Simultaneous Confidence Intervals by Iteratively Adjusted Alpha for Relative Effects in the One-way Layout

Martin J. Wolfsegger
Department of Biostatistics, Baxter AG, Vienna, Austria
Thomas Jaki
Department of Statistics, University of South Carolina, Columbia

November 7, 2007

This article is published in Statistics and Computing, 16(1):1523, 2005.
DOI: 10.1007/s11222-006-5197-1. The original publication is available at
www.springerlink.com

Abstract

A bootstrap based method to construct $1-\alpha$ simultaneous confidence intervals for relative effects in the one-way layout is presented. This procedure takes the stochastic correlation between the test statistics into account and results in narrower simultaneous confidence intervals than the application of the Bonferroni correction. Instead of using the bootstrap distribution of a maximum statistic, the coverage of the confidence intervals for the individual comparisons are adjusted iteratively until the overall confidence level is reached. Empirical coverage and power estimates of the introduced procedure for many-to-one comparisons are presented and compared with asymptotic procedures based on the multivariate normal distribution.

Keywords: Bootstrap; Nonparametric, Simultaneous Confidence Intervals

1 Introduction

Consider a control treatment labeled k and test treatments labeled $1, 2, \ldots, k-1$ where $k \geq 3$. Let $\{x_{ij}\ (1 \leq j \leq n_i)\}$ be a random sample of size n_i from treatment i $(1 \leq i \leq k)$. We assume that the x_{ij} 's come from a continuous distribution F_i $(1 \leq i \leq k)$. Relative effects can be used (e.g. Munzel and Hothorn, 2001) to estimate effects between the control and test treatments

$$\widehat{p}_{ik} = \frac{1}{n_i}\left(\overline{R}_k - \frac{n_k+1}{2}\right) \qquad (1 \leq i \leq k-1) \tag{1}$$

where $\overline{R}_k$ denotes the mean rank of the sample $\boldsymbol{x}_k$ of the pooled and ranked samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$.

Munzel and Hothorn (2001) presented an asymptotic approach for relative effects for two types of hypothesis. Their procedures use the multivariate normal distribution as the asymptotic distribution of a maximum statistic where the exact critical values can be derived using the algorithm of Genz (1992). The straightforward bootstrap version can markedly loose power in the case of skewed and contaminated distributions as observed in their simulation study. Here, a different approach is presented to construct simultaneous confidence intervals for relative effects for comparisons of $k-1$ treatments against a control.

2 Multivariate Distribution of Relative Effects

Using complete rerandomization to estimate a multivariate distribution as proposed by Miller (1981) is not appropriate for single step procedures as pointed out by Petrondas and Gabriel (1983). We use the following bootstrap algorithm to estimate the multivariate distribution of the relative effects.

Algorithm 1

1. Let $\boldsymbol{x}_i^*$ be a bootstrap sample from $\boldsymbol{x}_i \; \forall \; i = 1, \ldots, k$.
2. Calculate and store $\widehat{p}_{ik}^* \; \forall \; i = 1, \ldots, k-1$.
3. Repeat B times.

The whole scheme results in B points in a $k-1$ dimensional surface representing the joint distribution of $\widehat{\boldsymbol{p}}^* = \left(\widehat{p}_{1k}^*, \widehat{p}_{2k}^*, \ldots, \widehat{p}_{(k-1)k}^*\right)$. The $k-1$ dimensional sampling cumulative distribution function is estimated by

$$\widehat{H}\left(\boldsymbol{t}\right) = \frac{1}{B} \sum_{b=1}^{B} \left\{ \begin{array}{ll} 1 & \text{if } \widehat{p}_{1k}^{*(b)} \leq t_1, \ldots, \widehat{p}_{ik}^{*(b)} \leq t_i, \ldots, \widehat{p}_{(k-1)k}^{*(b)} \leq t_{k-1} \\ 0 & \text{otherwise} \end{array} \right\} \tag{2}$$

