

Response Similarity Analysis

Larry Nelson

Curtin University of Technology

Document date: 27 July 2005

website: www.larrynelsonstuff.com/lertap/

UPDATE 2019: this document is very outdated. The criticisms of Harpp-Hogan cheating-detection methods, as mentioned below, are incomplete, and are essentially now recognized as inaccurate, even by Professor Wesolowsky.

As of July 2019, recommended references are found at the bottom of this Lertap5 [webpage](#) on "RSA". Especially visit at least two of the references found in the "related tidbits" section at the bottom of the webpage: [this modification](#) of a journal article published in the year 2006, and two practical real-life examples [starting here](#).

Early in 2005 my good friends at [Assessment Systems Corporation](#) put up a link on their "New" page to a new (I thought it was new) software package called "*Integrity*".

It turns out that Integrity is a classical item analysis program which has, as one of its main objectives, an ability to estimate how much cheating there may appear to have been as students sat an exam.

At about the same time a colleague at a major Australian testing centre e-wrote to ask if perhaps Lertap could not benefit from some sort of cheat-checker, and, by the way, had I seen the work done by George Wesolowsky in this area?

So it was that on a recent fishing excursion to Australia's North West Cape, I sat inside the van as others risked their lives outside, perched on rocks, trying to reel in a meal (meals on reels).

I GPRS-ed to a [Telstra](#) server from my iBook, and started to access the Internet in order to launch a fresh career in what I decided to refer to as **RSA**, response similarity analysis. (With apologies to others who may have been using this term before I was born, just a few years ago on a sunny Thursday in Indiana.)

What did I learn? GPRS can be expensive if you use it without watching your packet count (smile).

SCheck.exe

This is the name of a program written by Professor Wesolowsky of McMaster University, Canada, based on his year 2000 article in the Journal of Applied Statistics (see [references](#), and [website](#)).

I liked the paper, finding myself drawn by the arguments and methods which Prof. W recommends. Accordingly, I decided to make Lertap SCheck-friendly as soon I had eaten my fill of fish, and had access to a computer faster than the 12" iBook I was carrying on that trip.

So it came to be that, in mid-July, 2005, Lertap obtained an SCheck interface, that is, an ability to create a file suitable for direct input to the SCheck.exe program.

This document describes how to get Lertap to create its file for SCheck.exe, tries to help explain how to interpret SCheck's output, and then gets into Lertap's own RSA routine, boldly suggesting how to employ an RSA index which Professor Wesolowsky hopes you will leave unemployed (personal e-correspondence, Wesolowsky to Nelson, July 2005).

The RSAdata worksheet, and the SCheckData.DAT file

[Lelp](#), the Lertap help file, provides a list of the steps required to set things up for SCheck, and also for "RSA", Lertap's initial version of a response similarity analysis.

Here's what Lelp said as of July, 2005:

Summary of RSA steps

To review, here are the steps required in order to have Lertap do its RSA magic:

1. You have to say "yes" to RSA in the right spot in Lertap's System worksheet. As this topic went to press, the right spot was row 25, column 2.
2. You must go to the Run menu, and click on "Output item scores matrix". This will produce the RSAdata worksheet, and also the SCheckData.DAT file. You'll be able to see the RSAdata worksheet right away as it will form part of your Excel workbook, but the SCheckData.DAT file becomes a separate entity, a file on its own, stored on your computer's hard disk. Where? Well, if you had saved your workbook prior to taking this step, it'll be saved in the same folder as your workbook (otherwise you may have to dig around to find it).
3. Next, back to the Run menu, and a click on "Response similarity analysis" if you want Lertap to make its RSAtable and RSAcases worksheets.
4. If you want to use Professor Wesolowsky's SCheck.exe program, start SCheck.exe, and get it to work with the SCheckData.DAT file created by Lertap.

Please to refer to Lelp for more information, and for sample screen shots. If you're connected to the Internet, a [click here](#) should transport you to the relevant topics within Lelp.

Inputting SCheckData.DAT to SCheck.exe

The SCheckData.DAT file created by Lertap is constructed so as to comply with the data formatting specifications of an SCheck.exe "DAT" file.

When I downloaded the SCheck.exe program package from McMaster University, the program's user guide was found in a file called `ReadMeSCheck.pdf`. The appendix to this guide provides information about the DAT file format:

Example of a .DAT file with responses in the 1-5 range:

```
9706600 , , , 3...1.1...2.....5..1..
9799221 , , , 3222....222.2222..224..33..
9735555 , AARDVARK , SD, 3.2.1.112.....4.2
9719999 , AHURA-MAZDA , S , 12....1....12..211.2.....5
9707777 , APOLONIUS , D , 3222..112.2.2.22...2..1.21.
9717777 , ASMODEUS , Z , 1.2.1.11..2..2.211.2...331.
```

The first field is the ID (up to seven characters), the second is the name, the third can be initials, and the fourth is the block of responses.

