Psychonomic Bulletin & Review
2009, 16 (3), 600-608
doi:10.3758/PBR.16.3.600

Item performance in visual word recognition

Arnaud Rey and Pierre Courrieu
CNRS and Université de Provence, Marseille, France

Florian Schmidt-Weigand
Universität Kassel, Kassel, Germany

and

Arthur M. Jacobs
Freie Universität Berlin, Berlin, Germany

Standard factorial designs in psycholinguistics have been complemented recently by large-scale databases providing empirical constraints at the level of item performance. At the same time, the development of precise computational architectures has led modelers to compare item-level performance with item-level predictions. It has been suggested, however, that item performance includes a large amount of undesirable error variance that should be quantified to determine the amount of reproducible variance that models should account for. In the present study, we provide a simple and tractable statistical analysis of this issue. We also report practical solutions for estimating the amount of reproducible variance for any database that conforms to the additive decomposition of the variance. A new empirical database consisting of the word identification times of 140 participants on 120 words is then used to test these practical solutions. Finally, we show that increases in the amount of reproducible variance are accompanied by the detection of new sources of variance.

The precision of theoretical accounts in the field of visual word recognition has significantly increased over recent years. Indeed, cognitive modelers have proposed several detailed descriptions of the structure and dynamics of the reading system (e.g., Ans, Carbonnel, & Valdois, 1998; Coltheart, Rastle, Perry, Langdon, & Ziegler, 2001; Grainger & Jacobs, 1996; Harm & Seidenberg, 2004; Perry, Ziegler, & Zorzi, 2007; Plaut, McClelland, Seidenberg, & Patterson, 1996; Seidenberg & McClelland, 1989). The fine-grained precision of these models has led to the development of so-called *computational models of reading* that can generate precise quantitative predictions. As a consequence, by making fine-grained assumptions about the cognitive architecture of visual word recognition, theorists have also remarkably increased the resolution of theoretical predictions.

This progress in theory has been accompanied by a corresponding gain of precision for empirical data. In a seminal study, Spieler and Balota (1997) asked 31 participants to read aloud a list of 2,870 English monosyllabic words and compared the mean naming latency for each item with the predictions of two computational models of word reading (i.e., Plaut et al., 1996; Seidenberg & McClelland, 1989). The results were somewhat surprising, since both of these models accounted for only a small amount of the item variance (3.3% for Plaut et al.'s model, 10.1% for Seidenberg and McClelland's). Spieler and Balota also noticed that the models explained the amount of variance less well than did the linear combination of three simple linguistic predictors: log frequency, word length, and neighborhood density (which accounted for 21.7% of the variance). Finally, when variables related to onset phonemes were added to the analysis, the simple predictors were able to account for 42% of the item variance. Item-level data therefore seem to provide a critical test for computational models of reading.

Seidenberg and Plaut (1998) claimed, however, that two reasons might explain the relatively low item variance accounted for by these models. First, item means are affected by several factors that are not addressed by these models. For example, they do not specify the processes involved in letter recognition or in the production of articulatory output. Balota and Spieler (1998) noticed, however, that the performance of these models remains surprisingly weak, since they fail to explain more variance than do three simple predictors (i.e., log frequency, word length, and neighborhood density) that are, in principle, captured by these models. Their second, and probably more critical, argument is based on the fact that item data include a substantial amount of error variance. The question is how substantial this amount of error variance is. Comparing Spieler and Balota's database with a similar database recorded by Seidenberg and Waters (1989),[1] they found a .54 correlation between item latencies in the two databases. This relatively low correlation indicates that a large amount of the variance in one database is absent from the other.

In the present study, in line with Seidenberg and Plaut's (1998) criticism, we address the issue of error variance in item databases (for a similar approach, see Rouder & Lu, 2005). More specifically, we propose practical solutions to estimate the amount of error variance as a function of the number of participants. Increasing the number of participants obviously decreases the amount of error variance (related to noise) while preserving the amount of reproducible variance (related to items). This outcome might appear trivial, but, paradoxically, none of the existing databases has seriously considered this issue.

In the next sections, we first provide a simple analysis of this problem. Second, we present a new empirical database consisting of the word identification scores of 140 participants on 120 words, and we use it to quantitatively estimate the amount of variance that should be accounted for as a function of the number of participants. Then, we propose a method to estimate the amount of reproducible variance from any database, and we give an

A. Rey, arnaud.rey@univ-provence.fr

example. Finally, we show that an increase in reproducible variance is accompanied by the detection of new sources of variance.

Problem Analysis

Let I be a population of items, let P be a population of participants, and let x be an experimental measure (e.g., response time) on $I \times P$. The usual additive decomposition model is

$$x = \mu + \alpha + \beta + \varepsilon, \tag{1.0}$$

where μ is the mean value of x on $I \times P$ and α, β, and ε are three independent random variables with mean 0 and variance σ_α^2, σ_β^2, and σ_ε^2, respectively.[2] The variable α is the *participant effect*, which takes a constant value for each given participant; β is the *item effect*, which takes a constant value for each given item; ε is considered as random noise. It is clear that the variable β, whose values characterize the items, is the variable of interest in this study.