where the superscript (b) refers to the b$^{\text{th}}$ bootstrap replication. The estimate of the cumulative distribution function $\widehat{H}_{ik}$ for the individual $\widehat{p}_{ik}^*$ $(1 \leq i \leq k-1)$ can be derived from $\widehat{H}(\boldsymbol{t})$ to be

$$\widehat{H}_{ik}\left(t_i\right) = \frac{1}{B} \sum_{b=1}^{B} \left\{ \begin{array}{ll} 1 & \text{if } \widehat{p}_{1k}^{*(b)} < \infty, \ldots, \widehat{p}_{ik}^{*(b)} \leq t_i, \ldots, \widehat{p}_{(k-1)k}^{*(b)} < \infty \\ 0 & \text{otherwise} \end{array} \right\}. \tag{3}$$

The main idea of this method is to select a $k-1$ dimensional subspace with probability $1-\alpha$ from $\widehat{H}(\boldsymbol{t})$ by solving the equation

$$P\left(t_i^{\left(\frac{\alpha}{2}\right)} \leq \widehat{p}_{ik}^* \leq t_i^{\left(1-\frac{\alpha}{2}\right)} \; \forall \; i = 1, \ldots, k-1\right) = 1-\alpha \tag{4}$$

for $\boldsymbol{t}^{\left(\frac{\alpha}{2}\right)} = \left(t_1^{\left(\frac{\alpha}{2}\right)}, \ldots, t_{k-1}^{\left(\frac{\alpha}{2}\right)}\right)$ and $\boldsymbol{t}^{\left(1-\frac{\alpha}{2}\right)} = \left(t_1^{\left(1-\frac{\alpha}{2}\right)}, \ldots, t_{k-1}^{\left(1-\frac{\alpha}{2}\right)}\right)$. By adding up all 2^{k-1} corners, the equation above can be denoted as

$$\widehat{G}\left(\boldsymbol{t}^{\left(\frac{\alpha}{2}\right)}, \boldsymbol{t}^{\left(1-\frac{\alpha}{2}\right)}\right) = \sum_{e_i = t_i^{\left(\frac{\alpha}{2}\right)} \text{ or } t_i^{\left(1-\frac{\alpha}{2}\right)}} (-1)^{z(e_1,\ldots,e_{k-1})} \widehat{H}\left(e_1, \ldots, e_{k-1}\right) = 1 - \alpha \tag{5}$$

where $z\left(e_1, \ldots, e_{k-1}\right)$ represents the number of e_i that equal $t_i^{\left(1-\frac{\alpha}{2}\right)}$. Solving this non-linear equation for $2\left(k-1\right)$ parameter of interest yields at least $k-1$ solutions.

3 Numerical Root Finding

The balanced tail probability criteria and the balanced coverage probability as mentioned by Tu and Zhou (2000) are used to find a unique solution for equation 4. This leads to a system of non-linear equations

$$P\left(t_i^{\left(\frac{\alpha}{2}\right)} \leq \widehat{p}_{ik}^{*} \leq t_i^{\left(1-\frac{\alpha}{2}\right)} \ \forall \ i = 1, \ldots, k-1\right) = 1 - \alpha \tag{6}$$

$$P\left(t_i^{\left(\frac{\alpha}{2}\right)} \leq \widehat{p}_{ik}^{*}\right) = P\left(t_i^{\left(1-\frac{\alpha}{2}\right)} \geq \widehat{p}_{ik}^{*}\right) = P\left(t_j^{\left(\frac{\alpha}{2}\right)} \leq \widehat{p}_{jk}^{*}\right) = P\left(t_j^{\left(1-\frac{\alpha}{2}\right)} \geq \widehat{p}_{jk}^{*}\right)$$

where $1 \leq i < j \leq k-1$.