Missing names and initials are represented by spaces, field length does not matter. Correct answers are dots. The number is the position of the incorrect response. Duplicate answers is coded '*'. No answer is coded '-'.

Lertap does not strictly adhere to the format. It puts the Lertap ID in the first field, and this may be longer than seven characters. It turns out that this does not affect SCheck.exe's output adversely.

SCheck wants to see a "name" in the second field, and initials in the third. Lertap doesn't do this – it puts `DataRowNNN` in the second field, and leaves the third entirely empty.

The NNN in `DataRowNNN` is a number which indicates the row number in Lertap's Data worksheet where the corresponding record of item responses may be found.

When it comes to the last SCheck field, Lertap's exactly follows what SCheck.exe wants. However, it should be noted that No answer is coded '-' may mean more in Lertap – if a student's response to an item is not one of the item's options, Lertap says the student has an "other" response, where "other" may mean no answer, or, perhaps, a data processing error. For example, if item responses are to

come from the set {A,B,C,D,E}, and Lertap encounters a response of F, or e, or * (asterisk), then an "other" response has been found.

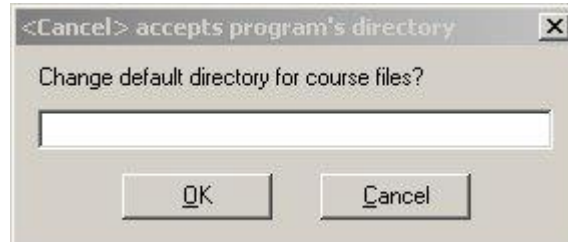
When it comes to setting things up for SCheck, Lertap translates "other" to what's expected by SCheck: a code of '-'. Note that SCheck treats a code of '-' as a wrong answer.

So, now that you're full bottle on the format of the DAT file, you'll no doubt be on the edge of your chair, wondering how you get the SCheck.exe program to use the SCheckData.DAT file produced by Lertap.

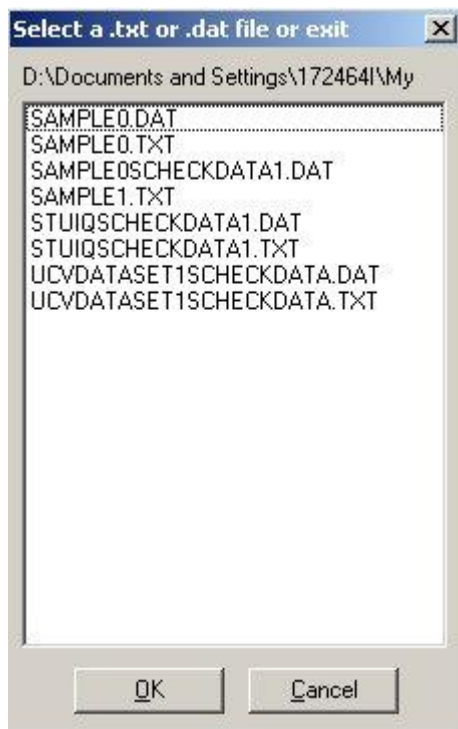
What I do is copy the file to the folder containing the SCheck.exe program. Before doing so, I rename the file. One of my favourite data sets, for example, is called StuIQ. What Lertap calls SCheckData.DAT gets renamed to StuIQSCheckData.DAT, and is then copied to the folder containing SCheck.exe.

Running SCheck.exe

Okay, here we are. Let's fire up SCheck.exe and get it to crunch some data. You sit there with your bottle of fresh country rain water, while I get set to capture a whole mess of screen shots.

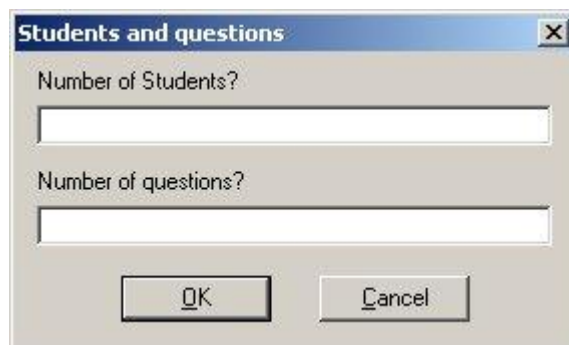


Since I have copied my DAT file to the directory (folder) where SCheck.exe resides, I can click the Cancel button.

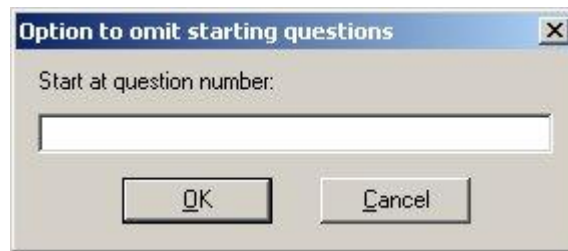


In the list of files above, those with the shortest names are one which come as samples with the SCheck.exe program. The others are ones I've made, using Lertap.

