Interactive Log-Linear and Contingency Table Analysis with ILOG

Roger Bakeman
Georgia State University

Augusto Gnisci
Second University of Naples

Byron F. Robinson
Robinwood Consulting

January 12, 2015

Author Note

Roger Bakeman, Department of Psychology, Georgia State University; Augusto Gnisci, Department of Psychology, Second University of Naples.

Correspondence concerning this article should be addressed to Roger Bakeman, 1339 Miller Ave NE, Atlanta, GA 30303, USA, or Augusto Gnisci, Department of Psychology, Second University of Naples, Viale Ellittico, 31, 81100 Caserta, Italy, or Byron F. Robinson, Robinwood Consulting, 4550 Jett Rd, Atlanta, GA 30327, USA. Electronic mail may be sent via Internet to bakeman@gsu.edu, augusto.gnisci@unina2.it, or bfrobins@bellsouth.net.

Abstract

Log-linear analysis is a useful but under-used technique for investigators who categorize some entity (persons, dyads, various kinds of events such as bids for attention or turns of talk in conversations, etc.) on several dimensions using nominal scales and then tally the results in a multi-dimensional contingency table. In searching for the most parsimonious yet tolerably fitting model for such data, log-linear analysis can provide straightforward answers to a wide array of research questions. Log-linear analysis specifically and contingency table analysis generally can be facilitated by an interactive computer program such as the ILOG program described here. ILOG allows the user to re-order searches for a fitting model interactively and collapse, expand, and otherwise manipulate multi-way contingency tables in ways that most standard statistical programs do not. It is available for free download at www.gsu.edu/~psyrab/ilog/.

Keywords:

Quantitative research; log-linear analysis; contingency table analysis; categorical data; nominal data; contingency tables; multi-dimensional tables; two-dimensional tables; frequency tables; hierarchical models; parsimonious models; Pearson chi-square; log-likelihood chi-square; chi-square difference test; free statistical software.

Interactive Log-Linear and Contingency Table Analysis

In a line that has been quoted by many writers since, the ancient Greek poet Archilochus wrote that the fox knows many things, but the hedgehog knows one big thing. Log-linear analysis is a hedgehog among statistical techniques: It does one thing very well, which is analyze the counts of multi-dimensional contingency tables.

Investigators with ordinal and interval-scale data will need to look elsewhere for more generalized analytic methods, but investigators who categorize some entity (persons, dyads, various kinds of events such as bids for attention or turns of talk in conversations, etc.) on several dimensions using nominal scales and then tally the results in a multi-dimensional contingency table can find log-linear analysis a straightforward way to address their research questions (e.g., see Bakeman & Robinson, 1994; Wickens, 1989).

There are at least three reasons why log-linear analysis is seldom used. First, as just noted, is its specialization; its use is limited to contingency table analysis. Second, although most statistical packages include one or two log-linear analysis programs, they are not inherently interactive and, as we argue here, log-linear analysis is facilitated with an interactive computer program. Third, the applied statistics taught to behavioral scientists generally tend to de-emphasize analysis of nominal data; for example, most introductory statistic tests in the behavioral and social sciences relegate chi-square analysis—which, as we will show, is log-linear analysis for two-dimensional tables—to a final chapter, one that, in the press of time, is often ignored by instructors.

Our primary point is that log-linear analysis is a potentially useful but often-overlooked technique in behavioral research (education, psychology, sociology, etc.). Here we provide an introductory tutorial for log-linear analysis and demonstrate how easily it can be effected with ILOG, the interactive computer program described here. In particular, ILOG allows you to define a series of hierarchic log-linear models and re-order searches for a fitting model interactively—and also collapse, expand, and otherwise manipulate multi-way contingency tables—in ways that most standard statistical programs do not. This interactive capability facilitates both analysis and interpretation.

Structuring Contingency Table Data

Before any computer analysis can begin, files containing the relevant and appropriately structured data need to be prepared, a point so obvious it probably goes without saying. The most common format is a simple grid, a two-dimensional table with rows indicating cases (persons, etc.; *subjects* in older literature) and columns indicating variables. Examples of this format include the data sheets of most statistical programs and the individual sheets of most spread sheet programs—although proprietary programs typically obfuscate this simple format with binary files, files that require specialized software to decipher. For text files—files that consist of lines of easily-readable text—one common convention is to separate columns with tab characters (another common convention uses a comma to separate values) and most proprietary programs allow for

the import and export of such tab-delimited files. This makes tab-delimited files a useful format for the exchange of grid-organized data.

Entering data into ILOG: Direct entry.

There are two ways to enter data into ILOG. The first is by direct entry. When ILOG first opens, the default data display is a 2×2×2 contingency table whose cells all contain zero (see Figure 1). Its three factors (dimensions) are named A, B, and C and their levels are named A1 and A2, B1 and B2, and C1 and C2.

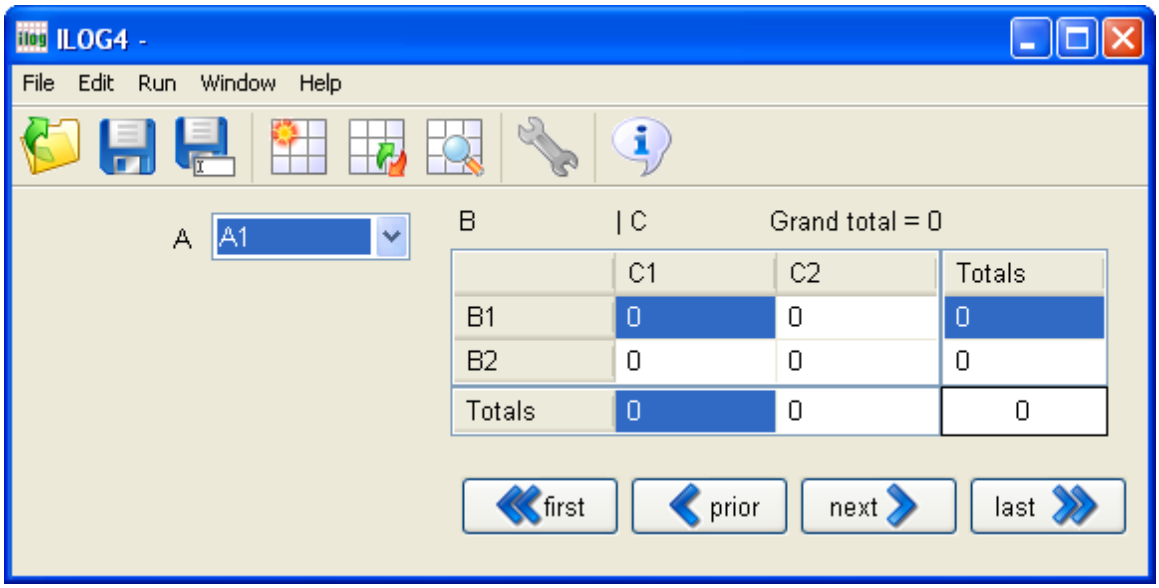


Figure 1. Main window for ILOG4, when first opened, showing tool bar icons and the default data display for a 2×2×2, A by B by C table. The display shows the B by C table for level A1. To display the B by C table for level A2, either select *next* or select level A2 for factor A from the drop down list box to the left.

