How Lertap and Iteman Flag Items

Last updated on 24 June 2012

Larry Nelson, Curtin University

This little paper has to do with two widely-used programs for classical item analysis, Iteman and Lertap. I am the author of the latter.

Both of these programs produce *very* detailed summaries of how test items have performed. Because these summaries often have so much minute detail, Iteman and Lertap also have brief reports for test items, small tables meant to make it possible to quickly grasp how well items have functioned. The idea here is that users can, if they wish, spare themselves from having to read and decipher table after table of detailed items statistics – they can instead have a look at the brief summary and get a quick idea of how things went.

The way these two programs present their brief summaries differs quite markedly. Iteman's summary does not cover all items, only those that may have problems. If a test's items are problem-free, Iteman says nothing at all. Lertap, on the other hand, always creates a brief summary, called a "Stats_b" report, and it always makes mention of each and every test item.

In both programs "flags" are used to denote problems. The flags in the two tables below are referenced throughout this document.

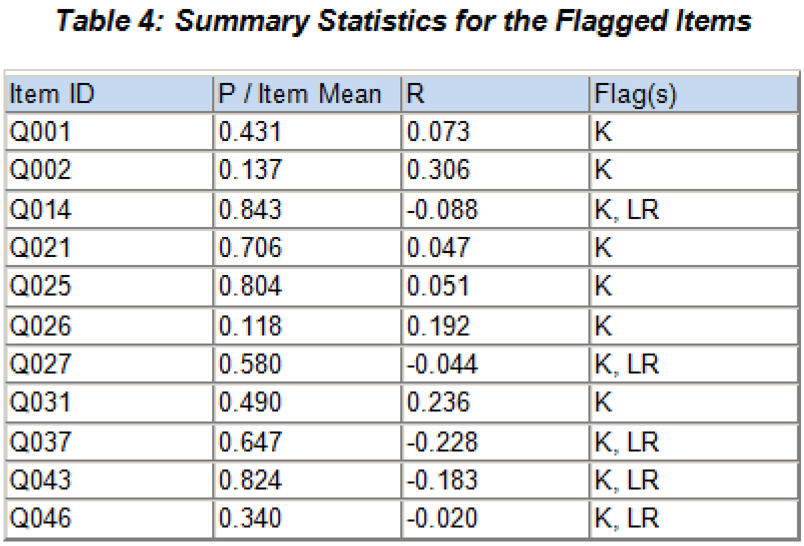
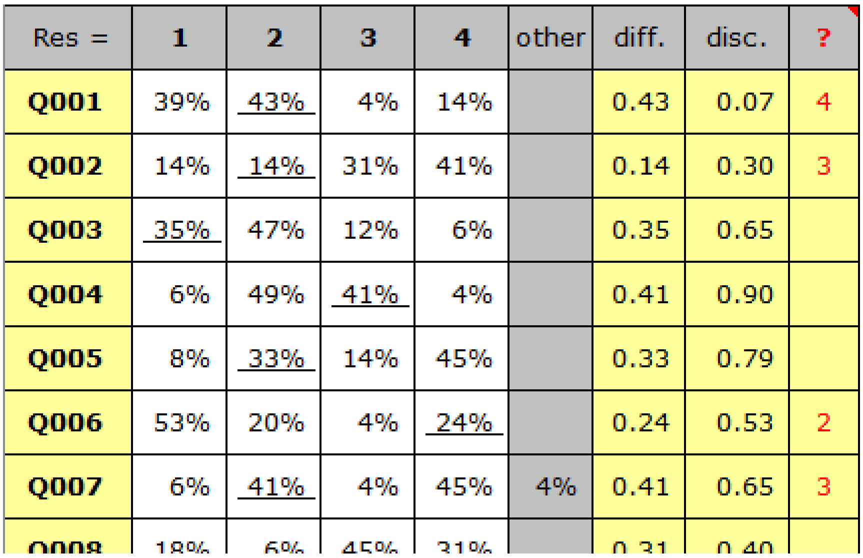