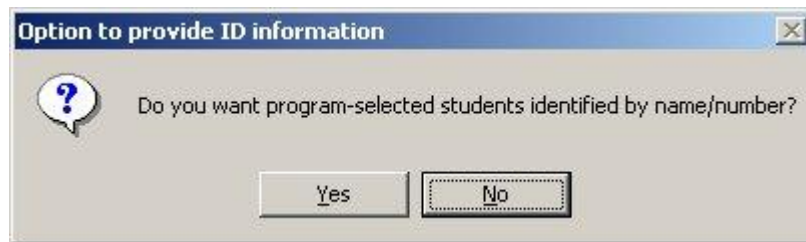
Let's say the file I want to work with is third in the list above, that is, Sample0SCheckData1.DAT (too bad SCheck converts all file names to uppercase). I mouse on it, and click OK.



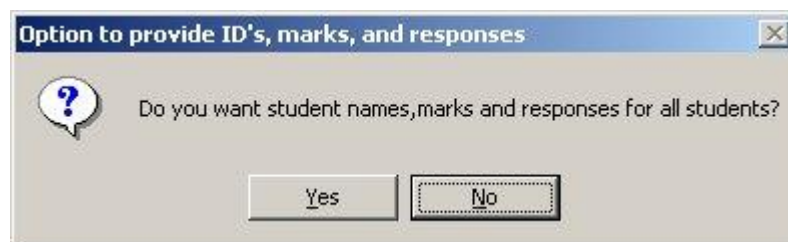
I click the Cancel button (above). (Note: the program points out that I can select the default response to its dialog boxes by simply pressing the <Enter> key. If the default answer is what I want, I can just press the <Enter> key instead of mousing to the corresponding response option, and clicking on it.)



I click the Cancel button (above).



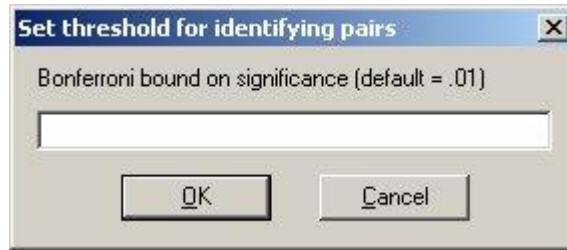
Here, above, I'm going for Yes. I want to see student name/number.



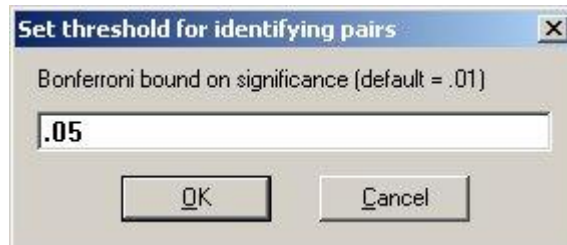
No, not required. My good buddy Lertap has given me this sort of information.



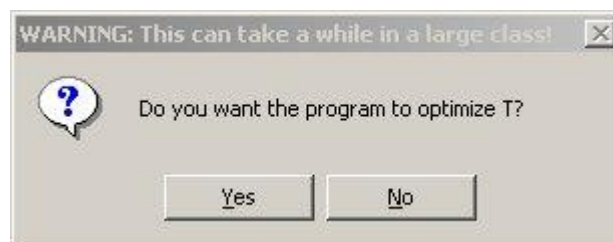
This option (above) is powerful. If your interest is really in detecting which students may have colluded in their test responses, you'd want to look into this option. For me, right now, my response is going to be No – my interest is in gauging the extent to which students in a given exam room may have somehow shared answers; I'm not particularly interested in special pairs of students.