If you intend to analyze a 2×2×2 table, you could proceed directly to enter counts in the appropriate cells. Usually, however, you would begin by selecting **Run > Define a New Table**, which lets you define the number of factors, the number of levels for each, and factor and level names that reflect your data. You can then enter counts directly into the cells of the contingency table displayed on your computer screen.

The order of the factors matters. An order may seem arbitrary, but necessarily factors must be listed in some order and that order affects how tables are displayed. It should also reflect how you think about your factors. The factor you think of as prior to the others should be listed first, followed by the other factors in order, with the factor you think of as the outcome—the factor you want to explain—coming last. This assures that tables are displayed in a way that makes sense.

For a two-dimensional table, it is conventional to think of rows as representing the antecedent (or *given*) factor and columns as representing the outcome (or *target*) factor. With more than two dimensions, a contingency table is made up of several separate two-dimensional tables. Let *a*, *b*, *c*, etc. represent the number of levels for factors A, B, C, etc. Then a three-dimensional table can be represented with *a* separate $b \times c$ tables, a four-dimensional table with *a* times *b* separate $c \times d$ tables, and so forth. The rows of the two-dimensional tables ILOG displays represent the next to last factor and the columns represent the last factor. Making the last factor listed represent the outcome of interest ensures that the columns of the two-dimensional tables displayed on the computer screen will represent outcome.

Entering data into ILOG: File entry.

Instead of entering data directly, data can be read from a tab-delimited text file; thus selecting **File > Open an Existing Data File** is the second way to enter data into ILOG.

The first line of this file consists of column headings: the first is ID, the next are the names of your factors, and the final is COUNT, all separated with tabs (you can use words other than ID and COUNT if you wish). For the remaining lines, the first column contains an identifier for each line; it can be anything you want. Let *N* represent the number of factors; then the next *N* columns contain names for a particular factor level. The final column contains the count for that particular combination of level names (thus each line contains $N + 1$ tabs).

Several lines could contain the same level names, in which case the counts would accumulate in the designated cell (i.e., the cell indicated by that combination of level names). Thus, if your data are already entered in another program's datasheet or a spread sheet, a tab-delimited file exported from these programs can be imported directly into ILOG (**File > Open an Existing Data File**). Moreover, if you entered data directly into ILOG, it can be saved to a file (**File > Save the Current Table**) and that file imported into a statistical or spread sheet program.

As an example, consider Bakeman and Brownlee's (1982) study of object struggles in toddlers and preschool children during free play. They asked observers (working from video records) to detect *possession struggles*—i.e., times when one child (the *holder*) possessed an object and another (the *taker*) attempted to take it away—and to code each possession struggle on four dimensions:

1. *Age*—whether the children were observed in the toddler or the preschool classroom,
2. *Dominance*—whether the taker had been judged dominant to the holder,
3. *Prior Possession*—whether the taker had had prior possession of the contested object within the previous minute, and
4. *Resistance*—whether the holder resisted the taker's attempt (the last three were coded yes or no).

Bakeman and Brownlee regarded *Resistance* as the outcome of interest, reasoning that holders would be less likely to resist if they believed the taker had a claim on the object, presumably through prior possession, or if the taker were dominant. Thus the factors are ordered Age of children, Dominance of taker, Prior Possession of taker, and Resistance of holder. The data for their study were organized as a 2×2×2×2 table, the tab-delimited version of which is shown in Figure 2. We use these data subsequently to illustrate various ILOG procedures.

ID	→	Age	→	DomiT	→	PriorT	→	ResistH	→	COUNT
1	→	todler	→	yes	→	yes	→	yes	→	19
2	→	presch	→	yes	→	yes	→	yes	→	6
3	→	todler	→	no	→	yes	→	yes	→	16
4	→	presch	→	no	→	yes	→	yes	→	9
5	→	todler	→	yes	→	no	→	yes	→	42
6	→	presch	→	yes	→	no	→	yes	→	18
7	→	todler	→	no	→	no	→	yes	→	61
8	→	presch	→	no	→	no	→	yes	→	27
9	→	todler	→	yes	→	yes	→	no	→	7
10	→	presch	→	yes	→	yes	→	no	→	5
11	→	todler	→	no	→	yes	→	no	→	4
12	→	presch	→	no	→	yes	→	no	→	6
13	→	todler	→	yes	→	no	→	no	→	30
14	→	presch	→	yes	→	no	→	no	→	5
15	→	todler	→	no	→	no	→	no	→	13
16	→	presch	→	no	→	no	→	no	→	4

Figure 2. The first line contains names for the four factors of this 2×2×2×2, Age by Dominance by Prior Possession by Resistance contingency table. The remaining 16 data lines contain level names for each factor—these uniquely identify a cell in the table—along with a count for that particular cell. Items on each line are separated with tabs, shown here as an arrow. The file could have more than 16 data lines, in which case counts for additional lines that contained the same level names would accumulate in the designated cell. In the extreme, for each cell there could be as many lines as there are counts in that cell, with each line having the same level names but a count of 1; such a file might result when exporting from a statistical program like SPSS.

When these data are opened in ILOG (and we urge you to try it), they would be displayed on the screen as shown in Figure 3.

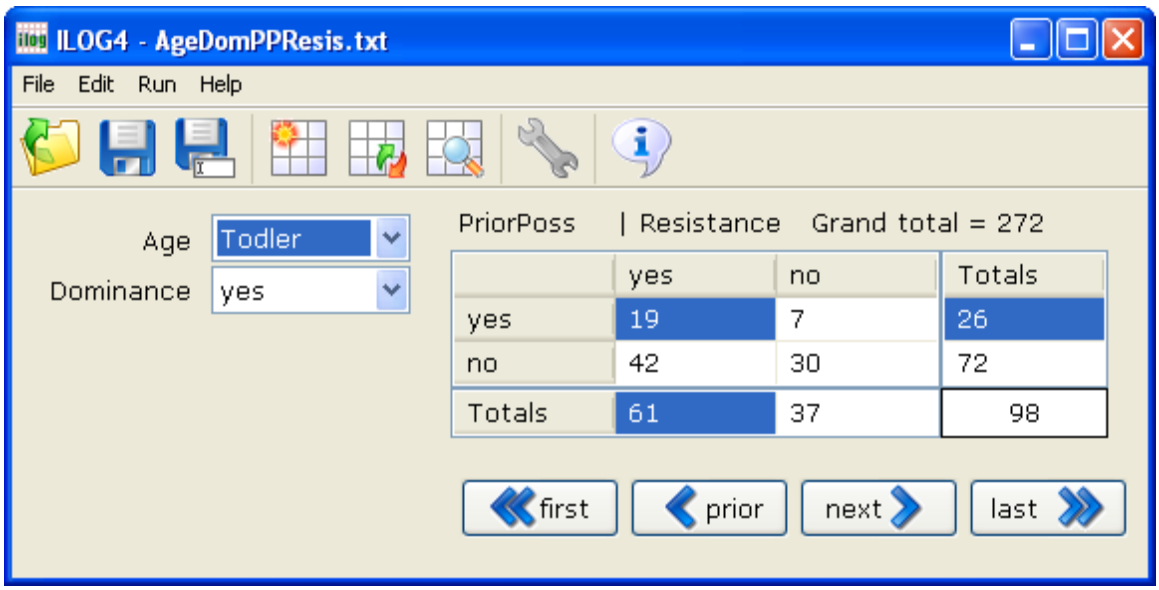


Figure 3. Main window for ILOG4, after importing the 2×2×2×2, Age by Dominance by Prior Possession by Resistance contingency table from the Bakeman & Brownlee study.

