>22.6</td>
<td>26.4</td>
</tr>
</tbody>
</table>

Suppose now the value of participant 3 to stimulus 3 (i.e., the value 18 in the middle of the table) is missing. A simple and straightforward way to estimate this value (Kirk, 1982) is to take into account (1) the grand mean, (2) the mean response level of participant 3, and (3) the mean response level to stimulus 3. So, in an RT study, one would take into account the overall RT, whether the participant was fast or slow, and whether the item was easy or difficult. This can be done by adding the row mean to the column mean and subtracting the grand mean. The various means are not exactly the ones listed in Table 1, because the missing value 18 is not included. Thus, the estimated value of participant 3 to stimulus 3 would be $(12+26+17+19)/4 + (27+22+29+32+34)/5 – 22.65 = 18.5 + 28.8 – 22.65 = 24.65$.

The best estimate would take into account that participant 3 had the second lowest values, but that stimulus 3 received the highest values. An estimated value of 24.65 fits well within the other values of S3 (it is the second lowest, as is true for the mean values across the participants) and reasonably well within the other values of P3 (only S2 has a higher value for this participant). As a matter of fact, the estimated value fits better within the table than the actually observed value! This is part of the
problem with the simple algorithm: It provides an error-free estimate of the missing value, whereas all other cells contain noisy data. In the end, this reduces the error variance in the data matrix (certainly when more than one observation is missing), which artificially inflates the estimated reliability of the data. This is not what we want: the goal of missing data imputation is to make valid and efficient inferences about the population of interest (Schafer & Graham, 2002, p 149). Therefore one should not only focus on the mean of the imputed values, but also on the variance and, in more elaborate imputation algorithms, the correlation with other variables in the data. So, a better algorithm takes into account the degree of error variance in the data, in order not to inflate the estimated reliability of the data.

Before we detail our search for a better algorithm, we first want to describe how a mixed effect model can replace the simple estimate based on the means of columns and rows. Mixed effects (lme) models see observed values as the outcome of a combination of fixed effects (usually the independent variables in the design), random effects (from the participants and the stimuli), and error (Baayen, Davidson & Bates, 2008). In the simple matrix of Table 1, there would be no fixed effects (as no variable has been manipulated), but each participant would have their random intercept (starting level of performance), each stimulus would have their random intercept, and the specific observations would be subject to error variance. More specifically, the model assumes that each observed value equals a global intercept + a participant intercept + a stimulus intercept + error and it will search for the best parameters associated with each component. Once the parameters are determined, they can be used to predict the observed values and (hence) any missing values. Because the model we are using does not contain any fixed effects, it is more correct
to call it a random effects model than a mixed effects model (although the calculations are the same).

Random effects models are too complicated to calculate by hand, but they are part of many statistical packages, such as SPSS or R. The starting point always is to rewrite the data matrix in long format (i.e., the format used for regression analysis). So, Table 1 would be rewritten as:

<table>
<thead>
<tr>
<th>Part</th>
<th>Stim</th>
<th>Obs</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>S1</td>
<td>22</td>
</tr>
<tr>
<td>P1</td>
<td>S2</td>
<td>23</td>
</tr>
<tr>
<td>P1</td>
<td>S3</td>
<td>27</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P6</td>
<td>S4</td>
<td>24</td>
</tr>
<tr>
<td>P6</td>
<td>S5</td>
<td>26</td>
</tr>
</tbody>
</table>

Then we estimate the best intercepts for each participant and each stimulus. We will do this with the R-package lme4 (Bates, Maechler & Bolker, 2013), as this is becoming the standard in experimental psychology. The following command is required:

```r
fit.toy <- lmer(Obs ~ 1 + (1|Part) + (1|Stim), data.toy)
```

This command fits a model with a global intercept (the first 1), a participant intercept (1|Part), and a stimulus intercept (1|Stim). The outcome can be seen with the command `summary(fit.toy)`

This gives the output:

```
Linear mixed model fit by REML
Formula: Obs ~ 1 + (1 | Part) + (1 | Stim)
```
To find the intercepts of each participant and stimulus, we use the command ranef(fit.toy), which results in:

$Part
(Intercept)
P1  2.2584921
P2 -4.5169841
P3 -3.0113228
P4  2.1079259
P5  0.1505661
P6  3.0113228

$Stim
(Intercept)
S1 -2.9063824
S2  0.3963249
S3  3.0384907
S4 -0.2642166
S5 -0.2642166