One can derive from x another measure, denoted $x^{(n)}$, which is the arithmetic mean of x over n randomly selected distinct participants, and then obtain from Equation 1.0 the following decomposition:

$$x^{(n)} = \mu + \alpha^{(n)} + \beta + \varepsilon^{(n)}, \tag{1.1}$$

where the random variables $\alpha^{(n)}$, β, and $\varepsilon^{(n)}$ are always independent with 0 means, but their variances are now σ_α^2/n, σ_β^2, and σ_ε^2/n, respectively. When n increases, the contributions of $\alpha^{(n)}$ and $\varepsilon^{(n)}$ to the variance of $x^{(n)}$ clearly decrease. These reductions in the amounts of variance related to participants and noise lead to an increase in the amount of reproducible variance related to items.

One way to estimate the evolution of reproducible variance as a function of the number of participants is to compare two independent realizations of $x^{(n)}$. The amount of variance that is common to these two groups provides a good estimate of the total amount of reproducible variance, and it can be estimated by the squared correlation between two independent groups of n participants. Starting from Equation 1.1, a simple derivation, which is stated in Appendix A, leads to the following equations, which relate the population correlation coefficient r to the number of participants n and to the ratio between σ_β^2 and σ_ε^2.

To simplify the notation, one can define the ratio

$$q = \frac{\sigma_\beta^2}{\sigma_\varepsilon^2}, \tag{2.0}$$

so that the correlation between two independent realizations of $x^{(n)}$ is

$$\rho = \frac{nq}{nq + 1}, \tag{2.1}$$

which implies that

$$q = \frac{\rho}{n(1 - \rho)} \tag{2.2}$$

and also that

$$n = \frac{\rho}{q(1 - \rho)}. \tag{2.3}$$

Clearly, given any two of the three quantities ρ, q, and n, one can easily find the third.

In practice, one does not know the population parameters (ρ or q), and one must estimate at least one of them from a finite sample of the measure x on a sample of m items by n participants. As we shall see below, in the sections on estimation with large and small samples, the sample of x can be used to estimate ρ by means of Pearson's r correlation statistic.[3] Before this discussion, we present the experimental data that will be used to calculate the estimates.

The Database

The present database has two primary characteristics. First, it was collected using a standard perceptual identification task, the luminance-increasing paradigm (Rey, Jacobs, Schmidt-Weigand, & Ziegler, 1998; Rey & Schiller, 2005). As in most perceptual identification paradigms, participants generate a simple motor response as soon as they have identified a target stimulus. This experimental procedure therefore simplifies the model-to-data connection, since it can be assumed that word identification times can be directly compared with word identification latencies in a localist connectionist model like that of Grainger and Jacobs (1996). Second, 140 participants were recorded in this experiment. This large number makes it possible to estimate the amount of error variance in item mean latencies by comparing independent groups of participants consisting of 20, 30, . . . , up to 70 participants.

Participants. One hundred forty-four undergraduate students at Arizona State University participated in the experiment in partial fulfillment of a course requirement.[4] All of them were native English speakers and had normal or corrected-to-normal vision.

Stimuli. The words used in the experiment were a random sample of 120 monosyllabic, five-letter English words taken from a list of all monosyllabic five-letter words reported in the CELEX lexical database (Baayen, Piepenbrock, & van Rijn, 1993). The random selection was applied to provide a representative distribution for a maximum number of statistical word features.

Procedure. The experiment was run on an IBM PC 486 DX2 computer.[5] The stimulus words were typed in lowercase using letters created from table zero of the computer BIOS, in which each letter is defined in an 8 × 14 pixel matrix. To obtain a progressive increase in bottom-up information, the screen contrast was set to its maximum value. The background therefore was as dark as possible, and the stimulus luminance was as bright as possible. The experiment was done in a dark room lit only by a lamp placed behind the participants, to keep the keyboard visible without causing reflections on the screen.

