Paper ## P911

Using SAS® to Perform Robust *I*-Sample Analysis of Means Type Randomization Tests for Variances

Peter Wludyka, University of North Florida, Jacksonville, FL

ABSTRACT

A SAS macro for performing Analysis of Means (ANOM) type randomization tests for testing the equality of *I* variances is presented. Randomization techniques for testing statistical hypotheses can be used when parametric tests are inappropriate. Suppose that *I* independent samples have been collected. Randomization tests are based on shuffles or rearrangements of the (combined) sample. Putting each of the *I* samples "in a bowl" forms the combined sample. Drawing samples "from the bowl" forms a shuffle. Shuffles can be made with replacement (bootstrap shuffling) or without replacement (permutation shuffling). The tests that are presented offer two advantages. They are robust to non-normality and they allow the user to graphically present the results via a decision chart similar to a Shewhart control chart. The decision chart facilitates easy assessment of both statistical and practical significance. Selected results from a Monte Carlo study used to identify robust randomization tests that exhibit excellent power when compared to other robust tests will be presented.

INTRODUCTION

Often it is useful to test whether each of *I* populations have the same variance. This homogeneity of variance (HOV) hypothesis may be written

$$H_0 : \sigma_1^2 = \ldots = \sigma_I^2 \qquad (1)$$

where σ_i^2 is the variance of the i^{th} population. The alternative hypothesis is $H_A :$ not H_0. This paper is concerned with is the case where $I > 2$; that is, where three or more populations are being compared. The focus will be on one-way balanced designs, but the discussions extend to unbalanced and more complex designs.

RANDOMIZATION TESTS

There are two types of randomization tests: exact randomization tests and approximate randomization tests. Randomization tests are based on shuffles (resamplings or rearrangements) of the (combined) sample. Suppose that *I* samples of size *n* have been selected. The combined sample is formed by putting each of the *I* samples "in a bowl." Let *X* denote the combined sample. "Drawing from the bowl" forms shuffles. The shuffles can be made with replacement (called a bootstrap shuffle) or without replacement (called a permutation shuffle).

ANALYSIS OF MEANS TYPE RANDOMIZATION TESTS FOR VARIANCES

Four ANOM type randomization tests for variances have been proposed by Bernard and Wludyka (to appear). They are randomization versions of the Analysis of Means for Variances (ANOMV), the normal based test proposed by Wludyka and Nelson (1997A). These tests can be used when normality cannot be safely assumed. Each test can be performed by plotting the *I* sample variances (or standard deviations) on a decision chart or equivalently by evaluating an empirical p-value. Using the decision chart, the HOV hypothesis is rejected if any sample variance plots outside the decision lines.

Either permutation shuffles or bootstrapping shuffles can be used. The tests are RANDANOMV-D, RANDANOMV-DD, RANDANOMV-R and RANDANOMV-RD. Each test can be classified as either a difference test or a ratio test. The difference tests are RANDANOMV-D and RANDANOMV-DD, and the ratio tests are RANDANOMV-R and RANDANOMV-RD. In the DD and RD tests, deviations from the sample mean are shuffled instead of the original observations.

See Bernard (1999) for Monte Carlo results which show that randomization tests are as powerful and robust as typical robust alternatives, such as Levene's test.

In this paper only RANDANOMV-R will be considered. For a thorough treatment of the other approaches see Bernard (1999). This test can be used when there are *k* samples (populations) of size *n*. This test should not be used when the populations have markedly different means. See Bernard and Wludyka (to appear) or Bernard (1999) for alternatives. Note that if the means are known, the mean of each population can be subtracted from each value prior to applying RANDANOMV-R.

RANDANOMV-R

RANDANOMV-R, which is based on the ratio of each sample variance to the sum of the sample variances, is the randomization version of the Analysis of Means for Variances (ANOMV), presented by Wludyka and Nelson (1997A). ANOMV is a test that is suitable when the *I* populations can safely be assumed to be normal. For a SAS macro to perform this test see Wludyka (1999).

There are two equivalent methods for performing this test. The p-value method and the decision chart method. Each requires that the following steps be performed. The SAS® program in this paper uses permutation shuffling. Either that or bootstrap shuffling can be used. A Monte Carlo study has shown that, in general, permutation shuffling produces a more powerful test.

