Can We Rely on IRR? Testing the Assumptions of Inter-Rater Reliability

Brendan R. Eagan, Bradley Rogers, Ronald Serlin, Andrew R. Ruis, Golnaz Arastoopour Irgens, and David Williamson Shaffer
beagan@wisc.edu, bjrogers2@wisc.edu, rcserlin@wisc.edu, arruis@wisc.edu, arastoopour@wisc.edu, dws@education.wisc.edu
University of Wisconsin–Madison

Abstract: Researchers use Inter-Rater Reliability (IRR) to measure whether two processes—people and/or machines—identify the same properties in data. There are many IRR measures, but regardless of the measure used, however, there is a common method for estimating IRR. To assess the validity of this common method, we conducted Monte Carlo simulation studies examining the most widely used measure of IRR: Cohen's kappa. Our results show that the method commonly used by researchers to assess IRR produces unacceptable Type I error rates.

Keywords: inter-rater reliability, coding, code validation, Cohen's kappa

Introduction

Inter-Rater Reliability (IRR) measures whether two processes identify the same properties in data. That is, it determines whether codes (or annotations or categorizations) are applied in the same way by two coders. In the context of Computer Supported Collaborative Learning (CSCL), it is often difficult, if not impossible, for a person to code an entire dataset. In these cases, researchers typically code a *test set*, or a subset of the data, and measure the IRR of the raters on the test set as a proxy for what their agreement would be if they were to code the entire dataset. But this raises a question: *Can we assume that the IRR measured for a test set generalizes to an entire dataset, or to a larger set of similar data?*

Prior work in CSCL on IRR is primarily concerned with the question of *which* IRR measure to use. Here we ask *how* IRR measures are used, and whether they are used appropriately. To investigate whether or not IRR measures are used appropriately, we conducted two Monte Carlo studies with the most popular IRR measure used in CSCL: Cohen's kappa.

Theory

In CSCL research, assessing the reliability of coding schemes using IRR is a *consensus estimate* (Stemler, 2004). There are many possible measures of IRR, for any IRR measure, the same basic method is used. For a given code: (1) A definition for the code is written. (2) A measure of IRR is chosen and a minimum threshold for acceptable agreement is set. (3) A test set of a specified length is randomly selected from the dataset. (4) Two independent raters code the test set based on the definition. (5) The agreement of their coding is calculated using the chosen IRR measure. (6a) If the IRR calculated is below the minimum threshold: the raters discuss their coding decisions; (I) they resolve their disagreements, often by changing the conceptual definition of the code; and (II) the raters repeat steps 3, 4, and 5. (6b) If the IRR calculated is above the minimum threshold, researchers conclude that the raters agree on the meaning of the concept, and the coding is considered to have construct validity. The two raters can then independently code the rest of the data.

We conducted a meta-analysis of four research journals in which CSCL research is commonly published: IJCSCL, JLS, JEDM, and JLA. We searched 225 IJCSCL articles from 2006 through 2016, and 491 JLS articles from 1997 through 2016 using the following search terms: inter rater, interrater, inter-rater, intra class, intraclass, intra-class, and reliability. We also read all 46 articles in JEDM from 2009 through 2015 and all 102 articles in JLA from 2014 through 2016. This meta-analysis found that more than 97% of CSCL research articles appear to follow this method. In what follows we refer to this progression as the *Common Method for IRR Measurement* (CIM).

When this method is described explicitly, it is clear that there is an implicit assumption when using the CIM: namely, that the IRR measured in the test set applies more broadly to data not contained in the test set.

We tested this assumption using a Monte Carlo method. *Monte Carlo* (MC) studies are one method commonly used to investigate the performance and reliability of statistical tests used in educational and psychological research (Harwell, 1992). In MC studies, researchers generate an *empirical sampling distribution*: a large number of simulated datasets and calculate a test statistic for each one. Type I and Type II error rates can

thus be computed empirically and used to evaluate the performance of statistical tests under different assumptions about the properties of the population from which samples are drawn.