The participants were seated 50 cm in front of the computer screen. The experiment started with a training session in which 6 of the 12 training items were presented. Data recording began with the 6 remaining training items, and, without transition, the 120 experimental trials were presented in a randomized order for each participant. Each trial began with a 1-sec presentation of a fixation mark ("+") in the center of the screen, which was replaced immediately by the target word. However, the target word

was initially written in black, just like the background, and so remained invisible to the participants. By incrementing the value of one of the RGB (red, green, blue) counters every 100 msec, the luminance of the target word increased progressively. Every counter was initially set to 0. After 100 msec, the red counter was set to 1 (with the green and blue counters remaining at 0). After another cycle (i.e., 200 msec after stimulus presentation), RGB was set to 1–1–0, then to 1–1–1 after three cycles, to 2–1–1 after four, and so forth. As soon as the participants could identify the target word, they interrupted the luminance-increasing process by pressing the space bar. Immediately after this response, the item was replaced by a pattern mask #####, which contained two more # characters than there were letters in the target word. Finally, the participants had to type in the word they had seen and press "return" to start the next trial. The screen remained black for 500 msec before the fixation point appeared again. For each trial, the response time was recorded as the time between the onset of the luminance-increasing procedure and the pressing of the space bar. The participants were told to concentrate on accuracy rather than on speed. Each experimental pass lasted about 25 min.

Results. After correcting obvious typing errors, 467 errors (2.7% of the data) remained. Four participants produced more than 10% errors and were excluded for this reason from further analysis. A trimming procedure excluded response times more than three standard deviations above and below a participant's mean (0.9% of the data). The resulting database was composed of 120 (words) $\times$ 140 (participants) word identification times, which included about 4% missing data.

Estimating the Amount of Reproducible Variance

Using this database of 120 items $\times$ 140 participants, it is now possible to estimate parameter q from Equation 2.1 by conducting a Monte Carlo study in the following way. Among the 140 participants, we randomly selected two independent, equally sized groups. Item means for each group were calculated, and correlation coefficients (r) were computed between the two groups on these item means. This procedure was repeated 1,000 times and for various group sizes (i.e., for 1, 5, 10, 15, . . . , 70 participants per group) to generate distributions of r (i.e., 1,000 correlations for each group size).

To test model validity, we computed the q value (Equation 2.1) that minimizes the standard prediction error of the observed correlations. One can easily obtain a first approximation of q—say, q_0—by applying Equation 2.2 (in which one replaces ρ with r) to the mean correlation observed for each group size, and then averaging all resulting q values. In the present case, we obtained $q_0 = 0.0618$. This first approximation was used as the starting point for a local search (MATLAB fminsearch procedure) to minimize the standard prediction error (i.e., the square root of the mean quadratic error). The obtained result was $q = 0.0607$, providing a standard prediction error of 0.0044. Figure 1 shows the mean observed correlations (with standard deviations) and the correlations predicted by Equation 2.1 (using the q value above) as functions of the number of participants per group. The result is that the predicted values are practically indistinguishable from the observed ones.

Using the q value above with Equations 2.1 and 2.3, one can now calculate the amount of reproducible variance obtained with a database composed of n participants, or the number of participants who are engaged in an experiment, in order to obtain a given amount of reproducible variance. For instance, using Equation 2.1 with $q = 0.0607$ and $n = 140$, one obtains $r = .89$, which means that by averaging the data of the 140 participants, one obtains an item data vector with 80% reproducible variance. Similarly, if one desires 90% reproducible variance, the corresponding r is $\sqrt{0.9} = .9487$, and via Equation 2.3, one finds that about 304 participants would be required.

Now, it is clear that another experiment, using a different task and different experimental conditions, would probably provide a different value for q. However, we can say that any experimental variable that conforms to the additive decomposition model (Equation 1.0) necessarily conforms to Equations 2.1–2.3, with an appropriate q value.[6]

Practical Method for Small Samples of Participants

The method used in the previous section is suitable for large samples of participants. However, experimenters commonly use participant samples of only 20–40. Thus, there is clearly a need to develop practical methods to estimate the percentage of variance that is reproducible when the number of participants is not large.

The proposed solution is similar to the one adopted in the section above, and it uses a Monte Carlo approach, which has the advantage of being distribution-free, thus avoiding the need for unverifiable hypotheses concerning the Gaussian nature of the variables. The principle used here is *permutation resampling* (Good, 1994; Opdyke, 2003). We describe this method hereafter and, in Appendix B, show an implementation in MATLAB code that is easy to use in practice. We then provide a model for the implementation details.

Given a data table of m items $\times$ n participants, first choose a group size n_g that is the greatest integer such that $n_g \leq n/2$. Then, randomly sample two independent groups of n_g participants, calculate item means for each group, and compute the correlation coefficients r between the resulting item means. When this has been repeated T times, one can sort the obtained r values in increasing order and easily find in this array the limits for any chosen confidence level. However, the obtained r estimates concern samples of n_g participants. To obtain the corresponding estimates for the whole sample of n participants, one can compute the q values corresponding to the obtained r values by using Equation 2.2, with n_g as the sample size parameter, and then use Equation 2.1 to compute the r values corresponding to the q values with sample size n. As for the choice of T, $T > 1{,}900$ provides precise enough estimates for most applications (Opdyke, 2003), so one can use $T = 2{,}000$.

