SRB

SHIKEN RESEARCH BULLETIN

Volume 16 • Number 2 • November 2012

Contents

JALT
Testing and Evaluation SIG

Foreword

Aaron Olaf Batty
TEVAL SIG Publications Chair,
SRB General Editor

In this issue…

In this issue of *SRB,* we have an article on an innovative new method of scoring yes/no vocabulary tests, a timely and relevant paper on the reliability of the TOEIC Bridge, an introduction to a fantastic piece of free statistics software, an interview with one of the most interesting people in language testing today, and, of course, another installment of JD Brown's indispensible statistics column. We also have a new section where we feature the recent publications of the TEVAL SIG members, as well as the customary list of upcoming language testing events. Personally, I am glad to present such a strong issue to the TEVAL SIG membership at this time because…

A fond farewell… again

Due to my acceptance to a language-testing-related PhD program, as well as expanding teaching and research commitments, I am sorry to report that this is my last issue of *SRB* as General Editor. I have really enjoyed this position, and I am proud of the work I have done. I have worked hard to modernize this publication over the last year, and have made some real progress in that regard—progress that will continue under the next General Editor, my personal pick for the position: the astoundingly diligent and capable Jeffrey Stewart, whom I have been privileged to have as my Associate Editor for this past year. As Jeff has been my partner in all the editorial decisions of the past year, the direction of the publication will continue on its current course under his editorship. I will, of course, continue to support the publication as a member of the editorial committee. Unfortunately, however, I will just be too busy for the next few years to devote the time and attention necessary to deliver the caliber of publication that I think TEVAL deserves. Thanks for the support over the last year, and I am sure you will offer the same to Jeff.

We hope you enjoy this issue of *SRB*.

Optimizing scoring formulas for yes/no vocabulary tests with linear models

Raymond Stubbe
raymondstubbe@gmail.com
Kyushu Sangyo University

Jeffrey Stewart
jeffjrstewart@gmail.com
Kyushu Sangyo University, Cardiff University

Abstract

Yes/No tests offer an expedient method of testing learners' vocabulary knowledge, although a drawback of this method is that since the method is self-report, actual knowledge cannot be confirmed. "Pseudowords" have been used within such lists to test if learners are reporting knowledge of words they cannot possibly know, but it is unclear how to use this information to adjust scores. Although a variety of scoring formulas have been proposed in the literature, empirical research (e.g., Mochida & Harrington, 2006) has found little evidence of their efficacy.

The authors propose that a standard least squares model (multiple regression), in which the counts of words reported known and counts of pseudowords reported known are added as separate predictor variables, can be used to generate scoring formulas that have substantially higher predictive power. This is demonstrated on pilot data, and limitations of the method and goals of future research are discussed.

Background

Yes/No checklists of English vocabulary knowledge (YN tests) can be used as a quick, psychometrically reliable self-report method of checking students' knowledge of large numbers of words in a relatively short period of time (Meara & Buxton, 1987). However, an obvious pitfall of such an approach is that tests reliant on self-reporting may produce results that differ from reality. Indeed, recent research by Stubbe and Yokomitsu (2012) reveals that students who completed a YN test of English words and were then asked to provide Japanese definitions for words they had reported knowledge of could on average only provide correct definitions for half. In many cases, overestimation of vocabulary size was due to confusion of an unknown word (e.g., *root*) with a known, but untested word with similar orthography (e.g., *route*). Such a disparity between YN test scores and translation tests scores is not uncommon. Waring and Takaki (2003) reported a 70% drop on scores from a recognition test to a similar (L2 to L1) translation test.

Pseudowords (e.g. "steck" or "noof") are commonly used in YN tests to detect false reports (Read, 2000); if a student claims to know a word that does not actually exist, it calls their reports of knowledge on real words into question. SLA literature contains a number of different formulas for YN tests that use the number of pseudoword checks, called false alarms (FA), to adjust the proportion of words considered to be known. For example, Huibregtse, Admiraal and Meara (2002) studied the following four correction formulas: $h - f$ (Anderson & Freebody, 1983), *cfg* (*correction for guessing*, Meara & Buxton, 1987), Δm (Meara, 1992) and *Isdt* (Huibregtse, et al., 2002).

However, it remains unclear which, if any, of these scoring formulas is preferable for improving the predictive power of YN tests. Mochida and Harrington (2006) proposed a research design for examining and comparing the efficacy of these various formulas:

- Give learners a self-report YN test.
- As a follow up, give them a conventional test of the same words.
- Score the self-report YN test by a variety of scoring formulas that use pseudoword false alarms to calculate the true number of words known, and see which formula results in the highest correlation to scores on the conventional test.

Mochida and Harrington examined the above formulas by correlating resulting YN test scores to scores on the Vocabulary Levels Test (Nation, 1990), but did not find substantial differences in correlations regardless of the adjustment formula used. Eight years following Huibregtse, et al., (2002), Schmitt noted that "it is still unclear how well the various adjustment formulas work" (Schmitt, 2010, p. 201). Possibly due to such uncertainties about the efficacy of these formulas, some researchers simply disregard individuals' YN test results when FAs exceed a given threshold. For example, Schmitt, Jiang and Grabe (2011) discarded any YN test forms reporting more than 3 false alarms (10% of the total of 30 pseudowords used in that study).

Can information gleaned from pseudowords be of use in adjusting YN test self-reports to better reflect the number of words a student is likely to know if they are tested on meanings of each word individually? In this research report, we will detail a statistical model built using experimental data that demonstrates that this is likely the case. Steps necessary for its creation can be described as follows:

- Follow Mochida and Harrington's (2006) research design, so that the resulting data set contains: a) a list of words students claim to know via a YN test; b) a count of pseudowords students claim to know (false alarms); and, c) a list of the same words that students actually know as determined by a conventional test.
- In addition to running a simple bivariate correlation between words self-reported as known (the predictor variable) and words actually known (the dependent variable), we can go a step further: Using multiple regression, add the count of pseudoword false alarms as a second predictor variable used to predict words actually known.

In addition to reporting an R^2 value indicating how well the two variables contained on the YN test predicted actual knowledge together, the model can also return a prediction expression, for example (Equation 1):

$$True\ number\ of\ words\ known = 8 + (0.7)(YN\ Score) - (2.4)(False\ Alarms) \quad (1)$$

Here, 8 words is the intercept. For every word reported known on the YN test, we add 0.7 words truly known. For every false alarm, we subtract 2.4 words. Such a prediction expression is effectively an optimized "scoring formula", at least in regards to the sample of students and the particular YN test it was derived from. Although it is unlikely the formula will be optimal for another sample or YN test without modification, such models provide an empirical basis for the creation of scoring schemes. If the prediction expression can be shown to hold well over multiple samples, it can be used with confidence with this same YN test when testing comparable demographics.

This short paper will detail preliminary steps taken to develop such a scoring formula using data from one of the authors' previous studies (Stubbe & Yokomitsu, 2012), and propose methods for examining the validity of self-reports on individual words and the predictive power of individual pseudowords. Further development will be detailed in future studies.

Method

Participants were second year students enrolled in a compulsory English course at a private Japanese university, with *TOEIC® Bridge* scores ranging from 100 through 140. Unfortunately, two of the students reported 10 and 11 false alarms, each. As they would have considerable leverage on scoring formulas arrived at in this study they were removed, reducing the sample size to 69. Participants first took a YN test of 120 words with 32 pseudowords, and afterwards provided definitions for those same words in a translation test (L2 to L1). YN tests were marked using a scanner, while the translation test was hand marked by two native Japanese speakers, with an inter-rater reliability of .97

Preliminary analysis

Descriptive statistics for both the YN and translation tests are presented below.

Table 1. YN and Tr test descriptive statistics

Test	Mean	SD	Range	Low	High	Reliability
YN Items	59.56	15.49	71	20	91	0.93
YN FAs	0.96	1.56	7	0	7	0.65
Tr Items	29.94	9.48	38	11	49	0.89

Note: Reliability = Cronbach's alpha.

The correlation between the YN test form and the translation test was quite low, at only 0.60. To determine if a scoring formula using the experiment's pseudoword data could explain more variance, we ran a standard least squares model in the statistical software package JMP 8 (SAS, 2009) using false alarms and YN test scores as separate variables. Residuals were random and normally distributed, and there was not substantial collinearity between the two predictor variables (R = .23).

The mean response score was 30.55 (of the 120 items). The overall regression model was significant ($F(2, 66) = 27.2$, $p < .0001$), with an R^2 of 45.2%. The intercept of this model and both independent variables were significant ($p = 0.0201$ and ≤ 0.0011, respectively). Resulting variable weights (beta p) are presented in Table 2.

Table 2. Regression model variable weights and p-values

Variable	weight	p
Y-intercept	8.14	.0201
YN scores	0.41	.0001
FA scores	-1.94	.0011

R = .67

This suggests that a scoring formula that adjusts scores using false alarms counts could indeed be useful in estimating true vocabulary size. The optimal scoring formula for the YN test suggested by this initial model can be expressed as (Equation 2):

$$True\ number\ of\ words\ known = 8.14 + (0.41)(YN\ Score) - (1.94)(False\ Alarms) \quad (2)$$

However, while this represents a sizeable improvement on a prediction model that does not use pseudoword data ($R^2 = 35.6\%$, Table 6), the accuracy of prediction of vocabulary knowledge given self-reports on the YN test remains underwhelming. An R^2 of 45.2% is only equivalent to a correlation of about 0.67.

Improving predictive power with item analysis

To improve the predictive power of the YN test, we decided to look at specifically which words could be reliably predicted as known given self-reports on the YN test. We did this by examining phi (dichotomous) correlations between students' self-reports on given words and whether or not the same word was confirmed as known on the subsequent translation test.

The words in Table 3 had high phi correlations between self-report of knowledge and demonstrated knowledge on the translation test, meaning the self-report test appears to be a fairly valid predictor of actual knowledge of them.

The words in Table 4 had very *low* point phi correlations to the translation test results, meaning that there was little correlation between claims that these words were known and actual demonstrations of knowledge on the translation test.

Table 3. Real words with high phi correlations

Item	Phi correlation
salmon	0.57
chapel	0.56
crystal	0.51
narrow	0.51

Table 4. Real words with low phi correlations

Item	Phi correlation
maker	-0.11
concerto	-0.11
overall	-0.11
convenience	-0.23

It is easy to see why these words were falsely reported as known. For example, *convenience* was confused with the loan word *konbini* (convenience store), and *overall* was taken to have a literal meaning. This suggests that removing words that students have difficulty self-assessing knowledge on should improve the predictive power of the self-report test.

It is also possible to examine the efficacy of pseudowords used. Much in the same way language testers examine the point-biserial correlations of test items to overall test score to choose effective questions for a test, we can examine the degree to which pseudowords have *negative* point-biserial correlations with total scores on the translation tests in which students actually pro-

vide definitions of words reported as known. By doing this, we can design YN tests with pseudowords that have empirically validated predictive power.

