Paper 210-2012

MV_META: A SAS® Macro for Multivariate Meta-Analysis

Julie A. Gloudemans, University of South Florida, Tampa, FL
Corina M. Owens, University of South Florida, Tampa, FL
Jeffrey D. Kromrey, University of SouthFlorida, Tampa, FL

ABSTRACT

Meta-analysis of multiple outcomes and multiple treatments from a single study require more sophisticated models than the typical meta-analytic models that assume independence of effect sizes. Three different approaches have been suggested to accommodate dependent effect sizes: a multivariate multi-level approach (Kalaian & Raudenbush, 1996), a robust variance estimation strategy (Hedges, Tipton, & Johnson, 2010), and the traditional univariate random effects approach (Hedges & Olkin, 1985). This paper presents a SAS macro that calculates multivariate meta-analysis confidence intervals, mean effect sizes, and estimated effect size variances for each outcome variable given a sample of effect sizes and sample sizes. This paper includes a demonstration of the macro, sample inputs and output, and an examination of the accuracy and precision of the three approaches based on a simulation study.

Keywords: DEPENDENT EFFECT SIZES, RANDOM EFFECTS, CONFIDENCE INTERVALS, BASE SAS®, SAS/STAT®, SAS/IML®

INTRODUCTION

Meta-analysis provides tools for the synthesis of research, specifically the use of quantitative procedures to statistically combine the results of studies (Cooper & Hedges, 2009). Meta-analyses are typically conducted with the assumption that the effects sizes are from independent studies, and consequently are statistically independent. However, many studies do not meet the assumption of independence. Studies can use multiple treatments compared to a common control group resulting in correlations between effect sizes. Additionally studies can have multiple outcome variables of interest. The traditional approaches of meta-analysis used to statistically combine and analyze effect sizes are fixed and random effect models. Each of these approaches assumes independence of studies.

FIXED AND RANDOM EFFECTS MODELS FOR META-ANALYSIS

Models for meta-analysis may be roughly divided into those based upon fixed effects and those based upon random effects (Field, 2001; Hedges, 1994; Hedges & Vevea, 1998; Raudenbush, 1994). A fixed effects model assumes a common effect size across all studies (Shadish & Haddock, 2009). In other words, in a fixed effects model it is assumed that one true effect size exists in the population, with variability being only due to sampling error. In contrast, under a random effects model one would not assume that one population effect size exists but rather a distribution of population effect sizes exists. Therefore differences in effect sizes are based on underlying population differences and are not just due to sampling error. In addition, when no random variation exists or the variance component is zero the random effects model simplifies to the fixed effects model.

Because sample effect sizes obtained for a meta-analysis typically present different magnitudes of estimation error, weighted means and variances are used to obtain the estimates of population effect sizes and confidence bands. For fixed effects models, these weights are given by

$$v_i = \frac{1}{\sigma_i^2}$$

where $\sigma_i^2 =$ the estimation variance of the ith effect size.

In contrast, for random effects models, the weights used are given by

$$w_i = \frac{1}{\sigma_i^2 + \tau^2}$$

where $\tau^2 = \sigma_\theta^2 =$ the population variance in effect sizes.

Because τ^2 is not known, it must be estimated from the observed sample effect sizes (see Biggerstaff & Tweedie, 1997; Friedman, 2000; Hedges & Olkin, 1985).

MULTIVARIATE APPROACHES FOR META-ANALYSIS
Other approaches have been proposed to account for dependency between effect sizes. Two methods are a multivariate multi-level approach (Kalaian & Raudenbush, 1996) and robust variance estimation strategy (Hedges, Tipton, and Johnson, 2010). Kalaian and Raudenbush (1996) used the multi-level approach to estimate two levels: the within-study (or the effect sizes from each study) and the between-study (or the effect sizes across the different studies). The combined model, as stated by Kalaian and Raudenbush, is:

$$d_i = X_i W_i \gamma + X_i u_i + e_i$$

where d_i is the effect size estimates, X_i is an indicator of estimating a true effect size, W_i represents model covariates, and u_i and e_i represent errors. Variances and covariances are estimated using restricted maximum likelihood (REML). The true effect sizes are estimated using empirical Bayes.

Hedges, Tipton, and Johnson (2010) described an approach that allows for dependent effect sizes without having the within-study covariance structure by using a robust variance estimate. The model uses a vector (**T**) that represents a sample of estimates for m clusters.

$$\begin{bmatrix} T_1 \\ \vdots \\ T_m \end{bmatrix} = \begin{bmatrix} X_1 \\ \vdots \\ X_m \end{bmatrix} \beta + \begin{bmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_m \end{bmatrix}$$