On the basis of this information, it is easy to calculate the predicted value of participant 3, stimulus 3. It is: $22.400 - 3.011 + 3.038 = 22.427$

The value of 22.427 is the value we would predict on the basis of the full dataset. To estimate the missing observation of participant 3 stimulus 3, we have to redo the analysis without the line P3 S3 18 (i.e., the missing value). The algorithm lme4 has no problems with missing observations and it will result in the estimate: $22.619 - 2.147 + 4.090 = 24.562$ (remember that the estimate on the basis of the row and column means was 24.65).
Again, the estimate is “too good”, because it does not include noise. As attentive readers may have guessed, however, the random effects algorithm provides us with information to estimate (and hence to use) the noise, but before we can do so, we need to make sure that the model provides valid information and is not introducing unwanted biases. This is investigated in the next section.

### Measurement Reliability

As indicated above, a good imputation method preserves the noise in the data. A straightforward way to estimate the systematic and error variance in a set of measurements is to estimate the reliability of the measurements. A reliability index tries to assess how much a measurement will correlate with a future measurement of the same construct. A simple measure often used in megastudies is the split-half reliability (e.g., Courrieu & Rey, 2011; Keuleers et al., 2010, 2012). It can easily be calculated by computing the stimulus means for one random half of the participants and correlating these with the stimulus means computed for the remaining participants. This value gives the amount of systematic variance in the measurement.

A problem with the split-half reliability is that its value will vary (slightly) dependent on the random split of participants. So, ideally one has to average the index across a large number of random splits. A better alternative in this respect is the Intraclass Correlation Coefficient (ICC), which gives a single analytic estimate of the amount of systematic variance in a set of measurements. Overviews are given in Shrout and Fleiss (1979) and McGraw and Wong (1996). ICCs come in different flavors, which makes them more flexible in use, but also gives them a more complicated initial impression than split-half correlations. First, they can estimate either the reliability of
a single measurement (i.e. the reliability of the item scores of one single participant in
an experiment) or the reliability of the average of several ratings (i.e. the reliability of
the item means over all participants). Second, the reliability can be computed for
studies that use nested designs (each item is seen by several participants, one-way
models) and for studies that use crossed designs (each item is seen by all
participants, two-way models). In the latter case, the distinction can be made
between absolute agreement and consistency. Absolute agreement requires two
measurements to be exactly the same to get an ICC of 1. Consistency allows a
constant shift of one of the measurements (e.g. 1, 2, 3 and 2, 3, 4 are not in perfect
agreement but are completely consistent). \(^2\) Table 2 gives an overview of the different

Table 2. Overview of the different types of ICC measures described in McGraw and
Wong (1996). A-type ICCs correspond to type 2 ICCs in Shrout and Fleiss (1979), C-
type ICCs correspond to type 3 ICCs in Shrout and Fleiss.

<table>
<thead>
<tr>
<th>ICC</th>
<th>model</th>
<th>type</th>
<th>unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICC(1)</td>
<td>one-way</td>
<td>agreement</td>
<td>single</td>
</tr>
<tr>
<td>ICC(A,1)</td>
<td>two-way</td>
<td>agreement</td>
<td>single</td>
</tr>
<tr>
<td>ICC(C,1)</td>
<td>two-way</td>
<td>consistency</td>
<td>single</td>
</tr>
<tr>
<td>ICC(k)</td>
<td>one-way</td>
<td>agreement</td>
<td>average</td>
</tr>
<tr>
<td>ICC(A,k)</td>
<td>two-way</td>
<td>agreement</td>
<td>average</td>
</tr>
<tr>
<td>ICC(C,k)</td>
<td>two-way</td>
<td>consistency</td>
<td>average</td>
</tr>
</tbody>
</table>

\(^2\) Notice that consistency is not the same as correlation: 1, 2, 3 and 1.5, 2, 2.5 are perfectly correlated but not consistent.
The use of the various ICCs can be illustrated with the word recognition megastudies that have been published so far. In the first megastudies, different groups of participants saw different files of stimuli. This was true for the English Lexicon Project (Balota et al., 2007) and the French Lexicon Project (Ferrand et al., 2010). For these databases with a nested design, only the one-way model can be used, in which the participant variance cannot be separated from the error variance. The index ICC(1) then gives the reliability of the individual RTs; the index ICC(k) gives the reliability of the mean RTs of the words calculated across the k participants to responded to each word. As a rule, ICC(k) will be higher than ICC(1) because of the averaging across k observations. The distinction between ICC(1) and ICC(k) depends on the goal of the analysis. When the researcher wants to test a regression model on the average word performance, ICC(k) is of importance. When one wants to compare the correlations between different runs of a simulation model each simulating a single participant to those of human performance, it is more appropriate to look at ICC(1) (Seidenberg & Plaut, 1998).