The YN data was revised to include only the 40 words with the highest phi correlations to translation test results, and the nine pseudowords with the strongest negative point-biserial correlations to overall translation test scores. These nine point-biserials (on pseudowords such as *curify, lannery, noot,* and *skene*) ranged from -0.22 to -0.42, while the average for the other 23 pseudowords was -0.03 (essentially, no predictive power whatsoever). The regression model was re-run to see if this item analysis improved its predictive power. The mean response score became 13.07 (of the 40 items). The overall regression model was significant ($F(2, 66) = 47.7$, $p < .0001$), with an R^2 of 59.1% and the following variable weights (beta p):

Table 5. Revised regression model variable weights and p-values

Variable	weight	p
Y-intercept	3.26	.0214
YN scores	0.51	.0001
FA scores	-2.39	.0001
R = .77		

The R^2 was improved, and is now equivalent to a correlation of 0.77, and both the intercept ($p = 0.0214$) and the predictor variables ($p < 0.0001$) are significant. The scoring formula the model implies for this revised YN test is:

True knowledge of tested words = 3.26 + 0.51 × YN Score - 2.39 × False Alarms

The overall effectiveness of adding FA scores to the model can be seen below.

Table 6. R^2 values before and after entry of pseudoword predictor variable

Predictor variables	Original item list R^2	Revised item list R^2
YN scores only	35.6%	47.8%
YN scores + FA scores	45.2%	59.1%

Conclusions

This study has two findings of potential interest. First, much in the same way item analysis can aid in identifying items that perform well or poorly on a conventional test, they can assist in the creation of YN tests with greater predictive power. Self-reports of knowledge correlated to true knowledge more for some words than for others, and although most pseudowords used in this study had very little predictive power, a few had sufficiently negative correlations to true vocabulary knowledge to be of use in the model. For these reasons, we recommend that researchers subject the words and pseudowords they include on YN tests to item analysis, as is commonly done with more conventional test formats.

Second, although preliminary, this research indicates multiple regression may be of use in determining scoring formulas for self-report YN tests with pseudowords that can be empirically demonstrated to improve prediction of actual word knowledge, as measured by a separate test in which knowledge is confirmed by a human rater.

Of course, this finding is entirely preliminary. There are a number of remaining concerns. The formula must be tested on other samples to determine if it is generalizable to groups of learners other than the one examined for this particular experiment. Future directions will include finalizing such a model, and pitting it against existing scoring formulas. It should be noted that in all likelihood, any useful scoring formulas derived will only be applicable to the particular YN test used to develop it, and with demographics similar to the sample used in the research (i.e., Japanese university students). How generalizable such scoring formulas are to other populations has yet to be determined.

Finally, although item analysis and the scoring formula suggested by the model greatly improved the predictive power of the YN test, an R of 0.77 still seems fairly low. One possibility is that contrary to the findings of prior research, Japanese university students can overestimate their vocabulary knowledge, due to complicating factors such as English loanwords which have different usages in Japanese. Another possibility is that the sample examined in this data set gave less reliable self-reports due to their lower level of proficiency. For this reason further research must be conducted on learners with a wider range of proficiency.

References

Anderson, R. C., & Freebody, P. (1983). Reading comprehension and the assessment and acquisition of word knowledge. In Hutson, B. A. (Ed.), *Advances in Reading/Language Research*, Vol. 2 (pp. 231–256). Greenwich, CT: JAI Press.

Huibregtse, I., Admiraal, W., & Meara, P. (2002). Scores on a yes–no vocabulary test: correction for guessing and response style. *Language Testing, 19*, 227-245. doi:10.1191/0265532202lt229oa

Meara, P. 1992: *New approaches to testing vocabulary knowledge.* Draft paper. Swansea: Centre for Applied Language Studies, University College Swansea.

Meara, P., & Buxton, B. (1987). An alternative to multiple choice vocabulary tests. *Language Testing, 4*(2), 142–154. doi:10.1177/026553228700400202

Mochida, A., & Harrington, M. (2006). Yes/No test as a measure of receptive vocabulary. *Language Learning, 23*, 73-98.

Nation, I.S.P., 1990: *Teaching and learning vocabulary*. Boston, MA: Heinle and Heinle.

Read, J. (2000). *Assessing Vocabulary.* Cambridge: Cambridge University Press. doi:10.1017/CBO9780511732942

SAS Institute Inc. (2009). JMP Version 8. SAS Institute Inc., Cary, NC, 1989- 2009.

Schmitt, N. (2010) *Researching Vocabulary: A Vocabulary Research Manual.* NY: Palgrave Macmillan. doi:10.1057/9780230293977

Schmitt, N., Jiang, X., & Grabe, W. (2011) The percentage of words known in a text and reading comprehension. *Modern Language Journal, 95*, 26-43. doi:10.1111/j1540-4781.01146x

Stubbe, R. & Yokomitsu, H. (2012) English Loanwords in Japanese and the JACET 8000. *Vocabulary Education and Research Bulletin, 1*, 10-11.

Waring, R. & Takaki, M., (2003). At what rate do learners learn and retain new vocabulary from reading a graded reader? *Reading in a Foreign Language*, *15* (2), 130-163.

Examining the reliability of a TOEIC Bridge practice test under 1- and 3-parameter item response models

Jeffrey Stewart
jeffjrstewart@gmail.com
Kyushu Sangyo University, Cardiff University

Aaron Gibson
aaronlgibson@gmail.com
Kyushu Sangyo University

Luke Fryer
lukekutszikfryer@gmail.com
Kyushu Sangyo University

Abstract

Unlike classical test theory (CTT), where estimates of reliability are assumed to apply to all members of a population, item response theory provides a theoretical framework under which reliability can vary by test score. However, different IRT models can result in very different interpretations of reliability, as models that account for item quality (slopes) and probability of a correct guess significantly alter estimates. This is illustrated by fitting a TOEIC Bridge practice test to 1 (Rasch) and 3-parameter logistic models and comparing results. Under the Bayesian Information Criterion (BIC) the 3-parameter model provided superior fit. The implications of this are discussed.

Reliability under classical test theory

Test reliability refers to the internal consistency of a test. Ideally, a test taker who has not changed in language ability will receive similar scores on forms of a test regardless of how many times it is taken. Under Classical Test Theory (CTT), test reliability is most often measured by statistics such as Cronbach's Alpha and KR 20, but is perhaps most practically assessed using the Standard Error of Measurement (SEM). In simple terms, the SEM reflects the degree to which a test score may vary between test administrations by chance. Essentially a "standard deviation of error", it can be interpreted as a confidence interval for the learner's theoretical "true score". For example, if a student receives a scale score of 500 on a test and the test's SEM is 70, it can be said that if the test is retaken without improving in ability, the student can be expected to receive a score of 500 ±70 points 68% of the time. Setting an interval of two standard errors (±140 points) will yield a confidence interval of approximately 95% (for more information on the SEM, please refer to Brown, 1999).

While these statistics provide useful rules of thumb, it should be noted that under CTT, measures of reliability are assumed to apply to all test takers, regardless of the score they receive. Strictly speaking, this assumption is unrealistic. In situations where two test takers receive very high scores, it could be the case that the test questions used were not difficult enough to suitably challenge them, which could give us less confidence in score comparisons between them than we

would have between two test takers of average ability, where the majority of questions were likely of suitable difficulty.

Another concern regarding how test reliability could change depending on student ability involves the use of multiple-choice questions. Suppose four students take a multiple-choice test of 100 questions, each with 4 options, and the test has an SEM of 5 points. One student receives a score of 65, one of 70, one of 20 and one of 25. Technically, the score difference between the first pair of students and second pair of students is identical at 5 points. But we have good reason to be less confident in the reliability of the scores for the second pair of students, because both received scores under the threshold of chance, as a score of 25 is possible simply by filling out answers at random. In cases where questions are too difficult to make a selection with any confidence, students become more likely to guess. Although multiple-choice formats likely affect reliability regardless of student ability, this practice could lead to more error for lower level students, where guessed answers could constitute a higher proportion of their total scores.

Reliability under item response theory

An advantage of Item Response Theory (IRT) is that under it, it is possible to examine how measures of test reliability change as a function of learner ability level (Embretson & Reise, 2000, p.185). We can determine not just how reliable a test will be overall, but how reliable it will be for particular groups of students with similar levels of language proficiency. This information can be used, for example, to determine if a test has suitable reliability at a proposed cut score between pass and fail, or if a new test is necessary for a special group of students of higher or lower ability than usual.

Under IRT, a number of factors can contribute to how reliable a test is considered to be for a given score (or, to use IRT terminology, the amount of "information" or precision the test provides). The following outline will focus on central concepts; Partchev (2004) provides details and formulas.

Item difficulty

Under the 1-parameter Rasch model, once misfitting items have been removed a primary determinant of reliability is how closely the difficulty of items used matches the ability level of the students tested. Items that students have a 0.5 probability of answering correctly are considered to provide the most information, and reciprocally to produce the smallest standard errors. The further items are in difficulty from a student's ability level, the less information the item will give about the student, and the greater standard errors of a student's ability estimate will become.

Item discrimination

In addition to item difficulty, the two-parameter logistic (2PL) model uses item discrimination to calculate the information items provide about student ability. Although items are still believed to provide maximum information when students have a 0.5 probability of answering them correctly, if difficulties are equal, items with low discrimination are considered to provide less information than items with high discrimination.

Guessing

In addition to item difficulty and discrimination, the three-parameter logistic (3PL) model also uses the likelihood a student of very low ability will choose a correct answer to determine item information. If a student's probability of correctly answering a question is as low as the probabil-

ity of correctly guessing it by chance, the item is not considered to provide any information about student ability. An interesting aspect of the 3PL model is that rather than providing maximum information for students with 0.5 odds of correctly answering, maximum information falls at the midpoint between the odds of a correct guess by a very low level student and 1. For example, if an item has a 0.2 probability of being answered correctly even by a very low level student, the item is considered to provide maximum information for students who have a 0.6 probability of answering it. The practical result of this is that unlike under CTT and Rasch frameworks, tests that result in mean scores of 50% are not always optimal; due to the fact that some questions will be answered correctly by chance, the ideal mean score can be somewhat higher.

Given these different considerations, estimates of test information and resulting reliability can vary greatly depending on the item response model used. Although person ability and item difficulty estimates are typically very highly correlated regardless of the IRT model used (Stewart, 2012), the same cannot be said about test information and resultant reliability, which can vary substantially between models (De Ayala, 2009). Model fit must be examined to determine which model best describes a test.

Aims

In this paper we will examine a practice form of a well-known and widely used test of English language proficiency, the TOEIC Bridge test (ETS, 2010) under CTT and 1- and 3-parameter item response models, in order to demonstrate how estimates of reliability differ under each framework. We will then conduct a model fit comparison to determine which IRT model is most appropriate for the data, detail how test reliability varies by student ability level under the chosen model, and explain the significance of the findings in practical terms.

