

A SAS[®] Macro to Calculate Exact Confidence Intervals for the Difference of Two Proportions

Paul R. Coe, Dominican University, River Forest, IL

Abstract

When comparing two proportions, it is often desirable to have more information than simply the p-value from a hypothesis test. The SAS[®] procedure FREQ provides both an approximate and an exact p-value, as well as an approximate confidence interval for the ratio of the two proportions. Calculation of the difference of two proportions and any confidence interval for this difference, however, must be done in a DATA step. I have written a SAS[®] macro to calculate exact confidence intervals for the difference of two proportions based on an algorithm previously only implemented in FORTRAN (Coe and Tamhane 1993). I have also written a SAS[®] macro to calculate exact confidence intervals for a single proportion. As well as being more accurate, exact confidence intervals have the added advantage over approximate confidence intervals of being well-defined even if one or both of the observed proportions is close (or equal) to 0 or 1. The algorithm used constructs confidence intervals which are, in general, less over-conservative (and consequently narrower) than those calculated by the commercially available software package StatXact[®].

Introduction

A common problem arising

especially in biostatistical applications is how to compare two “success” probabilities, p_1 and p_2 , based on the observed values of two independent binomial random variables, $X_1 \sim B(n_1, p_1)$ and $X_2 \sim B(n_2, p_2)$. If a hypothesis test of $H_0: p_1=p_2$ vs. a one or two-sided alternative is all that is required, approximate and exact methods are readily available. The SAS[®] procedure FREQ, for example, provides a Pearson’s X^2 statistic (and related p-value) and a p-value for Fisher’s exact test, among others. However, it is often desirable to have more information than simply the p-value from a hypothesis test.

The common estimators used for comparing p_1 and p_2 are: Berkson’s simple difference $\Delta = p_1 - p_2$, the relative risk $\rho = p_1/p_2$, and the odds ratio $\psi = p_1(1 - p_2)/p_2(1 - p_1)$. Standard approximate methods exist for constructing confidence intervals (C.I.’s) for the difference Δ which, although not directly implemented in SAS[®], can easily be written in a DATA step. Approximate methods also exist for calculating C.I.’s for the relative risk ρ , one of which is used in the SAS[®] procedure FREQ. Exact C.I.’s can be constructed for ψ using a conditional argument based on Fisher’s exact test (Cornfield 1956). This has not been implemented in SAS[®], but could be

done in a DATA step without too much difficulty. Methods for constructing exact C.I.'s for Δ and ρ also exist, but are much more computationally intensive.

I have written a SAS[®] macro to calculate C.I.'s for Δ using one of these methods (Coe and Tamhane 1993). In this paper I will briefly describe the method used to calculate the exact confidence intervals for Δ , compare it with other methods for calculating exact confidence intervals for Δ , and then describe the SAS macro that I have written to implement the method. Related macros for computing exact confidence intervals for a single proportion p and the relative risk ρ will also be described.

Methodology

The method used for constructing the exact C.I.'s for Δ (Coe and Tamhane 1993, hence forward known as CT) can be thought of as an extension of Sterne's (1954) method (as modified by Blyth and Still 1983) for constructing $(1 - \alpha)$ -level exact C.I.'s for a single binomial "success" probability p . Sterne's method constructs short C.I.'s for p by first constructing, subject to certain constraints, smallest possible acceptance regions having probability content at least $1 - \alpha$ for given values of p , and then inverting these acceptance regions to obtain the desired C.I.'s. The values of p are usually chosen as a uniform partition of the parameter space $[0, 1]$, with the fineness of the partition determining the accuracy of the resulting C.I.'s. (A uniform partition using 100 values of p starting with $p = 0.005$ in steps of 0.01 would yield two

decimal place accuracy.)

The extension to constructing C.I.'s for the difference of two proportions $\Delta = p_1 - p_2$ is complicated because of two reasons. Firstly, the sample space is two-dimensional, and hence no simple linear order exists among the sample outcomes. Secondly, there are two parameters, namely Δ , the parameter of interest, and p_1 , a nuisance parameter. Therefore, when finding the smallest possible acceptance region for a given value of Δ in the Sterne-spirit (with probability content at least $1 - \alpha$), it is necessary to consider the infimum of the probability content over the range of possible values of p_1 . Nonetheless, as in the single p problem, the method yields suitably short C.I.'s for the difference Δ . More details on the algorithm can be found below in the section on its implementation using SAS[®].