Solving this system of non-linear equations for $\boldsymbol{t}^{\left(\frac{\alpha}{2}\right)}$ and $\boldsymbol{t}^{\left(1-\frac{\alpha}{2}\right)}$ results in $k-1$ lower and upper bounds of the $k-1$ simultaneous bootstrap confidence intervals.

For numerical root finding of this system of non-linear equations, the well known bisection method for numerical root finding in single variable non-linear equations can be applied. The bisection method uses two initial guesses α_e and α_s which represent the type I errors for the $k-1$ simultaneous confidence intervals. Let $\alpha_s = \alpha$ and $\alpha_e = \frac{\alpha}{k-1}$, the Bonferroni corrected alpha level, which is a conservative upper bound. The Bonferroni inequality is discussed in standard text books, e.g. Hochberg and Tamhane (1987) and Hsu (1996). The $k-1$ α levels for the individual simultaneous confidence intervals are adjusted iteratively until the specified overall $1-\alpha$ level is reached.

The following algorithm shows how to get the $k-1$ dimensional upper and lower bounds for the simultaneous bootstrap confidence intervals at level $1-\alpha$.

Algorithm 2

1. Let $\alpha_s = \alpha$, $\alpha_e = \alpha/\left(k-1\right)$ and $\alpha_m = \frac{1}{2}\left(\alpha_s + \alpha_e\right)$.

2. Get confidence limits of the $k-1$ individual inferences using α_s, α_e and α_m as per-comparison error rates. This can be done by calculation of lower $\left(\frac{\alpha_s}{2}, \frac{\alpha_e}{2}, \frac{\alpha_m}{2}\right)$ and upper $\left(1-\frac{\alpha_s}{2}, 1-\frac{\alpha_e}{2}, 1-\frac{\alpha_m}{2}\right)$ quantiles for all $k-1$ relative effects $\widehat{p}_{ik}$ using corresponding the one-dimensional marginal bootstrap distributions $\widehat{H}_{ik}$ $(1 \leq i \leq k-1)$.

   - $t_i^{\left(\frac{\alpha_s}{2}\right)} = \widehat{H}_{ik}^{-1}\left(\frac{\alpha_s}{2}\right)$ and $t_i^{\left(1-\frac{\alpha_s}{2}\right)} = \widehat{H}_{ik}^{-1}\left(1-\frac{\alpha_s}{2}\right)$.
   - $t_i^{\left(\frac{\alpha_e}{2}\right)} = \widehat{H}_{ik}^{-1}\left(\frac{\alpha_e}{2}\right)$ and $t_i^{\left(1-\frac{\alpha_e}{2}\right)} = \widehat{H}_{ik}^{-1}\left(1-\frac{\alpha_e}{2}\right)$.
   - $t_i^{\left(\frac{\alpha_m}{2}\right)} = \widehat{H}_{ik}^{-1}\left(\frac{\alpha_m}{2}\right)$ and $t_i^{\left(1-\frac{\alpha_m}{2}\right)} = \widehat{H}_{ik}^{-1}\left(1-\frac{\alpha_m}{2}\right)$.

3. Calculate the experimental coverage of the $k-1$ comparisons under α_s, α_e and α_m used as per-comparison error rates for the individual inferences.

   - $P_s = \widehat{G}\left(\boldsymbol{t}^{\left(\frac{\alpha_s}{2}\right)}, \boldsymbol{t}^{\left(1-\frac{\alpha_s}{2}\right)}\right)$.
   - $P_e = \widehat{G}\left(\boldsymbol{t}^{\left(\frac{\alpha_e}{2}\right)}, \boldsymbol{t}^{\left(1-\frac{\alpha_e}{2}\right)}\right)$.
   - $P_m = \widehat{G}\left(\boldsymbol{t}^{\left(\frac{\alpha_m}{2}\right)}, \boldsymbol{t}^{\left(1-\frac{\alpha_m}{2}\right)}\right)$.