Later megastudies, such as the Dutch Lexicon Project (Keuleers et al., 2010) and the British Lexicon Project (Keuleers et al., 2012), used a crossed design in which the same participants saw (nearly) all stimuli. In such designs it is possible to separate participant variance from noise variance and one can use the measures ICC(C,1) to estimate the reliability of the individual observations and ICC(C,k) to estimate the reliability of the word means calculated across k participants. Shrout & Fleiss (1979) noted that the reliabilities ICC(C,1) and ICC(C,k) tended to be higher than ICC(1) and ICC(k).
In all megastudies it has been found that the reliability of the measures is higher when the RTs are standardized across test blocks, because this takes away practice effects and other noise due to the moment of testing. When the data are expressed as z-scores, the ICCs based on consistency effectively become ICCs based on agreement, as each participant has the same mean (i.e., a z-score of 0).

**Computation of Intraclass Correlation Coefficients**

Because ICCs give a single estimate of measurement reliability (and hence the ratio of specific vs. error variance), we will use them instead of split-half correlations. As it happens, they are also quite easy to calculate both in ANOVAs and random effects models.

One-way ICCs can be expressed in terms of variance components as

\[
\text{ICC}(k) = \frac{\sigma_s^2}{\left(\sigma_s^2 + \sigma_w^2 / k\right)},
\]

where \(\sigma_s^2\) is the stimulus variance, \(k\) the number of observations per stimulus,\(^3\) and \(\sigma_w^2\) the inseparable participant and error variance. Traditionally the variance components are computed using the mean squares of a one-way ANOVA with stimulus as the only factor:

\[
\sigma_s^2 = \frac{bms - wms}{k},
\]

\[
\sigma_w^2 = wms,
\]

where \(bms\) is the between stimulus mean squares and \(wms\) the within stimulus mean squares.

\(^3\) ICC(1) can be derived by setting \(k=1\) in the formulas.
The variance components can also be estimated using a random effects model with a global intercept and a random intercept per stimulus. In that case $\sigma_s^2$ corresponds to the variance of the stimulus random intercept and $\sigma_p^2$ to the error variance.

The two-way ICCs can be expressed in terms of variance components as

$$\text{ICC}(A, k) = \frac{\sigma_s^2}{\left(\sigma_s^2 + (\sigma_p^2 + \sigma_e^2) / k\right)}$$

$$\text{ICC}(C, k) = \frac{\sigma_s^2}{\left(\sigma_s^2 + \sigma_e^2 / k\right)},$$

where $\sigma_s^2$ is the stimulus variance, $k$ the number of observations per stimulus, $\sigma_p^2$ the participant variance and $\sigma_e^2$ the error variance. Here, the variance components can be computed from the mean squares of a stimulus x participant two-way ANOVA:

$$\sigma_s^2 = \frac{bms - ems}{k}$$

$$\sigma_p^2 = \frac{jms - ems}{n}$$

$$\sigma_e^2 = ems$$

where $bms$ is the between stimulus mean squares, $jms$ the between participant mean squares, $ems$ the error mean squares and $n$ the number of stimuli. For instance, applied to Table 1, $bms = 61.80$, $jms = 84.88$, and $ems = 20.98$, as can be seen in the ANOVA table produced by any good statistical software package.

The two-way ICCs can also be obtained from a random effects model with three intercepts: a global intercept, a random intercept per stimulus and a random intercept per participant. In that case $\sigma_s^2$ corresponds to the variance of the stimulus random effect, $\sigma_p^2$ corresponds to the variance of the participant random effect, and $\sigma_e^2$ corresponds to the error variance. The output of the lmer algorithm discussed above gives us the values of $\sigma_s^2 = 6.80$, $\sigma_p^2 = 12.78$, and $\sigma_e^2 = 20.98$. 
McGraw and Wong (1996) provide formulas for confidence intervals around the ICC and for $F$ tests comparing the ICC to a user-specified constant. In McGraw and Wong the formulas are specified in terms of mean squares, but they can easily be translated in terms of variance components, so that they apply to both ANOVA based ICCs and random effects based ICCs.