Other methods for constructing exact C.I.'s for Δ have been developed by Thomas and Gart (1977), Santner and Snell (1980), Santner and Yamagami (1993), and Cytel Software Corporation (1995). After pointing out the unconservative nature of the Thomas and Gart method, Santner and Snell proposed three new methods for constructing C.I.'s for Δ . Their first method (called the conditional method or CM) involves obtaining C.I.'s for Δ from the conditional C.I.'s for ψ derived by Cornfield (1956). Their second method (called the tail method or TM) is an extension of Clopper and Pearson's (1934) tail intervals for a single p . Their third method (called the partition method or PM) is an extension of the Sterne method for a single p . Of

these three, only TM is a viable method; it is computationally feasible and gives reasonable C.I.'s for Δ , but it is somewhat over-conservative for the same reason that the Clopper and Pearson method is over-conservative for the single p problem (see Blyth and Still 1983). CM is computationally easiest of the three, but the resulting C.I.'s are far too wide; e.g., CM yields C.I.'s for Δ that always include 0 regardless of the data or the confidence level. Thus the null hypothesis $H_0: p_1 = p_2$ will always be accepted if one uses these C.I.'s for testing purposes. PM does not appear to be computationally feasible nor programmable because of its complexity; the only example Santner and Snell give for this method is for the case $n_1 = n_2 = 2$, which they solve by hand.

Santner and Yamagami (1993) have given a computationally feasible extension of the Sterne method (as modified by Crow 1956) for which they have written a FORTRAN program. Their method (hence forward called SY) and the CT method both improve upon TM. Intervals constructed using the SY method are shorter than CT intervals for "extreme" values of Δ (close to ± 1) but are longer for "middle" values of Δ (closer to 0) for the same reason (as pointed out by Blyth and Still) that the Crow intervals for p are shorter when p is close to 0 or 1 and are longer when p is closer to 1/2. This has the consequence that the maximum as well as the average interval lengths are increased for the SY intervals (as they are for the Crow intervals). Certain regularity properties which are guaranteed in the C.I.'s constructed

using the CT method also may not hold for SY method C.I.'s based on Crow (see Coe and Tamhane 1993 and Blyth and Still 1983). However, the SY method is computationally faster than the CT method. The statistical software package StatXact-3 for Windows[®] contains a new algorithm for constructing exact C.I.'s for Δ which is computationally faster than either the CT or SY method. However, this method is based on Santner and Snell's TM, modified using an idea suggested in Burger and Boos (1994). This modification allows for a faster algorithm computationally, but still produces C.I.'s which are over-conservative (and, thus, unnecessarily wide). The SAS[®] macro that I have developed is based on the CT method.

Implementation

The SAS[®] macro that I have written is called CIDELTA and has five parameters. The first two parameters, N1 and N2, are the sample sizes n_1 and n_2 of the two groups to be compared. The third parameter, CONFLVL, is the desired confidence level of the intervals written as a decimal (i.e., 0.95 for 95% C.I.'s). The fourth parameter, DECPLAC, is the number of decimal places to use when calculating and reporting the C.I.'s. The fifth and final parameter, DATAOUT, is the name to be used for the output data set from the macro. The format of the output data set is described in the next paragraph. Thus, a call to the SAS[®] macro to construct 95% C.I.'s, calculated to 3 decimal places, for a trial with 10 items (patients, subjects, experimental units, etc.) in one group and 25 in the other,

with the results stored in the SAS® data set CI_Out, would look like:

```
CIDELTA(10, 25, 0.95, 3, CI_Out); .
```

You may notice that the macro does not require the observed number of “successes” for the two groups, x_1 and x_2 , as inputs. This is because the macro calculates C.I.’s for all possible outcomes of a trial with sample sizes N_1 and N_2 . The output data set will have $(N_1 + 1) * (N_2 + 1)$ observations, one for each possible outcome of the trial. (x_1 can take on values between 0 and N_1 , and x_2 can take on values between 0 and N_2 .) Thus, in the above example the data set CI_Out would have 286 observations. Each observation will have four variables: S1, S2, LB, and UB. S1 and S2 are the observed number of successes for groups 1 and 2, respectively, and LB and UB are the lower and upper bounds, respectively, of the $100 * CONFLVL$ % confidence interval for this outcome.

It may seem wasteful to calculate C.I.’s for all possible outcomes of the experiment when typically only one is necessary, but this is a natural byproduct of the CT algorithm. To obtain a $(1-\alpha)$ -level exact C.I. for Δ , the CT algorithm first constructs $(1-\alpha)$ -level acceptance regions for different values of Δ (a uniform partition of the parameter space $[-1, 1]$ of Δ , with the fineness of the partition based on the desired number of decimal places of the resulting C.I.’s). Once these acceptance regions are constructed, it is almost as easy to invert them to obtain a C.I. for all possible outcomes as it is for a single outcome. (For the theoretical basis of

this inversion, see Lehmann 1986, p. 90. For a more detailed description of the CT algorithm, see Coe and Tamhane 1993.) There probably is a way to curtail the process of constructing acceptance regions for values of Δ once enough of them have been constructed to give a C.I. for a specified outcome, but this curtailment has not been considered in the current macro.