4. If $P_s P_m < 0$, let $\alpha_e = \alpha_m$. If $P_e P_m < 0$, let $\alpha_s = \alpha_m$.

5. Let $\alpha_m = \frac{1}{2}(\alpha_s + \alpha_e)$.

6. Repeat 2 to 5 until α_m lies within a chosen tolerance.

The simultaneous confidence intervals for the relative effects p_{ik} at level $1-\alpha$ equal

$$p_{ik} \in \left[t_i^{\left(\frac{\alpha_m}{2}\right)}; t_i^{\left(1-\frac{\alpha_m}{2}\right)}\right] \qquad (1 \leq i \leq k-1). \tag{7}$$

4 Simulations

Simulations were performed to study the behavior of the introduced method of iterative simultaneously adjusted alpha (ISAA) in the many-to-one design. The Behrens-Fisher type procedure with Satterthwaite t-approximation and the Steel type procedures (Munzel and Hothorn, 2001) were used for comparison as implemented in the R package npmc Version 1.0 (Helms and Munzel, 2001). With the nominal experimental error rate α level being 0.05 (i.e. experimental coverage equals 0.95) the case of $k-1=3$ is used for simplicity.

Coverage probabilities and power estimations are reported for the general unbalanced design for different sample sizes. Normal distributions, log-normal distributions, uniform distributions and contaminated normal distributions with 10% one-sided or two-sided outliers were used. The term contam-one refers to a N(0,1) distribution with 10% outliers taken from a N(3,1) distribution. The

term contam-two refers to a N(0,1) distribution with 10% outliers taken from a N(-3,1) distribution and 10% outliers taken from a N(3,1) distribution.

For each parameter setting with preselected sample sizes, 10000 simulation runs were carried out. All-pairs power estimates in the case $k - 1 = 3$ for one selected expected value profile $(0, 0, \delta, 0)$ are also reported. Within each simulation four pseudo random samples from pre-specified distributions were generated. For the introduced method, 5000 bootstrap replications were used.

To ensure a fair comparison between the methods, coverage and power estimations were calculated on basis of the same simulation runs. All simulations were performed in R.

4.1 Experimentalwise Coverage and All-Pairs Power

In the balanced design with variance homogeneity the coverage probabilities of the ISAA procedure are similar to the coverage probabilities of the Behrens-Fisher type procedure which is more liberal than the Steel type procedure.

The Steel type procedure becomes markedly liberal and conservative in the balanced design with variance heterogeneity and in the general unbalanced design. The coverage probabilities of the Behrens-Fisher type procedure remains nearly constant and the ISAA procedure turns out to be more liberal than the Behrens-Fisher type procedure.

In the balanced design with variance homogeneity the ISAA procedure provides higher power than the two asymptotic procedures in case of contaminated normal distributions. For normal, uniform and log-normal distributions, the highest power was observed with the Behrens-Fisher type procedure where the ISAA procedure is still more powerful than the Steel type procedure.

In the balanced design with variance heterogeneity and in the general unbalanced design the ISAA procedure is superior to the Behrens-Fisher type procedure in terms of power. In designs with lower variability in the control group than in the treatment groups, the Steel type procedure has the highest power among the three procedures considered.

4.2 Per Comparison Coverage for Individual Inferences

Standard deviations were used to measure the balance of coverage in individual pairs while the experimental error is controlled. Both asymptotic procedures are superior in terms of balance of coverage of individual inferences than the ISAA procedure using standard normal (0.0007 with the Steel type procedure, 0.008 with the Behrens-Fisher type procedure and 0.0012 with the ISAA procedure) and log-normal distributed data (0.0007 with the Steel type procedure and the Behrens-Fisher type and 0.0012 with the ISAA procedure). In case of uniform distributed data, the ISAA procedure turned out to be better in terms balance of coverage in individual pairs than both asymptotic procedures (0.0014 with the Steel type procedure, 0.0011 with the Behrens-Fisher type procedure and 0.0009 with the ISAA procedure).