Table 4: Summary Statistics for the Flagged Items

Item ID	P / Item Mean	R	Flag(s)
Q001	0.431	0.073	K
Q002	0.137	0.306	K
Q014	0.843	-0.088	K, LR
Q021	0.706	0.047	K
Q025	0.804	0.051	K
Q026	0.118	0.192	K
Q027	0.580	-0.044	K, LR
Q031	0.490	0.236	K
Q037	0.647	-0.228	K, LR
Q043	0.824	-0.183	K, LR
Q046	0.340	-0.020	K, LR



Res =	1	2	3	4	other	diff.	disc.	?
Q001	39%	43%	4%	14%		0.43	0.07	4
Q002	14%	14%	31%	41%		0.14	0.30	3
Q003	35%	47%	12%	6%		0.35	0.65	
Q004	6%	49%	41%	4%		0.41	0.90	
Q005	8%	33%	14%	45%		0.33	0.79	
Q006	53%	20%	4%	24%		0.24	0.53	2
Q007	6%	41%	4%	45%	4%	0.41	0.65	3
Q008	18%	6%	45%	31%		0.31	0.40	

The top box above, “Table 4”, was made by Iteman 4.2. The lower box was created by Lertap 5.9.2.1.

Lertap’s ? column is similar to Iteman’s Flag(s) column. Both columns are drawing attention to items which may have a problem.

In the case of Q001, Iteman’s Flag is a K. The Iteman manual has this to say about K (and LR, another flag code seen in the Iteman output):

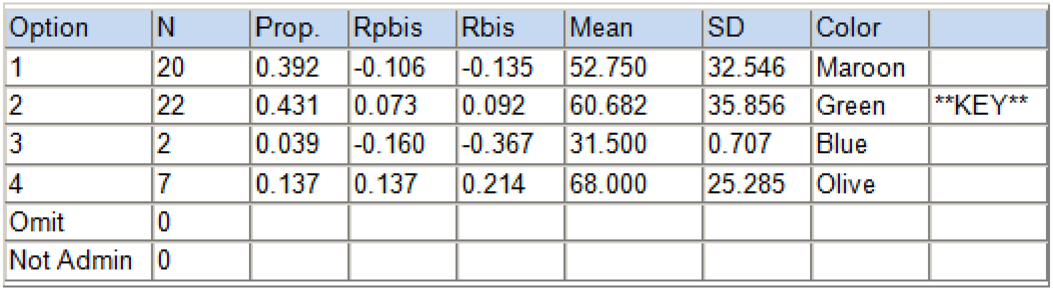
> K = Key error (r_{pbis} for a distractor is higher than r_{pbis} for key)
> LR = Low r_{pbis}

Lertap’s flag for Q001 is a 4, meaning that something may be wrong with this option. Here’s what Lertap documentation says about this column:

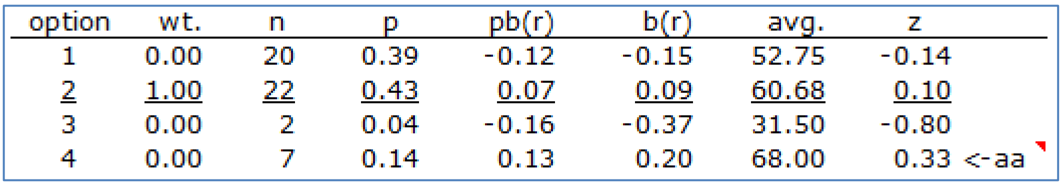
> One or more of an item's options will appear in the ? column whenever one of these conditions is met: no-one selects the option, the option is the correct answer but was selected by students with below-average criterion scores, or the option corresponds to a distractor (an incorrect option) selected by students with average or above-average criterion scores. The idea here is (basically) that we want an item's correct option to be selected only by the strongest students, while each of the distractors is selected by less than average students. Note that it is often the case that there will be many items with entries in the ? column when the number of students who take the test is small, say less than 50 or so. When there aren't many students, an item with four or more options will very frequently have "dead" distractors, wrong answers which were not selected by anyone; these will show up in the ? column.

Okay? I’ll now present results for a few items to show how Iteman and Lertap differ in their use of flags. Lertap, as I think you’ll see, has a flagman who’s a bit more active than Iteman’s.