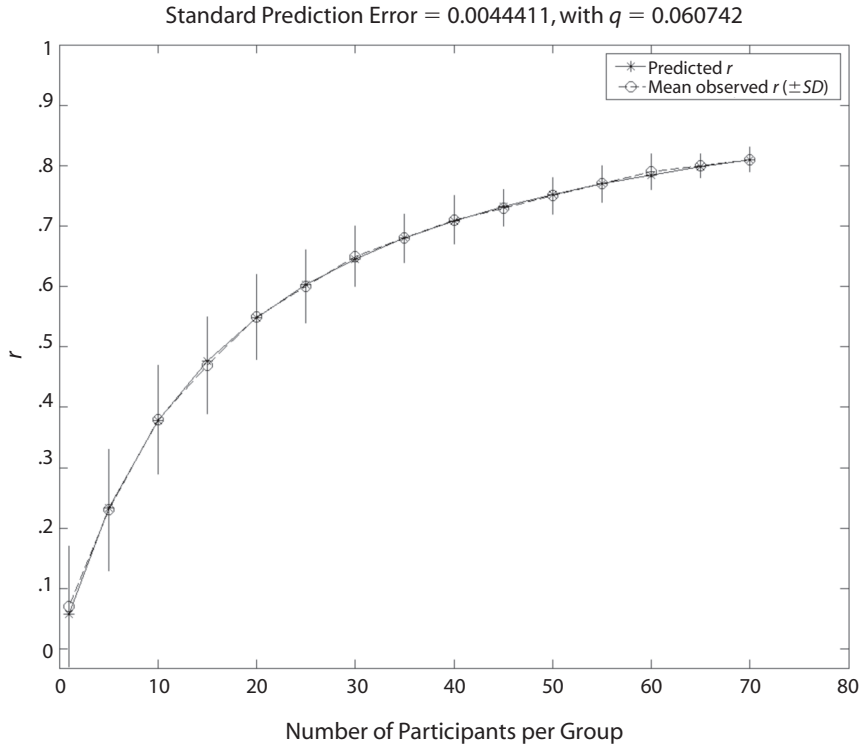

Figure 1. Means (with error bars for standard deviations) of the observed correlation coefficient distributions, with the predicted correlations (Equation 2.1), as a function of the number of participants per group.

To test the above method, we randomly selected from our database 14 subsamples whose size (n) varied from 10 to 140 participants (in steps of 10). The permutation resampling procedure (the function permuqr listed in Appendix B) was applied to each subsample, using a 95% confidence interval. Figure 2 shows the obtained mean r values, with confidence limits, as a function of the number of selected participants. Also plotted are the r values corresponding to the reference value $q = 0.0607$ (the best estimate of q for the whole database). As the figure shows, the estimates vary randomly in the neighborhood of the reference values, and they closely converge to these values as the number of participants increases. In these examples, the reference values were always within the 95% confidence interval provided by the method. We performed 25 independent replications of this experiment, corresponding to a total of 350 tests of the procedure. Globally, the reference values fell outside the 95% confidence interval only 8 times (and always for $n \leq 40$). This frequency of about .02 is smaller than, but not too far from, the expected .05 risk.

These observations suggest that the mean r is a reliable estimate for $n \geq 100$. However, for small samples of participants, it is better to use the lower limit of the 90% confidence interval—that is, the quantile of probability .05, hereafter denoted $r(.05)$—and then to provide the user with a statement in the form Prob[$r > r(.05)$] = .95.

We rapidly illustrate this approach using the MATLAB function permuqr, listed in Appendix B. First, we randomly selected a sample of 30 participants from our database; the resulting data table was named RT30. Then we applied the permuqr function as follows,

```
[q, confq, r, confr, ndr] = permuqr(RT30, 0, 0.90),
```

obtaining the output

```
q = 0.0713, confq = [0.0523; 0.0968], r = .6814,
   confr = [0.6106; 0.7439].
```

As we can see in this example, the q parameter was overestimated; however, the reference value (0.0607) is within the 90% confidence interval. The lower confidence limit of r is .6106, so one can state: Prob($r > .6106$) = .95. In other words, one can guarantee with 95% confidence that the reproducible percentage of item variance in the sample average vector is greater than 37.27%. This tells us that a model that accounts for about 40% of item variance, given this sample of 30 participants, is reasonably good in terms of its performance predictions. In a similar way, we can consider the upper confidence limit of r—that is, .7439—and state that Prob($r < .7439$) = .95. In other words, a model that accounts for more than 55.34% of the item variance (in our example) probably overfits the data by using too many free parameters, and thus actually accounts for a substantial part of the random noise.