The TOEIC Bridge test

The TOEIC Bridge (ETS, 2010a) is a test of emerging English language ability for learners with proficiency too low to be measured by the better-known TOEIC Test. Items used are similar in format and content to those of the TOEIC Test, but of lesser difficulty. Like the TOEIC Test, it has Listening and Reading sections, though with only 50 items each and 100 total, as opposed to 100 items each for 200 total. Although it is not yet popular with many private test takers, it is increasingly used by institutions such as junior high schools, high schools and low-level universities; of the approximately 198,000 people who took the test in 2009, 98% took the TOEIC Bridge IP (ETS, 2010b), which is delivered to such institutions and administered on-site rather than at a testing ground operated by ETS. Scores are derived from raw scores on the test, without penalty for guessing. Scores for its two sections are converted to scale scores ranging from 10-90. Total scale scores for both sections range from 20-180 (ETS, 2007a). In 2009, the mean total scale score for TOEIC Bridge IP Test (which constitutes the vast majority of tests taken) was 118.1 (ETS, 2010b).

In order to prepare for an official administration of the TOEIC Bridge IP used as an achievement test, 1071 first and second year students at a private university took an official ETS TOEIC Bridge practice test included in a test preparation workbook (ETS, 2008a). The mean scale score for the students at the private university on the official test, written shortly after, was approximately 120, close to the average reported by ETS for all TOEIC Bridge IP test takers the previous year (ETS, 2010b).

Initial results

The practice test had a KR-20 reliability statistic of 0.85, and a correlation of 0.81 to scores on the official test, which students took shortly after. The test's raw score mean was 50.14, with a high raw score of 86 and a low score of 22. The standard deviation was 11.72. As per Brown (1999), this results in a standard error of measurement of approximately 4.5 points.

The mean item difficulty was 0.5, which, as with the Rasch and 2PL IRT models, is considered optimal for norm-referenced tests under CTT (Brown, 2005), as such items aid in producing a normal distribution of scores. This results in a mean score near 50%, which, being the midpoint of possible scores, is typically considered ideal for norm-referenced tests under CTT. As the mean TOEIC Bridge score of test takers was close to the average score for all TOEIC Bridge IP test takers as reported by ETS, indicating that the sample's ability level was close to the mean of the average TOEIC Bridge IP test taker, it is possible that this is by design. However, the dispersion of scores of the current sample was quite narrow, with a standard deviation of 11.72, and two standard deviations out yielding a score range between roughly 27 and 73. As this score range covers almost 98% of the tested population, only approximately 50% of the potential scores are applicable to the majority of learners tested.

Table 7. Mean item difficulty by test section and part

Test Section	Part	Mean	K	SD	Min.	Max.
Listening	I	0.63	15	0.26	.24	.97
	II	0.55	20	0.20	.27	.96
	III	0.41	15	0.11	.20	.59
	Total	0.53	50	0.21	.20	.97
Reading	IV	0.45	30	0.15	.21	.87
	V	0.50	20	0.23	.20	.89
	Total	0.47	50	0.18	.20	.89
Total Test		0.50	100	0.20	.20	.97

Examining test reliability under Rasch and 3PL IRT models

The test data was analyzed under 1- and 3-parameter logistic item response models using JMP 8. Item parameter estimates were then used to calculate test information functions (IRT reliability estimates) for both models in Microsoft Excel, using formulas detailed by Parchev (2004), and graphed using guidelines by Kim (2004). The "Test Information" on the vertical axis refers to the precision of the test for given levels of student ability, listed on the horizontal axis. An ability (or "theta") level of 0 indicates the student average, as the theta mean is person-centered. The greater the test information at a given ability level, the more reliable the test is considered to be at that point.

The test's test information function (TIF) differed markedly between models. Under the Rasch model, the TIF appears to be ideal, with maximum information provided for test takers of average ability of 0, corresponding to the mean score of nearly 50% under classical test theory analysis; under both the Rasch model and CTT, test reliability is optimal for the majority of students.

However, under the 3PL model, which estimates the probability that a test taker of very low ability will correctly guess an answer by chance, the TIF is considerably more uneven, with maximum information given for test takers with ability estimates between approximately 1-2.5 logits. Much of the discrepancy can be accounted for by the 3PL model's use of the likelihood of a very low level student guessing the answer by chance in its estimate of reliability. If the model is accepted, the implication is that a somewhat easier version of the test would have maximum reliability for the majority of the students tested in this study.

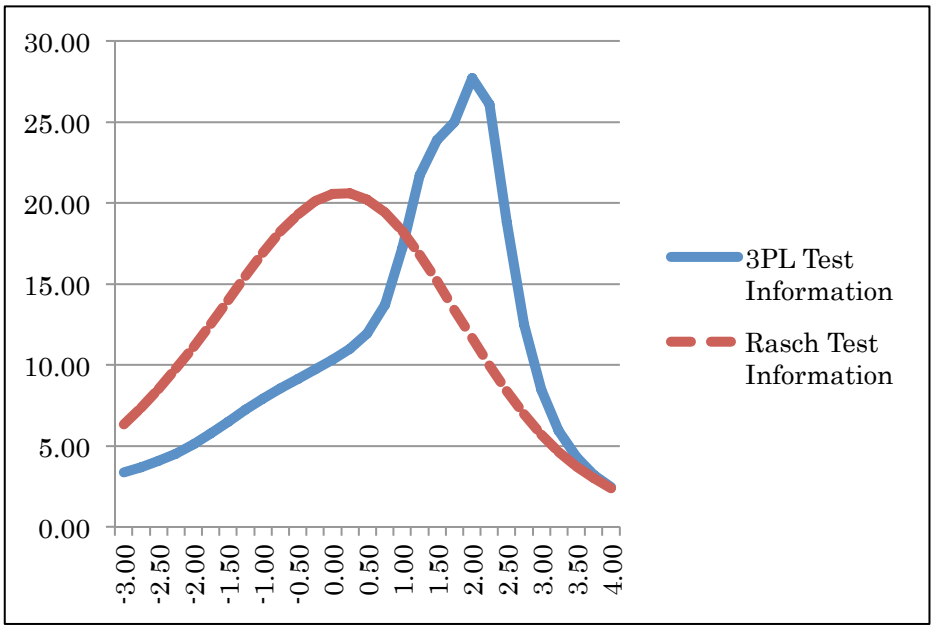

Figure 1. **Test information functions for Rasch and 3PL models**

Model selection

The two models tell us different stories about the test's reliability. How can we tell which is closer to the truth? To do this, we must make model fit comparisons, which can be performed in the statistical software program R (R Core Development Team, 2008), using the IRT package LTM (Rizpolous, 2006). In this issue, software columnist Aaron Batty explains how to get started with R using the LTM-compatible graphical user interface RKWard (though this particular feature of LTM must still be requested by command line).

When assessing model fit, it is important to consider that, to at least a negligible degree, nested models with more parameters (such as the 3PL when compared to the 1PL) will nearly always demonstrate superior fit, but that these solutions may simply represent an overfit of the model to the data set, and a solution that will not necessarily be generalizable to other samples (Zucchini, 2000). However, an earlier analysis by the authors on the practice test data set demonstrated that under several criteria, including the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), which penalize more complex models, the 3PL model displayed superior fit to the 1PL Rasch model. Therefore, the 3-parameter model can be considered to give a more accurate description of the test form's properties.

Table 8. Likelihood ratio table for Rasch and 3PL models

Model	AIC	BIC	Log Lik.	LRT	df	p
Rasch	114105.6	114558.3	-56961.81			
3PL	112644.2	114002.2	-56049.10	1825.43	182	<0.001

Examining test reliability under the selected model

Test information for different levels of student ability can be used to calculate standard errors (see Partchev, 2004), which are shown in the table below. Student ability under the 3PL model was equated to TOEIC Bridge Scale Scores, albeit crudely, using equipercentile equating. It should be noted these are rough estimates, as only a subsample of 491 for which the scores of both tests were available was examined, and score distributions were not smoothed. For a primer on more sophisticated methods of equipercentile equating, please refer to Livingston's eminently readable guide on the subject (2004), or Kolen & Brennan's authoritative book on test equating (2004).

Table 9. Standard errors and estimated scale scores for 3PL model.

Estimated TOEIC Bridge Scale Score	82	84	88	96	102	110	118	128	136	142	150	154
3PL Theta	-3.0	-2.5	-2.0	-1.5	-1.0	-0.5	0.0	0.5	1.0	1.5	2.0	2.5
Theta Standard Error	0.54	0.50	0.44	0.39	0.36	0.33	0.31	0.29	0.24	0.20	0.19	0.23
Test Information	3.4	4.1	5.1	6.5	7.9	9.1	10.3	12.0	17.3	23.9	27.7	18.8

The results suggest that standard errors are lower for students with ability levels between 1-2 logits than they are for students of mean ability (0). Under the 3PL framework, then, the test may be most reliable for test takers with scale scores between roughly 130 and 150. Due to the limitations of the smaller examined sample, it is not possible to equate higher scale scores with this data set, although under the 3PL model, standard errors should increase for test takers of higher ability.

Practical implications

An implication of these results is that a somewhat easier version of the test may result in higher test reliability for the majority of students, as the item difficulty would closer match the sample's mean ability level. To relate these findings to raw scores, as the TOEIC Bridge test is multiple-choice, it is exceedingly difficult for a student who answers every question to receive a score of less than 20%, even if they do not actually know any of the answers. Consequently, a score of 0 does not represent the "true" minimum score of the test, meaning a score of 50% does not represent the midpoint between minimum and maximum scores, or an ideal point to center the score distribution. Instead, items that result in a mean raw score of roughly 60% may be closer to ideal.

Taking this information into account, we have found that when we make our own norm-referenced multiple-choice tests for this student population, even if IRT models are subsequently ignored in favor raw scores and classical analyses, tests with means of 60% do, in fact, appear to result in higher reliability than tests constructed from the same item bank with means of 50% when other item properties are equal. Due to such experiences, we have found that although the large sample sizes required for estimation of some IRT models (Over 1000 for the 3PL, for

example) can make such studies troublesome to conduct, analyses of reliability under IRT can result in meaningful improvements to language tests.

References

Brown, J. D. (1999). Standard error vs. Standard error of measurement. *Shiken, 3*(1) p.20-25

De Ayala, R. (2009). *The theory and practice of item response theory*. New York: Guilford Press.

Embretson, S. E., & Reise, S. P. (2000). *Item response theory for psychologists*. Mahwah, NJ: Lawrence Erlbaum.