Table 1: Empirical Experimentalwise Coverage and All-Pairs Power Using a Nominal Experimental Error Rate of 0.05

Parameter	Distribution		Sample size		ISAA procedure	Behrens-Fisher type procedure	Steel type procedure
			T	C			
Experimental-wise coverage	Normal						
	Std. dev.						
	T	C					
	1	1	25	25	0.9436	0.9424	0.9512
	1	3	25	25	0.9476	0.9522	0.9564
	3	1	25	25	0.9211	0.9410	0.9154
	1	1	25	50	0.9412	0.9444	0.9520
	1	3	25	50	0.9503	0.9509	0.9817
	3	1	25	50	0.9203	0.9407	0.8560
	Uniform		25	25	0.9445	0.9438	0.9532
	Log-normal		25	25	0.9423	0.9430	0.9507
	contam.one		25	25	0.9432	0.9433	0.9518
	contam.two		25	25	0.9444	0.9447	0.9525
All-pairs power	Normal						
	Std. dev.						
	T	C					
	1	1	25	25	0.8614	0.8676	0.8465
	1	3	25	25	0.2394	0.2263	0.2232
	3	1	25	25	0.2394	0.1991	0.2548
	1	1	25	50	0.9419	0.9407	0.9354
	1	3	25	50	0.4016	0.3998	0.2456
	3	1	25	50	0.2526	0.2074	0.3676
	Uniform		25	25	0.9143	0.9234	0.9057
	Log-normal		25	25	0.8648	0.8679	0.8487
	contam.one		25	25	0.6950	0.6949	0.6681
	contam.two		25	25	0.4967	0.4932	0.4724

T...Treatment; C....Control

Table 2: Emperical Per-Comparison Coverage for Individual Inferences Using a Nominal Experimental Error Rate of 0.05 and a Sample Size of 25 per Group

Distribution	Comparison	Coverage	ISAA procedure	Behrens-Fisher type procedure	Steel type procedure
Log-normal	T1 vs. C	Individual	0.9773	0.9779	0.9811
		Lower tail	0.4889	0.4890	0.4909
		Upper tail	0.4884	0.4889	0.4902
	T2 vs. C	Individual	0.9796	0.9793	0.9825
		Lower tail	0.4903	0.4904	0.4923
		Upper tail	0.4893	0.4889	0.4902
	T3 vs. C	Individual	0.9785	0.9786	0.9822
		Lower tail	0.4874	0.4876	0.4892
		Upper tail	0.4911	0.4910	0.4930
Standard normal	T1 vs. C	Individual	0.9810	0.9804	0.9837
		Lower tail	0.4896	0.4894	0.4912
		Upper tail	0.4914	0.4910	0.4925
	T2 vs. C	Individual	0.9787	0.9793	0.9825
		Lower tail	0.4894	0.4895	0.4914
		Upper tail	0.4893	0.4898	0.4911
	T3 vs. C	Individual	0.9799	0.9789	0.9824
		Lower tail	0.4904	0.4905	0.4917
		Upper tail	0.4895	0.4884	0.4907
Uniform	T1 vs. C	Individual	0.9777	0.9777	0.9808
		Lower tail	0.4893	0.4891	0.4907
		Upper tail	0.4884	0.4886	0.4901
	T2 vs. C	Individual	0.9793	0.9797	0.9828
		Lower tail	0.4898	0.4904	0.4918
		Upper tail	0.4895	0.4893	0.4910
	T3 vs. C	Individual	0.9793	0.9796	0.9834
		Lower tail	0.4892	0.4897	0.4913
		Upper tail	0.4901	0.4899	0.4921

T...Treatment; C...Control

5 Example

We use the data from Watson et al. (1987) discussed in Edwards and Berry (1987) as example to illustrate the presented approach. The full data set can be found in the appendix.

