

# PROC SQL for Exact Testing Trend in Proportions

Jonghyeon Kim, The EMMES Corporation, Rockville, MD

Neal Oden, The EMMES Corporation, Rockville, MD

Sungyoung Auh, National Institute of Neurological Disorders and Stroke, Bethesda, MD

## ABSTRACT

PROC SQL is SAS' implementation of the ANSI Standard Query Language (SQL), an ANSI standard database programming/query language. Syntax is simple and its utility is broad in scope. As another capability, we find that PROC SQL is a very useful tool to perform exact test for monotonic trend in mean responses of discrete random variables. In this paper, we show utilization of PROC SQL to construct a null configuration set and to compute p-value for conditional exact likelihood ratio test for trend in proportions.

## INTRODUCTION

Suppose that a clinical trial investigates the dose-response relationship of drug with  $K$  different levels of an investigational drug. Denote by  $x_k$  the number of successes in group  $k$  with  $n_k$  subjects and assume that  $x_k \sim B(n_k, P_k)$  a binomial distribution of success rate  $P_k, k = 1, \dots, K$ . Suppose that a trial tests the following two hypotheses:  $H_0 : P_1 = \dots = P_K$  versus  $H_1 : P \leq \dots \leq P_K$  with  $P_1 < P_K$ .

## COCHRAN ARMITAGE TEST STATISTIC

Despite of the unfavorable sensitivity to the score selection, the one-sided Cochran-Armitage (CA) test statistic is commonly used in practice to test linear trend in proportions. The CA test statistic is constructed for testing the slope in the linear regression of rates  $\hat{P}_k = x_k / n_k$  to the pre-specified dose score level  $d_k$  (Agresti, 1990). The test statistic is the most powerful in testing linear trend in proportions. The CA test statistic is given by

$$T_{CA}(\mathbf{x}) = \sum_{k=1}^K x_k (d_k - \bar{d}) / \sqrt{\hat{P}_0(1 - \hat{P}_0) \sum_{k=1}^K n_k (d_k - \bar{d})^2},$$

$$\text{where } \hat{P}_0 = x_+ / n_+, x_+ = \sum_{k=1}^K x_k, n_+ = \sum_{k=1}^K n_k \text{ and } \bar{d} = \sum_{k=1}^K n_k d_k / n_+$$

The level of significance is computed from asymptotic normal distribution in large samples and estimated by re-sampling method in small samples.

## LIKELIHOOD RATIO TEST STATISTIC FOR THE ORDER-RESTRICTED ALTERNATIVE

As an alternative method, likelihood ratio (LR) test statistic for the order-restricted alternative hypothesis has received a great deal of favorable attention. The test statistic does not require the choice of score and it has higher power than

the one-sided CA test statistic. More specifically, the LR test statistic is given by

$$T_{LR}(\mathbf{x}) = 2 \sum_{k=1}^K \{x_k \log(\hat{P}_k^* / \hat{P}_0) + (n_k - x_k) \log((1 - \hat{P}_k^*) / (1 - \hat{P}_0))\},$$

where  $\hat{P}_k^*$ 's are isotonic version of maximum likelihood estimates  $\hat{P}_k$ 's satisfying the order constraint  $\hat{P}_1^* \leq \dots \leq \hat{P}_K^*$  and computed by a pool adjacent violator algorithm. The level of significance is computed from chi-bar distribution (a mixture of chi-square distribution) in large samples and estimated by re-sampling method in small samples. We refer to Robertson, et al. (1988) for details.

In this paper, we utilize PROC SQL to construct exact tests for trend in proportions: (1) an exact likelihood ratio test for testing isotonic trends in proportions as well as (2) exact test for linear trend in proportions. Note that (1) PROC FREQ has option "EXACT TREND;" for exact test for linear trend in proportions, but (2) it does not allow users to input their own scores, but instead to choose one of 4 options: MODRIDIT, RANK, RIDIT, and TABLE. PROC MULTTEST procedure can be used for general score but only approximate p-values can be obtained.