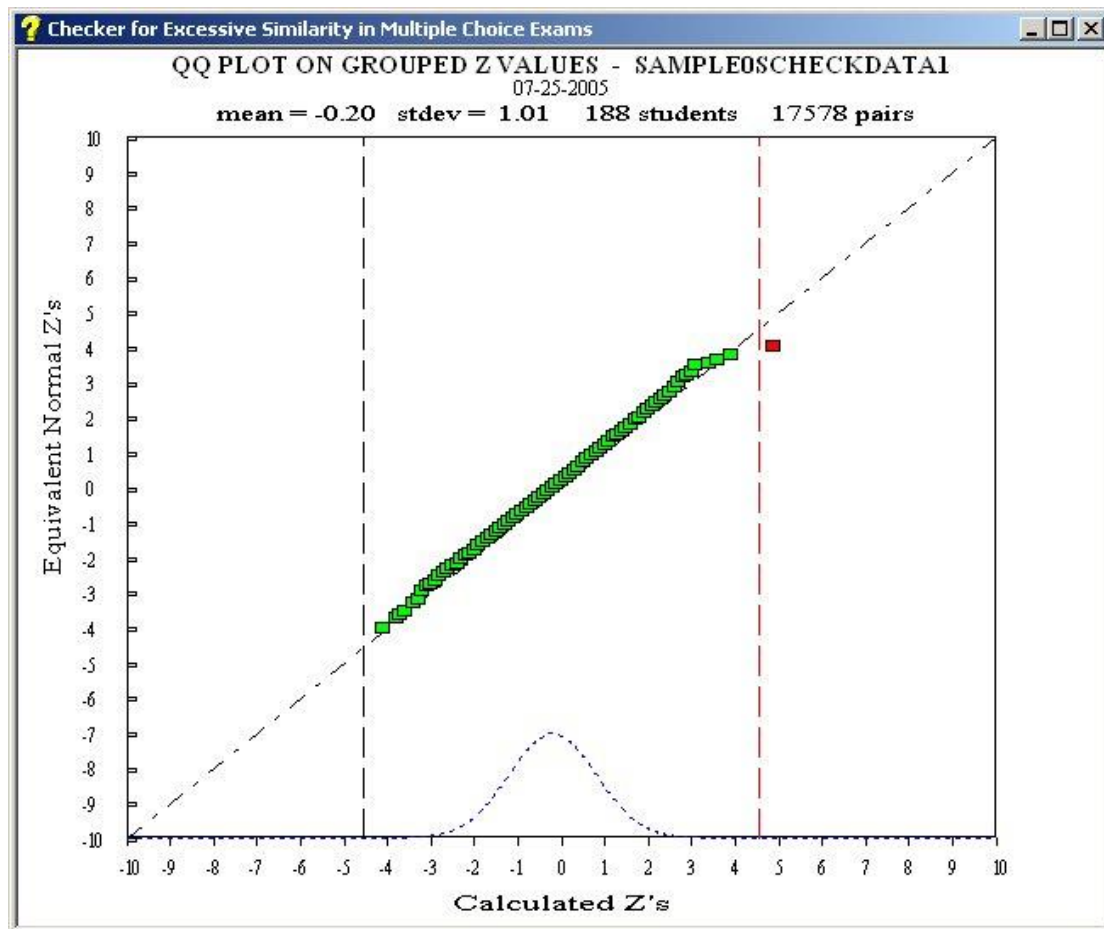
Ah, yes: at this point I'm *not* going for the default. Let me say again that my interest is in getting an estimate of the cheating which may have gone on in a given exam room. I know that Prof W's SCheck program does its level best to avoid false positives – it's biased in favour of the students; if anything it's more likely to give me an underestimate of the prevalence of cheating. So I'm going to relax the Type I error rate by typing .05 into the box:



Having put my .05-cents worth into the box above, I then click on Ok. (*Note: in a private e-communicé, Professor W has suggested that we might run with something greater than .05 – he suggests an appropriate value would be anything from .10 to 1.00.*)



It's not yet T-time for me; I'm going to click No above, and ride with SCheck's in-built default action.



A Q-Q plot, a quantile-quantile plot, may be used as a check on the goodness of a model. It's standard SCheck.exe output, and is another strength of the program.

If the model used by SCheck to represent student response patterns is accurate, the "blips" in the Q-Q plot, the calculated SCheck Zb values and their corresponding normal equivalents, may be expected to fall on a straight line. Departures from the line signal poor fit, and may lead us to suspect the presence of collusion. But: if the number of departures from the line is great, we might well suspect the model more than the students. (Personal correspondence from Professor Wesolowsky, 27 July 2005: Actually, aside from things like people trying to run an exam with ten students, I have only seen apparent model violations in your speeded exam example. This definitely does not mean that there are no model violations, only that the model seems to be very robust to abuse except when this abuse goes over the top. I reached this conclusion by checking seating adjacency to identify false positives. The rate of false positives has been remarkably consistent with that predicted by the model.)

In the graph above, the model seems to be becoming unstuck towards its extremes, towards its end, particularly, perhaps, at the

upper end. (Personal correspondence from Professor Wesolowsky, 27 July 2005: I must protest. Actually, the line is behaving admirably. The squares represent grouped data, and at the ends there are very few points in the group. Therefore, chance will give the ends a little jiggle or spread sometimes. Except for the red square, the line is very well behaved with even gaps from the verticals at both points. At the right end it dips a little but that is the fault of the red square. Remove the red pair from the data set and the green line will straighten out. The gap size is right for your choice of cutoff.)

One Zb value, the red one, is rather removed from the line, and also above the cutoff Zb value determined by SCheck.exe, represented by the dashed vertical coming up from the horizontal axis at about Zb = 4.5. (The cutoff value, listed before the Q-Q plot appears as the "Z treshold" (sic), is 4.54 in this example.)

After dismissing the graph, I note that SCheck.exe has produced an output file called Sample0SCheckData1.out, and has automatically opened the file for me, using the Windows Notepad program.