**Comparing ICCs based on ANOVAs and random effects models**

Whether ICCs are based on ANOVA or a random effects model makes no difference, as long as the data set is complete. Things may be different, however, when data are missing. Within the existing ANOVA method, the results depend on the implementation used. The ICC function from the R package `psych` (Revelle, 2013), for example, simply refuses to do any computations on a data matrix with missing values. The ICC function from the R package `irr` (Gamer, Lemon, Fellows & Singh, 2012), on the other hand, does a complete-case analysis (i.e., it will silently remove all items with missing values from the analysis).

Rather than trying to find out how each and every existing function works, we decided to program two new ANOVA methods in R that do not revert to a complete case analysis. The first implementation uses vectorized functions in R to compute the grand means, stimulus means and participant means from which the relevant sums of squares are computed. The second implementation mirrors the behavior of the algorithm provided by Courrieu and Rey (2011). It loops over all stimuli and participants, and computes the grand mean, stimulus mean and participant mean on
the fly. On full datasets both methods give the same result, but in R the former method is about 25 times faster.

Because the analyses we will report correspond largely to those reported by Courrieu and Rey (2011), we calculated split-half reliabilities in addition to the ICCs, as this was the measure used by Courrieu and Rey. For each dataset, we computed the split-half reliability for 100 random splits of the data. The average of the 100 replications was taken as the estimate of the reliability and the 2.5% and 97.5% percentiles formed the lower and upper bound of the 95% confidence interval. All split-half estimates were corrected for sample size using the Spearman-Brown prophecy formula. The correction for the proportion of missing data proposed by Courrieu and Rey was also computed.

The ICCs were also calculated on the basis of a random effects model. In such a model the variance components are estimated on the basis of likelihood methods rather than sums of squares (as in ANOVA). A priori this should be a better method, as likelihood based estimates are not biased by randomly missing data, whereas estimates based on sums of squares are (Schafer & Graham, 2002). Dempster, Laird and Rubin (1977) developed the Expectation Maximization (EM) algorithm that is generally applicable to estimation in the presence of missing data or latent variables. This procedure iterates between estimating the parameters based on the observed data and a current best guess of the missing data, and estimating the missing data given the current value of the parameters (actually, we do not need the missing values per se, only their sufficient statistics - see Little and Rubin 1977 for further

---

4 Calculations with 1,000 random splits per computation would have given more accurate estimates, but the split-half reliabilities now already took much more time to calculate than the ICCs.
elaboration). This process is repeated until convergence is reached and it should result in estimates that are not substantially biased (Collins, Schafer & Kam, 2001; Schafer & Graham, 2002).

Simulation 1: Behavior of Different ICC Estimates in the Presence of Missing Data

Method. An artificial dataset with observations from 40 participants and 1000 items was generated. The number of participants was similar to a typical lexicon project, but the number of items was lower to make the simulations feasible (please note that the number of participants influences k-type ICCs, the number of items does not). The data were generated according to the model

\[ x_{ij} = \mu + s_i + p_j + e_{ij}, \]

where \( \mu \) is the mean value of \( x \) (500), \( s_i \) is the effect of stimulus \( i \) (randomly distributed around zero with a standard deviation \( \sigma_s \) of 10), \( p_j \) is the effect of participant \( j \) (randomly distributed around zero with a standard deviation \( \sigma_p \) of 50) and \( e_{ij} \) is the residual (randomly distributed around zero with a standard deviation \( \sigma_e \) of 50).

In the full dataset, ICC(C,k=40) was 0.579. We then removed between 0 and 30% of the data at random (in 1% increments) and computed ICC(C,k) (1) after case deletion, (2) using the vectorized ANOVA method, (3) using the loop-based ANOVA method, and (4) using the random effects method. This was compared to split-half correlations without (5) and with (6) correction for the proportion of missingness, as proposed by Courrieu and Rey (2011) for this type of datasets. The whole process was replicated 100 times.
**Variables.** The behavior of the ICC(C,k) statistic was evaluated using the classical criteria of bias, consistency, mean square error (MSE) and coverage (Schafer & Graham, 2002). Bias is defined as the difference between the true value of the statistic and the average estimate of the statistic (here, we take the ICC of the complete data as the true value). Consistency is measured by the variance of the estimate around its mean. MSE is a measure of both bias and consistency: it is the average value of the squared difference between the true value of the statistic and the estimate of the statistic over all samples. Coverage is a measure of the quality of the confidence interval (CI) generated by a method. A good 95% Confidence Interval (CI) should contain the true value in about 95% of the replications.\(^5\) A good method should yield an ICC close to the ICC based on the entire dataset with a proper CI.