Dr. Watson studied the effects of different perfusates on the permeability of capillary walls in cats. A measure of this called the capillary filtration coefficient (CFC), reflects the rate at which liquid is taken up by the tissue via the capillaries. The four treatments (the perfusates) considered are composed of ingredients A, B, and I in the following way

1. A
2. A + B
3. A + I
4. A + B + I.

We used the treatment with the single ingredient A as control and 100,000 bootstrap replications for the ISAA method. The results were summarized in the following table

Table 3: Summary of Relative Effects of Different Perfusates

Comparison	Relative effect	95% Simultaneous CI's for relative effects
A + B versus A	0.800	0.638 to 0.929
A + I versus A	0.692	0.501 to 0.863
A + B + I versus A	0.924	0.823 to 0.991

CI ... confidence interval

The probability that the capillary filtration coefficients of ingredient A were tendentiously larger than those of the combination of ingredients A+B was 0.800 (95% CI: 0.638 to 0.929). Contrariwise, the probability that the capillary filtration coefficients of ingredients A+B were tendentiously larger than those of ingredient A was 1-0.800=0.200. The increased probability of tendentiously larger capillary filtration coefficients with ingredient A than with ingredients A+B was statistically significant at the 5% level, because the 95% confidence interval did not contain the value 0.5. With a probability of 0.692 (95% CI: 0.501 to 0.863), the CFC's were tendentiously larger with ingredient A than those of the combination of ingredients A+I. This effect was statistically significant at the 5% level, because the corresponding 95% confidence interval did not contain the value 0.5. With a probability of 0.924 (95% CI: 0.823 to 0.991), the CFC's were tendentiously larger with ingredient A than those of the combination of ingredients A+B+I. This effect was also statistically significant at the 5% level, because the corresponding 95% confidence interval did not contain the value 0.5.

6 Discussion

Summarizing, in case of contaminated normal distributions the ISAA procedure is favorable if a moderate liberality can be tolerated. In addition, we did not observe a big loss in power of our bootstrap procedure in case of skewed and contaminated distribution as observed in the simulations of Munzel and Hothorn (2001) using the straightforward bootstrap version of the maximum statistic.

The extension of this results for all-pairwise comparisons is still to be studied. The asymptotic properties of the ISAA procedure presented and the generalization of these properties for the class of U-statistics are also subject to further research.

Acknowledgement: The authors are grateful to Werner Engl, Ph. D. of the Department of Biostatistics, Baxter AG, for his criticism of this note. We also would like to thank the Statistics and Computing for publishing this article and giving helpful comments during the process of writing it.

References

Edwards D. and Berry J. J. 1987. The efficiency of Simulation Based Multiple Comparisons. Biometrics 43:913-928.

Genz A. 1992. Numerical computation of the multivariate normal probabilities. Journal of Computational and Graphical Statistics, 1:141-150.

Helms J. and Munzel U. 2001. R package npmc. http://www.cran.r-project.org, last visited at 2004-03-01, Version 1.

Hochberg Y. and Tamhane A. C. 1987. Multiple Comparison Procedures. J. Wiley and Sons, New York.

Hsu J. C. 1996. Multiple Comparison Procedures: Theory and Methods. Chapman and Hall, London.

Miller R. G. 1981. Simultaneous Statistical Inference. Springer, New York.

Munzel U. and Hothorn L. A. 2001. A Unified Approach to Simultaneous Rank Test Procedures in the Unbalanced One-way Layout. Biometrical Journal, 43(5):553-569.

Petrondas D. A. and Gabriel K. R. 1983. Multiple comparisons by randomization tests. Journal of the American Statistical Association, 78:949-957.

R Development Core Team. 2004. R: A language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Austria.

Tu W. and Zhou X.-H. 2000. Pairwise multiple comparison of the means of skewed data. Journal of Statistical Planning and Inference, 88(1):59-74.