The out file provides detailed information on each "hit" found by the program – these are the student pairs whose responses are judged to be "odd", or "excessively similar".

```

=====
** pair = 105  149 ** Harpp-Hogan stat = #wr.mat/#diff =    3.667
Zb =  4.841  'equivalent' z from the BVP model
  Significance of Zb on a pre-selected pair = 6.4E-7
  Significance bound (Bonferroni)
                    on program selected pairs = 1.1E-2
#matches = 46 | 49  (mu,s)=( 32.634,  2.910)
prop. right for 105 = 0.735          prop. right for 149 = 0.735
Quest. range = [ 1  49 ]          #students = 188
-----
  STUDENT  105   9049306  DataRow107
..b..a.a..  ....a.....  .e...bd...  .....c.  a..ee.d.b
-----
  STUDENT  149   9057663  DataRow151
..b..a.a..  b...a.....  .e....d...  .....c.  a..ee.c.b
-----

```

In this example, SCheck.exe has found only one student pair whose Zb statistic, SCheck's degree of similarity measure, falls beyond the cutoff value.

For the pair of students found in Lertap's Data worksheet, at rows 107 and 151, Zb = 4.841. The numbers 105 and 149 correspond to the IDs of these two students; had Lertap used names as IDs, then the names would appear instead of 105 and 149.

The Harpp-Hogan similarity statistic, the number of exact errors in common divided by the total number of differences in the two students' responses, is 3.667, as seen in the first line of results above. You can confirm this by looking at the rows of student responses found in the last lines. If you do this, squinting your eyes up and counting along the strings, you'll see that the students differed in only three (3) of their 49 responses, while having eleven (11) exact errors in common; Harpp-Hogan is thus 11/3, or 3.667. (Your much-loved Lertap help file, Lelp, goes through the calculation of the Harpp-Hogan index more patiently: [click here](#) for a good read if you're on line.)

The $(\mu, s) = (32.634, 2.910)$ part of the output above represents the expected chance mean and standard deviation of the number of observed response matches for the two students, given a set of 49 test items. (These values will vary from student pair to student pair as they depend on the estimated abilities of the students to answer the items correctly.)

SCheck.exe to get number of hits

I have mentioned that my main interest is not really in determining which pairs of students may have colluded, but how many such pairs there may have been in a given test situation.

If I gave my 49-item multiple-choice test on classical test analysis methods to 300 students on the 4th of July, 2005, and if students sat the test in three different exam rooms, is there any evidence to suggest that the student responses in each of the three venues were suspiciously similar? And, were there differences in the number of possibly-suspect pairs among the exam rooms – Room A is much smaller than the other two, with desks closer together – might this have been a problem?

I'd like to get a simple number of hits from SCheck.exe. Can I?

Yes. All I have to do is scroll down to the bottom of the out file. For example:

SUMMARY

=====

mean of Z's = -0.1069 stdev= 1.5587

The number of pairs checked is 41328

The Bonferroni cut-off Z is 4.710

The entered Bonferroni cut-off significance bound is 0.0500

The estimated actual scanning significance cut-off is 1.0E+0

The execution time was 6.78 seconds

Number of observations below -4.71 or above 4.71 is 45

The program is running on an NT computer

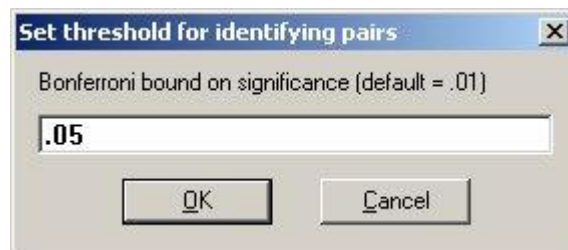
This software is not to be used, copied, or distributed without the direct permission of G.O. Wesolowsky (wesolows@mcmaster.ca)

In this example, the number of what I've called "hits" is 45.

SCheck.exe usage suggestions

SCheck.exe is clearly a powerful program. If you agree with me, but say that you find it difficult to use and interpret, I'd have little trouble understanding that. (Personal correspondence from Professor Wesolowsky, 27 July 2005: I agree that correct interpretation is tricky, even for people with statistical backgrounds. It is the nature of the beast. I think correct interpretation is even more difficult with other methodologies, although they often compensate with simplistic outputs from which incorrect and simplistic conclusions can be drawn. I try to provide all the information that would be needed if one wanted to check every calculation and conclusion. I would disagree a bit about ease of use. If one has a .dat file, one can just keep hitting the enter key until a list of suspects pops up. For those without understanding, this is a very conservative and safe option. Also, one can customize one's output with the batchs.ini file and then there is only the work of choosing a file. Unfortunately, my dog no longer appears.)

What I'd suggest is that you use SCheck as I have above, putting a value of .05 in the threshold box if your interest is similar to mine: estimating how much colluding may have gone on in a test room. (Note that in personal e-correspondence, Wesolowsky to Nelson, it has been suggested that the threshold could be a set anywhere between .10 and 1.00 if our interest is of this nature. I have experimented with this, and agree: go for .10, then try 1.00 – see how this affects the number of suspect pairs.)



On the other hand, if your interest is in finding out who done it, let the program run with its default value of .01 in the threshold box.