Numerical Examples

As a numerical example, the SAS® code:

```
%INCLUDE CIDELTA;
%CIDELTA(2, 2, 0.95, 3, CI_Out);
PROC PRINT DATA=CI_Out;
```

would produce the following output (assuming that the SAS® macro CIDELTA was stored in an appropriate library):

OBS	S1	S2	LB	UB
1	0	0	-0.776	0.776
2	0	1	-0.975	0.502
3	0	2	-1.000	0.054
4	1	0	-0.502	0.975
5	1	1	-0.804	0.804
6	1	2	-0.975	0.502
7	2	0	-0.054	1.000
8	2	1	-0.502	0.975
9	2	2	-0.776	0.776

StatXact-3 for Windows® is a commercially available package that can produce exact C.I.’s for Δ . However, as I mentioned earlier, it uses an algorithm which produces C.I.’s that are, in general, over-conservative and, thus, unnecessarily wide. In the User Manual for this package, an exact C.I. for Δ is presented for an experiment with $n_1 = n_2 = 10$ which has 9 and 5 successes,

respectively, for the two groups. That 95% C.I. is (-0.08596, 0.7623). The 95% C.I. produced by my macro for the same outcome is (-0.007, 0.674). This is almost 20% narrower than the StatXact[®] C.I., with the same coverage probability.

Discussion

In a recent article in **Biometrical Journal**, Lee, Serachitopol, and Brown (1997) develop two new algorithms for constructing C.I.'s for Δ . They then compare C.I.'s generated using their new methods to those created using each of seven other algorithms (some exact and some using Normal approximation theory). The CT method using my SAS[®] macro is one of these seven algorithms. While acknowledging that the macro fails for large sample sizes, Lee, et. al. conclude (p. 406), "The Coe and Tamhane interval ... is the best method with guaranteed coverage rate when sample size is small."

In the future I hope to modify the current macro to accommodate larger sample sizes, particularly in the case where the observed number of "successes" in each of the two groups is either very small (near zero) or very large (near n_1 and n_2 , respectively). In these cases Normal approximation theory breaks down and exact C.I.'s are particularly desirable. I also hope to write a SAS[®] macro to construct exact C.I.'s for the relative risk ρ based on the algorithm in Coe and Tamhane (1993). I have written a SAS[®] macro to construct exact C.I.'s for a single proportion p based on the method of Blyth and Still (1983). It appears to work equally well for large and small

sample sizes. While this macro and the one for Δ are too large to include in these Proceedings, they are available from the author upon request.

References

- Blyth, C. and Still, H. A. (1983). "Binomial confidence interval," *Journal of the American Statistical Association*, **78**, 108-116.
- Clopper, C. J. and Pearson, E. S. (1934). "The use of confidence or fiducial limits illustrated in the case of the binomial," *Biometrika*, **47**, 381-391.
- Coe, P. R. and Tamhane, A. C. (1993). "Small sample confidence intervals for the difference, ratio, and odds ratio of two success probabilities," *Communications in Statistics (Simulation and Computation)*, **22**, 925-938.
- Cornfield, J. (1956). "A statistical problem arising from retrospective studies," *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability*, **4**, 135-148.
- Crow, E. L. (1956). "Confidence intervals for a proportion," *Biometrika*, **43**, 423-435.
- Cytel Software Corporation (1995). *StatXact-3 for Windows User Manual*, Cambridge.
- Lee, J. J., Serachitopol, D. M., and Brown, B. W. (1997). "Likelihood-Weighted Confidence Intervals for the Difference of Two

Binomial Proportions,” *Biometrical Journal*, **39**, 387-407.

Lehmann, E. L. (1986). *Testing Statistical Hypotheses*, Second Edition, New York: John Wiley.

Santner T. J. and Snell, M. K. (1980). “Small sample confidence intervals for $p_1 - p_2$ and p_1/p_2 in 2×2 contingency tables,” *Journal of the American Statistical Association*, **75**, 386-394.

Santner T. J. and Yamagami, S. (1993). “Invariant small sample confidence intervals for the difference of two success probabilities,” *Communications in Statistics (Simulation and Computation)*, **22**, 33-59.

Sterne, T. E. (1954). “Some remarks on confidence or fiducial limits,” *Biometrika*, **41**, 275-278.

Thomas, D. G. and Gart, J. J. (1977). “A table of exact confidence limits for differences and ratios of two proportions and their odds ratios,” *Journal of the American Statistical Association*, **72**, 73-76.

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Paul R. Coe
Department of Mathematics and
Computer Science
Dominican University
7900 W. Division St.
River Forest, IL 60305
coepaul@email.dom.edu