### EXACT TEST FOR TREND IN PROPORTIONS

Exact p-values of the following tests are computed by (1) constructing the complete enumeration of a configuration

set  $C$  that contains  $(y_1, \dots, y_K)$  with  $\sum_{k=1}^K y_k = \sum_{k=1}^K x_k = x_+$  and (2) computing the null probability of each element

in  $C$ . Note that under the null  $H_0 : P_1 = \dots = P_K = P$ ,  $\Pr\{X_1 = y_1, \dots, X_K = y_K\} = \prod_{k=1}^K \binom{n_k}{y_k} / \binom{n_+}{x_+}$ .

We use a WHERE clause in PROC SQL to list all the elements in  $C$  and then compute the corresponding probabilities. Here is an illustration of constructing a configuration set  $C$  using an animal carcinogen study data (See Table 1). Suppose that SAS datasets "temp<sub>k</sub>", ( $k=1, 2, 3, 4$ ) have 11 records of 2 column variables  $n$  and  $x$ , where column  $n$  has value of 10 and column  $x$  has values of 0, 1, ..., 10. Dataset "Out" for the configuration set  $C$  is created by

```
PROC SQL;
CREATE TABLE out AS
SELECT g1.x AS x1, g2.x AS x2, g3.x AS x3, g4.x AS x4,
       g1.n AS n1, g2.n AS n2, g3.n AS n3, g4.n AS n4,
FROM temp1 AS g1, temp2 AS g2, temp3 AS g3, temp4 AS g4
WHERE (g1.x+g2.x+g3.x+g4.x)=5;
```

SAS table "Out" has 56 records of  $x_1+x_2+x_3+x_4=5$ .

|   | x1 | x2 | x3 | x4 | n1 | n2 | n3 | n4 |
|---|----|----|----|----|----|----|----|----|
| 1 | 0  | 0  | 0  | 5  | 10 | 10 | 10 | 10 |
| 2 | 0  | 0  | 1  | 4  | 10 | 10 | 10 | 10 |
| 3 | 0  | 0  | 2  | 3  | 10 | 10 | 10 | 10 |
| 4 | 0  | 0  | 3  | 2  | 10 | 10 | 10 | 10 |
| 5 | 0  | 0  | 4  | 1  | 10 | 10 | 10 | 10 |
| 6 | 0  | 0  | 5  | 0  | 10 | 10 | 10 | 10 |
| 7 | 0  | 1  | 0  | 4  | 10 | 10 | 10 | 10 |

**COCHRAN ARMITAGE TEST STATISTIC**

Permutation test for linear trend in proportions compares  $T_{CA}(\mathbf{y})$  constructed permutated datasets with observed

value of  $T_{CA}(\mathbf{x})$  constructed from the observed dataset. Since  $\sum_{k=1}^K y_k = \sum_{k=1}^K x_k = x_+$  is given, the comparison is

equivalent to comparison of  $\sum_{k=1}^K y_k d_k$  with  $\sum_{k=1}^K x_k d_k$ . P-value for testing linear trend in proportions is given by

$$\sum_{\mathbf{y} \in C_1} \Pr\{X_1 = y_1, \dots, X_K = y_K\} = \sum_{\mathbf{y} \in C_1} \prod_{k=1}^K \binom{n_k}{y_k} / \binom{n_+}{x_+}, \text{ where } C_1 = \left\{ \mathbf{y} : \sum_{k=1}^K y_k = x_+; \sum_{k=1}^K y_k d_k > \sum_{k=1}^K x_k d_k \right\}.$$

**LIKELIHOOD RATIO TEST STATISTIC FOR THE ORDER-RESTRICTED ALTERNATIVE**

Permutation test for monotonic trend in proportions compares  $T_{LR}(\mathbf{y})$  constructed permutated datasets with

observed value of  $T_{LR}(\mathbf{x})$  constructed from the observed dataset. P-value for testing monotonic trend in proportions is given by

$$\sum_{\mathbf{y} \in C_1} \Pr\{X_1 = y_1, \dots, X_K = y_K\} = \sum_{\mathbf{y} \in C_1} \prod_{k=1}^K \binom{n_k}{y_k} / \binom{n_+}{x_+}, \text{ where } C_1 = \left\{ \mathbf{y} : \sum_{k=1}^K y_k = x_+; T_{LR}(\mathbf{y}) > T_{LR}(\mathbf{x}) \right\}.$$

**EXAMPLE**

We now illustrate the use of our macro with an animal carcinogen study data. SAS codes and the following example are given in Appendix. Programs are available from authors upon request.