ETS. (2007a). *TOEIC® Bridge Examinee Handbook*. Retrieved September 10, 2010, from http://www.ets.org/Media/Tests/TOEIC_Bridge/pdf/TOEIC_BridgeExam.pdf

ETS. (2008a). *TOEIC Bridge kōshiki wākubukku*. Tokyo: Kokusai Bijinesu Komyunikēshon Kyōkai TOEIC Un'ei Iinkai.

ETS. (2010a). *About the TOEIC Bridge.* Retrieved September 10, 2010, from http://www.toeic.or.jp/toeic_en/bridge/about.html

ETS. (2010b). *TOEIC Bridge® Data & Analysis 2009.* Retrieved 10 September 2010, from http://www.toeic.or.jp/bridge/pdf/data/Bridge2009_DAA.pdf

Kim, J (2004) An Excel manual for item response theory. Retrieved 20 August 2012, from http://education.gsu.edu/coshima/EPRS8410/Sarah_Project1%2012%203%202004.pdf

Kolen, M. J., & Brennan, R. L. (2004). *Test equating, scaling, and linking.* New York, NY:Springer.

Partchev, I. (2004). A visual guide to item response theory. Retrieved 18 August 2012, from www.metheval.uni-jena.de/irt/VisualIRT.pdf

R Development Core Team. (2008). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from http://www.R-project.org

Rizpolous, D. (2006). ltm: An R package for Latent Variable Modeling and Item Response Theory Analyses. *Journal of Statistical Software, 17*(17), 1-25.

Stewart, J. (2012) Does IRT provide more sensitive measures of latent traits in statistical tests? An empirical examination. *Shiken Research Bulletin, 16*(1). 15-22.

Zucchini, W. (2000). An Introduction to Model Selection. *Journal of Mathematical Psychology, 44*(1), 41-61. doi:10.1006/jmps.1999.1276

The psycholinguistic approach to speaking assessment

An interview with Alistair Van Moere

Aaron Batty
abatty@sfc.keio.ac.jp
SRB General Editor

Alistair Van Moere is the vice president of product and test development at Pearson's Knowledge Technologies Group, and is responsible for the development, delivery, and validation of their automated language tests, including the Versant test (previously known as PhonePass). Versant is a telephone- or computer-mediated, computer-scored test of speaking proficiency, available in six languages, including English. Although computer scoring is widely used for multiple-choice test formats, Versant is revolutionary in its use of computer scoring of speaking proficiency. Van Moere and his colleagues have written extensively on the validation of the Versant test and its methods of automated scoring (e.g. Bernstein, Van Moere, & Cheng, 2010; Van Moere, 2010), and he has most recently raised eyebrows with his calls for a "psycholinguistic" approach to speaking assessment from the pages of Language Testing (2012).

Prior to his employment at Pearson, Alistair was instrumental in the development of the Kanda English Proficiency Test at Kanda University of International Studies in Chiba, and drew from his experience with its speaking test in his PhD work under Charles Alderson at Lancaster University—work which won him the Jacqueline Ross TOEFL Dissertation Award in 2010 (a portion of which can be read in Van Moere, 2006).

We were pleased that Alistair was willing to take some time out of his busy schedule to talk to us about the psycholinguistic approach to language assessment, the Versant tests, and the communicative/psycholinguistic divide.

In your recent article in *Language Testing*, you advocate a psycholinguistic approach. What exactly does this mean?

The theory underlying the psycholinguistic approach is that the speed and accuracy with which learners process language, even relatively simple language, tells us as much about their proficiency as more complex, communicative tasks can do. We don't actually need to elicit a variety of complex structures on a variety of topics while pretending to be in a variety of interactional situations in order to predict how well learners can handle themselves in a conversation.

A psycholinguistic approach refers to a particular way of (a) eliciting, and (b) scoring spoken responses. Elicitation uses constrained question-and-answer tasks, such as listen to a sentence and repeat it. Scoring involves three elements: accuracy, fluency and complexity. Let me give you an example. I'll say three sentence fragments and then ask you say them back in a correct, meaningful order: "to the cleaner / I gave my baby / to look after". Right. Native speakers will make sense of the meaning in a few milliseconds and then reel this out with barely a thought. When it comes to learners, though, some will say a fragment or two then trail off because they couldn't construct the meaning, some will say it correctly but slowly while they are processing it, some will say it

correctly but stammer over the articulation of certain words because their cognitive resources are so busy concentrating on meaning or word order. Every delay, stammer, stretched syllable, mispronunciation, missed word or missed word-ending, provides insight into the learner's psycholinguistic competence. And there's actually an interplay between accuracy, fluency and complexity. For example, learners tell us that when they focus on articulation their pronunciation or fluency scores go up, but they find it harder to focus on meaning and their accuracy scores go down, and vice versa. Naturally, as complexity or sentence familiarity get harder, both accuracy and fluency suffer.

The important thing here is that the tester, by means of controlled input and expected output, has a very standardized arena in which to judge the test-taker's speech for accuracy, fluency and complexity. This is in contrast to communicative tests, where test-takers have more control over the situation and may use various strategies to obscure or enhance the appearance of their linguistic range.

And how do you score psycholinguistic assessments?

Well the first thing to note is that psycholinguistic assessments do not require automated scoring. Sure, automated scoring makes the whole thing easier, but a teacher can administer these item-types in a five-minute test and rate responses for pronunciation, accuracy, and fluency, either on the spot or from recordings.

In our Versant tests, we score learner performances using our automated models. In the example of "I gave my baby to the cleaner," for example, we would conduct extensive field testing using that item. That is, we would gather the responses from numerous native and proficient speakers who performed on the sentence, and the responses from numerous learners of all proficiencies and many L1s. The native and proficient speakers provide us with the parameters of acceptability in the utterance, i.e. where linking or reduction occurs, where it's acceptable to pause inter-word, and so on. The learner responses tell us how difficult or complex the sentence is based on the number of word errors or other mistakes. As we have large item-banks we use partial credit Rasch modeling to determine item difficulty estimated from response data provided during field testing. The whole approach is empirical and quantifiable, and provides very granular scores. I tried to provide a readable explanation on this in my 2010 paper.

[Further information can be found in Pearson Education, Inc., 2011a. –Ed.]

What exactly do you mean by language assessments with "automated scoring"?

It refers to artificial intelligence systems that have been developed to assign scores like a human rater. We score both spoken and written responses and we use speech recognition, natural language processing and statistical modeling techniques. We do it by capturing test-taker performances from field testing, and having them marked by a pool of expert raters. Then, we analyze the data to establish which features of the performance are associated with high or low scores, and ultimately train an engine to assign scores like a human rater.

It's quicker and more reliable than human scoring, as long as the test is properly designed. I explain how it's all done in a series of short video clips here:

http://goo.gl/zZAjP

Could you give our readers a quick idea of how you scale these data? What model(s) do you use? How does the computer assign scores?

Wow, those are big questions and it's difficult to give a brief answer, but here goes. The spoken Versant tests last anything from 8 to 25 minutes and produce scores on a 20 – 80 scale for each of 4 subskills: Sentence Mastery, Vocabulary, Fluency, and Pronunciation. The overall score is a weighted combination of these subcores. For pronunciation and fluency, a panel of expert human raters applies rating criteria on a 0 – 6 scale to each field test response. Speech processors extract information such as the speed and delay in the response, the position and length of pauses, the segmental forms of words, the pronunciation of word segments. These features are used in models to predict the pronunciation and fluency of new, unseen responses. Sentence mastery and vocabulary are a combination of Rasch difficulty estimates from constrained items and predicted human judgments on criteria with a 0 – 6 scale. Rasch logits and criteria judgments are combined and put onto a normal distribution with 20 – 80 scale, according to global large-scale field testing. It's explained more fully in the Bernstein, Van Moere and Cheng (2010) article.

The Versant test uses computer speech recognition to assess learner speaking proficiency. According to the Pearson website, this technology was developed in-house. Allowing that speech recognition software has improved greatly over the years, most commercially-available packages are still pretty bad at it. What makes you (and Pearson) confident that you are getting accurate readings from your software, when, for example, the technology behind Apple's Siri, which has been in development for 40 years, has trouble understanding anyone who doesn't have a North American accent?

There several differences between Siri and the Pearson system. One advantage our speech recognizer has over other systems is that we've optimized it for learner speech. It means that we've developed it to recognize the pronunciations that learners have, so for example we can credit the learner for the content of their speech and debit them for mispronunciations, and the automated scoring does not confuse the two.

But the real difference between Siri and Pearson's system is that Siri doesn't know what you are trying to say, whereas the Pearson system usually does. Not only that, but during our test development we have presented the test items to a sample of learners and gathered their responses, so we know how test-takers respond to each item and what kind of words they use. Some item-types elicit highly constrained responses (e.g. read this passage; repeat this sentence). In this case, we are not conducting speech recognition so much as alignment of the test-taker's response with an existing set of response parameters. Other item types require the test-taker to construct a response, but they are bound by certain input or topics (e.g. describe this image; summarize this lecture). In this case the recognizer can anticipate certain frequently used words and word strings.

At Pearson our approach is to design tests that are compatible with the automated scoring technology. That means we maximize the things that the machine is good at and minimize what it isn't good at. It means carefully selecting our tasks and content. Frankly, it sometimes also means restricting our construct. But that's not necessarily a big limitation; every test is a compromise between length, reliability, task variety and construct.

We have other proprietary techniques embedded in the processing of data which ensure a good match between human evaluations of speech and the machine scores. But the proof is in the validation data. Machine scores correlate with human judgments at around 0.97 and 0.98, so the system is highly accurate.

What do you say to those who criticize your approach to speaking testing as too "form-focused"? Isn't the ability to communicate one's ideas more important than correct grammar or the ability to parrot back something they have heard? In real life, no one asks you to do anything remotely like the tasks you describe in your most recent *Language Testing* article (sentence repeat and sentence build). Why bother testing something that no one will ever be expected to do?

A big section of the article on the psycholinguistic approach is devoted to explaining why these tasks are more reflective of real-life communication than most people realize. That's why I review the literature on memory, retrieval, and chunking. Language production is highly memory-based, and when constructing sentences we frequently re-use chunks of language that we've previously encountered in speech because it's more cognitively resource-efficient.

It's also a misconception that sentence repetition is the same as "parroting"; I doubt that many informed academics believe that anymore. There is a large body of literature on this in the field of second language acquisition (where it is referred to as elicited imitation) and the task is generally acknowledged to involve comprehension of meaning and subsequent reproduction of meaning through form.

That said, it's true that the psycholinguistic approach is form-focused. When I said earlier that every mispronunciation or missed word ending provides information about language proficiency, well, that drives some communicative language testers absolutely loopy. For them, spoken performance testing is about conveying a meaningful message or demonstrating communication strategies, and not about dropping a suffix or measuring how long test-takers grope for vocabulary. And I don't disagree. But what we've found is that even minor errors of form can tell us a lot about the speaker's automaticity with the language. They are powerful indications of processing competence and can be excellent predictors of language proficiency and conversational ability, from beginner to advanced. Versant tests have been likened to other predictive tests in the field of medicine. For example, checking blood pressure takes only a minute and predicts very well whether the patient would get out of breath walking up a mountain. This saves you actually having to follow them up a mountain with your clipboard!