Be sure to look at the nifty Q-Q plot. You want the bulk of the cases to fall on or real close to the straight line. If there are many cases falling away from the line, not just a smattering at the ends, then suspect the model used by SCheck, not the students – I know professor Wesolowsky would like to hear from you if you have this sort of outcome, and, in fact, he's provided the following comment (personal correspondence, 27 July 2005):

I haven't ever seen this, except at the right end, and then this is due to cheating. Even your speeded tests have a connected line, except that it has an "impossible" slope (much greater than 1) and a very unusual (impossible) mean. However, I think this is an abomination, and of no concern in ordinary tests. The only other possibility is that someone is using a very small class (say, less than 15). Even here, the scatter of points is likely not due to the model but because the averages of small samples have more variation than averages of large samples, and the green squares represent averages.

Using Lertap's in-built RSA

Let's close the door now, and have a little hush-hush just among ourselves, without George Wesolowsky listening in. I am going to suggest you might make use of something which Prof W would put in with "... other methodologies ... with simplistic outputs from which incorrect and simplistic conclusions can be drawn...."

In the process of reading George's journal article, and following up on some of the references, I was attracted to work done by Harpp and Hogan (see [references](#)).

H & H have derived an empirical estimate of collusion, and I have used it as the basis for the initial version of an in-built Lertap response similarity analysis routine, RSA.

Your ever-present, always-faithful, esteemed and trusted friend, Lelp, has persact details on how to use Lertap's RSA. A little [click here](#) will let you Lelp, providing, of course, that you're on line to the Internet.

Lertap's RSA produces two worksheets, RSAtable and RSAcases.

Here's a snippet from an RSAtable:

Microsoft Excel - GeorgeSample0.xls

File Edit View Insert Format Tools Data Window Documents To Go Help

Lertap5 RSA table, created on: 25/07/2005.

H-H	f	%	cf	c%	Each symbol represents
0.1	107	1.0%	107	1.0%	
0.2	2,831	27.3%	2,938	28.3%	
0.3	3,659	35.2%	6,597	63.5%	
0.4	2,040	19.6%	8,637	83.1%	
0.5	1,082	10.4%	9,719	93.6%	
0.6	379	3.6%	10,098	97.2%	
0.7	158	1.5%	10,256	98.7%	
0.8	73	0.7%	10,329	99.4%	
0.9	20	0.2%	10,349	99.6%	
1.0	19	0.2%	10,368	99.8%	
1.1	13	0.1%	10,381	99.9%	
1.2	1	0.0%	10,382	99.9%	
1.3	2	0.0%	10,384	99.9%	
1.4	1	0.0%	10,385	99.9%	
1.5	2	0.0%	10,387	99.9%	
1.6		0.0%	10,387	99.9%	
1.7		0.0%	10,387	99.9%	
1.8		0.0%	10,387	99.9%	
1.9		0.0%	10,387	99.9%	
2.0		0.0%	10,387	99.9%	
2.1		0.0%	10,387	99.9%	
2.2		0.0%	10,387	99.9%	
2.3		0.0%	10,387	99.9%	
2.4		0.0%	10,387	99.9%	
2.5		0.0%	10,387	99.9%	

Ready

The table above is based on the Sample0SCheckData1 data set used with SCheck.exe way back at the start of this epistle. This data set is one which comes as a sample when you download and unpack the SCheck system from George Wesolowsky's [website](#). It's based on the responses of 188 students to 49 multiple-choice items.

Let's see ... for 188 students there will be $(188)(187)/2 = 17,578$ student pairs to look at. When you crank up Lertap's RSA, it starts by comparing the item responses of the first student with those of the second student. If the two students have a number of exact errors in common, EEIC, which is at or above the minimum value set in Lertap's system worksheet, Lertap forms their H-H index, EEIC over D, the total number of differences in their responses.

This value then gets tallied in the appropriate band in the RSatable worksheet.

Next, if the H-H value is at or above the cutoff value set in Lertap's system worksheet, the pair of students will have a record of their responses written to the RSAcases worksheet.

Lertap then goes on to compare the first student with the third, then with the fourth, and so on. Once finished with the first student, it then goes back to the second student, comparing his or her responses to those of the third student, then to the fourth student, and, well, you get the picture, eh?