The two boxes below show Iteman (top box) and Lertap (bottom box) summaries for Q001. These are examples of those "very detailed summaries" of item statistics mentioned at the start of the paper. The idea is that many users will look at such statistics for an item only when the item has been flagged. As you look at these tables, remember that Iteman has raised its K flag for this item, while Lertap raised a flag of 4.



Option	N	Prop.	Rpbis	Rbis	Mean	SD	Color	
1	20	0.392	-0.106	-0.135	52.750	32.546	Maroon	
2	22	0.431	0.073	0.092	60.682	35.856	Green	**KEY**
3	2	0.039	-0.160	-0.367	31.500	0.707	Blue	
4	7	0.137	0.137	0.214	68.000	25.285	Olive	
Omit	0							
Not Admin	0							



option	wt.	n	p	pb(r)	b(r)	avg.	z	
1	0.00	20	0.39	-0.12	-0.15	52.75	-0.14	
2	1.00	22	0.43	0.07	0.09	60.68	0.10	
3	0.00	2	0.04	-0.16	-0.37	31.50	-0.80	
4	0.00	7	0.14	0.13	0.20	68.00	0.33	<-aa

Iteman uses **KEY** to delimit the keyed-correct response to an item. Lertap indicates the same thing by the use of underlining.

"Rpbis" in Iteman is the same statistic as "pb(r)" in Lertap: the point-biserial correlation between the item and the criterion score. "Rbis" and "b(r)" stand for the biserial correlation.

"Mean" and "avg." indicate the average criterion score earned by the N (Iteman) or n (Lertap) students who selected the corresponding option. What's the "criterion score"? It's usually the total test score, but both programs make it possible to use an "external criterion", which could be, for example, GPA (grade-point average).

Lertap has two columns without an Iteman equivalent. "wt." indicates the number of points a student will get for selecting one of the item's options. "z" is a "standard score" found by converting the criterion scores to "z scores" with a mean of zero and standard deviation of one. When "avg." equals the average of the criterion scores, z will be zero. When "avg." is one standard deviation below the criterion average score, z will be -1.00. One standard deviation above the criterion average equates to a z score of +1.00.

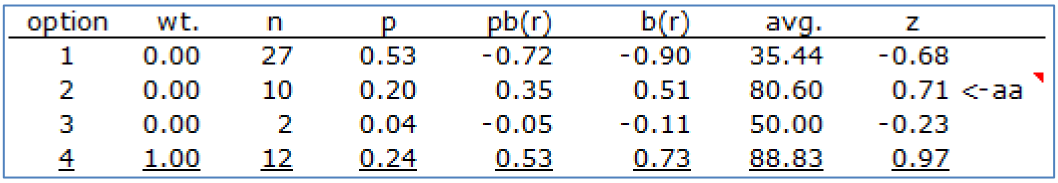
For Q001, those 7 students who selected option 4 had the highest "avg.", even higher than the 22 students who selected the keyed-correct option (2). Note that this pattern is also reflected in the pb(r) values, and it's why Iteman has applied the K flag to this item, suggesting that the item has been incorrectly keyed.

Lertap's flag of 4 for this item is more subtle. Lertap is not suggesting that the item has been mis-keyed; it's only drawing attention to the fact that, in this case, the students who selected option 4, a distractor, had an above-average "avg." score (this is indicated by **<-aa** for "above average"). They were strong students, stronger, in fact, than those who selected option 2 (based on a comparison of z values).

Both programs are pointing to a potential problem with Q001. I feel that Lertap's approach is perhaps technically preferable; this might well turn out to be an item with two correct answers, an item with two "keys", as it were. How will we know if option 4 can also be seen as a correct answer? By talking with the item writer, who, in turn, may ask the 7 students why they considered this option to be correct. Something like this is often done; generally, the students will often point to another way of interpreting the item and its options – when this happens, it's often said that the item may be "ambiguous" in its present state. Q001 may require revision before being used again.

Do these programs allow an item to have more than one keyed-correct answer? Yes. (In Lertap, it's done through the use of *mws statements.)

Now turn to item Q006. Iteman has no flag for this item, but Lertap has indicated that option 2 may merit a closer look.