In the following we will focus on ICC(C,k), as this corresponds to the mean split-half reliability over all possible splits and to Cronbach’s \(\alpha\). Biases in the other ICC indices should be similar, as they are functions of the same variance components that are estimated.

**Results of the case deletion method.** A first inspection of the data revealed that the case deletion method could only handle very small amounts of missing data. If the percentage of missing values exceeded 10%, the ICC could not be computed for some of the data sets. If the percentage of missing values exceeded 20%, the ICC could not be computed for any of the data sets (See Figure 1). Because case

---

\(^5\) It might be objected that in our simulations we do not draw randomly from a known population, but we remove data randomly from a single sample of that population. This means we cannot test whether coverage of the method on complete data is adequate (but we do know that on complete data the different ANOVA implementations and the random effects method give exactly the same result). What we can do is look at how the size of the confidence intervals grows or shrinks with increasing levels of missingness.
deletion performed so badly, the results of this method are plotted separately in Figure 2. For the percentages of missing values over which it could be calculated, the estimate is not biased (i.e., the median estimate is close to the real value of 0.579), but the variance of the estimate increases rapidly as the percentage of missing observations accrues.

Figure 1: Proportion of data sets for which the ICC could be computed after case deletion as a function of the proportion of missing values.

Figure 2: Boxplots of estimated ICCs after case deletion as a function of the proportion of missing values. Notice that the estimated ICC values are not biased (i.e., lie around the value for the complete dataset), but are rather variable, in particular when the percentage of missing values is above 5%. Also notice the rather large number of outliers (black dots).
Results of other methods to deal with missing values. The other methods all produced valid estimates of the ICC in all runs of the simulation. We present the outcome of the analyses in terms of bias, consistency, RMSD and coverage. Figure 3 shows the bias, variance and MSE of the different methods to calculate ICC.

Figure 3: Quality of the various point estimates of ICC as a function of the proportion of missing values. Anova 1 = the vectorized ANOVA method to estimate missing values, Anova 2 = the loop-based ANOVA method, lme = the random effects method, split-half = the analysis as recommended by Courrieu and Rey (2011), corrected split-half = the analysis with a correction factor as recommended by Courrieu and Rey.
When we look at the bias measure, it is immediately clear that the random effects method produces the required reliability estimate independent of the percentage of missing data (at least up to 30% missing data). The split-half correlation underestimate the reliability of the data when data are missing, as reported by Courrieu and Rey (2011), but the correction formula they proposed does not seem to be strong enough: Even after correction some negative bias remains. The reliability estimates based on the ANOVA methods are the worst, because they not only overestimate the ICC of the dataset (going as high as .9 where .6 is the true value),
but they additionally result in underestimated CIs (Figure 4) suggesting that the estimates are less variable than they really are. For the split-half methods the confidence intervals become wider. These methods seem to lose power, which is natural when we have less data. Again, the random effects method is the only one that seems to be robust against the percentage of missing data.

For all methods, the variance of the estimate seems to be largely unaffected by missingness, (at least in our simulation, there is no guarantee that this would not change if we used less stimuli or participants). The bottom panel of the plot shows the MSE, which is a compound measure of (absolute valued) bias and variance.

Figure 4: Size of the 95% confidence interval around ICC as a function of the proportion of missing values.

Conclusions simulation. The results of our simulation can be summarized as follows. First, the behavior of ANOVA based ICC estimates depends on the implementation. One implementation (case deletion) results in reasonable estimates,
but is only useful when the percentage of missing values does not exceed 5%.

Another implementation (the vectorized ANOVA method) showed a strong positive bias and over-confidence. The loop-based method, proposed by Courrieu and Rey (2011) was the best, but even then performed much worse than the random effects model. Second, the split-half reliability measure showed the negative bias predicted by Courrieu and Rey, but their correction formula was not strong enough to overcome the problem. Moreover, the correction formula should not be applied to any of the other methods, as none of them showed a negative bias.

The most positive outcome of the simulation was that the random effects based ICC was not biased when the missing data were randomly distributed. This indicates that the random effects model can be used to impute missing values. The parameter estimates provided by the model remain stable up to 30% missing observations. So, in the next section we develop an imputation scheme based on the same random effects model as the one used to estimate the ICC. We will then additionally show that this imputation model does not bias the ICC.