1. Calculate the mean of each sample, $\bar{x}_i$

2. Calculate $z_{ij} = x_{ij} - \bar{x}_i$

3. Calculate $S_i^2 = \sum_{j=1}^{n_i} z_{ij}^2 \Big/ (n-1)$ and $\sum_{i=1}^{I} S_i^2$

4. For the initial sample, calculate

$$AD_{\max} = \max\left(S_i^2 \Big/ \sum_{i=1}^{I} S_i^2 \right)$$ and

$$AD_{\min} = \min\left(S_i^2 \Big/ \sum_{i=1}^{I} S_i^2 \right)$$

5. Randomly shuffle the original data x_{ij} some number of times, NS

6. After each shuffle, calculate $S_i^2 = \sum_{j=1}^{n_i} z_{ij}^2 \Big/ (n-1)$ and

$\overline{S^2} = \sum_{i=1}^{I} S_i^2 \Big/ I$, where $\bar{x}_i$ and $z_{ij} = x_{ij} - \bar{x}_i$ are based on the current shuffle.

7. Calculate $AD_{\max}^{q} = \max\left(S_i^2 \Big/ \sum_{i=1}^{I} S_i^2 \right)$ and

$$AD_{\min}^{q} = \min\left(S_i^2 \Big/ \sum_{i=1}^{I} S_i^2 \right)$$

To use the p-value method:

8. If $AD_{\max}^{q} > AD_{\max}$ then $ngmx = ngmx + 1$

9. If $AD_{\min}^{q} < AD_{\min}$ then $ngmn = ngmn + 1$

10. If p-value-high = $\left((ngmx + 1) \Big/ (NS + 1) \right) < \dfrac{\alpha}{2}$

or p-value-low = $\left((ngmn + 1) \Big/ (NS + 1) \right) < \dfrac{\alpha}{2}$, then

hypothesis (1) is rejected.

Note that *ngmx* and *ngmn* are used to count the number of times that the shuffled values are more extreme those from the initial sample (the unshuffled data).

To use the decision chart method:

For level of significance α , the sample variances are plotted with decision lines

$$UDL = \left(\sum_{i=1}^{I} S_i^2 \right) \left(AD_{\max}^{\left(1-\alpha/2\right)} \right) \tag{2}$$

$$CL = \overline{S^2} \tag{3}$$

$$LDL = \left(\sum_{i=1}^{I} S_i^2 \right) \left(AD_{\min}^{\left(\alpha/2\right)} \right) \tag{4}$$

where the quantiles $AD_{\max}^{\left(1-\alpha/2\right)}$ and $AD_{\min}^{\left(\alpha/2\right)}$ are found using equations (6) - (9). The equal variance hypothesis is rejected whenever at least one sample variance plots outside the decision lines.

$AD_{\max}^{\left(1-\alpha/2\right)}$ can be found by ordering the set $\mathrm{A} = \left\{ AD_{\max}^{q} \mid q = 1, \ldots, NS \right\}$. Denote the r^{th} largest value in *A* by $AD_{\max}^{[r]}$. Let $AD_{\max}^{\left(1-\alpha/2\right)}$ be the $\left(NS - \left[\left| (NS+1)\frac{\alpha}{2} - 1 \right|\right] \right)^{th}$ largest value in set A , where $\left[\left| X \right|\right]$ is the greatest integer in X . (5)

That is

$$AD_{\max}^{\left(1-\alpha/2\right)} = AD_{\max}^{[M]} \tag{6}$$

where

$$M = NS - \left[\left| (NS+1)\frac{\alpha}{2} - 1 \right|\right] \tag{7}$$

$AD_{\min}^{\left(\alpha/2\right)}$ can be found by ordering the set $\mathrm{B} = \left\{ AD_{\min}^{q} \mid q = 1, \ldots, NS \right\}$. Denote the rth smallest value in B by $AD_{\min}^{[r]}$. Let $AD_{\min}^{\left(\alpha/2\right)}$ be the $\left(NS - \left[\left| (NS+1)\left(1 - \frac{\alpha}{2}\right) - 1 \right|\right]' \right)^{th}$ smallest value in set B , where $\left[\left| X \right|\right]'$ is the smallest integer in X . That is,

$$AD_{\min}^{\left(\alpha/2\right)} = AD_{\min}^{[M]} \tag{8}$$

where

$$M = NS - \left[\left| (NS+1)\left(1 - \frac{\alpha}{2}\right) - 1 \right|\right]' \tag{9}$$

THE %RANOMV MACRO

The %RANOMV macro can be used to perform the RADANOMV-R test for the equality of k variances. The variable *ns* is used identify the number of shuffles --- typically 1000.