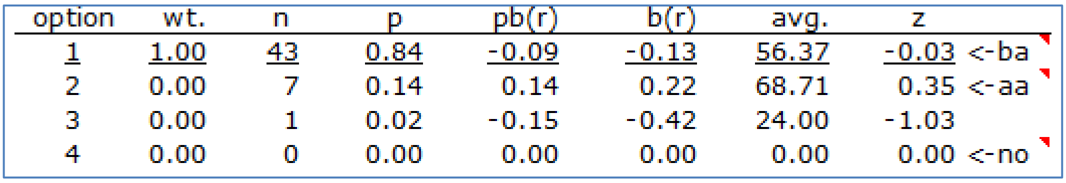
option	wt.	n	p	pb(r)	b(r)	avg.	z	
1	0.00	27	0.53	-0.72	-0.90	35.44	-0.68	
2	0.00	10	0.20	0.35	0.51	80.60	0.71	<-aa
3	0.00	2	0.04	-0.05	-0.11	50.00	-0.23	
4	1.00	12	0.24	0.53	0.73	88.83	0.97	

Can you see why Lertap has flagged option 2? The 10 students who selected it were above-average; the z score corresponding to their "avg." criterion score was 0.71, which, as z scores go, is quite respectable.

Iteman has not flagged this item as Iteman is not quite as fussy as Lertap. As long as pb(r) for the keyed-correct response is positive, and higher than the other pb(r) values, Iteman will not raise a flag.

Which program is correct? Well, it may not surprise you to find that I favour Lertap. We don't want distractors to be taken by above-average students. This being the case, thank you Lertap for waving a flag for Q006's option 2.

Now, for another example, we could look at Q014. As shown at the start of the paper, Iteman had two flags for this item, K and LR. (K for a Key problem, LR for low pb(r)). Here's Lertap's summary for this item:



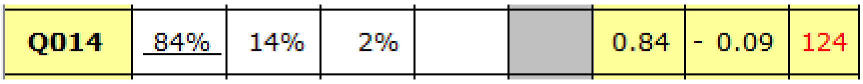
option	wt.	n	p	pb(r)	b(r)	avg.	z	
1	1.00	43	0.84	-0.09	-0.13	56.37	-0.03	<-ba
2	0.00	7	0.14	0.14	0.22	68.71	0.35	<-aa
3	0.00	1	0.02	-0.15	-0.42	24.00	-1.03	
4	0.00	0	0.00	0.00	0.00	0.00	0.00	<-no

The keyed-correct answer is 1, selected by 43 of the students. Their avg. criterion score was 56.37; the z score of -0.03 corresponding to this avg. figure is negative, meaning that this avg. was below the overall average on the criterion score (this is what **<-ba** is signalling). This is unwanted; those who select the keyed-correct option should be the strongest students, the above-average ones. We expect Lertap to flag option 1 because of this.

Then, look at option 2, selected by 7 students. This is a distractor. We want distractors to be selected by weak students. Were these 7 students weak? No. They've got a positive z score, and their avg. is not only above average, but it is also higher than those who took option 1, the keyed-correct response. We expect Lertap to flag option 2 with **<-aa**.

Finally, no-one took the last option, 4. Lertap will flag this, too as it's a non-functioning option. (The **<-no** seen above indicates that no-one selected the option.)

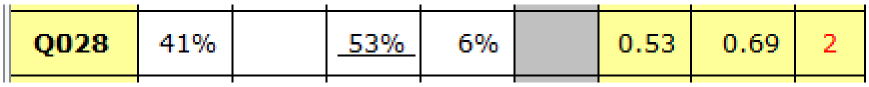
Here's what Lertap said in its brief statistics report:



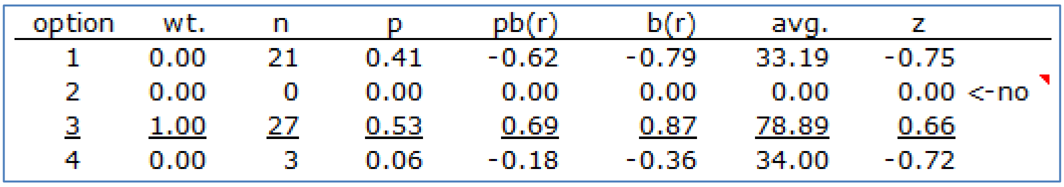
Q014	84%	14%	2%			0.84	- 0.09	124

In this case, Iteman, with its K and LR flags for Q014, is doing fairly well, sort of on-song with Lertap. Both programs have flagged the item. Iteman raised two flags. Lertap raised three.