MC studies thus require construction of simulated datasets that reflect the properties of the distribution being modeled. In the case of IRR, MC studies require a specific type of simulated dataset, a *simulated codeset* (SCS) that models data coded by two raters. Such sets consist of binary ordered pairs—(1,1); (1,0); (0,1); and (0,0)—where the first number represents whether the first rater applied the code and the second number represents whether the second rater applied the code.

Parameters need to be specified to produce simulated data that more closely reflect the data produced by trained raters. This simulated data can then be used to investigate the performance and reliability of various IRR measures, allowing researchers to test the extent to which the CIM produces generalizable results.

In what follows, we describe a series of MC studies that assess the performance of the CIM using the most commonly employed IRR measure in CSCL: Cohen's kappa (hereafter, kappa), which we chose based on our meta analysis (described above) that showed kappa was used in 40% of articles that computed IRR.

We consider two conditions. First, we examine the case in which there is a large dataset (on the order of 10,000 items) and two raters code a small sample of the data as a test set. Second we considered cases, where the initial dataset is smaller (on the order of 1,000 items), and thus two raters are able to code a very large portion of the data (up to 50%). In each case, we ask whether the CIM produces acceptable Type I error rates, which we take here as <0.05.

Methods

Generation of simulated codesets

We identified four parameters necessary for generating SCSs: base rate, SCS length, kappa, and precision. (1) **Base Rate:** The frequency with which a code is applied by a single rater. (2) **SCS Length**: The total number of items in the SCS. Measures of inter-rater reliability are almost always invariant to permutation of the excerpts being coded; therefore, these first two parameters allow us to simulate the codes of the first rater as a series of 1s of length *base rate* × *simulated codeset length* followed by a series of 0s of length *(1 – base rate)* × *simulated codeset length*. To compute the simulated codes for the second rater, we need two additional parameters. (3) **Kappa**: We used kappa (Cohen, 1960) to specify the overall level of agreement between the two raters. (4) **Precision**: The base rate and SCS length produce a unique set of codes for the first rater. However, one can produce multiple sets of codes for the second rater for any given kappa because kappa does not distinguish between positive and negative agreements. To address this, we used *precision*, which measures the likelihood the first rater thought the code was present if the second rater thought the code was present.

These four parameters identify a unique set (ignoring permutations) of ordered pairs $\{(f_i,s_i)\}$ that represent the codes for the first rater, f_i, and the codes for the second rater, s_i, for each item *i* in the SCS. Our meta-analysis of CSCL and related research provided limited guidance on appropriate ranges for these parameters for the purpose of modeling what two raters in the field would produce when coding qualitative data. Therefore, for our MC simulations, we empirically derived conservative estimates of what two trained human raters would reasonably produce for base rate, kappa and precision, based on the performance of raters observed in our own lab. For example, we typically find base rates for discourse codes in the range of 0.01 to 0.30. While base rates for codes are not typically reported in studies, we believe that these rates are not atypical in CSCL research. Simulated data generation parameter ranges were: base rate (0.01, 0.05, 0.10, 0.20, 0.30, 0.50; simulated codeset length (10,000 [MC Study 1] & 1,000 [MC Study 2]); kappa (0.30 – 1.00): precision (0.60 – 1.00). Simulated codeset length was held constant in both MC study 1 and MC Study 2.

To construct a SCS, we thus (a) chose a base rate and SCS length to calculate the number of 1s and 0s produced by the first rater, (b) randomly selected a value from our range of kappas, and (c) randomly selected a precision from the estimated range until it formed a valid (mathematically possible) combination with the kappa previously selected.

MC simulation construction

Using the SCS generation method described, we developed a *simulated IRR measurement* (SIM) method to model the CIM based on three additional parameters: (1) **Test Set Length:** We specified a *test set length* as in the CIM (CIM Step 3). A review of the literature indicated that researchers use a variety of test set lengths. For example, De Laat and Lally's (2004) used a sample of 10% of their dataset of 160 messages. In contrast, McKenzie and Murphy (2000) chose to sample one-third of the 151 messages containing 271 message units. None of the researchers justified the choice of a particular test set length. In MC study 1 (SCS length = 10,000), we used test set lengths of 20, 40, 80, 160, 200, 400, and 800. In MC study 2 (SCS length = 1,000), we used test sets lengths