The animal carcinogen study data was cited in Corcoran and Mehta (2002) to show the inaccuracy of large sample based trend test. Forty mice were divided into four equal groups. Each group was treated with a different dose of an animal carcinogen as a result of which some mice developed a tumor. The data are displayed in Table 1.

Table 1: Dose-Response Data for Animal Carcinogenicity Study

| Response Status | Dose $d_k$<br>assigned to all mice in group $k$ |           |           |            | Total |
|-----------------|-------------------------------------------------|-----------|-----------|------------|-------|
|                 | $d_1 = 0$                                       | $d_2 = 1$ | $d_3 = 5$ | $d_4 = 50$ |       |
| Tumor           | 1                                               | 0         | 1         | 3          | 5     |
| No Tumor        | 9                                               | 10        | 9         | 7          | 35    |
| Total           | 10                                              | 10        | 10        | 10         | 40    |

Using our SAS macro %*isotrendexact*(indat=animal); for 1-sided Cochran Armitage linear trend test, we obtain conditional exact p-value= 0.054638 and asymptotic p-value=0.025749 with the given dose scores

$(d_1, d_2, d_3, d_4) = (0, 1, 5, 50)$  and conditional exact p-value= 0.105017 and asymptotic p-value=0.067240

with the given dose scores  $(d_1, d_2, d_3, d_4) = (0, 1, 2, 3)$ . This example clearly illustrates us how sensitive significance level of CA linear trend test can be to the pre-specified dose scores. On the other hand, for the likelihood ratio test for ordered alternative hypothesis, we obtain conditional exact p-value=0.10859 and asymptotic p-value=0.06933.

## CONCLUSION

PROC SQL is an extremely powerful procedure. Currently, most applications of this procedure have been focused on data manipulation. We view it as another tool for learning exact test for trends in proportions. We write user-friendly SAS macro, so that SAS users can easily apply the above methods to their statistical tasks with good accuracy. Current macro can be extended to exact trend likelihood ratio test for correlated binary data (Le CT, 1988; Corcoran, et al., 2001; Kim, et al., 2007). To overcome the conservativeness of conditional exact test, we have currently developed macro for confidence-interval based unconditional exact test for testing trend in proportions (Berger and Boos, 1994; Freidlin B and Gastwirth, JL, 1999) and many-to-one proportion comparison (Dunnett, 1955; Koch and Hothorn, 1999). Finally, we are testing SAS version 9.2 whether user-written function by PROC FCMP (e.g., isotonization of estimates) can be incorporated in PROC SQL, which speeds up the computation.

## REFERENCES

- Agresti A (1990), *Categorical Data Analysis*, New York: John Wiley & Sons, Inc.
- Robertson T, Wright FT and Dykstra RL (1988), *Order Restricted Statistical Inference*, New York: John Wiley & Sons, Inc.
- Corcoran CD, and Mehta CR (2002), Exact level and power of permutation, bootstrap, and asymptotic tests of trend. *Journal of Modern Applied Statistical Methods* 1: 42-51.
- Le CT (1988), Testing for linear trends using correlated otolaryngology or ophthalmology data. *Biometrics* 44: 299-303.
- Corcoran CD, Ryan L, Senchaudhuri P, Mehta CR, Patel N, and Molenberghs G (2001), An Exact Trend Test for Correlated Binary Data. *Biometrics* 57: 941-948.
- Kim J, Oden N, and Auh S (2007), Exact Likelihood Ratio Trend Test Using Correlated Binary Data. Presented: *International Biometrics Society ENAR Meeting, Society, ENAR Meetings in Atlanta, GA, March 2007*.
- Berger RL and Boos DD (1994), P values maximized over a confidence set for the nuisance parameter. *Journal of the American Statistical Association* 89: 1012-1016.
- Freidlin B, and Gastwirth JL (1999), Unconditional Versions of Several Tests Commonly Used in the Analysis of Contingency Tables. *Biometrics* 55: 84-89.
- Dunnett CW (1955), A multiple comparison procedure for comparing several treatments with a control. *Journal of the American Statistical Association*, 50:1096-1121.
- Koch HF, and Hothorn LA (1999), Exact unconditional distributions for dichotomous data in many-to-one comparisons. *Journal of Statistical Planning Inference* 82: 83-99.