Next, consider these Lertap summaries for Q028. Lertap has flagged this item while Iteman has not:



Q028	41%		53%	6%		0.53	0.69	2



option	wt.	n	p	pb(r)	b(r)	avg.	z	
1	0.00	21	0.41	-0.62	-0.79	33.19	-0.75	
2	0.00	0	0.00	0.00	0.00	0.00	0.00	<-no
3	1.00	27	0.53	0.69	0.87	78.89	0.66	
4	0.00	3	0.06	-0.18	-0.36	34.00	-0.72	

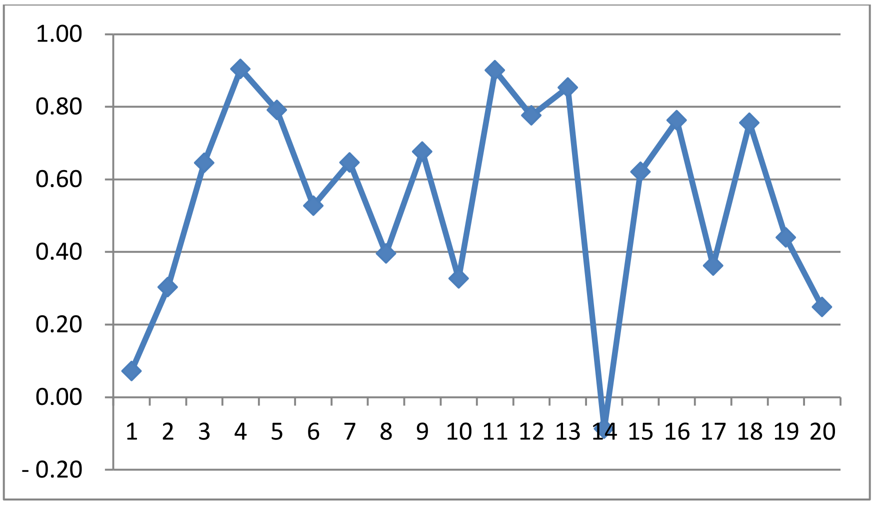
Iteman does not have a flag for non-functioning distractors, but Lertap does. Lertap's flag of 2 for item Q028, and its flag of 4 for Q014, are pointing to inactive distractors. When only a few students have sat the test, a flag of this sort is almost to be expected. It's when a hundred or more students have taken the test that we'll generally not want to get these flags.

Flagmented?

So, there you have it. Of the two programs, Lertap can be said to be more diligent when it looks for possible item problems and goes about waving flags.

Two points which I feel worthy of special mention: (1) Lertap always makes its brief summary of item performance, but Iteman only does so when items have a problem. I feel that having the brief summary always on hand is reassuring; it can be comforting to scan down Lertap's **?** column and find it entirely void of flags. And, at the same time, we get a quick view of summary item statistics for every item. (*There is no equivalent in Iteman unless each item has a problem.*) Point (2): Lertap includes the **<-aa**, **<-ba**, and **<-no** flags in its detailed item performance summaries; *Iteman's detailed performance tables do not have an equivalent*.

There is yet another point. Having items statistics in a table such as seen in Lertap's brief summary lends itself to quick graphs. For example, the plot below was created by selecting the "disc." values from Lertap's brief summary for the first twenty items, and then getting Excel to make a line chart from them. The plot has disc. values along the y-axis, with item numbers along the x-axis.