Increasing the Amount of Reproducible Variance

If we assume that the amount of reproducible variance in naming with 30 participants is close to the value ob-

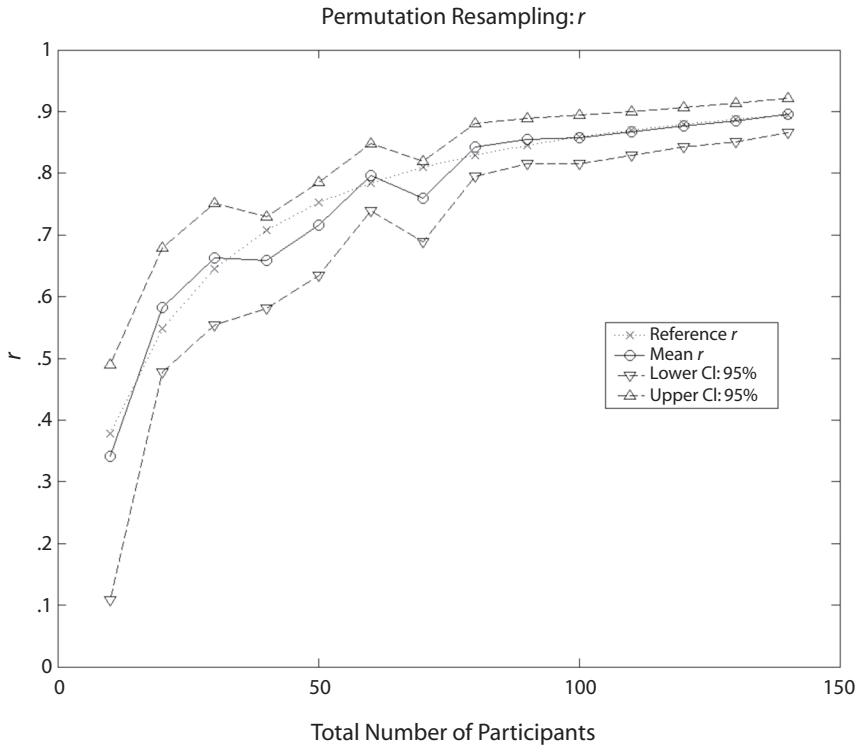

Figure 2. Mean and 95% confidence limits of the permutation resampling *r* distribution as a function of the total number of participants. The reference *r* values correspond to $q = 0.0607$.

tained in the present perceptual identification task (i.e., around 40%), cognitive modelers may come up against a critical problem. Indeed, we have already mentioned that Spieler and Balota (1997) reported that phonemic features, together with word frequency, neighborhood density, and length, accounted for 42% of the variance of item-naming latencies. Similarly, Perry et al. (2007), when testing the CDP+ model against the same item databases, were able to account for a similar amount of variance. From these results, one might conclude that psycholinguistic research has fully solved the problem of visual word recognition processes, since all of the reproducible variance at the level of items has been accounted for. The only remaining debate would concern the format of the explanation: Should we prefer a simple, linear model description or a sophisticated computational account?

A solution to this potential dilemma would be to increase the amount of reproducible variance by simply recording the performance of more participants. However, although the reproducible variance would then increase, the amount of variance explained by the linear or the computational model could likewise increase. If, by increasing the number of participants, one obtains 60% of the reproducible variance at the level of items, the linear model might also account for about 60% of the variance, and this result would probably mark the end of psycholinguistic research. Alternatively, as the amount of reproducible variance increases, other sources of variance might also become visible, such as variance related to morphological, syntactic, or semantic processes. This second, more optimistic outcome would then open the race for a new generation of more sophisticated models.

To determine which of these two outcomes is the correct one, we used the Monte Carlo study described above, in which mean item latencies were calculated for different sizes of participant groups. We then systematically correlated these item means to the log frequency of items. Figure 3 displays the evolution of correlation coefficients as a function of the number of participants per group when independent groups are compared (i.e., for estimating the amount of reproducible variance) and when item means are correlated with the log frequency.

The result is that an increase in the amount of reproducible variance is not accompanied by a proportional increase in the variance explained by log frequency. For example, with groups of 30 participants, on average 41% of the variance is reproducible, and 18% of the item variance is accounted for by log frequency. With groups of 70 participants, these values are now 66% and 23%, respectively. Thus, when increasing from 30 to 70 participants, an increase of 25% is observed in the reproducible variance, whereas an increase of only 5% is obtained in the variance accounted for by log frequency. This result suggests that the increase in reproducible variance allows for capturing new sources of variance that were initially not visible.