**Imputation Based on the Random Effects Model**

As described in an earlier section, the variance components on which the ICC estimates are based can be estimated using a random effects model with a fixed intercept, a random intercept per stimulus and a random intercept per item. The model can be written as

\[ x_{ij} = \mu + s_i + p_j + e_{ij} \]
where $\mu$ is the mean value of $x$, $s_i$ is the effect of stimulus $i$ (normally distributed with mean 0 and variance $\sigma^2_s$), $p_j$ is the effect of participant $j$ (normally distributed with mean 0 and variance $\sigma^2_p$) and $e_{ij}$ is the residual (normally distributed with mean 0 and variance $\sigma^2_e$). The parameters of this model are the global intercept and the three variances mentioned above. However, the fitted model also contains estimates of each stimulus and participant effect $s_i(i=1..n)$ and $p_j(j=1..k)$, as we showed earlier when we used the random effects model to estimate the missing value of the toy dataset.

For the observed data, we know all four numbers on the right side of the equation: the estimates of the global intercept, stimulus intercept, participant intercept and the error term (the difference between the observed and the fitted value). For the missing data, we only miss the error term. The imputation scheme we propose is very simple: we impute each missing value $x_{ij}$ with the sum of the estimated global intercept $\mu$, the estimated stimulus intercept $s_i$, the estimated participant intercept $p_j$ and a random number from a normal distribution$^6$ with mean 0 and variance $\sigma^2_e$. Applied to the toy corpus, the missing value of P3 S3 would be any random value from the normal distribution with mean $= 24.56$ and variance $= 20.98$.

This imputation scheme does not alter the expected value of the stimuli or the participants (the error we add is drawn from a normal distribution with mean 0, so changes to the mean of each stimulus or participant cancel each other out over repeated sampling). It should also not influence the ICC of the data: ICC is a function

$^6$ The imputation algorithm can easily be adjusted for skewed distributions by sampling noise values $e_{ij}$ from the observed (and skewed) residuals instead of from the estimated normal distribution of the residuals.
of $\sigma_s^2$, $\sigma_p^2$ and $\sigma_e^2$ and the imputation scheme does not alter any of these. In the next section we describe a simulation study to verify this prediction.

**Simulation 2: Behavior of different ICC estimates after data imputation**

To each data set from simulation 1, we applied the imputation scheme described above and traced the ICC estimates in the same way as in simulation 1.

Figure 5 shows the bias, variance and MSE after imputation. All ICC methods gave the same estimates after imputation. This is how it should be, as all methods yielded the same estimates for the complete dataset. There is also no increase in bias or variance as a function of the proportion of initially missing values. Only the corrected split-half method shows a positive bias. Again this is normal: the correction tried to compensate for missing values. But after imputation there are no missing values anymore; so, there is no need compensate twice for missing values.

Figure 6 shows that not only the ICC values restore to normal after imputation of the missing observation, but also the confidence intervals around the ICCs. Again, the results for the ANOVA based and random effects based ICCs are identical. The confidence intervals around the split-half correlations are somewhat smaller, but of the same order and only slightly influenced by the proportion of initially missing data.

Figure 5: Quality of the ICC estimates after imputation of the missing values with our random effects model as a function of the proportion of initially missing values.
Figure 6: Size of the 95% confidence interval around ICC after imputation as a function of the proportion of initially missing values.
Application to the lexicon projects

Now that we have good ways to estimate ICC and impute missing data, the procedures can be applied to the various existing megastudy datasets. In particular, we applied the methods to the Dutch Lexicon Project (Keuleers et al., 2010), the British Lexicon Project (Keuleers et al., 2012), the English Lexicon Project (Balota et al., 2007), and the French Lexicon Project (Ferrand et al., 2010). We started from the raw data of each project and preprocessed all projects in the same way. First, RTs from error trials were set to missing. Items with an average accuracy below 65 percent were then removed (when there are 40 responses per word, a percentage correct that is smaller than 65% does not differ significantly from chance). Next we applied hard RT cutoffs of 200 and 2000 ms. Further outliers were deleted per block per participant, using a boxplot rule that is adjusted for skewed distributions (Hubert and Vandervieren, 2004). In the Dutch and British Lexicon Projects blocks comprised 500 trials, in the English and French Lexicon Projects blocks comprised 250 trials. The number of items and observations in each lexicon project is listed in Table 2.
Table 2: Design variables of the lexicon projects. The British Lexicon Project consisted of two lists. These were analyzed separately and indicated by “a” and “b”.

In the elp there were 815 participants and 3375 trials per participant, in the flp there were 974 participants and 2000 trials per participant.