Watson, P. D., Wolf, M. B., and Beck-Montgomery I. S. 1987. Blood and isoproterenol reduce capillary permeability in cat hindlimbs. The American Journal of Physiology, 252:H47-H53.

Appendix

Function in R for many-to-one-comaprisons of relative effects in the one-way layout using the ISAA method.

```
##############################################################################
# Program name: releff.R
# Last modification Date: 2005-07-01
# Author: Martin J. Wolfsegger & Thomas Jaki
# Program Version: 0.1
# R Version: >=2.0.0
#
# Notes: function to calculate simultaneous confidence intervals for
#        relative effects for many-to-one comparisons in the one-way
#        layout by resampling with iteratively adjusted alpha
#
# Input:
#
# x ... a numeric vector of responses
# g ... a vector encoded as factor containing the class-levels;
#       the lowest factor level will be used as control for
#       many-to-one comparisons
# nsample ... number of bootstrap replications; default=1E4
# alpha ...  nominal experimental error rate; default=0.05
# tol ... absolute error tolerance for numerical root finding;
#         default=1E-6
#
# Output:
#
# list containing of
# estimate ... data frame of relative effects
# conf.int ... data frame of simultaneous confidence intervals for
#              relative effects
#
# Version history: Inital release
##############################################################################

# start of function
releff <- function(x, g, nsample=1E4, alpha=0.05, tol=1E-6){

   # function to select lower and upper bounds by ISAA method
   "subspace" <- function (G, alpha, tol) {

      # multivariate cumulative distribution function
      "CDFMult" <- function(G, border) {
         m <- ncol(G)
```

```
   sim <- nrow(G)
   for (i in 1:m) {G <- subset(G, G[, i] <= border[i])}
   return(nrow(G)/sim)
}

# function to identify 2^m corners of m dimensional interval
"corners" <- function(lb, ub) {
   m <- length(lb)
   limit <- matrix(nrow = 2^m, ncol = m)
   index <- 2^m
   for (i in 1:m) {
      value <- lb[i]
      index <- index/2
      count <- 1
      for (j in 1:(2^m)) {
         if (count == index + 1) {
            ifelse(value == lb[i],
                   value <- ub[i],
                   value <- lb[i])
            count <- 1
         }
         limit[j, i] <- value
         count <- count + 1
      }
   }
   return(limit)
}

# function to identify algebraic sign for adding up 2^m corners
"sign" <- function(limit, lb) {return((-1)^sum(lb == limit))}

# function to calculate probability of m dimensional interval
"interval" <- function(G, lb, ub) {
   m <- ncol(G)
   limit <- corners(lb = lb, ub = ub)
   prob <- 0
   for (i in 1:(2^m)) {
      prob <- prob + sign(limit = limit[i, ], lb = lb) *
              CDFMult(G = G, border = limit[i, ])
   }
   return(prob)
}

# define objects
G <- as.data.frame(G)
m <- ncol(G)
```

```
   alpha.s <- alpha/2
   alpha.e <- alpha/(2 * m)
   alpha.m <- (alpha.s + alpha.e)/2
   alpha.h <- alpha.m + 1
   lb <- list(e = array(1:m), s = array(1:m), m = array(1:m))
   ub <- list(e = array(1:m), s = array(1:m), m = array(1:m))

   # iterative root finding
   while (abs(alpha.h - alpha.m) > tol) {