Modifying an Existing Contingency Table

It is often useful to modify an existing contingency table before further analysis. Contingency table data may be available to you in spread sheet or datasheet form, but not structured exactly as you wish, or with names for factors and levels other than those you prefer. To modify an existing table in ILOG, select **Run > Modify This Table**.

This procedure lets you edit factor and level names, thus changing existing names to the ones you want. It also lets you reorder factors so that your presumed output factor comes last; and also delete factors, thereby reducing the number of dimensions of the contingency table. Additionally, you can reorder levels, lump levels together (useful if some levels have few counts), insert new levels (useful if additional data become available), or delete existing ones (useful if you want to discard cases assigned to a particular level).

These functions give you considerable freedom in setting up particular analyses or overcoming technical problems (e.g., eliminating an entire level of a variable in case of few counts). In sum, ILOG provides a number of ways to manipulate and modify an existing contingency table. Exploring these functions with actual data should give you a sense of ILOG's flexibility and power in this regard.

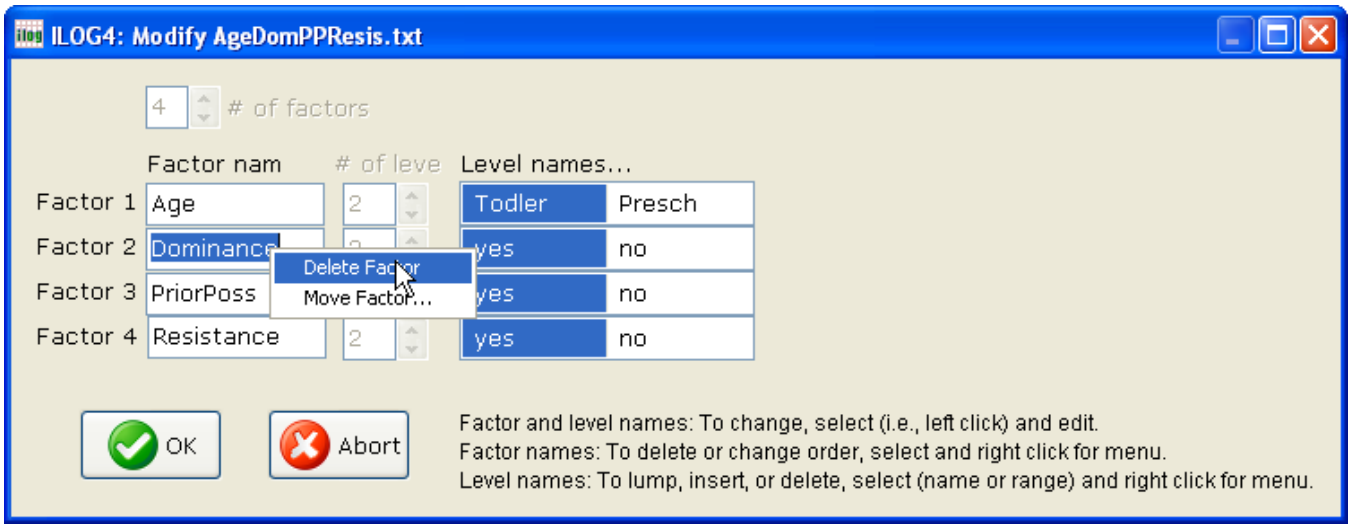


Figure 4. Modifying a table to delete the Dominance factor. Select Dominance factor, right click for context menu, then select *Delete Factor*.

For example, if we had imported the data file described earlier, but decided we did not want to consider Dominance further, we could delete the Dominance factor as follows: select *Dominance*, right click for context menu, then select *Delete Factor* (see Figure 4). The resulting 2x2x2 table would be displayed by ILOG as shown in Figure 5. Note that the count for toddlers with prior possession showing resistance is 35, the sum of the two cells with (19) and without dominance (16) from the data file.

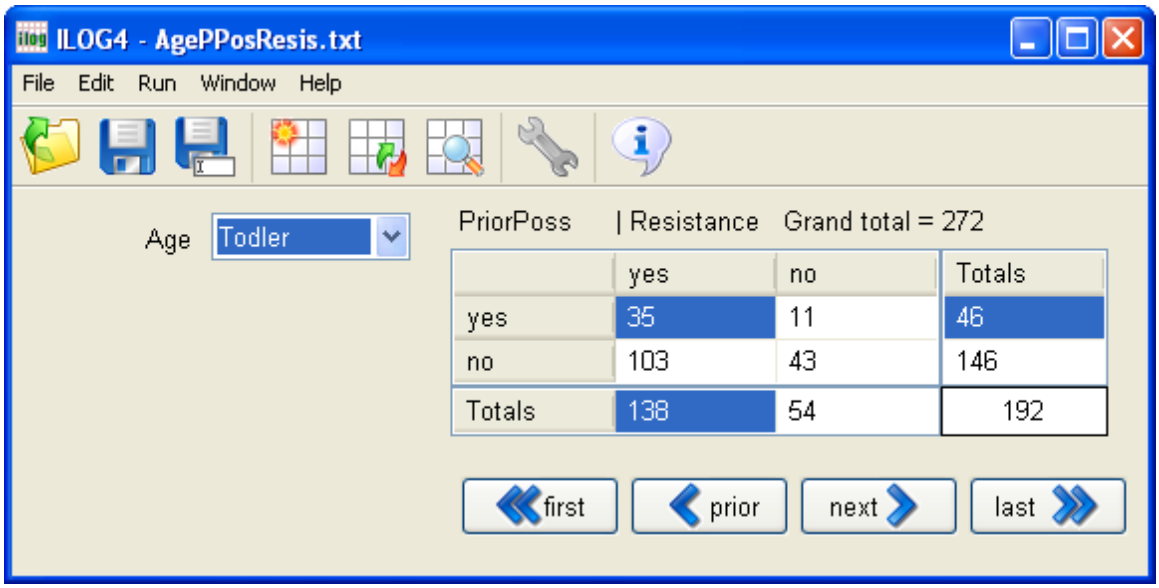


Figure 5. Main window for ILOG4, after deleting the dominance factor, which results in a 2×2×2, Age by Prior Possession by Resistance contingency table.

Analyzing Two-Dimensional Tables

The data for any *N*-dimensional table could be analyzed using nothing more than the two-dimensional chi-square analyses of introductory statistics courses. Examining the constituent two-dimensional tables derived from a larger *N*-dimensional table would usually be characterized as piece-meal analysis, but could be justified as follow-up analyses, guided by log-linear results (as we show later)—much in the same way as, with continuous data, follow-up analyses are used to explicate significant analysis of variance interactions. Examining separate two-way tables can be useful in itself, but because ILOG users will most likely pursue such an examination as follow-up to log-linear analyses, it makes some sense to defer discussion of this topic until after log-linear analysis has been introduced.