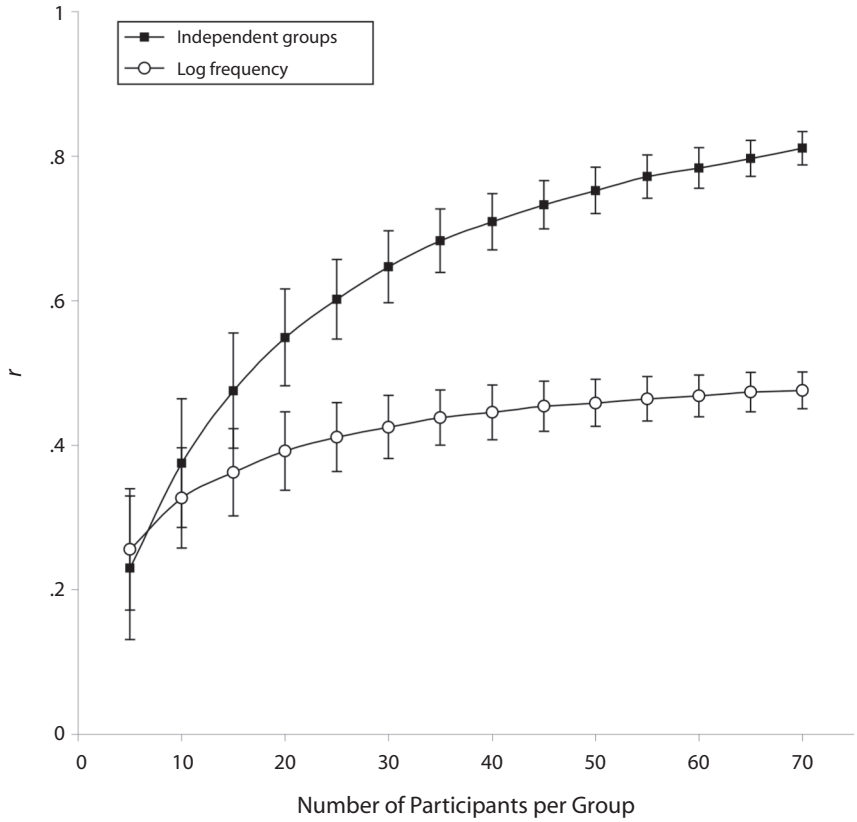

Figure 3. Means (with error bars for standard deviations) of correlations between independent groups composed of 5–70 participants (line with black squares) and between item means, computed for groups of 5–70 participants, and log frequency (line with white circles).

Discussion

Starting with a mathematical description of the reliability of item-level databases, we have proposed a method of estimating the amount of variance that models should account for when they are tested against a database with *n* participants. When *n* is sufficiently large (i.e., larger than 100), we have shown that the function relating the amount of reproducible variance and the number of participants in a given experimental paradigm can be approximated precisely. When *n* is relatively small, calculating confidence intervals using a permutation resampling method is still possible and is useful for estimating the boundaries of the amount of reproducible variance.

Following Balota and Spieler (1998) and Seidenberg and Plaut (1998), the present study provides new arguments concerning the amount of variance that models should account for. The main result concerns the relation between reproducible variance and the total number of participants involved in the computation of item means. On the basis of a common statistical model, we can confidently state that the present set of item mean response times (computed on the basis of the performance of 140 participants and recorded in this specific experimental setup[7]) is composed of 80% reproducible variance and 20% error variance. This information is of major importance, because one can now clearly evaluate the descriptive adequacy of computational models and the amount of variance that a given hypothesized cognitive architecture can account for.[8]

Conclusion

One can assume that the interaction between a given word, characterized by a set of properties (e.g., visual, phonological, or semantic), and the population of adult readers (supposed to share a similar cognitive architecture for processing written words) can be quantified by a measure of the processing time required to read the word in a given experimental situation. Likewise, if one considers a list of such words, it is a priori possible to rank those words according to their processing time and to evaluate the ability of computational models to reproduce this ranking. Using this item-level ranking might be misleading, however, if intra-item variability is greater than between-item variability. In this case, item-level databases only reflect general trends, and the fine-grained ranking of items remains hidden in an undesirable source of error variance.

The aim of the present study was to quantify the respective amounts of reproducible and error variance to determine the amount of variance that models should account for in item-level databases. The methodology we presented offers such quantification, together with practical solutions for estimating the amount of reproducible variance for any database. The conclusion is that collect-

ing large-scale databases composed of both many items and many participants will provide genuine challenges to future generations of computational models of word recognition.

AUTHOR NOTE

We are grateful to David A. Balota, Stephen D. Goldinger, Jeffrey N. Rouder, and one anonymous reviewer for their helpful comments. Correspondence should be sent to A. Rey, Laboratoire de Psychologie Cognitive, CNRS–Université de Provence, 3 place Victor Hugo, 13331 Marseille Cedex 3, France (e-mail: arnaud.rey@univ-provence.fr).

REFERENCES

ANS, B., CARBONNEL, S., & VALDOIS, S. (1998). A connectionist multiple-trace memory model for polysyllabic word reading. *Psychological Review*, **105**, 678-723.

BAAYEN, R. H., PIEPENBROCK, R., & VAN RIJN, H. (1993). The CELEX lexical database (CD-ROM). Philadelphia: Linguistic Data Consortium, University of Pennsylvania.

BALOTA, D. A., & SPIELER, D. H. (1998). The utility of item-level analyses in model evaluation: A reply to Seidenberg and Plaut. *Psychological Science*, **9**, 238-240.