      alpha.h <- alpha.m
      for (i in 1:m) {
         lb$e[i] <- as.real(quantile(G[, i], alpha.e))
         lb$s[i] <- as.real(quantile(G[, i], alpha.s))
         lb$m[i] <- as.real(quantile(G[, i], alpha.m))
         ub$e[i] <- as.real(quantile(G[, i], 1 - alpha.e))
         ub$s[i] <- as.real(quantile(G[, i], 1 - alpha.s))
         ub$m[i] <- as.real(quantile(G[, i], 1 - alpha.m))
      }
      P.e <- interval(G = G, lb = lb$e, ub = ub$e) - (1 - alpha)
      P.s <- interval(G = G, lb = lb$s, ub = ub$s) - (1 - alpha)
      P.m <- interval(G = G, lb = lb$m, ub = ub$m) - (1 - alpha)
      if (P.s * P.m < 0) {alpha.e <- alpha.m}
      if (P.e * P.m < 0) {alpha.s <- alpha.m}
      alpha.m <- (alpha.s + alpha.e)/2
   }
   return(list(lb = lb$m, ub = ub$m))
}

# function to calculate relative effects
"effects" <- function(x, n){
   m <- length(x)-n
   x <- rank(x)
   return(1/m*(mean(x[(m+1):(n+m)])-1/2*(n+1)))
}

# define ojects
k <- nlevels(g)
varnames <- levels(g)
rnames <- paste(varnames[2:k], "-", varnames[1], sep = "")
ssizes <- as.vector(tapply(x, g, length))
G <- matrix(nrow=nsample, ncol=k-1)

# calculate relative effects
observed <- tapply(x, g, sample, replace=FALSE)
observed <- lapply(observed[2:k], append, observed[[1]])
estimate <- as.matrix(sapply(observed, effects, ssizes[1]))
```

```
   # generate joint bootstrap distribution of relative effects
   for (i in 1:nsample){
      bootsamp <- tapply(x, g, sample, replace=TRUE)
      bootsamp <- lapply(bootsamp[2:k], append, bootsamp[[1]])
      G[i,] <- sapply(bootsamp, effects, ssizes[1])
   }

   # get bounds
   bounds <- subspace(G, alpha=alpha, tol=tol)

   # define output objects
   conf.int <- data.frame(lower=as.matrix(bounds$lb),
                          upper=as.matrix(bounds$ub))
   estimate <- data.frame(estimate)
   rownames(conf.int) <- rnames
   rownames(estimate) <- rnames
   return(list(estimate=estimate, conf.int=conf.int))
}
# end of function

## example from Watson et al. (1987)
cfc <- c(0.0156, 0.0118, 0.0130, 0.0082, 0.0209, 0.0222, 0.0158, 0.0119,
         0.0126, 0.0200, 0.0195, 0.0185, 0.0123, 0.0162, 0.0144, 0.0100,
         0.0143, 0.0277, 0.0116, 0.0342, 0.0200, 0.0219, 0.0195, 0.0178,
         0.0181, 0.0177, 0.0088, 0.0089, 0.0094, 0.0100, 0.0104, 0.0105,
         0.0106, 0.0106, 0.0107, 0.0110, 0.0112, 0.0113, 0.0118, 0.0119,
         0.0130, 0.0132, 0.0140, 0.0140, 0.0141, 0.0145, 0.0146, 0.0148,
         0.0150, 0.0191, 0.0100, 0.0102, 0.0109, 0.0115, 0.0118, 0.0120,
         0.0124, 0.0124, 0.0129, 0.0131, 0.0132, 0.0135, 0.0140, 0.0144,
         0.0151, 0.0154, 0.0155, 0.0162, 0.0197, 0.0211, 0.0061, 0.0063,
         0.0070, 0.0073, 0.0075, 0.0078, 0.0081, 0.0081, 0.0086, 0.0086,
         0.0086, 0.0087, 0.0088, 0.0091, 0.0092, 0.0093, 0.0099, 0.0100,
         0.0102, 0.0102, 0.0106, 0.0108, 0.0108, 0.0109, 0.0109, 0.0109,
         0.0111, 0.0115, 0.0116, 0.0125, 0.0135, 0.0140, 0.0148)
trt <- as.factor(c(rep(1,26), rep(2,24), rep(3,20), rep(4,33)))

# set seed for bootstrap resampling
set.seed(12345)

# function call and output
res <- releff(x=cfc, g=trt, nsample=1E5)
print(round(res$estimate, 3))
print(round(res$conf.int, 3))
```