Nonetheless, we discuss examining two-way tables first because of the way it introduces concepts essential to log-linear analysis, but in the more familiar context of two-dimensional chi-square. To invoke the ILOG procedure that gives counts and other statistics for two-dimensional tables, select **Run > Examine as Two-Way Tables**.

You then name the factors for the rows and columns of the two-dimensional table you wish to examine and indicate whether other factors should be *pooled* (tables are collapsed over such factors) or *listed* (separate two-dimensional tables for each level of such factors are displayed, or combinations of levels if more than one factor is listed). Appropriate statistics for each two-dimensional table are then displayed: observed and expected frequencies and adjusted residuals for each cell, chi-square for each table, and the odds ratio and Yule's Q for 2×2 tables (for detailed definitions of these and other cell and table statistics see Bakeman & Gottman, 1997; Bakeman & Quera, 2011).

For example, for the object struggle data, we might request that rows represent *Prior Possession* and columns *Resistance*, listed separately by *Age*, but pooled over *Dominance* (see Figure 6). Analyzing the toddler and preschool tables separately shows a significant chi-square for preschoolers but not toddlers (p = .013 vs. .47).

For preschoolers, the odds that the holder resisted when the taker had prior possession were less than when the taker did not have prior possession: OR = 0.27 [0.09, 0.78)—95% confidence interval (CI) in brackets. In contrast, for toddlers, the odds that the holder resisted when the taker had prior possession were somewhat greater than when the taker did not have prior possession, but not significantly so: OR = 1.33 [0.62, 2.86).

Because we checked *odds ratio* for *Write checked stats*, these statistics were displayed in the ILOG results window (see Figure 7).

In sum, a piece-meal analysis would suggest that a taker's prior possession decreased the holder's resistance for preschoolers, but had little effect on toddlers. This piece-meal analysis, however, only tells us that the prior possession-resistance association was statistically significantly just for preschoolers (the 95% CI excluded 1) but not for

toddlers. In particular, it does not tell us whether the difference between the toddler and preschool effect was itself statistically significant. For that a log-linear analysis is needed.

Nonetheless, the **Examine as Two-Way Tables** procedure provides a powerful way to explore a multi-way contingency table.

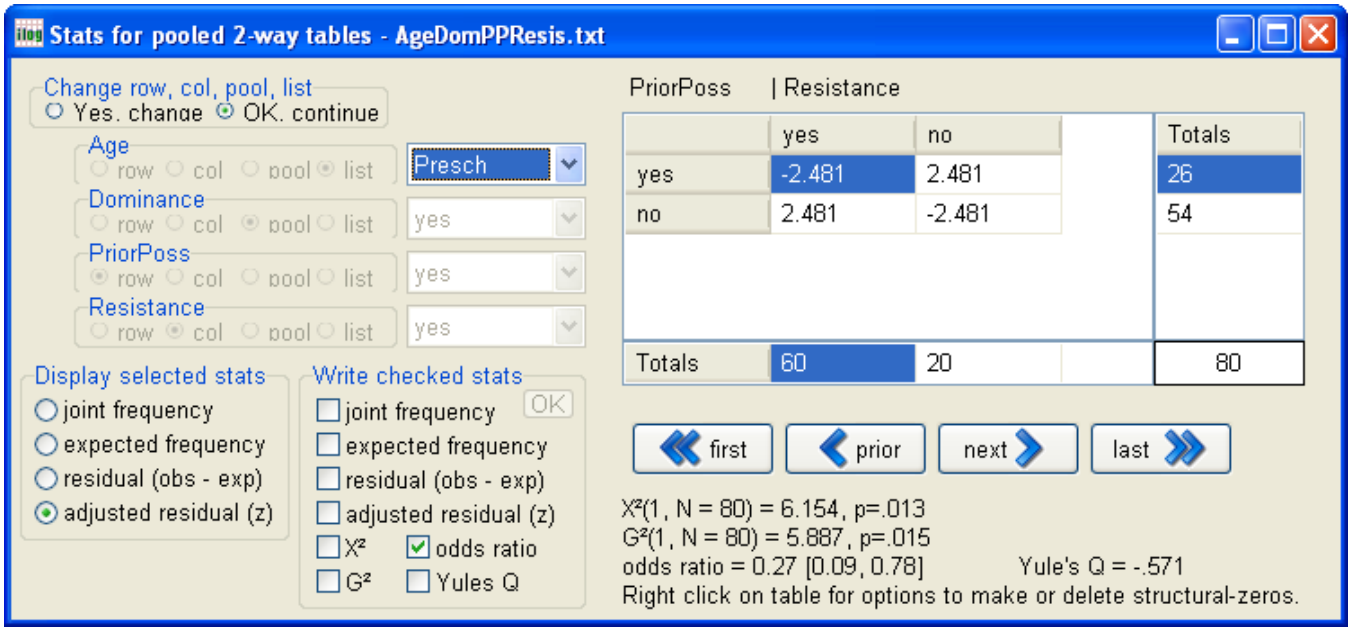


Figure 6. Window for *Examine as Two-Way Tables* with object struggle data. Age selected as list, dominance as pool, PriorPoss as row, and Resistance as column.

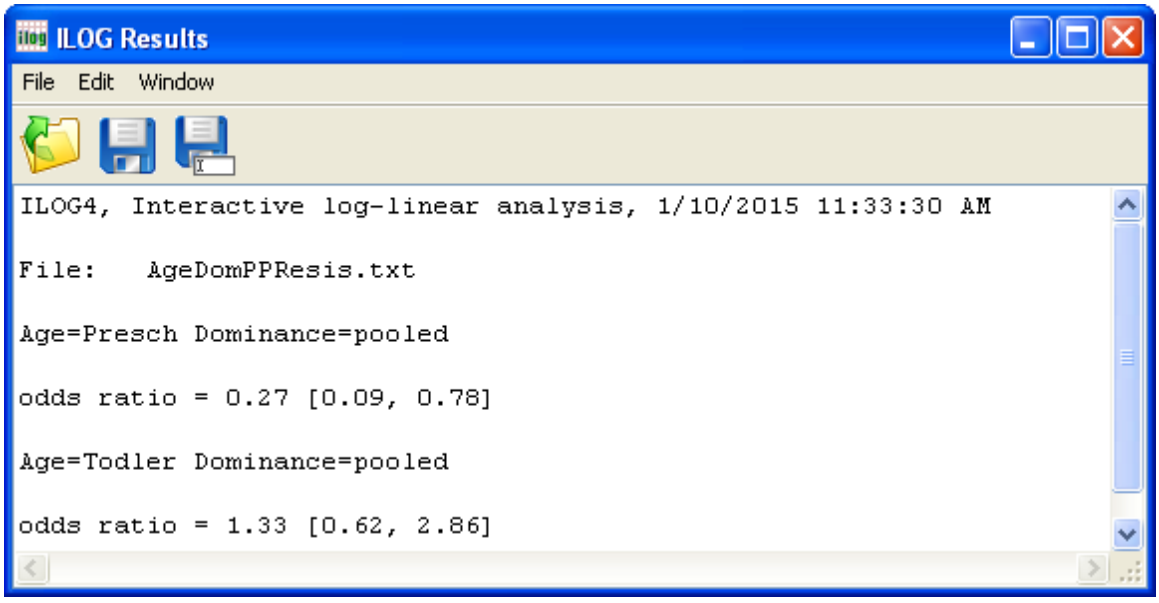


Figure 7. Results window when *odds ratio* was checked for *Write checked stats.*

Log-Linear Analysis of *N*-Dimensional Tables

Log-linear analysis is used to analyze, not just two-dimensional tables, but contingency tables of more than two dimensions. As such, it can be regarded as an *N*-dimensional extension of the chi-square analyses of introductory statistics courses. Among the standard references are Bishop, Fienberg, and Holland (1975) and Fienberg (1980), although more accessible alternatives are Bakeman and Robinson (1994), Kennedy (1992), and Wickens (1989), with Wickens being especially thorough and clear.