COLTHEART, M., RASTLE, K., PERRY, C., LANGDON, R., & ZIEGLER, J. (2001). DRC: A dual route cascaded model of visual word recognition and reading aloud. *Psychological Review*, **108**, 204-256.

GOOD, P. (1994). *Permutation tests: A practical guide to resampling methods for testing hypotheses*. New York: Springer.

GRAINGER, J., & JACOBS, A. M. (1996). Orthographic processing in visual word recognition: A multiple read-out model. *Psychological Review*, **103**, 518-565.

HARM, M. W., & SEIDENBERG, M. S. (2004). Computing the meanings of words in reading: Cooperative division of labor between visual and phonological processes. *Psychological Review*, **111**, 662-720.

OPDYKE, J. D. (2003). Fast permutation tests that maximize power under conventional Monte Carlo sampling for pairwise and multiple comparisons. *Journal of Modern Applied Statistical Methods*, **2**, 27-49.

PERRY, C., ZIEGLER, J. C., & ZORZI, M. (2007). Nested incremental modeling in the development of computational theories: The CDP+ model of reading aloud. *Psychological Review*, **114**, 273-315.

PLAUT, D. C., MCCLELLAND, J. L., SEIDENBERG, M. S., & PATTERSON, K. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review*, **103**, 56-115.

REY, A., JACOBS, A. M., SCHMIDT-WEIGAND, F., & ZIEGLER, J. C. (1998). A phoneme effect in visual word recognition. *Cognition*, **68**, B71-B80.

REY, A., & SCHILLER, N. O. (2005). Graphemic complexity and multiple print-to-sound associations in visual word recognition. *Memory & Cognition*, **33**, 76-85.

ROUDER, J. N., & LU, J. (2005). An introduction to Bayesian hierarchical models with an application in the theory of signal detection. *Psychonomic Bulletin & Review*, **12**, 573-604.

SEIDENBERG, M. S., & MCCLELLAND, J. L. (1989). A distributed, developmental model of word recognition and naming. *Psychological Review*, **96**, 523-568.

SEIDENBERG, M. S., & PLAUT, D. C. (1998). Evaluating word-reading models at the item level: Matching the grain of theory and data. *Psychological Science*, **9**, 234-237.

SEIDENBERG, M., & WATERS, G. S. (1989). Word recognition and naming: A mega study. *Bulletin of the Psychonomic Society*, **27**, 489.

SPIELER, D. H., & BALOTA, D. A. (1997). Bringing computational models of word naming down to the item level. *Psychological Science*, **8**, 411-416.

ZIMMERMAN, D. W., ZUMBO, B. D., & WILLIAMS, R. H. (2003). Bias in estimation and hypothesis testing of correlation. *Psicológica*, **24**, 133-158.

NOTES

1. In this study, 30 McGill University undergraduates named aloud 2,900 monosyllabic English words.

2. It is not necessary to assume that the random variables are Gaussian, but one can assume that the variances are finite and that $\sigma_{\varepsilon}^2 > 0$.

3. Indeed, if an m-dimensional vector $\mathbf{B}$ is hypothesized to be an approximation of β (or of any affine function of β) for the item set under consideration, a common procedure consists of averaging the n columns of the data table and comparing the resulting m-dimensional vector to $\mathbf{B}$ by means of Pearson's r statistic. However, even if $\mathbf{B} = \beta$, there is no chance that $r = 1$, because of the data random variance. In the best case, one could expect a correlation on the order of ρ, as defined by Equation 2.1, given that r is known to be a consistent asymptotically unbiased estimator of ρ (Zimmerman, Zumbo, & Williams, 2003).

4. We are indebted to Guy Van Orden, who allowed one of us to run the experiment in his laboratory.

5. We thank David Chesnet and Jonathan Grainger for providing us with the computer program to implement the luminance-increasing procedure.

6. This is visibly the case for our data, and note that the additive decomposition model has for many years been the most commonly used model for the analysis of experimental data (usually with the additional assumption that the variables are Gaussian or conform to the conditions of the central-limit theorem). If this model is grossly false for common experimental tasks and variables, this is a serious problem that greatly exceeds the focus of the present study.

7. Although the estimations generated with the present database provide a concrete example of estimating the amount of reproducible variance from any database having either a small or a large sample, such estimations may vary greatly from one database to another. Notably, there might be important differences between experimental paradigms (e.g., perceptual identification vs. naming), and the estimations given here therefore cannot generalize from one experimental setup to another.

8. This does not mean that models cannot handle error variance. Solutions have been proposed in which error variance or noise could be simulated by adding, for example, a variable response mechanism based on a normally distributed parameter (see, e.g., Grainger & Jacobs, 1996). Here, we simply wish to dissociate the modeling of reading processes, which can theoretically be considered as free of error variance, from the addition of noise within cognitive models. One may, however, argue that modeling should necessarily incorporate the presence of noise in the studied systems.