The RSAcases worksheet looks like this:

Responses	Correct	EEIC	D	H-H index
..2..1.1.....1.....5...24.....3.1..55.4.2	36	11	3	3.67
..2..1.1..2...1.....5....4.....3.1..55.3.2	36			
..2..1.1..1..11...5.....4.5.1...2..3.....2.5	35	11	7	1.57
..2..1.1.1...11.....1.4.5.1...4..3.....1.215	33			
..22.1.1.....1.....2.31.4.5.2.....3.4..23.2.2	32	11	7	1.57
..2.21.1.....35.4.5.2.....3....43.2.2	35			
..2..121.....5....4...5.....3.....3...2	39	7	5	1.40
..2..12...1.....4.5.2.....3.....3...2	39			
..2.2121.....1.....4...4.....2....3.2.2	37	8	6	1.33
..2.2121...1.....4...3.....4....3...2	38			
..2..12...1.....4.5.2.....3.....3...2	39	8	6	1.33
..2.2121.1.....5.2.....3.1...3...2	37			
..2221.1.....2.....1.5..4.1.23.....3...41.2.2	32	10	8	1.25
..2.21.1.....35.4.5.2.....3....43.2.2	35			
..2..121.....5...1.4.5.4.....2.2	38	7	6	1.17
..2..121.....5....4...5.....3.....3...2	39			
..2.2.....1.....1.....4.5.43.....3.....3.2	38	7	6	1.17
..2...1...1...1.....4...43.....3.....2.2	39			
..2.2121.....1.....4...4.....2....3.2.2	37	7	6	1.17
..2..121.....5....4...5.....3.....3...2	39			
..2..121.....5....4...5.....3.....3...2	39	7	6	1.17
..2.2121...1.....4...3.....4....3...2	38			
..2..12...1.....4.5.2.....3.....3...2	39	7	6	1.17
..2.2121...1.....4...3.....4....3...2	38			
..2..121.....5...1.4.5.4.....2.2	38	8	7	1.14

I've had to scroll this worksheet so that you can see the crucial H-H index column, and, in the process, the first three columns have disappeared to the left.

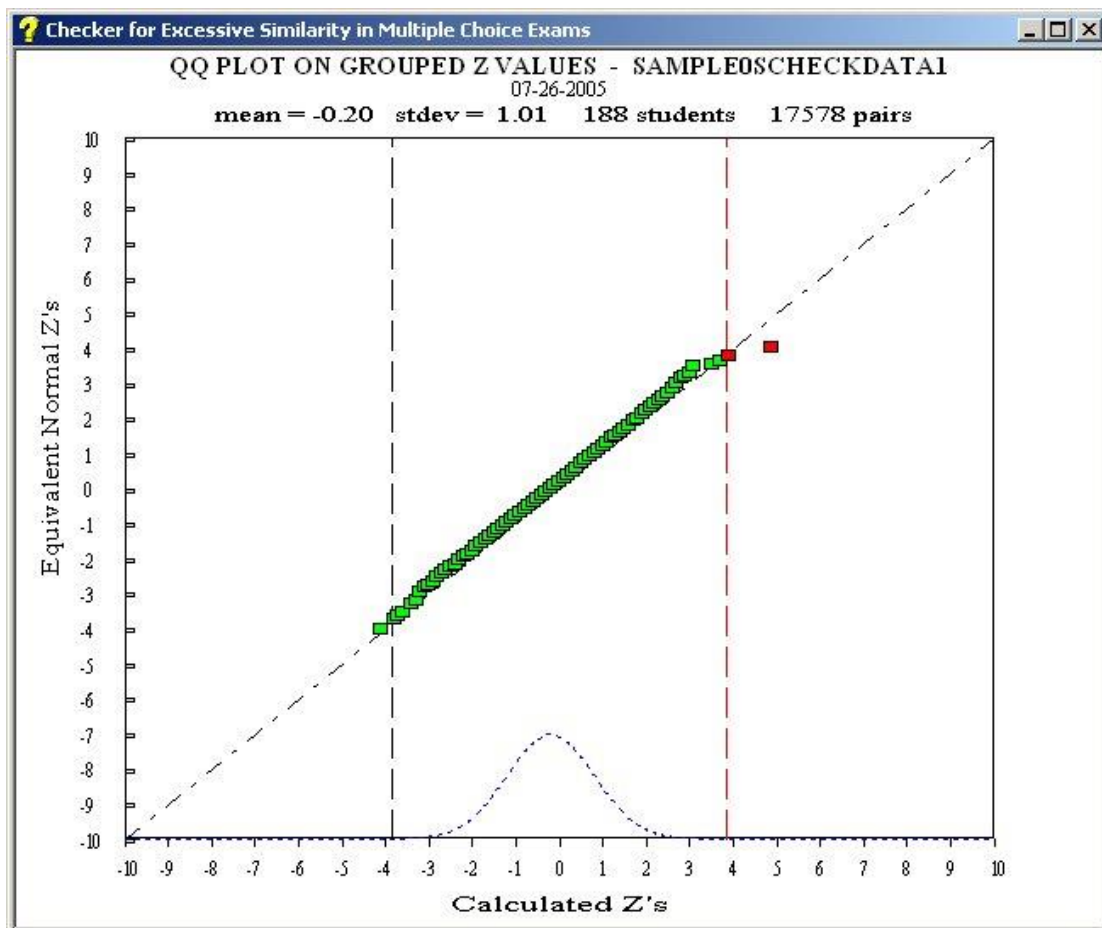
I have mentioned that Lertap's RSA is based on the work of Harpp and Hogan (and Jennings – see [references](#)). They have reported on their extensive experiments with similarity measures, and write "In

virtually all cases to date where the exam has ~30 or more questions, has a class average of <80% and where the minimum number of EEIC is 6, this parameter has been nearly 100% accurate in finding suspicious pairs” (Harpp, Hogan & Jennings, 1996, pp.349-350).