A typical log-linear analysis begins by defining a series of hierarchical models. A series of models is hierarchic when higher level models include all terms present in lower order models, and lower level models omit one or more terms from the model immediately preceding it. As terms are deleted—deleting higher-order terms before lower-order ones—more parsimonious models result, that is, each model in the series is less complex than the one before it.

The goal of log-linear analysis is to identify the simplest model that still provides an acceptable fit to the data. Thus the best fitting model combines parsimony and information. Models generate expected counts for the cells of the contingency table. Less complex models, having fewer terms, are less constrained and so have more degrees freedom—and consequently the counts they generate fit the observed data less well. The question is, how much less? How bad is a less well-fitting model?

A model's goodness-of-fit—more accurately, badness-of-fit—is assessed with the likelihood ratio chi-square (symbolized G^2), an alternative computation for the more familiar Pearson chi-square ($\chi 2$) that, for technical reasons associated with its decomposition, is preferred in log-linear analysis. Both G^2 and χ^2 express the discrepancy between the data collected and a hypothesized model that indicates how variables are related; in other words, they reflect the difference between the observed counts and the expected counts generated by a particular model. The greater the difference between the observed and expected counts, the larger G^2 and χ^2 become.

The bigger G^2 is, the worse the model fits. The first model in the hierarchic series—the *saturated model*—generates expected frequencies that match the observed ones exactly and, for that reason, fits the data perfectly: Its $G^2 = 0$ with 0 degrees of freedom. The question then becomes whether a more parsimonious model, one with a larger G^2, will still fit acceptably.

A common criterion for a *tolerably fitting model* is the significance of its G^2: If G^2 is small and *not* significant, $p > .05$, the discrepancies between the observed cell counts and those generated by the model must be relatively small, and so we conclude that the model fits acceptably. However, given large counts, this criterion may be too strict because even relatively small deviations from expected will result in a G^2 significantly different from zero.

A second criterion, useful when counts are especially large, is the magnitude of Q^2, which is a comparative fit or reduction in error index analogous to the R^2 of multiple regression. Knoke and Burke (1980) suggested that any model whose Q^2 is greater than .90 provides satisfactory fit, even if its G^2 differs significantly from zero. Q^2 is the proportion of a specified base model's G^2 that is accounted for by the model in question, specifically:

$$Q^2 = \frac{G^2_{base} - G^2_{model}}{G^2_{base}}.$$

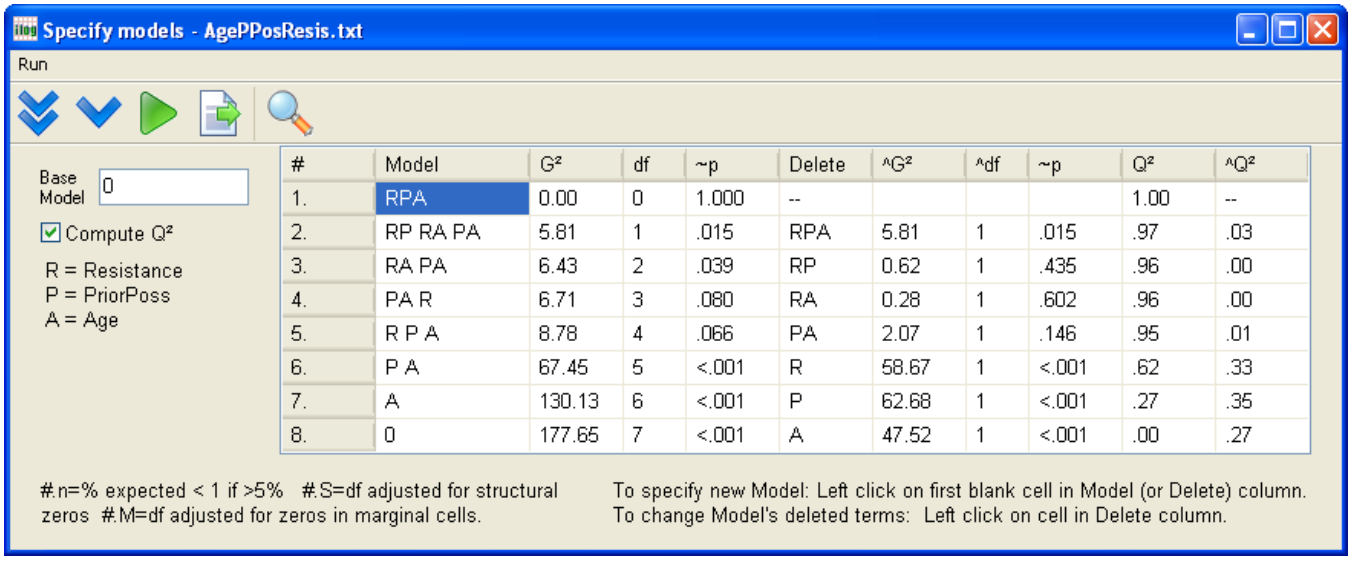
In other words, Q^2 indicates how much initial failure to fit is reduced by a particular model. When the terms in this model account for over 90% of the baseline model's failure to fit we conclude that the model fit is acceptable and that the terms deleted to form the model are not consequential. Selecting an appropriate baseline model can be something of an art. Absent a rationale, a safe choice is the equiprobable or *null model*, the model that predicts that all cells will have the same count. When one factor is clearly regarded as the outcome, another choice is the *outcome-and-design model*, as described shortly.

Bracket notation. An especially convenient way to specify log-linear models is with bracket notation. As an example, consider the prior possession study whose four dimensions are A, D, P, and R (for Age, Dominance, Prior Possession, and Resistance). As previously noted, the most complete model is called the saturated model. Using bracket notation, it can be represented with the single 4-way term [RPDA]—here we have reversed the order of the factors so that, as in a multiple regression equation, the presumed outcome variable is listed first (ILOG automatically reverses the order of factors when making models). The saturated model is conventionally shown as a single term—letters representing each of the factors enclosed in square brackets. In fact, it also includes implicitly all possible lower-order terms, in this case the four 3-way terms—[RPD][RPA][RDA][PDA]; the six 2-way terms—[RP][RD][RA][PD][PA][DA]; the four 1-way terms—[R][P][D][A]; and a constant, indicated here as the null model—[0], the model that predicts equal counts for each cell and so maximizes G^2 (i.e., failure to fit).