Data Preparation

The input data file must be contain two variables: a classification variable that identifies each of the *k* populations and a measurement variable. The basic idea is that ns permutation shuffles of the data will be made. This is easily achieved by “reading” the data set ns times and assigning a shuffle variable (1,…,ns) and a uniform random (pseudo) variable (in the data step in the program this variable is denoted shufno) to each data point. The data set is then sorted by the random number (shufno) within shuffle, so that there are then ns random permutation samples. These samples are used to gauge how unusual the original sample is with regard to the ratio of smallest variance to the sum of the variances and the ratio of the largest variance to the sum of the variances. See the INPUT DATA comment in the source code.

%RANOMV Output

The decision table can be used to decide the hypothesis: if any of the STD0 (sample standard deviations) plot outside the UDL (upper decision line) or LDL (lower decision line) then reject at pre-specified alpha. See Tables 1 below.

Table 1: Decision Table

```
  RANDANOMV decision table for equality of 5 variances
    OBS   TOOL    UDL        CL         LDL       STD0

     1      1    5.50998   3.57459   1.56243   1.20078
     2      2    5.50998   3.57459   1.56243   1.49026
     3      3    5.50998   3.57459   1.56243   3.84371
     4      4    5.50998   3.57459   1.56243   4.13798
     5      5    5.50998   3.57459   1.56243   5.32247
```

A decision chart is also quite easy to interpret: since the standard deviation for tool 1 plots below the LDL the equal variance hypothesis is rejected.

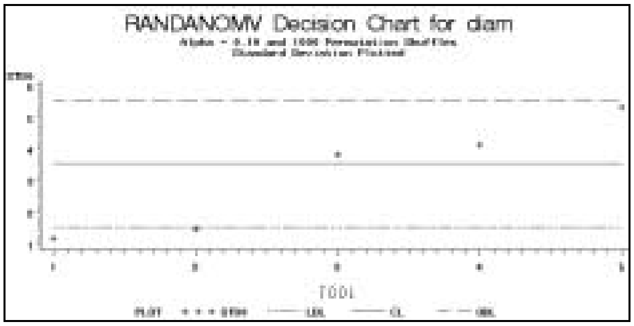

Figure 1: Decision Chart

One may also use the p-value method. The output in Table 2 illustrates the idea. The key point is that the HOV (homogeneity of variance) hypothesis is rejected whenever either p-value is less than alpha/2. Since PVALLOW (the lower p-value) is less than 0.05/2 = 0.025 the hypothesis is rejected.

Table 2: P-values

```
                 Empirical p-values should be compared to alpha/2

                       OBS   PVALLOW    PVALHI

                        1     0.017        0.449
```