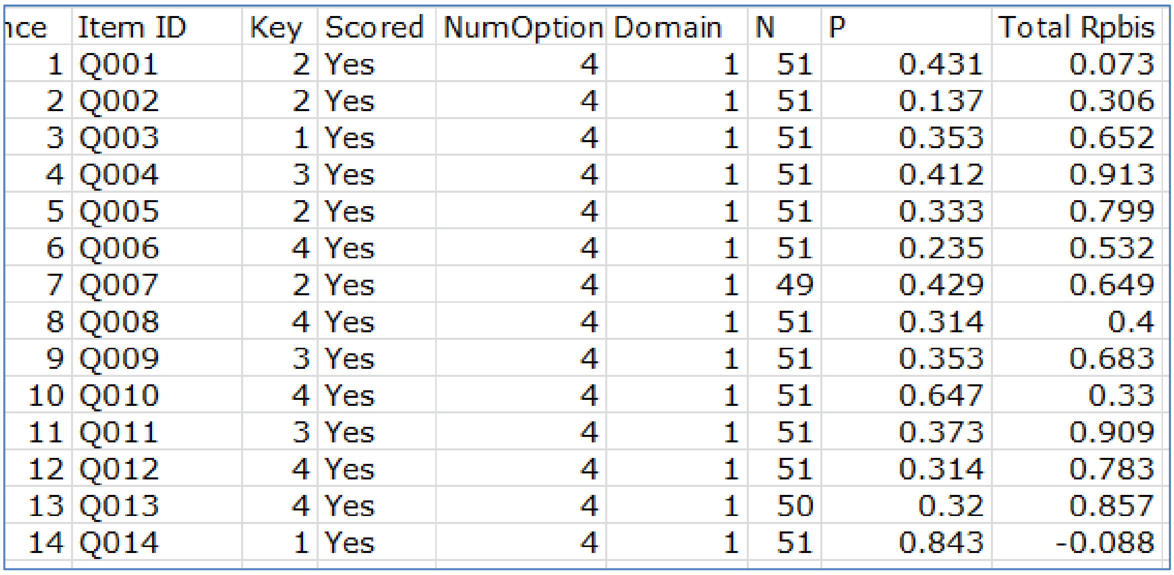


In Lertap, “disc.” stands for “discrimination”. It’s identical to pb(r) for an item’s keyed correct option, and thus is the same as the Rpbis figure seen in Iteman.

A test with good reliability will have items whose disc. values are all positive. And the higher the better. If all items have a disc. of at least 0.30, our chances of having a test reliability in excess of 0.80 will generally be good. My little Excel graph readily indicates that most of these twenty items have disc. figures above 0.30, but a few do not. The disc. value for the first item (Q001) is very low while, for Q014, it’s negative (highly unwanted).

Lertap is an Excel application. As such it lends itself to graphs of all sorts.

It’s possible to get this same graph from Iteman too, but not as quickly. Iteman’s main output is an RTF file (rich text file), ready for viewing in an application such as Word. But, behind the scenes, Iteman also outputs CSV (comma-separated values) files, and one of them has item statistics. An example is seen below. The “Total Rpbs” column is equivalent to Lertap’s disc. column.



nce	Item ID	Key	Scored	NumOption	Domain	N	P	Total Rpbis
1	Q001	2	Yes	4	1	51	0.431	0.073
2	Q002	2	Yes	4	1	51	0.137	0.306
3	Q003	1	Yes	4	1	51	0.353	0.652
4	Q004	3	Yes	4	1	51	0.412	0.913
5	Q005	2	Yes	4	1	51	0.333	0.799
6	Q006	4	Yes	4	1	51	0.235	0.532
7	Q007	2	Yes	4	1	49	0.429	0.649
8	Q008	4	Yes	4	1	51	0.314	0.4
9	Q009	3	Yes	4	1	51	0.353	0.683
10	Q010	4	Yes	4	1	51	0.647	0.33
11	Q011	3	Yes	4	1	51	0.373	0.909
12	Q012	4	Yes	4	1	51	0.314	0.783
13	Q013	4	Yes	4	1	50	0.32	0.857
14	Q014	1	Yes	4	1	51	0.843	-0.088

Excel loves to open CSV files and, consequently, I could get the same plot by selecting down the Total Rpbis column seen here (providing I know what Iteman has called the CSV file, and where it has been saved).