So you think there is something wrong with the communicative approach to testing?

The communicative approach is a work in progress, and we need to keep researching it. For me, the biggest limitation is the lack of empirical evidence underpinning our models of communicative competence. The field has defined various abilities or competences, but there is no way to prioritize them in terms of importance or measure exactly how they interact. For example, take organization. Now that seems to me to be an ability that is borrowed from L1. If you have good organizational skills in L1 then you can transfer it over to L2. But if you don't have it in L1, it's hard to see how you can ever have it in L2. Now you might say that organization is language specific – for example, in China when you write an essay you don't write your thesis statement in your introduction, but rather hold it back until the conclusion, and sort of hit your reader with it as the culmination of your argument. But that is just a different form of organization that you can apply in L1 if you wanted to. It's not something that you could only apply in one language and not another language. So, much of organization is a language-independent skill.

Pragmatics is another area that I struggle with. It seems to me to consist mostly of grammar and vocabulary plus emotional intelligence. You are only able to express yourself in an appropriate register if you have the emotional intelligence to understand what is appropriate and the grammar to carry it out. So it appears to me that a large part of pragmatics consists of context-specific

grammar structures (form) in the target language plus a non-linguistic trait that is transferable across languages, in addition to a sociolinguistic ability of selecting the right grammar and vocabulary for the right situation. So, I think research needs to focus on identifying the contribution of the different components of CLA [(Communicative Language Ability) –Ed.] and finding out how much of language proficiency is attributable to core language skills (e.g. grammar, vocabulary, automaticity), how much is attributable to non-linguistic, transferable skills (e.g. critical thinking, organization, emotional intelligence) and how much is genuinely an ability that exists in the L2 illocutionary and sociolinguistic competences, irrespective of the L1.

How have your views on spoken language testing evolved?

Although I still believe in communicative frameworks to help us define constructs and design tests, my research experiences have led me to focus more on concrete, core skills that are essential for meaningful communication.

When I was at Kanda University we ran group discussion tests for thousands of students for placement and progress-monitoring. I thought it was a very useful communicative test—three or four students discussed a prompt and were evaluated by two raters using an analytic scale. The performances provided seemingly rich, social interactions. I researched this for my PhD, and as I conducted more analyses I gradually became doubtful about the whole approach. In a study that I co-authored with Miyoko Kobayashi, we found that a large portion of the score variance was attributable simply to the amount that students spoke, even after controlling for proficiency. So if two students were the same proficiency level, the one that spoke more scored significantly higher. In a study I conducted with William Bonk, we discovered that outgoing students scored higher than average or shy students. Gary Ockey subsequently ran this study more rigorously than Bonk and I, and he published it in Language Testing, but his findings were basically the same. Then during my PhD I looked more carefully at the score reliability and found that students' scores really varied quite a lot over consecutive test occasions, and so were too unreliable for anything but low-stakes decision-making. I also found through discourse analysis that the task itself elicited an alarmingly limited range of functional interactions.

So, I was getting despondent about these findings and having a kind of crisis of faith about our ability to reliably measure talk-in-interaction. At that time I started working on the Versant tests. I wasn't at all convinced at first. Like many people, my reaction was: "This is just repeating sentences. It's not assessing communicative skills." But it took me about a year to change my mind. I had to understand the scoring, and analyze the data for myself. And I saw the tremendous reliability, and also how the Versant scores were correlated with interview tests. Then I began to realize that we can complement the communicative approach with more reliable, more standardized tests. So that's why I advocate a psycholinguistic approach in addition to a communicative approach.

[For a full report on the validation and reliability of Versant tests, see Pearson Education, Inc., 2011b –Ed.]

Thank your for taking the time to speak with us. Your ideas are always intriguing. To wrap up, what can we expect to see from you in the future?

We are developing a four-skills test for young learners on the iPad which involves very interactive, tactile item types, including speaking and writing activities. We've just launched a 4-skills, computer-based placement test for universities which assesses CEFR A1 through B2, and which provides immediate scores. We are also in the process of launching a business English test for Ja-

pan and Korea, which is going to be just 90 minutes for all 4-skills, and will be practical, reliable, and very accessible.

In R&D we are also working on test security features such as speaker verification. This helps you ensure that the person sitting in front of you holding a test certificate is indeed the same person that took that test and earned that score. You could have them jump on the phone or computer, offer a speech sample, and we'd compare it to the test and took and confirm whether it's an exact match or not. There is a lot happening and over the next few years I've no doubt that we're going to wow people with automated scoring technology even more than now.

References

Bernstein, J., Van Moere, A., & Cheng, J. (2010). Validating automated speaking tests. *Language Testing*, *27*(3), 355–377. doi:10.1177/0265532210364404

Pearson Education, Inc. (2011a). Research. *Pearson – Versant spoken language tests, patented speech processing technology, and custom text services*. Corporate Website. Retrieved October 15, 2012, from http://www.versanttest.com/technology/research.jsp

Pearson Education, Inc. (2011b). *Versant™ English Test: Test description and validation summary*. Palo Alto, CA: Pearson. Retrieved from http://www.versanttest.com/technology/VersantEnglishTestValidation.pdf

Van Moere, A. (2006). Validity evidence in a university group oral test. *Language Testing*, *23*(4), 411–440. doi:10.1191/0265532206lt336oa

Van Moere, A. (2010). Automated spoken language testing: Test construction and scoring model development. In L. Araújo (Ed.), *Computer-based Assessment (CBA) of foreign language speaking skills, Joint Research Centre scientific and technical reports* (pp. 84–99). Brussels: Publications Office of the European Union. Retrieved from http://publications.jrc.ec.europa.eu/repository/handle/111111111/15037

Van Moere, A. (2012). A psycholinguistic approach to oral language assessment. *Language Testing*, *29*(3), 325–344. doi:10.1177/0265532211424478

Software Corner:
RKWard: Installation and use

Aaron Olaf Batty
abatty@sfc.keio.ac.jp
SRB General Editor

Many readers of *SRB* are likely familiar with the free, open-source statistical software package R. R offers a staggeringly wide range of analyses and outputs, besting every other general statistics package on the market, for free. This value, however, comes at a cost that may seem too steep for many users: it is command-line only. One must learn to write R code to take advantage of this software, and although the code itself is relatively memorable and intuitive, it cannot compete with the ease of use offered by the graphical user interfaces (GUIs) of competing packages, such as SPSS/PASW—a software package whose price can run over a thousand US dollars.

What is RKWard?

RKWard seeks to lower the usability barrier to entry into statistical analysis with R by providing a GUI frontend that hides much of the code from the use (although it is easy to inspect in the console for those who wish to do so). Furthermore, it streamlines the sometimes-maddening process of importing data into R, and can read SPSS and Stata files natively in addition to standard text formats. Finally, its general look-and-feel will be immediately familiar to those who have used SPSS, further easing entry for novices. It has been in development since 2005, and achieved its first stable release in 2011.

Although it uses the KDE desktop environment, which is usually found on Linux systems, it is not limited to computers running Linux/Unix. It also has a Windows version, and since the Macintosh OS is based on FreeBSD (another Unix variant), it can also be used on the Mac, perhaps making RKWard the only free, GUI-based, cross-platform statistical package available.

RKWard not only performs traditional statistical analyses (e.g., descriptive statistics, correlation, *t*-tests, regression), but, of particular interest to the *SRB* readership, with the LTM package installed, it can also perform a wide range of item response theory (IRT) analyses. For free.

However, as is the case with many FOSS (free, open-source software) packages, getting started can still be somewhat daunting. This goal of this article is to walk you through obtaining the software, installing the software on either a Windows or a Macintosh computer, importing a data file, and becoming familiar with the interface.

Obtaining and installing the software

As of late October, the stable version of RKWard is version 0.6.0. It is easily installed on Windows and MacOS X 10.7 “Lion” or above, and less-easily installed on MacOS X 10.6 or earlier. For step-by-step screencasts of each of these three installation scenarios, please see my YouTube channel, under the name “AaronBattyVideos”.

Windows

1. *Obtain the software*

The RKWard project page for Windows can be found at the (shortened) URL below (or linked from the installation video on YouTube):

http://goo.gl/FPTNM

Scroll down the page to the section titled "Standard Installation: Complete installation bundle" and click the link to the "binary installation bundle" to download the installer (134 MB).

2. *Install the software*

Once the file has downloaded, simply double-click it to install. You will need to give it permission, etc. When it concludes, click "Close", and a new shortcut will be on your desktop. This can be moved to any location you wish, of course, but before launching the software:

3. *Set to run as administrator*

There can be some trouble importing data files if RKWard is not run as administrator (NOTE: This does not apply to systems running Windows XP). To permanently set that option, right-click the shortcut, select "Properties," and in the "Shortcut" tab (which is most likely what the Properties dialog opened to), click the "Advanced" button. In the window that appears, tick the "Run as administrator" box. Click "OK" on the windows to dismiss them. See Figure 1.

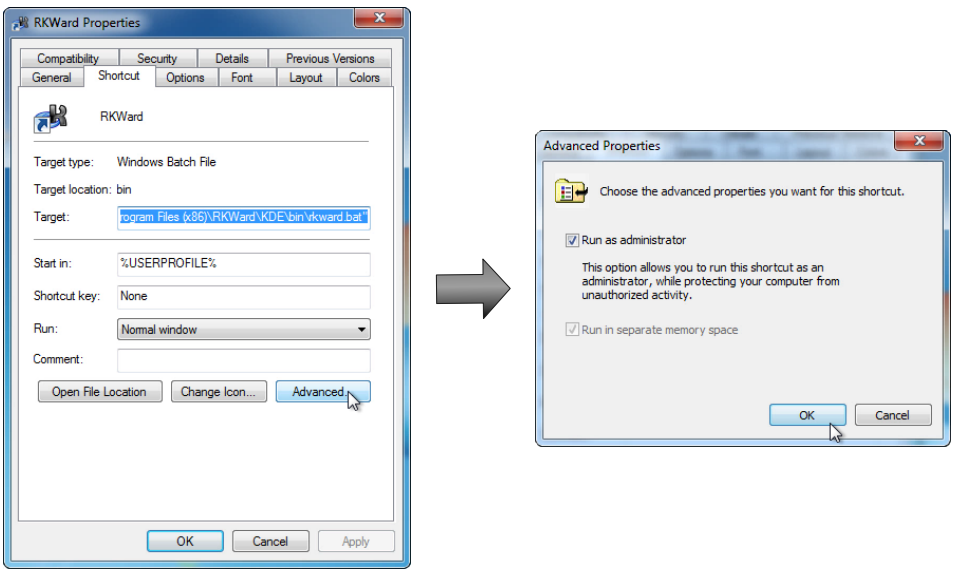

Figure 1. **Setting the RKWard shortcut to run as administrator.**

4. *Launch RKWard*

Simply double-click the shortcut to launch the program. You will need to verify its permission to run as an administrator. You will also likely receive a pair of errors about a KDE file being unwritable, but I have yet to find any problem with this, and they only appear on the first run after install. Simply click "OK" to dismiss these (see Figure 2).