of 2%, 4%, 8%, 16%, 20%, 40%, and 50% of the SCS length. (2) **Replicates**. We empirically derived the number of *replicates*, or the number of times we needed to simulate the CIM to be confident in our calculation of error rates. To do so, we incrementally increased the number of replicates until the standard deviation of the Type I error rates decreased to less than or equal to 0.01. This result was achieved for all of the simulation in our MC studies with 12,000 replicates. (3) **Thresholds**: We used a threshold of 0.65 for kappa, which is consistent with the most commonly used threshold (Cohen, 1960; Viera and Garrett, 2005).

To complete the MC studies, we applied the SIM method as follows: (1) We chose a base rate and test set length and created 12,000 sets using our SCS generation method—this simulates the coding of the data (CIM step 4). (2) We computed kappa for each SCS, which represents the true IRR rates for two coders. (3) We randomly selected a *test set* from each SCS at the given test set length, which represented the number of excerpts the raters actually coded—that is we took a sample of the dataset (CIM step 3). (4) We computed kappa on the test set (CIM Step 5). (5) We computed the Type I error rate (*false positives*, or all test sets with IRR above the corresponding threshold) for kappa (CIM Step 2 & 6b)—where a Type I error was defined as a case where the agreement measured by the IRR test statistic in the test set was above the threshold of 0.65 and the actual agreement in the SCS was below the threshold. We repeated the SIM process for all combinations of base rates and test set lengths.

Findings

RQ1: Does the CIM using kappa produce acceptable (< 0.05) Type I error rates when two raters code a small subset of the data? In MC Study 1, we conducted 42 simulations, each containing 12,000 SCS with lengths of 10,000, using base rates from 0.01 to 0.50 and test set lengths from 20 to 800 (see Table 1). Of these 42 simulations, only 4 had Type I error rates less than 0.05. These 4 had test set lengths of 400 or higher, and base rates of 0.20 or higher. The remaining 38 studies all had Type I error rates greater than 0.05. Of those 38 studies, 15 studies had Type I error rates greater than 0.20. This suggests that the CIM for kappa produces valid results only for large test sets with base rates that may be larger than are typically seen in CSCL research.

Table 1: SIM method using kappa Type I error rates - MC Study 1 (simulated codeset length = 10,000)

		Test Set Length						
		20	40	80	160	200	400	800
Base Rate	0.01	0.304	0.355	0.367	0.383	0.364	0.297	0.199
	0.05	0.255	0.347	0.280	0.210	0.182	0.123	0.073
	0.10	0.228	0.256	0.179	0.132	0.118	0.078	0.061
	0.20	0.216	0.196	0.132	0.097	0.083	0.053	***0.039**
	0.30	0.229	0.168	0.110	0.077	0.0728	0.050	***0.035**
	0.50	0.204	0.136	0.095	0.073	0.059	***0.044**	***0.034**

RQ2: Does the CIM using kappa produce acceptable (< 0.05) Type I error rates when two raters code a large subset of the data? In MC Study 2, we conducted 42 simulation studies, each containing 12,000 SCS with lengths of 1,000, using base rates from 0.01 to 0.50 and test set lengths from 2% (20) to 50% (500) of the SCS length. Of these 42 simulations, all but 6 had Type I error rates greater than 0.05. All of these 6 used test set lengths of 40% (400) or higher, and base rates of 0.20 or higher. Many of the remaining simulation studies had Type I error rates greater than 0.20. This suggests that the CIM using kappa produced valid results only for large test sets with base rates that may be larger than are typically seen in CSCL research.

Discussion

The results of our MC studies show that the CIM has high Type I error rates: greater than 0.05 except in the few cases where codes have very high base rates *and* test sets that are larger than those typically found in CSCL research. In many cases, Type I error rates are near or above 0.30, meaning a third of the test sets generated a kappa that exceeded the threshold, but the kappa of the entire dataset did not. In over one third of the cases we examined, Type I error rates were greater than 0.20.