## ACKNOWLEDGMENTS

This work was supported in part by the National Eye Institute Support contract (N01-EY-7-0001) at the EMMES Corporation.

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Jonghyeon Kim  
 The EMMES Corporation  
 401 North Washington Street, Suite 700  
 Rockville, MD 20850  
 Work Phone: 301-251-1161 Ext 233  
 Fax: 301-251-1355  
 Email: jhkim@emmes.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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

## APPENDIX

```
%macro isotrendexact(indat=);
/* indat = input data that has 3 columns yes, nsize, dosescore */
/* Compute pvalue for (1) Cochran Armitage Test Statistic for Linear Trend in Porportions and
(2) Likelihood Ratio Test Statistic for Isotonic Trend in Proportions */
/* Written by Jonghyeon Kim and Neal Oden on 4/14/2006 */
/* Modified on 7/10/2008 */

data _null_;
    set &indat end=eof;
    if eof then call symput('ngroups',trim(left(_n_)));
run;

data trend;
    set &indat end=eof;
    array x{&ngroups} x1-x&ngroups;
    array n{&ngroups} n1-n&ngroups;
    array d{&ngroups} d1-d&ngroups;

    x[_n_]=yes;
    n[_n_]=nsize;
    d[_n_]=dosescore;
    retain x1-x&ngroups n1-n&ngroups d1-d&ngroups;

    array y{&ngroups} y1-y&ngroups;
    array w{&ngroups} w1-w&ngroups;
    array z{&ngroups} z1-z&ngroups;
    array g{&ngroups} g1-g&ngroups;
    array wycusum{&ngroups};
    array wcusum{&ngroups};

/* level probabilities of simple order with equal weights */
/*      array level3pr{3} level3pr1-level3pr3 (0.33333 0.50000 0.16667);*/
/*      array level4pr{4} level4pr1-level4pr4 (0.25000 0.45833 0.25000 0.04167);*/
```

```

/*      array level5pr{5} level5pr1-level5pr5 (0.20000 0.41667 0.29167 0.08333 0.00833);*/
/*      array level6pr{6} level6pr1-level6pr6 (0.16667 0.38056 0.31250 0.11806 0.02083 0.00833);*/

```

```

/* limiting level probabilities of simple order with arbitrary weights */

```

```

array level3pr{3} level3pr1-level3pr3 (0.37500 0.50000 0.12500);
array level4pr{4} level4pr1-level4pr4 (0.31250 0.47917 0.18750 0.02083);
array level5pr{5} level5pr1-level5pr5 (0.27344 0.45833 0.22396 0.04167 0.00260);
array level6pr{6} level6pr1-level6pr6 (0.24609 0.43984 0.24740 0.05990 0.00651 0.00026);

```

```

if eof then do;

```

```

    xtot=sum(of x1-x&ngroups);
    ntot=sum(of n1-n&ngroups);
    pbar=xtot/ntot;
    do i=1 to &ngroups;
        y[i]=x[i]/n[i];
    end;
    call symput('xtot',xtot);
    call symput('ntot',ntot);
    call symput("pbar",pbar);

```

```

/* Cochran Armitage Linear Trend Test Statistic */

```

```

dbar=0;
do i=1 to &ngroups;
    dbar=dbar+ d[i]*n[i];
end;
dbar=dbar/ntot;
castat=0;
ssd=0;
do i=1 to &ngroups;
    ssd=ssd+n[i]*(d[i]-dbar)**2;
    castat=castat+x[i]*d[i];
end;
castatstd=round((castat-xtot*dbar)/sqrt((pbar*(1-pbar)*ssd)),.0000001);
call symput("castat0",castat);
normalpca=1-probnorm(castatstd);

```

```

/* LRT statistic for Isotonic Trend */

```

```

isotonic=1;
do i=2 to &ngroups until(isotonic=0);
    if (y[i]<y[(i-1)]) then isotonic=0;
end;
do i=1 to &ngroups;
    g[i]=y[i];
end;
if isotonic=0 then do;