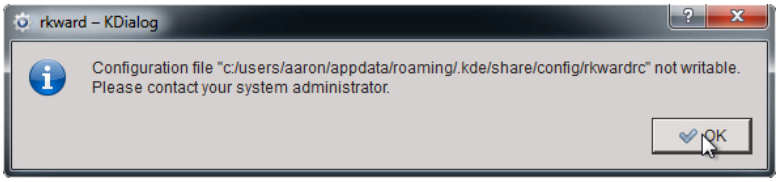

Figure 2. **One of two KDE error messages that may appear on the first run of RKWard. They can simply be dismissed.**

As soon as the software launches, a warning message will appear recommending that you disable native file dialogs (i.e., the familiar Windows file dialogs) for stability. The KDE file dialogs are not materially different from those of Windows, so there is no reason not to disable the native dialogs (see Figure 3).

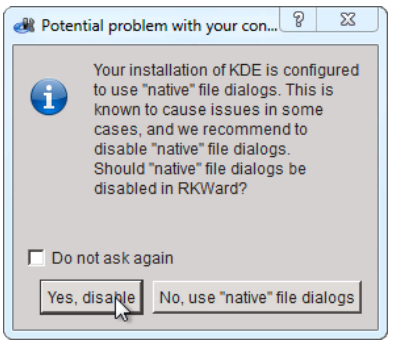

Figure 3. **Warning message regarding native file dialogs. Disable them.**

After disabling native file dialogs, the software will run normally. Skip ahead to the section on using RKWard.

MacOS X 10.7 "Lion" or 10.8 "Mountain Lion"

The process for installing RKWard on a Macintosh running one of the latest two versions of the Macintosh operating system is similar to that of Windows, althought the download is considerably larger.

1. Obtain the software

The RKWard project page for Windows can be found at the (shortened) URL below (or linked from the installation video on YouTube):

http://goo.gl/T6B9h

Scroll down the page to the section titled "Installing using the precompiled binary bundle (experimental!)". Although this method is officially experimental, I have used it on three Macintoshes without issue. If you are uncomfortable using an experimental installation method, the "proper" installation method is explained below in the section on installation on older Macintosh operating systems.

Click on the text "precompile a .dmg archive which includes KDE, R and RKWard" to visit the list of files for Macintosh.

You must download the approximately-915 MB file linked at the top of the page (see Fig. 4).

Figure 4. **The precompiled installation bundle for MacOS X 10.7 and 10.8. Please open this link in a new tab.**

2. Install the software

After the file downloads, simply mount the .dmg and double-click the installer, as you normally do to install software. (NOTE: If you are using OS X 10.8 "Mountain Lion", and you have Gate-Keeper enabled, you will need to right-click and select "Open" to start the installer.)

3. Launch RKWard

You will now find a folder named "RKWard" in your Applications folder. Open this and start RKWard by double-clicking.

MacOS X 10.6 "Snow Leopard" or older

Installation on older Macintosh operating systems requires the following steps:

1. Installation of Xcode from the OS installation disc, or from one of the discs that shipped with the computer, and the installation of relevant updates
2. Installation of MacPorts from the project's web page
3. Setting up the environment from Terminal
4. Building RKWard from the Terminal (this process may take several hours, but you do not have to be present, and can use the computer for other things while it works)
5. Running final dbus terminal commands
6. Launching RKWard

I will not go into detail on this process here. This is the process by which I first installed RKWard, and it was not very difficult. If you are extremely uncomfortable with the terminal, it may not be for you. However, just as in the case above, most of the commands can simply be copy/pasted from the RKWard web page (linked above), and once the installation is complete, it works perfectly with no further tweaking. This process is also possible with later versions of MacOS X. If you are interested, please view my screencast on YouTube, which should demonstrate how feasible this approach actually is.

Using RKWard

Starting RKWard

(NOTE: All following screencaps are from a Macintosh running RKWard; however, because RKWard runs in the KDE environment, the usage and the vast majority of the visual elements are identical to those when running under Windows.)

Upon starting RKWard, you will be given a choice of startup options (see Figure 5):

Figure 5. **Starting RKWard.**

- *Start with an empty workspace.* This starts a new session with no data table. This is ideal for situations when you are starting a new analysis and will be importing data.
- *Start with an empty table.* This starts a new session with an empty data table into which you can type your data. Pasting is also ostensibly supported, but is frustrating, and not recommended.
- *Load workspace from current directory.* If you have a saved workspace in the directory that RKWard is using, you may use this option.
- *Load an existing workspace.* If you have saved a previous session (a workspace), it will appear in the list below this option. This is analogous to SPSS's feature for opening previous datasets, but is superior in that it will also load the outputs as well, in a separate tab. If you find yourself using RKWard frequently, this allows you to have a list of separate projects saved with all of their datasets and outputs saved together in workspace files.

The default is "Start with an empty table," but in most cases it will be easiest to start with an empty workspace and import your dataset, which you have already formatted elsewhere (e.g., in Excel, SPSS, Stata, and/or a text editor). Regardless of the choice you make on startup, you will be greeted with a welcome/documentation page.

Importing data

Importing text/CSV files

If you have your data in Excel (or another spreadsheet package), simply export it as CSV (comma-separated values) or a tab-delimited text file (see Batty, 2012 for detailed instructions on exporting from Excel).

To import your dataset, go to the File menu and select:

File → Import → Import format → Import Text / CSV Data

It is also possible to simply select "Import data" from the "Import" sub-menu, but I find that specifying the file time from the outset saves a step.

The “Import Text / CSV data” window opens (see Figure 6). Click the folder button next do the red-highlighted “File name” field. A file picker dialog will open, from which you can select your CSV or other text-format data file. In this example, I will be using a CSV file.

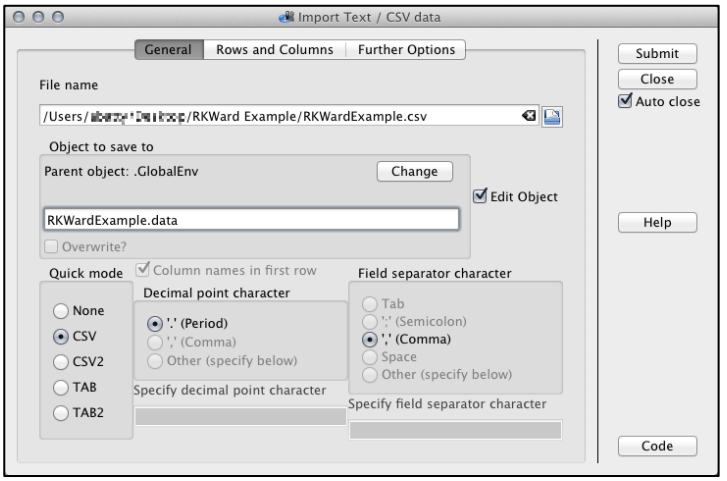

Figure 6. **“General” tab of the “Import Text / CSV data” dialog, with options set for a CSV data file.**

The “General” tab of the dialog has the following sections:

- *File name.* This is where you specify what file to load.
- *Object to save to.* This saves to the global environment by default (which is likely where you want to save your data), and allows you to rename the data table as it will appear in RKWard.
- *File format settings.* The bottom row of options tells RKWard how to read your data file. In most cases you can simply choose “CSV” or “TAB” and accept the default settings.

Other settings to help RKWard interpret your data file can be found on the “Rows and Columns” and “Further Options” tabs, but usually, as long as your variable names are in the top row of your text table, there is no need to change them.

After importing, you may change meta-information about the imported variables at the top of the data viewer (see Figure 7), although usually RKWard does a fairly job of understanding the data types upon import. This area has the following rows:

- *Name.* This is the short name for the variable.
- *Label.* Here you can specify a longer name/label for the variable.
- *Type.* There are four data types: Number (default), Factor (categorical data), String (nominal text), and Logical (true or false).
- *Format.* This is where you can change the alignment, number of decimals, etc.
- *Levels.* Levels of categorical variables.

In Figure 7, I am changing the type of my first variable to “Factor,” because it represents students of four different ability levels, not continuous data ranging from 1 through 4.

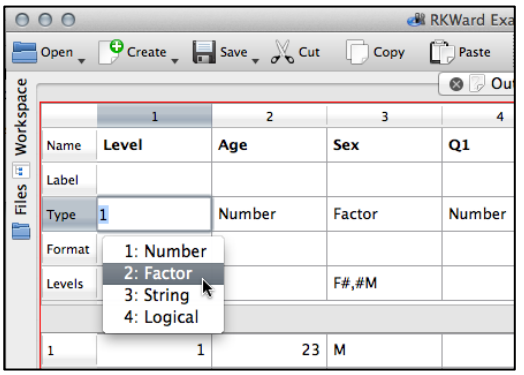

Figure 7. **The data-formatting section of the data table interface.**

Importing SPSS or Stata data files

Importing data files from other statistical applications is very straightforward, and follows the same basic procedure as importing text data files, but more information about the variables can be imported without RKWard having to interpret the data.

To import your dataset, go to the File menu and select:

File → Import → Import format → Import SPSS

The workspace browser

When you start RKWard, the welcome message, output, or data table will likely take up the entire window, but there are several other panes that can be very helpful:

- *Workspace pane.* This can be found on the upper left-hand corner of the browser, and displays the active packages and datasets in the session (see Figure 8). They can also be removed, renamed, etc. from this pane, so I usually leave it open.
- *Files pane.* The button for this is directly below that for the workspace pane, and displays the contents of the current working directory.
- *Command log pane.* The button for this resides at the lower left-hand corner of the window, and displays a log of the commands RKWard is issuing to R in the background. This can be useful if you are familiar with R, but if not can be ignored.
- *R Console pane.* The button for this is directly to the right of that for the command log pane. This opens a miniature R console for running analyses using the command line. This is especially useful for using commands that have not been incorporated into RKWard yet, without leaving RKWard.
- *Help search pane.* The button for this pane is located directly to the right of that for the R console. It displays the help files for the various R packages, which is of limited use for RKWard users.
- *R engine status and interrupt.* In the lower right-hand corner of the window, there is an "R" in a colored box. If the box is green, the R engine is idle; if it is yellow, it is starting; if it is red, it is working. This is extremely useful for determining whether the reason a result has not appeared is due to a mistake, or if R is still calculating it. To the right of the indicator is

a stop button, which interrupts R. This is useful in cases when an analysis is taking too long and you wish to cancel it.