APPENDIX A

Let us consider the bivariate distribution of pairs (X, Y), where X and Y are independent realizations of $x^{(n)}$; that is, the n participants are never the same for X and for Y. The population correlation between X and Y, varying the items, is given by

$$\rho(X,Y) = \mathrm{Cov}(X,Y) \Big/ \sqrt{\mathrm{Var}(X)\,\mathrm{Var}(Y)},$$

where, using Equation 1.1, one has

$$\mathrm{Cov}(X,Y) = \mathrm{Cov}\left(\beta + \varepsilon_X^{(n)}, \beta + \varepsilon_Y^{(n)}\right) = \mathrm{Var}(\beta) = \sigma_\beta^2,$$

because the terms that are constant with respect to the item variable (μ and $\alpha^{(n)}$) play no role in the correlation, and the variables β, $\varepsilon_X^{(n)}$, and $\varepsilon_Y^{(n)}$ are independent.

For the same reasons, one has also

$$\mathrm{Var}(X) = \mathrm{Var}\left(\beta + \varepsilon_X^{(n)}\right) = \mathrm{Var}(\beta) + \mathrm{Var}\left(\varepsilon_X^{(n)}\right) = \sigma_\beta^2 + \sigma_\varepsilon^2 \big/ n,$$

and similarly,

$$\mathrm{Var}(Y) = \mathrm{Var}\left(\beta + \varepsilon_Y^{(n)}\right) = \mathrm{Var}(\beta) + \mathrm{Var}\left(\varepsilon_Y^{(n)}\right) = \sigma_\beta^2 + \sigma_\varepsilon^2 \big/ n.$$

Thus, finally,

$$\rho(X,Y) = \frac{\sigma_\beta^2}{\sigma_\beta^2 + \sigma_\varepsilon^2 \big/ n} = \frac{n\sigma_\beta^2 \big/ \sigma_\varepsilon^2}{n\sigma_\beta^2 \big/ \sigma_\varepsilon^2 + 1}.$$

Not surprisingly, the expression above is similar to that of an intraclass correlation coefficient.

APPENDIX B

Here is MATLAB code of the permuqr function, which provides expected q and r values, with confidence intervals of chosen probabilities confp, from a data table x. For ease of reading, structural coding is set in boldface and comments in italics.

```
function [q,confq,r,confr,ndr] = permuqr(x,missing,confp,dr)
% Permutation Resampling to estimate q, r, and confidence
% intervals of given probabilities "confp" (row vector),
% from the m-items by n-participants data table x,
% where "missing" is the code for missing data in x.
% The first (second) row of "confq" corresponds to the
% lower (upper) confidence limit(s), similarly for "confr."
% An optional desired r (dr) provides the necessary n (ndr)
% r^2 is the reproducible proportion of item variance when
% one averages the n columns of x.
resample = 2000; % T > 1900 (see Opdyke, 2003)
[m,n] = size(x); confp = 1-confp; % Proba to alpha
ng = fix(n/2); % Number of participants per group
rt = zeros(resample,1);
for t = 1:resample
ok = false;
while ~ok
xp = x(:,randperm(n)); % Random participant permutation
ng1 = zeros(m,1); mg1 = ng1; ng2 = ng1; mg2 = ng1;
for i = 1:m
for j = 1:ng  % First group
if xp(i,j) ~ = missing
ng1(i) = ng1(i) + 1; mg1(i) = mg1(i) + xp(i,j);
end
end
for j = (ng + 1):(2*ng) % Second group
if xp(i,j) ~ = missing
ng2(i) = ng2(i) + 1; mg2(i) = mg2(i) + xp(i,j);
end
end
end
if (min(ng1) > 0) && (min(ng2) > 0), ok = true; end
end
mg1 = mg1./ng1; mg2 = mg2./ng2;
```

APPENDIX B (Continued)

```
rr = corrcoef([mg1, mg2]); rt(t) = rr(1,2);
end
q = rn2q(mean(rt),ng); r = qn2r(q,n); rt = sort(rt);
nconf = length(confp); confindex = zeros(2,nconf);
confindex(1,:) = round(resample*confp/2) + 1;
confindex(2,:) = round(resample*(1-confp/2));
confq = rn2q(rt(confindex),ng); confr = qn2r(confq,n);
if nargin = = 4, ndr = round(qr2n(q,dr)); else ndr = []; end
function q = rn2q(r,n) % Provides q given r and n
q = r./(n.*(1-r));
function r = qn2r(q,n) % Provides r given q and n
r = (n.*q)./(n.*q + 1);
function n = qr2n(q,r) % Provides n given q and r
n = r./(q.*(1-r));
```

(Manuscript received September 10, 2007;
revision accepted for publication January 23, 2009.)