```

```

    /* PAVA */

```

```

    do i=1 to &ngroups;
        w[i]=n[i];
    end;
    wycusum[1]=w[1]*y[1];
    do i=2 to &ngroups;
        wycusum[i]=wycusum[(i-1)]+w[i]*y[i];
    end;

    wg=0;
    do i=1 to &ngroups;
        wcusum[i]=w[i];
        if(i<&ngroups) then do;
            do j=(i+1) to &ngroups;
                wcusum[j]=wcusum[(j-1)]+w[j];
            end;
        end;
        do j=i to &ngroups;

```

```

                                z[j]=(wycusum[j]-wg)/wcusum[j];
                                end;
                                g[i]=z[i];
                                if(i<&ngroups) then do;
                                    do j=(i+1) to &ngroups;
                                        if(z[j]<g[i]) then g[i]=z[j];
                                    end;
                                end;
                                wg=wg+g[i]*w[i];
                                end;

                                end;

                                lrt=0;
                                do i=1 to &ngroups;
                                    if x[i]=0 then lrt=lrt+(n[i]-x[i])*log(((1-g[i])/(1-pbar)));
                                    else if 0<x[i]<n[i] then lrt=lrt+x[i]*log((g[i]/pbar))
                                                                + (n[i]-x[i])*log(((1-g[i])/(1-pbar)));
                                    else if x[i]=n[i] then lrt=lrt+x[i]*log((g[i]/pbar));
                                end;
                                lrt=2*lrt;
                                call symput('lrt0',lrt);

                                chibarp=0;
                                do i=2 to &ngroups;
                                    chibarp=chibarp+level&ngroups.pr[i]*(1-probchi(lrt,(i-1)));
                                end;
                                call symput('chibarp',chibarp);

                                end;
                                if eof;
                                keep x1-x&ngroups n1-n&ngroups d1-d&ngroups castat castatstd normalpca
                                    y1-y&ngroups isotonic g1-g&ngroups lrt chibarp;
                                run;

                                /* Log N choose X */
                                %macro lchoose(n, x);
                                (lgamma((&n)+1) - lgamma((&x)+1) - lgamma((&n)-(&x)+1))
                                %mend lchoose;

                                /* Creation of tables to Join in PROC SQL */
                                %macro makefiles;
                                %do i=1 %to &ngroups;
                                    data temp&i;
                                        set trend;
                                        n=n&i;
                                        d=d&i;
                                        mintemp=min(n&i,&xtot);
                                        do x = 0 to mintemp;
                                            p = x/n;
                                            logbinomcoeff=%lchoose(n, x);
                                            output;
                                        end;
                                        keep n x d p logbinomcoeff;
                                    run;
                                %end;
                                %mend makefiles;

                                %makefiles;

                                /* The following macros will be used in PROC SQL */

                                /* Cartesian Join */
                                %macro readtables(ngroups);

```

```

temp1 AS g1
%do i = 2 %to &ngroups;
    , temp&i AS g&i
%end;
%mend readtables;

/* Condition on Sufficient Statistic*/
%macro sum(ngroups,x);
(
    g1.&x
    %do i = 2 %to &ngroups;
        + g&i.&x
    %end;
)
%mend sum;

/* Keep variables in each file to be joined*/
%macro keep(ngroups,x);
    g1.&x AS &x.1
    %do i = 2 %to &ngroups;
        , g&i.&x AS &x.&i
    %end;
%mend keep;

/* Numerator of Cochran Armitage Statistic*/
%macro catestat(ngroups);
(
    g1.x*g1.d
    %do i = 2 %to &ngroups;
        + g&i.x*g&i.d
    %end;
)
%mend catestat;

/* Cochran Armitage Linear Trend Test */
PROC SQL;
    SELECT sum(Prob) INTO :condexactpca FROM
    (
        SELECT %catestat(&ngroups) AS CAstat, exp(%sum(&ngroups,logbinomcoeff)-
            %Ichoose(&ntot,&xtot)) AS prob
        FROM %readtables(&ngroups)
        WHERE %sum(&ngroups,x) = &xtot
    )
    WHERE CAstat >= &castat0;
quit;

/* Isotonic trend test */
PROC SQL;
    CREATE TABLE isoperm AS
    SELECT %keep(&ngroups,x), %keep(&ngroups,n),
        exp(%sum(&ngroups,logbinomcoeff)- %Ichoose(&ntot,&xtot)) AS prob
    FROM %readtables(&ngroups)
    WHERE %sum(&ngroups,x) = &xtot;
quit;

data isoperm;
    set isoperm end=eof;
    array x{&ngroups} x1-x&ngroups;
    array n{&ngroups} n1-n&ngroups;
    array y{&ngroups} y1-y&ngroups;
    array w{&ngroups} w1-w&ngroups;
    array z{&ngroups} z1-z&ngroups;
    array g{&ngroups} g1-g&ngroups;
    array wycusum{&ngroups};
    array wcusum{&ngroups};