Specifying models. ILOG lets you specify models in three different ways. As an example, imagine that we used the modify table procedure to delete the dominance factor from Figure 2's 2×2×2×2 table, resulting in a 2×2×2, Age by Prior Possession by Resistance table for analysis. To begin analyzing log-linear models in ILOG, select **Run > Specify Log-Linear Models**.

The *Specify Models* window that would open for the 3-dimensional table just described is shown in Figure 8. Initially, the first line would indicate the saturated model (without brackets), but all other lines would be blank. One way to specify a new model is to select (i.e., left click) the first blank cell in the model column; a list of terms will be

displayed and a new model will be built from the terms selected. A second way is to select a cell in the delete column; a list of terms will be displayed and a new model will be built, but with the selected terms deleted. A third way is more automatic. Selecting **Run > Make All-Level Models** generates a list of all possible models, from the saturated to the null, as shown in Figure 3, whereas selecting **Run > Make Next-Level Models** generates just models for the next level. For example, if only the 3-way saturated model is listed (and selected), **Make Next-Level Models** generates all models involving 2-way terms up to and including the model that consists of all 1-way terms (lines 2–5 in Figure 3).



#	Model	G²	df	~p	Delete	^G²	^df	~p	Q²	^Q²
1.	RPA	0.00	0	1.000	--				1.00	--
2.	RP RA PA	5.81	1	.015	RPA	5.81	1	.015	.97	.03
3.	RA PA	6.43	2	.039	RP	0.62	1	.435	.96	.00
4.	PA R	6.71	3	.080	RA	0.28	1	.602	.96	.00
5.	R P A	8.78	4	.066	PA	2.07	1	.146	.95	.01
6.	P A	67.45	5	<.001	R	58.67	1	<.001	.62	.33
7.	A	130.13	6	<.001	P	62.68	1	<.001	.27	.35
8.	0	177.65	7	<.001	A	47.52	1	<.001	.00	.27

Figure 8. Specify Models window for ILOG4, shown after opening a 2×2×2, Age by Prior Possession by Resistance contingency table (based on data in Figure 2), after checking *Compute* Q^2, and selecting **Run > Make all level models** and **Run > Compute model stats**. The figure shows the G^2, degrees of freedom, and approximate probability for each model and, for models after the saturated model, the term deleted from the previous model to create it and the ΔG^2 and its degrees of freedom and approximate *p*-value for the chi-square difference test.

You probably see a pattern here, especially if you recall permutation formulas or Pascal's triangle (Bakeman & Robinson, 2005). With two dimensions (factors) and including the saturated and the null, 4 hierarchical models are possible—[YX], [Y][X], [X], and [0]; with three dimensions (as here) there are 8, with four there would be 16, and generally with *N* factors there are 2^N possible hierarchical models. The number of *n*-way terms follows the binomial expansion (Pascal's triangle). For *N* = 2 coefficients are 1, 2, 1; for *N* = 3 they are 1, 3, 3, 1; for *N* = 4 they are 1, 4, 6, 4, 1; etc., where coefficients are the number of *n*-way terms, from the saturated to the null model (e.g., for *N* =4, one 4-way, four 3-way, six 2-way, four 1-way and one null term).

Two points should be emphasized. First, although you can generate 2^N hierarchical models, you are only interested in finding the most parsimonious—that is the last model in the hierarchic series that still fits tolerably well. Second, the order in which terms are

deleted from models that include all *n*-way terms (e.g., lines 2 and 5 in Figure 3) is arbitrary. You can accept the default order or specify your own, based on whatever order makes the most conceptual sense to you. By default, ILOG deletes terms in the left-to-right order you might expect: for example, for [ABC], first [AB] then [AC] then [BC]. But if you prefer a different order, you can simply select a delete term and change the terms deleted as described earlier.

After specifying a series of hierarchic models, next you would select **Run > Compute Model Stats**, which causes a variety of statistics to be displayed in the *Specify Models* window. Of immediate interest is the G^2 for each model, which is displayed along with its degrees of freedom and approximate p-value. Typically, your interest is locating the last model in the series with a non-significant *p*-value, that is, one for which $p > .05$. Alternatively, especially if the number of tallies is large, you may prefer to locate the last model in the series whose Q^2 is at least .90. Other entries indicate, for each model, the terms deleted from the previous model and the degrees of freedom and approximate *p*-value for the deleted term or terms. This constitutes a chi-square difference test (*partial* G^2 or ΔG^2—labelled ^G^2 in ILOG), appropriate when one model is nested in another, and indicates whether removing the deleted terms caused the fit of the model to deteriorate significantly.

Interpretation. Earlier we noted that the association between Prior Possession and Resistance was significant for preschoolers ($p = .015$) but not toddlers ($p = .47$), but that the piece-meal analyses could not tell us wjetjer the magnitude of the association differed between them. The log-linear analysis shown in Figure 8—after checking the *Compute* Q^2 box (for illustration; the number of tallies here is not especially large) and after selecting **Run > Compute Model Statistics**—provides an answer. We would conclude that only the saturated [RPA] model fits because its *p*-value is > .05 and the *p*-value for the next model in the series is less than .05 (assuming an alpha level of .05). In other words, this 3-dimensional table cannot be simplified. The association between Prior Possession and Resistance differed by Age.

Recasting these results in more familiar analysis of variance terms is helpful. If we identify one factor as the outcome, an *N*-dimensional contingency table can be described as an (*N*–1)-way analysis of variance. Imagine the factors are A, B, and Y, with Y the outcome, so that the saturated model is [YBA]; B could be a predictor and A could be a moderator variable (like Age or Gender). The [YB] term indicates a main effect for factor B, the [YA] term a main effect for factor A, the [BA] term simply indicates the design, and the [YBA] term indicates a B×A interaction. In particular, given three factors—one an outcome, one a predictor, and one a moderator—log-linear analysis provides an answer to a common question: Is the association between the outcome and predictor different for different groups, that is, is it moderated by group membership?

Base model. Thinking in analysis of variance terms also helps us determine an appropriate base model. When one variable is regarded as the outcome and others as design variables, an appropriate base model, using the notation of the previous paragraph, is [BA][Y]—this is the *outcome-and-design model* we mentioned earlier. Including [BA] in the base model signals that we are interested in associations between outcome and design variables, not in associations within the design variables; after all, they are often determined by the investigator, as when gender is one factor and we recruit equal numbers of males and females. For example, if we specified [PA][R] as the base model for the Figure 3 analyses (and not the null model as in Figure 3), the Q^2 for the [RP][RA][PA] model would be = (6.71 – 5.81)/6.71 = .13—which is further evidence that this is an ill-fitting model.

If you checked *Compute* Q^2, which requires that a base model be specified, ILOG assumes the base model is the last model listed in the hierarchic series. ILOG lets you state your base model (upper-left edit box in Figure 8). This is useful as a reminder if your base model is something other than the null. If the number of factors is 2 or 3, automatic model generation will stop with the specified base model, but if the number of factors is 4 or more, you will need to select terms to delete to insure that the last model specified is, in fact, your desired base model.