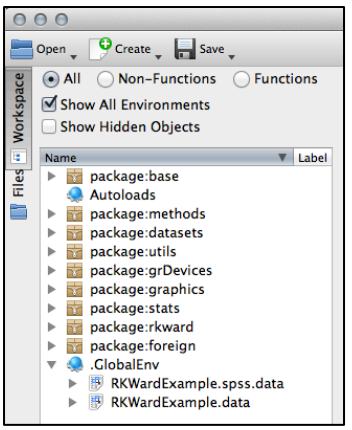

Figure 8. **Workspace pane.**

Running analyses and plots

All analyses are available from the "Analysis" menu, and are fairly straightforward to use. In cases where additional packages are necessary to run the analysis in question, a dialog will ask you to identify the nearest server, and download the required packages automatically. A wide variety of plots are also available under the "Plots" menu. Results will appear in the output tab.

t-tests

Since *t*-tests are extremely common, it is worth mentioning that RKWard does not allow independent sample *t*-tests of "long format" data, meaning two groups delineated by a value in a separate variable (e.g., "M" and "F"). To carry out this kind of analysis, place the values for the groups in their own variables (perhaps in different data tables to avoid confusion).

Exporting outputs

Unless you can use the results of analyses in other documents, they are of limited value. Luckily, RKWard's output file is HTML-formatted, making it very easy to work with. To export your output, Simply go to:

File → Export page as HTML

This will create an HTML file that can be opened in any browser to copy/paste out tables and/or graphics. It is important to note, however, that the graphics are located elsewhere on your computer, and you may want to make an additional copy of them before moving on to another project.

Saving your workspace

Finally, when quitting RKWard, it is a good idea to save your workspace. This preserves all the data, settings, packages, and outputs of your project in one small file, allowing you to return to it at any time. This function is accessible from the following:

Workspace → Save Workspace

The workspace can be saved anywhere you wish, and once it is saved, you can quit RKWard and return to it at any time.

Conclusion

This guide is intended to introduce a free, feature-packed software package for statistical analysis. It is hoped that this piece will encourage readers to install and explore the software over the next few months before the next issue of *SRB* is released, in which I will introduce the IRT functions available in RKWard via the LTM package for R.

References

Batty, A. O. (2012). Software Corner: jMetrik 2.1. *Shiken Research Bulletin*, *16*(1), 34–42.

Statistics Corner:

How do we calculate rater/coder agreement and Cohen's Kappa?

James Dean Brown
brownj@hawaii.edu
University of Hawai'i at Mānoa

Question:

I am working on a study in which two raters coded answers to 6 questions about study abroad attitudes/experience for 170 Japanese university students. The coding was done according to a rubric in which there were 4 – 8 possible responses per question. Since most—if not all—of the data is categorical, I have heard that Cohen's Kappa is the most common way of ascertaining inter-rater agreement. What is the best way to actually calculate that? Since more and more people are moving away from single-rater assessments to multi-rater assessments, this question should be relevant to *Shiken Research Bulletin* readers.

Answer:

In order to address your question, I will have to describe the *agreement coefficient* as well as the *Kappa coefficient*. I will do so with a simple example, then with the more complex data that you have in your study.

Simple agreement coefficient example

In the realm of ratings or codings (hereafter simply called codings) of various categories, an *agreement coefficient* can be used to estimate the proportion of codings assigned by two raters or coders (hereafter simply called coders) that coincide. In the simplest scenario, let's say that two coders listen to the interview data of 120 students who were interviewed just after returning from a study abroad experience. After listening to each interview, each of two coders is required to decide if the student was generally positive about the living abroad experience or generally negative. In other words, the coders are required to code each student's experience as positive or negative. Figure 1 illustrates how we need to lay out the results for the two coders in order to calculate an agreement coefficient.

In some cases, the codings agree between the two coders. When the two assigned codings for a student are both positive, that student is counted in cell A; when the two assigned codings for a student are both negative, that student is counted in cell B. The other cells indicate that the two coders disagreed in their codings (i.e., Coder A assigned a positive coding, but Coder B assigned a negative one, or vice versa). Notice that the row totals Row1 and Row2 are given to the right of Figure 1, and column totals Col1 and Col2 are given at the bottom. Notice also that the grand total (also affectionately known as *N*) is shown at the bottom right.

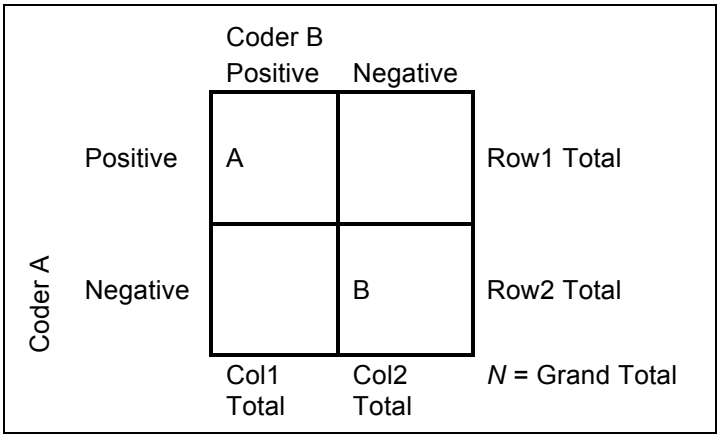

Figure 1. **Layout for positive and negative coding data for two coders.**

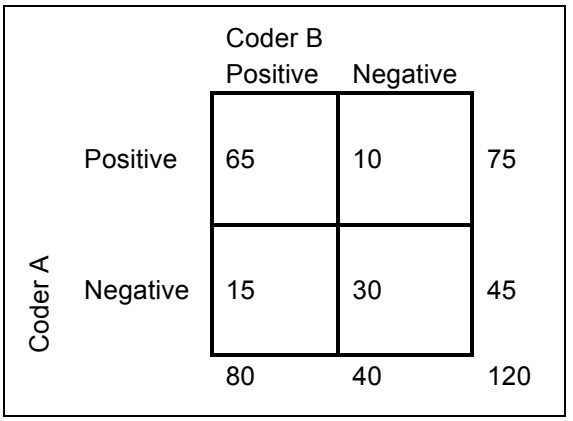

Figure 2. Sample data for positive and negative coding data for two coders.

Coming back to our imaginary scenario, notice in Figure 2 that 65 out of the 120 students were classified as positive by both coders, while 30 others were classified as negative by both coders. In addition, 25 students (10 + 15 = 25 students) are classified differently by the two coders.

With this information in hand, the following formula (Equation 1) can be used to calculate the agreement coefficient:

$$p_o = \frac{A + B}{N} \tag{1}$$

where:

p_o = agreement coefficient (or proportion observed)
A = number of agreed codings in cell A
B = number of agreed codings in cell B
N = total number of codings

Substituting the values found in Figure 2 into the equation, we get (Equation 2):

$$p_o = \frac{65 + 30}{120} = \frac{95}{120} = .7916666 \approx .79 \quad (2)$$

Thus, the agreement coefficient in this case is about .79, which means that the coders classified the students in the same way about 79% of the time. Note that by extension, the coders disagreed 21% of the time (100% - 79% = 21%).

If all students were coded exactly the same way by both coders, the coefficient would be 1.00 [e.g., $(A + B) / N = (70 + 50) / 120 = 1.00$], so 1.00 is the maximum value the agreement coefficient can have. However, unlike the reliability coefficients (for more on this concept, see Brown, 1997, 2002) that researchers often use, the agreement coefficient can't be lower than what would result by chance. Put another way, with 120 students, we might reasonably expect 30 students per cell by chance alone. This would result in a coefficient of .50 [$(A + B) / N = (30 + 30)/120 = 60/120 = .50$]. Thus, no agreement coefficient can logically be any lower than what would occur by chance alone—in this case, no lower than .50. This is very different from reliability estimates, which can go as low as .00.

Simple Kappa coefficient example

The *Kappa coefficient* (κ) arose (due to Cohen, 1960) to adjust for this chance-lower-limit problem by providing an estimate of the proportion of agreement in classifications beyond what would be expected to occur by chance alone. The adjustment is given in the following formula (Equation 3):

$$\kappa = \frac{(p_o - p_e)}{(1 - p_e)} \quad (3)$$

where:

κ = the Kappa coefficient
p_o = agreement coefficient (or proportion observed)
p_e = proportion classification agreement that could occur by chance alone,
In this case: $p_e = (Row1 \times Col1) + (Row2 \times Col2) / N^2$ [in this case]

Before calculating κ, a researcher must calculate p_o and p_e for the particular data involved. We have already calculated $p_o = .7916666$ for the data in Figure 2; the calculations for p_e for the same data are as follows (Equation 4):

$$p_e = \frac{(Row1 \times Col1) + (Row2 \times Col2)}{N^2} = \frac{(75 \times 80) + (45 \times 40)}{120^2} = \frac{6000 + 1800}{14400} = \frac{7800}{14400} = .5416666 \quad (4)$$

Notice that we are calculating p_e by (a) multiplying the total for the row in which cell A is found by the column for cell A (Row1 × Col1), (b) multiplying the row total for cell B times the column total for cell B (Row2 × Col2), then (c) adding the two results together [(Row1 × Col1) + (Row2 × Col2)], and (d) dividing the whole thing by N^2. In doing so, we are calculating the proportion of the expected frequencies for cells A and B.

Since the p_o for the data in Figure 2 was .7916666 and now we know that p_e is .5416666[1], Kappa turns out to be (Equation 5):

$$\kappa = \frac{(p_o - p_e)}{(1 - p_e)} = \frac{(.7916666 - .5416666)}{(1 - .5416666)} = \frac{(.25)}{(.4583334)} = .5454544 \approx .55 \quad (5)$$

This Kappa coefficient is an estimate of the coding agreement that occurred beyond what would be expected to occur by chance alone. It can also be interpreted as a proportion of agreement, .55 in this case, or as a percentage of agreement, 55% in this case. Unlike the agreement coefficient, Kappa represents the percentage of classification agreement beyond chance, so it is not surprising that Kappa is usually lower than the agreement coefficient. Like the agreement coefficient, Kappa has an upper limit of 1.00, but unlike the agreement coefficient, Kappa has the more familiar .00 lower limit.

Your more complex agreement coefficient example

In the case of your more complex coding criteria, your data should be laid out as shown in Figure 3 for your Question 1. Figure 4 shows the labels and examples you sent me for your Question 1 rubric. Notice that instead of cells A and B (as in the simpler example), we are focused here on agreements in cells A, B, C, D, E, and F.