Our results highlight a critical problem for CSCL researchers. Because the CIM does not control for Type I error rates, researchers must code a prohibitively large amount of data to obtain reliable IRR with the CIM. More generally, though, our results point to significant issues (significant in both the statistical and practical sense) with the reliability of the CIM. The problem, of course, is that the CIM assumes that a statistic (in this case, an IRR measure) computed on a sample (in this case, the test set) provides a good measure of the value of the statistic in some population (in this case, the rest of the data being coded).

A critical job of statistical methods is to establish whether such inferences are warranted given the properties of a sample. Thus, we believe the results here suggest that statistical methods need to be used to establish the reliability of coding regardless of the IRR measure used.

Although it is beyond the scope of this preliminary paper, we have developed such a method by treating code validation as a sampling problem and using a Monte Carlo hypothesis testing method to calculate a pseudo p-value, *Shaffer's rho*, that estimates the Type I error rate for an IRR measure given a test set coded by two raters. This method has been outlined in a working paper (Shaffer et. al., 2015) and is available as an R package. We will describe the method in detail in a subsequent publication, but briefly, *Shaffer's rho*: 1) Has acceptable type I error rates (< 0.05); 2) Can be used with any IRR measure; 3) Statistically tests whether an IRR measure generalizes to the entire dataset and population of interest; and 4) Allows for validation of low base rates codes, which has historically been difficult for researchers.

Whether researchers ultimately choose to adopt *rho* or another statistical test, the results here suggest that the current, widely-accepted approach to IRR should be used with caution in most circumstances that CSCL researchers are likely to encounter in their work. This issue will become only more critical as CSCL research continues to use datasets with tens or hundreds of thousands of items, making it impossible for human raters to code more than a tiny fraction of the data by hand.

References

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement 20*(1): 37–46.

De Laat, M., & Lally, V. (2004). It's not so easy: Researching the complexity of emergent participant roles and awareness in asynchronous networked learning discussions. *Journal of Computer Assisted Learning*, *20*(3), 165–171.

De Wever, B., Schellens, T., Valcke, M., & Van Keer, H. (2006). Content analysis schemes to analyze transcripts of online asynchronous discussion groups: A review. *Computers & Education*, *46*(1), 6–28.

Dunn, G. (1989). Design and analysis of reliability studies: The statistical evaluation of measurement errors. Oxford, UK: Oxford University Press.

Feinstein, A.R., & Cicchetti, D.V. (1990). High agreement but low kappa: I. The problems of two paradoxes. *Journal of Clinical Epidemiology, 43*(6), 543-549.

Harwell, M. R. (1992). Summarizing Monte Carlo Results in Methodological Research. *Journal of Educational Statistics*, *17*(4), 297–313.

Kolodner, J.L., & Gray, J. (2002). Understanding the affordances of ritualized activity structures for project-based classrooms. *Proceedings of the International Conference of the Learning Sciences,* Mahwah, NJ: Erlbaum, 221-228.

McKenzie, W., and Murphy, D. 2000. "I hope this goes somewhere": Evaluation of an online discussion group. Australian Journal of Educational Technology, 16(3), 239–257.

Shaffer, D.W., Borden, F., Srinivasan, A., Saucerman, J., Arastoopour, G., Collier, W., Ruis, A.R., & Frank, K.A. (2015). The nCoder: A Technique for Improving the Utility of Inter-Rater Reliability Statistics. Epistemic Games Group Working Paper 2015-01. University of Wisconsin–Madison.

Viera, A.J., & Garrett, J.M. (2005). Understanding interobserver agreement: The kappa statistic. *Family Medicine, 37*(5), 360-363.

Acknowledgments

We thank participating teachers and students. This work was funded in part by the National Science Foundation (DRL-0918409, DRL-0946372, DRL-1247262, DRL-1418288, DUE-0919347, DUE-1225885, EEC-1232656, EEC-1340402, REC-0347000), the MacArthur Foundation, the Spencer Foundation, the Wisconsin Alumni Research Foundation, and the Office of the Vice Chancellor for Research and Graduate Education at the University of Wisconsin–Madison. The opinions, findings, and conclusions do not reflect the views of the funding agencies, cooperating institutions, or other individuals.