Structural zeros, empirical zeros, and low counts. Cells may contain zero for different reasons. For example, if one factor is gender (male or female) and the other pregnant (yes or no), one of the cells will necessarily be zero—this is called a *structural zero* (any other value is logically impossible) and is indicated in ILOG, not with a 0, but with an asterisk. Structural zeros reduce degrees of freedom and ILOG makes the appropriate adjustments; for any given model, the degrees of freedom adjustment is displayed after the model number.

However, cells may also contain zeros simply because no cases were observed; usually such *empirical zeros* are not problematic if they are few in number. However, if many cells contain zeros or low counts, the expected frequencies computed may be low. If expected frequencies are less than 1 for more than 5% of the cells for any given model, ILOG displays the percentage after the model number. Guidelines vary, but Wickens (1989) has suggested that, for large two-way tables, it may be acceptable, if not desirable, for as many as 20% of the cells to contain expected frequencies less than one—but if the guideline is violated, the test should be abandoned. For multidimensional tables, you should remain wary if many cells are zero.

Empirical zeros can cause another problem. Depending on where they occur, they can cause some cells of the marginal tables used to compute expected frequencies to be zero and thereby reduce degrees of freedom (Wickens, 1989, pp. 120–124). As Wickens writes, “A pre-packaged computer program may or may not make these corrections automatically—one should check to be sure” (p. 120). ILOG does make these

adjustments and notes the number by which degrees of freedom were reduced (displayed after the model number of any affected models).

Deviant cells. A small G^2 for a model indicates that most cells fit well, that is, the expected frequencies generated by the model are fairly close to those observed, whereas a large G^2 indicates just the opposite. For the present example, the saturated model [RPA] fit the observed exactly, but it could be useful to examine why the [RP][RA][PA] model failed to fit. Selecting a particular model in the Specify Models window and then selecting **Run > Examine Selected Model**, opens a window that lets you examines the differences between observed and expected-by-the model frequencies for each cell. For this example, the largest standardized residual (+1.36) is for prior possession without resistance for preschoolers: 11 instances were observed but only 7.31 were expected by the [RP][RA][PA] model. For toddlers, the opposite was true; again 11 instances were observed but 14.7 were expected and, consequently, the standardized residual was negative (–0.96). Thus it is not surprising that the log-linear analysis suggested an interaction—a difference in the prior possession–resistance association between toddlers and preschoolers.

Once you have settled on a tolerably fitting model to interpret, next steps include explicating its terms. As with the significant main effects and interactions of analysis of variance, the included terms indicate how you should explicate the data. For the present example, because you decided that age moderates the prior possession–resistance association—i.e., you accepted the [RPA] model—then you would report that preschoolers were less likely to resist when the taker had prior possession than toddlers (58% vs. 76%). However, if the data had been different and the [RP][RA][PA] model had fit tolerably well, then you would report main effects for age and prior possession (no interaction), along with resistance percentages for age and for prior possession.

One final comment: A strength of ILOG is its ability to re-order models in a hierarchic series. In Figure 8, by default ILOG deleted the [RP] term before [RA], but if you wished to delete the [RA] term first, for whatever reason, you would select the [RP] term in the Delete column and select the [RA] term from the options presented. If the resulting series is hierarchic, ILOG will then re-compute statistics, as appropriate. The advantage of this ability to change the order in which terms are deleted in a hierarchic series becomes more apparent with 4-dimensional tables, for which there are four 3-way terms and six 2-way terms. As with any interactive computer program, exactly how this all works is understood best as you explore the program with your own data.

Winnowing a Two-Dimensional Table

A particularly useful feature of ILOG's *Examine Two-Way Tables* procedure is its winnowing ability (see Bakeman & Gottman, 1997, pp. 119–120; Bakeman & Quera, 2011, pp. 129–130 and 143–144). When the χ^2 or G^2 associated with a two-way table (either a simple two-way table or one created by pooling over levels of other factors) is large and its *p* value small (e.g., < .05), we say the test of independence (of rows and

column) has failed—only the saturated model fits. To interpret this result when there are more than two levels for the row and column factors, we could examine adjusted residuals (Bakeman & Quera, 2011); large ones indicate cells whose expected values deviate significantly from expected. However, adjusted residuals in a two-dimensional table form an interrelated web. If some are large, others necessarily must be small, and so which do we interpret? All of those, for example, larger than 1.96 absolute?

Winnowing offers a more economical approach to interpretation. We identify those cells that cause fit to fail; almost always this will be a smaller number than the number initially identified as large, which offers a more parsimonious interpretation. Winnowing consists of iteratively replacing selected cells (in an order you determine) with structural zeros until we find a table that fits tolerably. We then assume that the cells we replaced caused the bad fit of the table.

As an example, we consider unpublished data from a study of dinner conversation provided by Clotilde Pontecorvo (University of Rome). Turns of talk were coded for *Speaker* (Father, Mother, Target child, or Sibling) and *Action* (uses Knowledge, Relates, Entertains, Controls, or Manages) for six families. Only the saturated model fit, thus we concluded that no common pattern joined Speaker and Action—different Speakers were associated with different functions in different families. We had thought that fathers generally controlled and mothers managed, but these hypotheses were not supported.

In one family, $G^2(12, N = 330) = 78.3$, $p = <.001$, indicating that different speakers favored different actions: 7 of the 20 adjusted residuals exceeded 1.96 absolute (see figure 9). Using the *Examine Two-Way Tables* procedure, and replacing just four cells with structural zeros, produced a fitting model, $G^2(8, N = 212) = 14.716$, $p = .064$ (see Figure 10.

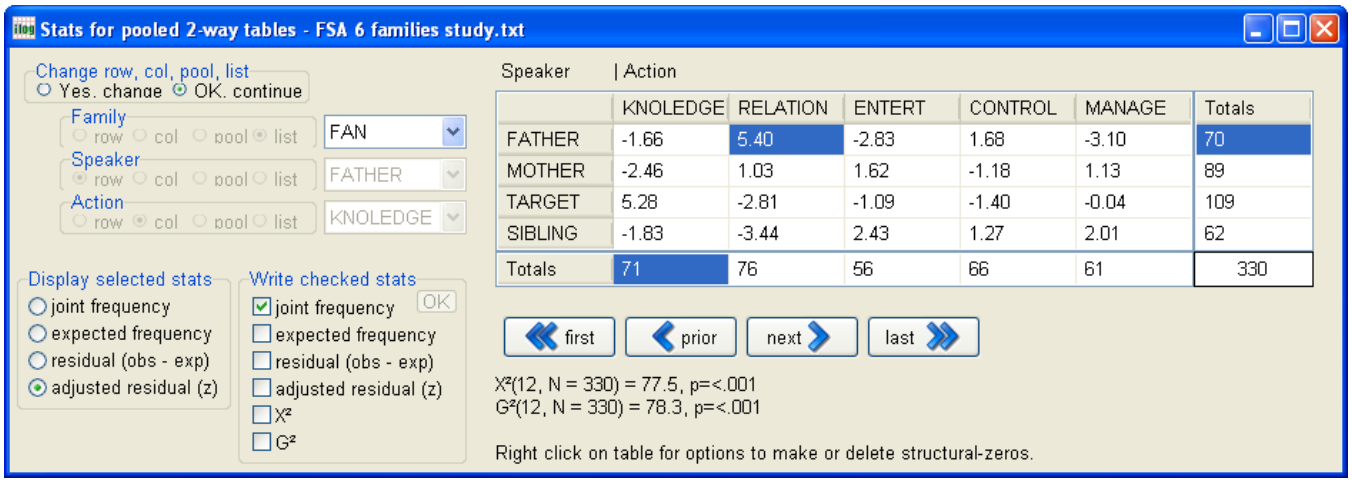


Figure 9. The *Speaker* by *Action* table for one family, showing adjusted residuals and chi-squares before winnowing.