<table>
<thead>
<tr>
<th>Words</th>
<th>Nonwords</th>
<th>Ratings/word</th>
<th>Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>blp a</td>
<td>14365</td>
<td>14337</td>
<td>40</td>
</tr>
<tr>
<td>blp b</td>
<td>14365</td>
<td>14343</td>
<td>38</td>
</tr>
<tr>
<td>dlp</td>
<td>14089</td>
<td>14089</td>
<td>39</td>
</tr>
<tr>
<td>elp</td>
<td>40481</td>
<td>40481</td>
<td>34</td>
</tr>
<tr>
<td>flp</td>
<td>38450</td>
<td>39363</td>
<td>25</td>
</tr>
</tbody>
</table>

After preprocessing we computed z-scores for RT and for Processing Rate (1/RT) per block per participant. The advantage of Processing Rate is that it is typically less skewed than RT, while it affords an interpretation of effects in terms of speed instead of time (Kliegl, Masson & Richter, 2010). For both variables, we computed imputed scores and ICCs. The ICCs are listed in Table 3. For the Dutch and British lexicon projects the ICCs and imputation were based on the two-way model (with a random effect for item and for participant), for the English and French lexicon projects the ICCs and imputation were based on the one-way model (with only a random effect for item).
Table 3: ICCs and 95% confidence intervals for the Dutch, British, English and French Lexicon Projects. The ICCs for the imputed scores are not listed, as they were identical to the standard ICCs up to at least three decimal positions.

<table>
<thead>
<tr>
<th>database</th>
<th>ICC</th>
<th>lexicality</th>
<th>RT</th>
<th>rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>blp a</td>
<td>ICC(C,1) words</td>
<td>0.142 [0.138, 0.145]</td>
<td>0.177 [0.173, 0.181]</td>
<td></td>
</tr>
<tr>
<td>blp b</td>
<td>ICC(C,1) words</td>
<td>0.156 [0.152, 0.159]</td>
<td>0.193 [0.189, 0.197]</td>
<td></td>
</tr>
<tr>
<td>dlp</td>
<td>ICC(C,1) words</td>
<td>0.143 [0.139, 0.146]</td>
<td>0.168 [0.165, 0.172]</td>
<td></td>
</tr>
<tr>
<td>elp</td>
<td>ICC(1) words</td>
<td>0.244 [0.242, 0.247]</td>
<td>0.270 [0.267, 0.273]</td>
<td></td>
</tr>
<tr>
<td>flp</td>
<td>ICC(1) words</td>
<td>0.193 [0.190, 0.195]</td>
<td>0.215 [0.212, 0.217]</td>
<td></td>
</tr>
<tr>
<td>blp a</td>
<td>ICC(C,k) words</td>
<td>0.868 [0.865, 0.871]</td>
<td>0.907 [0.893, 0.898]</td>
<td></td>
</tr>
<tr>
<td>blp b</td>
<td>ICC(C,k) words</td>
<td>0.875 [0.872, 0.878]</td>
<td>0.901 [0.899, 0.903]</td>
<td></td>
</tr>
<tr>
<td>dlp</td>
<td>ICC(C,k) words</td>
<td>0.866 [0.868, 0.870]</td>
<td>0.887 [0.885, 0.890]</td>
<td></td>
</tr>
<tr>
<td>elp</td>
<td>ICC(k) words</td>
<td>0.917 [0.915, 0.918]</td>
<td>0.926 [0.925, 0.927]</td>
<td></td>
</tr>
<tr>
<td>flp</td>
<td>ICC(k) words</td>
<td>0.857 [0.855, 0.859]</td>
<td>0.872 [0.871, 0.874]</td>
<td></td>
</tr>
<tr>
<td>blp a</td>
<td>ICC(C,1) nonwords</td>
<td>0.117 [0.114, 0.120]</td>
<td>0.140 [0.136, 0.143]</td>
<td></td>
</tr>
<tr>
<td>blp b</td>
<td>ICC(C,1) nonwords</td>
<td>0.141 [0.138, 0.144]</td>
<td>0.163 [0.160, 0.167]</td>
<td></td>
</tr>
<tr>
<td>dlp</td>
<td>ICC(C,1) nonwords</td>
<td>0.134 [0.131, 0.138]</td>
<td>0.152 [0.149, 0.156]</td>
<td></td>
</tr>
<tr>
<td>elp</td>
<td>ICC(1) nonwords</td>
<td>0.160 [0.158, 0.162]</td>
<td>0.167 [0.165, 0.169]</td>
<td></td>
</tr>
<tr>
<td>flp</td>
<td>ICC(1) nonwords</td>
<td>0.133 [0.131, 0.135]</td>
<td>0.146 [0.144, 0.148]</td>
<td></td>
</tr>
<tr>
<td>blp a</td>
<td>ICC(C,k) nonwords</td>
<td>0.841 [0.837, 0.845]</td>
<td>0.867 [0.863, 0.870]</td>
<td></td>
</tr>
<tr>
<td>blp b</td>
<td>ICC(C,k) nonwords</td>
<td>0.862 [0.859, 0.865]</td>
<td>0.881 [0.879, 0.884]</td>
<td></td>
</tr>
<tr>
<td>dlp</td>
<td>ICC(C,k) nonwords</td>
<td>0.858 [0.855, 0.862]</td>
<td>0.875 [0.872, 0.878]</td>
<td></td>
</tr>
<tr>
<td>elp</td>
<td>ICC(k) nonwords</td>
<td>0.866 [0.864, 0.868]</td>
<td>0.872 [0.870, 0.874]</td>
<td></td>
</tr>
<tr>
<td>flp</td>
<td>ICC(k) nonwords</td>
<td>0.794 [0.791, 0.796]</td>
<td>0.810 [0.808, 0.813]</td>
<td></td>
</tr>
</tbody>
</table>