```

```

retain condexactplrt 0;
do i=1 to &ngroups;
    y[i]=x[i]/n[i];
end;
isotonic=1;
do i=2 to &ngroups;
    if (y[i]<y[(i-1)]) then isotonic=0;
end;

do i=1 to &ngroups;
    g[i]=y[i];
end;
if isotonic=0 then do;
    /* PAVA */
    do i=1 to &ngroups;
        w[i]=n[i];
    end;
    wycusum[1]=w[1]*y[1];
    do i=2 to &ngroups;
        wycusum[i]=wycusum[(i-1)]+w[i]*y[i];
    end;

    wg=0;
    do i=1 to &ngroups;
        wcusum[i]=w[i];
        if(i<&ngroups) then do;
            do j=(i+1) to &ngroups;
                wcusum[j]=wcusum[(j-1)]+w[j];
            end;

            end;
            do j=i to &ngroups;
                z[j]=(wycusum[j]-wg)/wcusum[j];
            end;
            g[i]=z[i];
            if(i<&ngroups) then do;
                do j=(i+1) to &ngroups;
                    if(z[j]<g[i]) then g[i]=z[j];
                end;
            end;
            wg=wg+g[i]*w[i];
        end;
    end;

    lrt=0;
    do i=1 to &ngroups;
        if x[i]=0 then lrt=lrt+(n[i]-x[i])*log(((1-g[i])/(&1-&pbar)));
        else if 0<x[i]<n[i] then lrt=lrt+x[i]*log((g[i]/&pbar))
            + (n[i]-x[i])*log(((1-g[i])/(&1-&pbar)));
        else if x[i]=n[i] then lrt=lrt+x[i]*log((g[i]/&pbar));
    end;

    lrt=lrt*2;
    if (lrt>=&lrt0) then condexactplrt=condexactplrt+prob;
    if eof then call symput('condexactplrt',condexactplrt);
    keep x1-x&ngroups n1-n&ngroups y1-y&ngroups w1-w&ngroups isotonic lrt prob;
run;

%put _user_;

data trend;
    set trend;
    condexactpca=&condexactpca;
    condexactplrt=&condexactplrt;
    label normalpca="Asym P for CA";

```

```

label chibarp="Asym P for LR";
label condexactpca="Cond Exact P for CA";
label condexactplrt="Cond Exact P for LR";
run;

proc print data=trend;
var condexactplrt chibarp normalpca condexactpca d1-d&ngroups;
run;

%mend;

data animal;
input group dosescore no yes @@;
nsize=yes+no;
datalines;
1 0 9 1
2 1 10 0
3 5 9 1
4 50 7 3
;

run;

%isotrendexact(indat=animal);

data temp;
set animal;
do i=1 to no;
y=0;
output;
end;
do i=1 to yes;
y=1;
output;
end;
run;

/* Asymptotic P of CA Test */
proc multtest data=temp notables;
class group;
test ca(y / binomial continuity=0 uppertailed);
contrast 'CA Linear Trend' 0 1 5 50;
contrast 'CA Linear Trend' 0 1 2 3;
contrast 'CA Linear Trend' 1 2 3 4;
run;

/* Conditional EXACT P of CA Test */
proc freq data=temp;
tables group*y;
exact trend;
run;

```