%RANOMV SOURCE CODE

```
/*   RANOMV IS A TEST FOR THE EQUALITY
     OF K VARIANCES BASED ON K INDEPENDENT
     SAMPLES OF SIZE N   */
/********************************
        INPUT DATA
********************************/
data data1 ;
        pops = 7;
        samp = 10;
        ns =1000;
        do tool=1 to pops;
                do j=1 to samp;
                diam = tool*rannor(-1) + 100;
                do shuf = 1 to ns;
                        shufno = uniform(-1);
                        output ;
        end; end; end;
        drop j ;



/********************************
       DEFINE MACRO
********************************/

%macro ranomv(
   k=,     /* the number of populations being
compared */
   n=,     /* the sample size
*/
   alpha=, /* level of significance
*/
   ds=,    /* the data set containing the
observations */
   varname=,         /* the variable name for
the observations   */
   classvar=,    /* the variable name for the
populations    */
   ns =,   /* the number of shufles */
   tops=);       /*  1 = randanomv-R     */




/***************************
Print Data Set
***************************/



data basedat;
        set data1;
        if  shuf > 1 then delete;
        title 'basedat: original data set';
        proc print;  var &classvar &varname;
run;

/*********************************
   DETERMINE RANOMV CRITICAL VALUES
*********************************/
title 'initial data set';
       /* proc print data = &ds;*/
run;



data shufdat1;
      set &ds;
      proc sort; by shuf shufno;
      title 'shuffled data';
      /* proc print;*/
run;




data shufdat2;
        set shufdat1;
        drop &classvar;
        title 'shuffled data without tool';
        /*proc print;*/
run;



data origdat0;
        set data1;
        proc sort; by shuf;
        title 'original data with diam';
        /*proc print;*/
```

```
data origdat;
        set data1; drop &varname;
        proc sort; by shuf;
        title 'original data without diam';
        /* proc print; */



data shufdat3;
       merge shufdat2 origdat;
       title 'merged data';
       /* proc print;*/
run;


proc means data = shufdat3 noprint;
        by shuf &classvar; var &varname ;
output out = stats1 var = varx;
        title 'variance of each shuffled
sample';
        /*proc print data = stats1;*/
run;



proc means data=stats1 noprint;
        by shuf; var varx; output out =
stats2 sum = sumvarx;
        /*proc print data = stats2;*/
run;



data vardat;
        merge stats1 stats2; by shuf;
        varrat = varx/sumvarx;
        title 'vardat: variances of shuffled
samples along with sums and ratios';
        /* proc print data = vardat; */
run;




data admaxwk;
        set vardat;
        proc means noprint;
        by shuf; var varrat; output out =
admax max = maxrat;
        /* proc sort data = admax ; by
maxrat;*/
        title 'admax: admax distribution';
        /* proc print data = admax;*/
run;



data adminwk;
        set vardat;
        proc means noprint;
        by shuf; var varrat; output out =
admin min = minrat;
        title 'admin: admin distribution';
        /* proc sort data = admin; by minrat;
*/
        /*proc print data = admin;*/
run;



proc rank data=admin   out=rkadmin;
        var minrat;
        ranks rminrat;
        /* proc print data = rkadmin; */
run;



data critlow;
        set rkadmin;
        ranklow = &ns-floor((&ns+1)*(1-
&alpha/2)-1)-1;
        if rminrat > ranklow then delete; if
rminrat < ranklow then delete;
        lowcrit = minrat;
        title 'crit low';
        /*proc print data = critlow;*/
run;



proc rank data=admax   out=rkadmax;
        var maxrat ;
        ranks rmaxrat;
        title 'rkadmax:  ranked admax';
        /*proc print data = rkadmax; */
run;




data crithi;
        set rkadmax;
        rankhi = &ns-
floor((&ns+1)*(&alpha/2)-1);
        if rmaxrat > rankhi then delete; if
rmaxrat < rankhi then delete;
        hicrit = maxrat;
        title 'crithi: crit high';
        /*proc print;*/
run;




data critdum1;
        merge crithi critlow;
        codex =1;
        title 'critdum1: critcial values';
        /*proc print;*/
run;
data c2;set critdum1;codex=2;
data c3;set critdum1;codex=3;
data c4;set critdum1;codex=4;
data c5;set critdum1;codex=5;
data c6;set critdum1;codex=6;
data c7;set critdum1;codex=7;
data c8;set critdum1;codex=8;
data c9;set critdum1;codex=9;
data c10;set critdum1;codex=10;
data c11;set critdum1;codex=11;
data c12;set critdum1;codex=12;
data c13;set critdum1;codex=13;
data c14;set critdum1;codex=14;
data c15;set critdum1;codex=15;
data c16;set critdum1;codex=16;
data c17;set critdum1;codex=17;
data c18;set critdum1;codex=18;
data c19;set critdum1;codex=19;
data c20;set critdum1;codex=20;


data critvals;
        set critdum1 c2 c3 c4 c5 c6 c7 c8 c9
```

```
c10 c11 c12 c13 c14 c15 c16 c17 c18 c19 c20;
        if codex > &k then delete;
        title 'critvals: critical values';
        proc print data=critvals;  var
rankhi hicrit ranklow lowcrit ;


/*********************************