		Coder B						
		5	4	3	2	1	0	
Coder A	5	A						Row1
	4		B					Row2
	3			C				Row3
	2				D			Row4
	1					E		Row5
	0						F	Row6
		Col1	Col2	Col3	Col4	Col5	Col6	N

Figure 3. **Layout for categories 5, 4, 3, 2, 1, and 0 data for two coders for question 1 in your study.**

Your actual data are displayed in Figure 5. Notice that the agreements in cells A, B, C, D, E, and F (at 74, 21, 1, 9, 20, and 25, respectively) are fairly high relative to the other cells. That is an indication that the two raters are tending to agree with each other in their ratings.

[1] Note that, for the sake of accuracy, I keep all the decimal places that my 100-Yen-shop calculator gives me until the very last step where I round the result off to two-places in order to be consistent with APA formatting.

5 = A strong interest in SA	Ex: I'm very interested in abroad study because I want to speak English fluently. [T1]
4 = A mild interest in SA	Ex: A little. But, I like to go abroad. [K4]
3 = Neutral and/or ambivalent	Ex: It's so-so. I want to study abroad, but I don't have money. And you? [T80]
2 = Little interest in SA	Ex: I have just little interested about study abroad. [K28]
1 = A strong disinterest in SA	Ex: I'm nothing. I like Japan school. And you? [T90]
0 = No response	

Figure 4. **Coder options for question 1 in your study.**

		Coder B						
		5	4	3	2	1	0	
Coder A	5	74	0	0	0	0	0	74
	4	0	21	0	1	1	0	23
	3	1	5	1	3	0	0	8
	2	0	2	0	9	0	0	11
	1	0	0	0	2	20	0	22
	0	3	1	0	1	0	25	30
		78	78	29	10	16	21	25

Figure 5. **The actual data for categories 5, 4, 3, 2, 1, and 0 for two coders for question 1 in your study[2].**

With Figures 3 and 5 in hand, the calculation of the agreement coefficient is simple with the following equation (6):

$$p_o = \frac{A + B + C + D + E + F}{N} \tag{6}$$

where:

p_o = agreement coefficient (or proportion observed)
A to F= number of agreed codings in cells A to F
N = total number of codings

[2] You asked me in an aside if this could be a Likert item (see Brown, 2000, 2011). I'm inclined to say "no" because you really are interested in how consistently the coders judged data to be in these different categories (including "no response). If you had been asking the students themselves to rate their experience using this scale, then I would say that it should be analyzed as a Likert item. Even if all the above were not true, your other questions are all clearly nominal in nature, so it is probably best if you use agreement and Kappa to consistently analyze all of your questions the same way.

Substituting the values found in Figure 5 into the equation, we get (Equation 7):

$$p_o = \frac{74 + 21 + 1 + 9 + 20 + 25}{170} = \frac{150}{170} = .88235294 \approx .88 \quad (7)$$

Thus, the agreement coefficient is about .88, which indicates that the coders agreed in their classifications of students about 88% of the time (and that they disagreed 12% of the time).

Your more complex Kappa coefficient example

Recall that the Kappa coefficient arose to adjust for this chance-lower-limit problem by providing the proportion of consistency in classifications beyond what would be expected to occur by chance alone and that the adjustment is given in the following formula (Equation 8):

$$\kappa = \frac{(p_o - p_e)}{(1 - p_e)} \quad (8)$$

where:

κ = the Kappa coefficient
p_o = agreement coefficient (or proportion observed)
p_e = proportion classification agreement that could occur by chance alone
In this case:
p_e = [(*Row*1 × *Col*1) + (*Row*2 × *Col*2) + (*Row*3 × *Col*3) + (Row4 × Col4) + (*Row*5 × *Col*5) + (*Row*6 x *Col*6)] / N^2

Before calculating κ, we must calculate p_o for the particular data table involved. For the data in Figure 5, the calculations would be as follows (Equation 9):

$$p_e = \frac{(Row1 \times Col1) + (Row2 \times Col2) + (Row3 \times Col3) + (Row4 \times Col4) + (Row5 \times Col5) + (Row6 \times Col6)}{N^2} \quad (9)$$

Notice that this time we are calculating p_e by multiplying the row and column totals for cells A, B, C, D, E, and F and adding them up before dividing by N^2. In doing so, we are calculating the proportion of the expected frequencies for cells A through F. Substituting in the values from Figure 5, we get (Equation 10):

$$\begin{aligned} p_e &= \frac{(74 \times 78) + (23 \times 29) + (8 \times 10) + (11 \times 16) + (22 \times 21) + (30 \times 25)}{170^2} \\ &= \frac{5772 + 667 + 80 + 176 + 462 + 750}{28900} = \frac{7907}{28900} = .2735986 \end{aligned} \quad (10)$$

Given that p_o for the data in Figure 5 is .88235294 and p_e is .2735986, Kappa turns out to be (Equation 11):

$$\kappa = \frac{(p_o - p_e)}{(1 - p_e)} = \frac{(.88235294 - .2735986)}{(1 - .2735986)} = \frac{(.6087543)}{(.7264014)} = .8380411 \approx .84 \quad (11)$$

This Kappa coefficient shows that the proportion of coding agreement that occurred beyond what we would expect by chance alone is .84, so the percentage of agreement is a respectable 84%, and we know that it could have fallen anywhere between .00 to 1.00.

Conclusion

This column described how to calculate rater/coder agreement and Cohen's Kappa. I have shown here how to lay out the data and calculate agreement and Kappa coefficients for a simple set of data based on binary decisions by two coders, as well as for the data generated in your study for six-category decisions by two coders. I hope you see how you need to arrange your data for Questions 2 to 6 in order to calculate p_o, and more importantly, how to calculate p_e from the row and column totals associated with each agreement cell (regardless of the number of decisions involved) in the process of then calculating agreement and Kappa coefficients for each question in your study. Please note that I would generally report both the agreement and Kappa coefficients because they provide different types of information, both of which may be interesting to some readers.

References

Brown, J. D. (1997). Statistics Corner: Questions and answers about language testing statistics: Reliability of surveys. *Shiken: JALT Testing & Evaluation SIG Newsletter, 1*(2), 17-19. Retrieved from http://www.jalt.org/test/bro_2.htm

Brown, J. D. (2000). Statistics Corner. Questions and answers about language testing statistics: What issues affect Likert-scale questionnaire formats? *Shiken: JALT Testing & Evaluation SIG Newsletter, 4*(1), 18-21. Retrieved from http://www.jalt.org/test/bro_7.htm

Brown, J. D. (2002). Statistics Corner. Questions and answers about language testing statistics: The Cronbach alpha reliability estimate. *Shiken: JALT Testing & Evaluation SIG Newsletter, 6*(1), 14-16. Retrieved from http://www.jalt.org/test/bro_13.htm

Brown, J. D. (2011). Statistics Corner. Questions and answers about language testing statistics: Likert items and scales of measurement? *Shiken: JALT Testing & Evaluation SIG Newsletter, 15*(1), 10-14. Retrieved from http://www.jalt.org/test/bro_34.htm

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*, 37-46.

Where to Submit Questions:

Please submit questions for this column to the following e-mail or snail-mail addresses:

brownj@hawaii.edu

JD Brown
Department of Second Language Studies
University of Hawai'i at Mānoa
1890 East-West Road
Honolulu, HI 96822
USA

Your question can remain anonymous if you so desire.

TEVAL member publications

Starting from this issue of *SRB,* we will be listing the recent publications of TEVAL SIG members. If you have a publication that you would like to promote here, please send an email to the submission email found at the end of the issue.

Batty, A. (2012). Identifying dimensions of vocabulary knowledge in the Word Associates Test. *Vocabulary Learning and Instruction*, *1*(1), 70–77. doi:10.7820/vli.v01.1.batty

Cook, M. (2012). Exploring the Influence of Shadow Education on Testing Practices. *Journal of International Studies and Regional Development 3.* 1–9.

Cook, M. (in press). The Multipurpose English Entrance Examination: Beliefs of Expatriate ELT Faculty. *The Language Teacher*, *37*(1).

Hirai, M. (2012). Correlations between BULATS Speaking/Writing and TOEIC® scores. In R. Chartrand (Ed.), Proceedings of 11th Annual JALT Pan-SIG Conference, JALT.

Hirai, M. (in press). Importance of Business English Skills Assessment. *JALT Business English Journal, 1*(1).

Hirai, M. (2012). 受信型スキルテストで仕事における発信型能力を測れるか [Can receptive-skill tests measure productive skills in workplace?]. *BULATS 通信 [BULATS Tsuushin], 22.* http://www.eiken.or.jp/bulats/index.html

Hirai, M. (2012). A Proposed Set of Can-Do Statements for Technical English. Annual Report of JACET SIG on ESP, Vol.14, 2012 (to be published later this year)

Stewart, J. (2012). A multiple-choice test of active vocabulary knowledge. *Vocabulary Learning and Instruction*, *1*(1), 53–59. doi: 10.7820/vli.v01.1.stewart

Stewart, J., Batty, A. O., & Bovee, N. (2012). Comparing multidimensional and continuum models of vocabulary acquisition: An empirical examination of the Vocabulary Knowledge Scale. *TESOL Quarterly*. doi:10.1002/tesq.35

Stubbe, R. (2012) Do pseudoword false alarm rates and overestimation rates in yes/no vocabulary tests change with Japanese university students' English ability levels? *Language Testing.* 471–488. doi: 10.1177/0265532211433033

Stubbe, R. & Yokomitsu, H. (2012) English Loanwords in Japanese and the JACET 8000. *Vocabulary Education and Research Bulletin, 1*(1), 10–11.

Stubbe, R. (2012). Searching for an acceptable false alarm maximum. *Vocabulary Education and Research Bulletin, 1*(2), 7–9.

Upcoming language testing events

The 35th Language Testing Research Colloquium (LTRC): July 3 – 5, 2013

Abstract submissions: July 15 – November 15

Venue: Seoul National University, Seoul, Korea

Conference homepage: http://www.ltrc2013.or.kr/

The 10th Annual European Association for Language Testing and Assessment: May 23 – 26, 2013

Abstract submissions: October 21 – December 10, 2012

Venue: Istanbul Military Museum, Istanbul, Turkey

Conference homepage: http://ealta2013.sabanciuniv.edu/

Association for Language Testing and Assessment of Australia and New Zealand (ALTAANZ) First Annual Conference: November 9 – 10, 2012

Abstract submissions: (closed)

Venue: University of Sydney, Sydney, Australia

Conference homepage: http://www.altaanz.org/altaanz-conferences.html

—ALTAANZ is now an official international affiliate of JALT TEVAL SIG!

The 32nd Annual Language Testing Forum: November 16 – 18, 2013

Abstract submissions: (closed)

Venue: Bristol University, Bristol, UK

Conference homepage: http://www.bristol.ac.uk/education/ltf2012/

Shiken Research Bulletin Editorial Board

Submissions

If you have a paper that you would like to publish in *Shiken Research Bulletin,* please email it in Microsoft Word format to the General Editor at:

jaltteval+srb@gmail.com