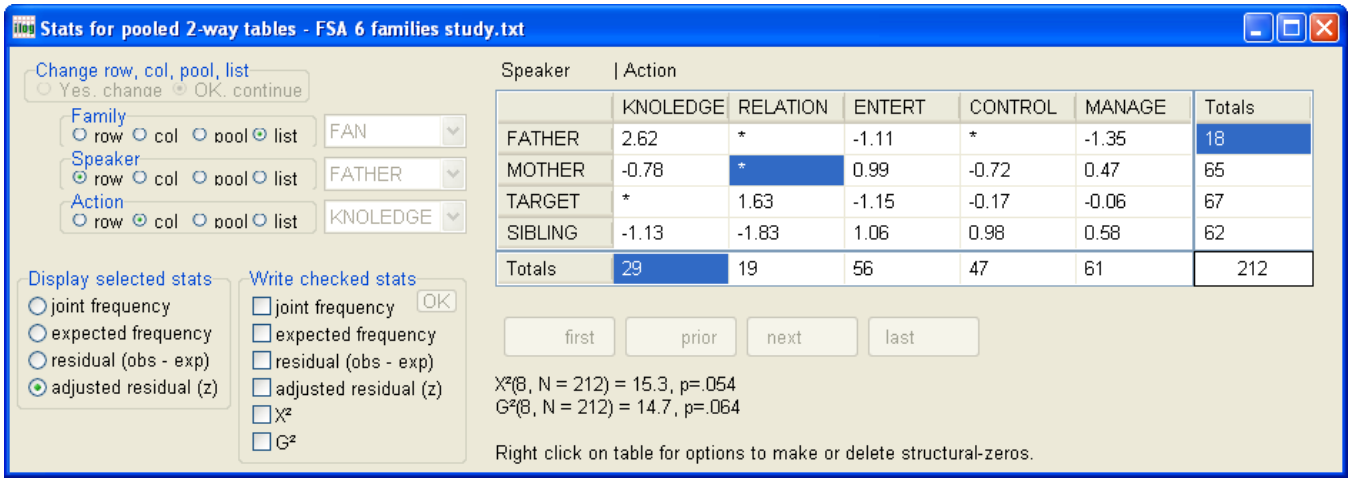


Figure 10. The *Speaker* by *Action* table for one family, showing adjusted residuals and chi-squares after winnowing (replacing four cells with structural zeros).

Specifically, to replace cells with structural zeros, after selecting *Examine as Two-Way Tables* in the main ILOG window, and after selecting the particular two-way table to examine (here, *Speaker* as row, *Action* as column, *Family* as list and selecting the particular family) in the *Examine as Two-Ways* window, position the mouse over the table, right click, and select *Let click make cell a structural zero* from the context menu. Then select (i.e., left click) each of the four cells: father uses relate and control, mother uses relate, and target child uses knowledge. Their counts will be replaced with asterisks (see Figure 10), indicating structural zeros, and the table chi-square will no longer be statistically significant. We conclude that for this family these four associations adequately captured their unique pattern.

More generally, winnowing is an economic way to identify those particular cells in any two-dimensional table that cause fit to fail and is easily effected in ILOG (clicking on particular cells replaces them with structural zeros and re-computes G^2).

Conclusion

We have presented a relatively brief introduction to log-linear analysis. It is by no means complete; several book-length treatments describe log-linear analysis with much more breadth and depth (e.g., Wickens, 1989). Our intent is to provide readers with enough of a sense of what log-linear analysis can do that they can then decide if it would serve them and if they want to learn more. Throughout we have noted how log-linear analysis and other analyses of contingency tables can be effected with an interactive computer program, ILOG. We find that the analysis of hierarchical log-linear models works best when approached interactively, which is what the ILOG program does. We encourage interested readers to enter their own data into ILOG, or use the data given in Figure 2, and then try running the various procedures. As is generally true, exploring the various options that a computer program permits is often an excellent way to learn more about both the analysis performed and the program's capabilities.

ILOG has several advantages. As noted earlier, standard statistical packages typically have one or two log-linear analysis routines. Often they produce many pages of output. As noted here, log-linear analysis typically proceeds by comparing models in a hierarchic series, searching for a model to interpret. This is inherently an interactive process, a process that an interactive program like ILOG greatly facilitates. The exploration required for interpretation of log-linear results likewise is more efficient when approached interactively, and again this is facilitated by the procedures ILOG provides (Examine the Selected Model, Examine as Two-Way Tables). Finally, ILOG lets you import tables that were exported by spread sheet or statistical package programs as tab-delimited files, manipulate and modify contingency tables with considerable flexibility (Modify This Table), export initial or modified tables as tab-delimited files that can be imported into spread sheet and statistical analysis programs, and read or paste its tab-delimited output into a standard spread sheet program for further manipulation and analysis.

ILOG4 was written in Pascal using the Embarcadero® Delphi® XE2 compiler and uses an Iterative Proportional fitting (IPF) algorithm to estimate expected frequencies. The program is available for download at no cost from http://www2.gsu.edu/~psyrab/ilog.

References

Bakeman, R., & Brownlee, J. R. (1982). Social rules governing object conflicts in toddlers and preschoolers. In K. H. Rubin & H. S. Ross (Eds.), *Peer relationships and social skills in childhood* (pp. 99–111). New York: Springer-Verlag.

Bakeman, R., & Gottman, J. M. (1997). *Observing interaction: An introduction to sequential analysis* (2nd ed.). Cambridge: Cambridge University Press.

Bakeman, R., & Quera, V. (2011). Sequential analysis and observational methods for the behavioral sciences. Cambridge: Cambridge University Press.

Bakeman, R., & Robinson, B. F. (1994). *Understanding log-linear analysis with ILOG: An interactive approach.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Bakeman, R., & Robinson, B. F. (2005). *Understanding statistics in the behavioral sciences.* Mahwah, NJ: Lawrence Erlbaum Associates.

Bishop, Y. M. M., Fienberg, S. R., & Holland, P. W. (1975). *Discrete multivariate analysis: Theory and practice.* Cambridge, MA: MIT Press.

Fienberg, S. E. (1980). *The analysis of cross-classified categorical data* (2nd ed.). Cambridge, MA: MIT Press.

Kennedy, J. J. (1992). *Analyzing qualitative data: Log-linear analysis for behavioral research* (2nd ed.). New York: Praeger.

Knoke, D., and Burke, P. J. (1980). *Log-linear models*. Newbury Park, CA: Sage.

Wickens, T.D. (1989). *Multiway contingency tables analysis for the social sciences.* Hillsdale, NJ: Lawrence Erlbaum Associates.