     DETERMINE DECISION LINES
*********************************/

data basedat2;
        set origdat0;
proc means data = basedat2 noprint;
        by shuf &classvar;
        var &varname ;
        output out = stats4a var = var0 std =
std0;
        title ' variances and standard
deviations of original data shuf replicated
';
        /*proc print data = stats4a; */
run;


proc means data=stats4a noprint;
        by shuf;
        var var0;
        output out = stats5a sum = sumvar0;
        /* proc print data = stats5a; */
run;

data vardat0;
        merge stats4a stats5a; by shuf;
        varrat0 = var0/sumvar0;
        title 'variances of original data
shuf repl samples along with sums and
ratios';
       /* proc print data = vardat0;*/
run;


data vardat0i;
        set vardat0;
        if shuf > 1 then delete;
        avgvar = sumvar0/&k;
        title 'vardat01i: variance data for
initial sample';
        /* proc print data = vardat0i;*/


data vardat1;
        merge vardat vardat0; by shuf
&classvar;
        title 'variances of original data +
shuf data repl samples along with sums and
ratios';
        /* proc print data = vardat1;*/
run;
data adminwk0;
        set vardat1;
        proc means noprint;
        by shuf; var varrat0; output out =
admin0 min = minrat0;
data admaxwk0;
        set vardat1;
        proc means noprint;
        by shuf; var varrat0; output out=
admax0 max = maxrat0;


data dldat1;
        merge critvals vardat0i;
```

```
        UDLVAR = sumvar0*hicrit;
        CLVAR = avgvar;
        LDLVAR = sumvar0*lowcrit;
        UDL = sqrt(UDLVAR);
        CL = sqrt(CLVAR);
        LDL = sqrt(LDLVAR);
        nameit = &classvar;
        title 'dldat1: data to determine
decision lines';
        /*proc print;*/
        title "RANDANOMV decision table for
equality of &k variances";
        proc print data = dldat1; var
&classvar UDL CL LDL std0 ;
run;

/*********************************
     OUTPUT ANOMV DECISION CHART
*********************************/
 proc gplot data=dldat1 ;

        plot std0*&classvar=4
                 ldl*&classvar=1
                 cl*&classvar=2
                 udl*&classvar=3
                 /overlay
                 haxis=axis2
                /* annotate=bars */
                legend;
        symbol1 c=BLUE,i=join, l=14, v=none;
        symbol2 c=BLUE, i=join, l=1, v=none;
        symbol3 c=BLUE, i=join, l=2 v=none;
        symbol4 c=BLACK, i=none, v=star;
        axis2 order=(1 to &k by 1) offset=(2)
label=(h=1.5);
        title1 "RANDANOMV Decision Chart for
&varname";
        title2 "Alpha = &alpha and &ns
Permutation Shuffles";
        title3 "Standard Deviation Plotted";
run;

/*********************
Calculate p-values
*********************/

data pvaldat;
        merge admax admin admin0 admax0;
        if maxrat > maxrat0 then sighi = 1;
else sighi =0;
        if minrat < minrat0 then siglo = 1;
else siglo = 0;
        sig = min(sighi+siglo,1);
        title 'pvalue data';
        /*proc print data=pvaldat; */


proc means data = pvaldat noprint;
        var sig; output out = pvaldat1 mean =
emppval;
        /*title1 "p-value for test for
equality of &k variances";
        titile2 "&n observations per group
and &ns shuffles";*/
        /*proc print data = pvaldat1;  var
emppval; */
proc means data = pvaldat noprint;
        var sighi; output out = pvaldat2 mean
= pvalhi;
        /*title1 "p-value for test for
equality of &k variances";
        titile2 "&n observations per group
and &ns shuffles";*/
```

```
        /*proc print data = pvaldat2;  var
pvalhi; */
proc means data = pvaldat noprint;
        var siglo; output out = pvaldat3 mean
= pvallow;
        /*title1 "p-value for test for
equality of &k variances";
        titile2 "&n observations per group
and &ns shuffles";*/
        /*proc print data = pvaldat3;  var
pvallow;*/
data pvalout;
        merge pvaldat2 pvaldat3;
        title 'Empirical p-values should
should be compared to alpha/2';
        proc print data=pvalout; var pvallow
pvalhi;
run;
%mend ranomv;
%ranomv(k=7,n=10,alpha=0.10,ds=data1,varname=
diam,classvar=tool,ns=1000,tops=1);
```

DOWNLOADING SAS® PROGRAM

The source code can be downloaded from the University of North Florida Center for Research and Consulting in Statistics web page (www.unf.edu/coas/math-stat/CRCS) as technical report #080100.

REFERENCES

A. J. Bernard and P. S. Wludyka, “Robust I-Sample Analysis of Means Type Tests for Variances”, *Journal of Statistical Computation and Simulation*, to appear.

P. S. Wludyka, “Using SAS to Perform the Analysis of Means for Variances Test”, *Conference Proceedings Southeast SAS Users Group*, 1999.

P. S. Wludyka and P. R. Nelson, “An Analysis of Means Type Test for Variances from Normal Populations,” *Technometrics*, 1997.

CONTACT INFORMATION

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

Peter Wludyka
University of North Florida
Jacksonville, Florida
Work Phone: 904-620-1048
Fax: 904-620-2818
Email: pwludyka@unf.edu
Web: www.unf.edu/coas/math-stat/~pwludyka