Datasets with the imputed data are made available via the authors' website (http://crr.ugent.be/vinci). We strongly recommend readers to use these for further analyses of the megastudy data. The trial datafiles contain data at the trial level and include the following fields:

- participant: participant index.
- list (BLP only): list index.
- block: block index.
- subblock: subblock index.
- trial_in_subblock: trial in subblock index (BLP and DLP only).
• trial_in_block: trial in block index.
• trial: trial index.
• observation: observation index, counts from 1 to the total number of observations in the respective lexicon project.
• warmup (DLP only): warmup block (1) or not (0) - block 1 is warmup.
• repetition (DLP only): repetition block (1) or not (0) - block 50 is a repetition of block
• item (factor): item index.
• spelling (character): item spelling.
• lexicality (factor): word (W) or nonword (N).
• r (DLP and BLP only): meaning of the response: word (W) or nonword (N).
• corr: correctness of the response, yes (Y) or no (N).
• corrn: correctness of the response, yes (1) or no (0).
• rate: response rate (n of responses per second).
• rt: rt in milliseconds.
• one: column of ones (1).
• is_missing: was the RT missing due to computer error?
• is_error: was the RT set missing because of a response error?
• is_bad item: was the RT set missing because the item score was below 65%?
• is_lt_200: was the RT less than 200ms?
• is_gt_2000: was the RT greater than 2000ms?
• is_lt_lower: was the RT less than the lower limit for this block?
• is_gt_upper: was the RT greater than the upper limit for this block?
• is_imputed: was the RT observed or imputed?
• lower: lower RT limit for this block.
• upper: upper RT limit for this block.
• rtR, rateR: raw RT and rate.
• rtC, rateC: cleaned RT and rate (bad RTs set to missing).
• rtI, rateI: imputed RT and rate.
• zrtC, zrateC: z-transformed cleaned RT and rate.
• zrtI, zrateI: z-transformed imputed RT and rate.

The item data file contains averaged data per word (and nonword) and has the following fields:

• item: item index.
• spelling: item spelling.
• lexicality: word (W) or nonword (N).
• nobs: number of observations for this item.
• acc.mean, acc.sd: mean and standard deviation of the item accuracy.
• rtC.mean, rtC.sd: mean and standard deviation of the cleaned RT (i.e., after the outlier detection, as described in the present ms).
• rateC.mean, rateC.sd: mean and standard deviation of the cleaned rate.
• zrtC.mean, zrtC.sd: mean and standard deviation of the z-transformed cleaned RT.
• zrateC.mean, zrateC.sd: mean and standard deviation of the z-transformed cleaned rate.
• plus the same for the imputed values. These variables carry the suffix “I” instead of the suffix “C”.
Conclusion

We have described how a random effects model analysis can be used to impute missing observations in megastudy datasets. This imputation is remarkably simple and turns out to be superior to existing methods (based on ANOVA tables). It also leads to a correct estimate of the reliability of the measures in the dataset, independent of the percentage of missing values (up to 30%). We have applied the method to the existing datasets, so that researchers now have access to full data matrices.
References


Bates, D., Maechler, M., & Bolker B. (2013). Lme4: Linear mixed-effects models using S4 classes. R package version 0.999999-2. http://CRAN.R-project.org/package=Lme4