You ought to read the Harpp Hogan Jennings papers. I think they did a thorough job.

Now, before proceeding to the next topic, let me point something out: running SCheck.exe and Lertap RSA on the same data set, Sample0SCheckData1, has produced markedly different results.

SCheck.exe will find only two hits in this data set. Here’s its Q-Q, working from a threshold of 1.00:



How many hits did Lertap RSA find? Thirty-nine (39)! To confirm this, look at the RSatable output above. Count the number of H-H values equal to or greater than one (1.00). You’ll see 38. Add to this the single case found in the H-H 3.6 band, not visible above.

This is quite a discrepancy, isn't it? Who's right? Well, I myself would be quite tempted to go with SCheck until I have calibrated Lertap's RSA. One factor, not mentioned to this point, is that the 49 items in the data set include 19 true-false items. When Harpp and Hogan did their research, they seem to have consistently used multiple choice items with four or five options. I suspect that the true-false items may be affecting the results reported here – SCheck will accommodate them, for sure, but I suspect we'd want to raise the EEIC minimum setting when true-false items are involved, something which might be expected to decrease the number of hits in Lertap RSA.

Insert here another note from SCheck HQ (personal correspondence, Wesolowsky to Nelson, 27 July 2005):

You are right. The HH index depends on the number of choices and becomes quite unreliable on true-false. It is also, in my experience, unreliable for large classes (200 or more). However, even though they don't make it totally clear in the second article, they **must** have sigma over 5. Their software clearly includes $\sigma > 5$ as a necessary condition. Their results are then reasonably consistent with Scheck on strong similarities but differ on marginal cases. However, one needs to borrow David Harpp to explain all the restrictions. Sigma is, very imprecisely stated, the z value of $\text{SUM}(2(\log(\text{Proportion of matching responses})))$.

While you have been reading this, no doubt having a nice cup of green tea and a biscuit, I have been running SCheck and Lertap RSA on quite a number of data sets.

For classroom tests from some university ed psych classes, with 50 to 60 items, all with four choices, I have not found any hits with either method.

In some other cases, ones with a similar number and type of items, I have found Lertap RSA to produce only a few hits, even when the number of students goes beyond 100. I have found a couple of data sets where SCheck and Lertap RSA are in good accord.

Unfortunately quite a number of my data sets are from speeded tests, that is, tests with a tight time limit, with more than half the class leaving the last several items unanswered. SCheck and Lertap RSA both count matches on unanswered items as error matches, and this might be expected to adversely skew the results, leading to more hits. (Note: I know that George Wesolowsky is already working on a modification to SCheck which will control for this situation.)

Why Lertap's RSA should not be used

Lertap's RSA is easy to use, and, according to wife and kids, creates some pretty output. But it doesn't have anywhere near the statistical rigor found in SCheck. I would use Lertap RSA cautiously until having more experience with it – if I were in an active testing environment, I'd attempt to calibrate the H-H index used in Lertap RSA on my data sets. I wouldn't be surprised to find it holding up as well as Harpp and Hogan have reported, but, again, I'd try to get some baseline data on Lertap RSA before coming to rely on it.

What is handy, I suggest, is Lertap's RSatable – not by coincidence, it resembles the figures seen in Harpp, Hogan, & Jennings (1996). Look for outliers in RSatable, for gaps; my experience to date is that the H-H values will dribble away, sort of like the eigenvalues in a scree plot – if something suspect has gone on in the exam venue, you'll likely see H-H cases which stand out, which are, for example, beyond the 2.00 band in RSatable. I'd bet you a six pack of one of my favourite beverages, that Lertap RSA and SCheck will almost always agree on the extremes. Where they'll disagree, *I suspect*, is on those cases corresponding to H-H values in the range 1.00 to 2.00.

By no means would I use Lertap RSA as a basis for accusing a pair of students of cheating. My hope is that it may come to be useful as an indicator of the possible presence of cheating in a given test venue, a tool which might lead you to say *"Hey, our invigilators need to be more invigilative"*, or *"Obviously having 50 students sit an exam in the Faulty waiting room, with some students sitting on the laps of others, is not on!"*

Now, I'll have that cup of tea while you go off and read some of the [references](#). And you should. (Check in at www.assess.com for additional references, including texts and software.)

Credits

Many thanks to George Wesolowsky, McMaster University, for providing comments on the first draft of this paper.

Future developments

Watch this spot.

If you have questions, or comments, and they're not nasty in nature, email them to: larry@lertap5.com. (Nasty comments should be posted by surface mail to Santa Claus at the South Pole.)

Larry Nelson
Curtin University of Technology
Perth, Western Australia