Where **X** is a vector of matrices that are the value of the covariates for each effect size, $\boldsymbol{\varepsilon}$ is a vector of residuals, and $\boldsymbol{\beta}$ is a vector of unknown regression coefficients that is estimated by weighted least-squares as

$$b = \left(\sum_{j=1}^{m} X_j' W_j X_j \right)^{-1} \left(\sum_{j=1}^{m} X_j' W_j T_j \right)$$

with W_j representing a diagonal vector of weights. The standard error of b_j when m is small is determined by:

$$S_j^R = \sqrt{\frac{m v_{jj}^R}{m - p}}$$

where v_{jj}^R are the elements of the vector $\mathbf{V^R}$:

$$V^R = \left(\sum_{j=1}^{m} X_j' W_j X_j \right)^{-1} \left(\sum_{j=1}^{m} X_j' W_j e_j e_J' W_J X_j \right) \left(\sum_{j=1}^{m} X_j' W_j X_j \right)^{-1}$$

MACRO MV_META
The macro MV_META was designed to estimate mean effect sizes, confidence intervals around each mean effect size, and estimated effect size variance for each outcome variable using the three approaches described. The macro allows meta-analysts to easily apply the most recent approaches to meta-analysis with dependent effect sizes. The macro was written in SAS/IML, SAS/STAT, and BASE SAS Software. The macro can accommodate any number of studies and any number of outcome variables. In addition, missing data among the sample of effect sizes (i.e., not all studies in the sample include all of the outcome variables) is accommodated.

Inputs to the macro include the name of the SAS dataset containing the effect size information and the names of the SAS variables containing the effect sizes, the sample sizes, the outcome variable name, and the study identifier. The dataset is transferred to SAS/IML for computation of the traditional univariate random effects analysis and the robust variance analysis suggested by Hedges, Tipton, and Johnson (2010). The MIXED procedure from SAS/STAT is used for the multi-level approach suggested by Kalaian and Raudenbush (1996). The meta-analytic results include descriptive statistics of about the studies and three tables of results using a file statement in a SAS/IML. The macro MV_META is copied below.

```
%macro MV_Meta(data = _LAST_, ID = IDN, ES = d1 d2 d3 d4 d5, NN = nt1 nc1 nt2 nc2 nt3 nc3 nt4 nc4 nt5 nc5);
options  ls = 132;
title ' ';

*+-------------------------------------------------------------------------------------------------------------------+
          Macro arguments are:
          DATA = SAS dataset containing the effect size data
          ID = Study ID number
          ES = Effect sizes included in the dataset
          NN = Group (treatment and control) sample sizes for each effect size included in the data set
 +-------------------------------------------------------------------------------------------------------------------+;
Proc iml;
* +------------------------------------------------------------------------------+
          Subroutine effect -- arranges effect sizes and sample sizes for
          input into proc mixed. As well as determining the weights
          and the number of effects per observation.

          Creates a2 with study id, one effect size, the corresponding
          sample sizes, type (based on the number of effect sizes per
          observation), weights, and number of effect sizes per
          observation.
* +------------------------------------------------------------------------------+;

start effect(id_vector, es_matrix, nn_matrix, es, one_n, two_n, type, w, noeffects, studyid);
create a2 var{studyid es one_n two_n type w noeffects};
do row = 1 to nrow (ES_Matrix);
 do col = 1 to ncol(ES_Matrix);
          es = ES_Matrix[row,col];
          one_n = NN_Matrix[row,2#col-1];
          two_n = NN_Matrix[row,2#col];
          type = col;
          studyid = id_vector[row,1];
          w = 1/(((one_n + two_n)/(one_n*two_n)) + ((es**2)/(2*(one_n+two_n))));
          noeffects = ncol(ES_Matrix);
          append;
 end;
end;
 call symput('noeffect',left(char(noeffects)));
finish effect;


* +-------------------------------------------------------------------------+
Subroutine parm -- determine variances and covariances for
provided effect sizes the PARMS STATEMENT in Proc Mixed.
If one or both of the corresponding effect sizes are missing
then it is coded as 0. The file parm2 is created with
observations of covp
* +-------------------------------------------------------------------------+;

start parm (ES_Matrix, NN_Matrix, covp);
create parm2 var{covp};
do i = 1 to nrow(ES_matrix);
          do m = 1 to ncol(ES_matrix);
                    do j = 1 to ncol(ES_matrix);
                              esize1 = ES_Matrix[i,j];
                              k = 2*j - 1;
                              l = 2*j;
                              one_n1 = NN_Matrix[i,k];
                              one_n2 = NN_Matrix[i,l];
                    if m = j then do;
                              covp = ((one_n1 + one_n2)/(one_n1*one_n2)) + ((esize1**2)/(2*(one_n1+one_n2)));
                              if covp = . then covp = 0;
                              append;
```

```
                        end;
                        if m > j then do;
                                esize2 = ES_Matrix[i,m];
                                n = 2*m - 1;
                                o = 2*m;
                                two_n1 = NN_Matrix[i,n];
                                two_n2 = NN_Matrix[i,o];
                                covp = (1/(one_n2+two_n2)) + ((esize1*esize2)/(2*(one_n1+one_n2+two_n1+two_n2)));
                                if covp = . then covp = 0;
                        append;
                        end;
                        end;
                end;
 end;
finish parm;


* +----------------------------------------------------------------------------+
   Transfer data from regular SAS into IML and populate matrices
* +----------------------------------------------------------------------------+;
use &data;
read all var {&ES} into ES_Matrix;
read all var {&NN} into NN_Matrix;
read all var {&ID} into ID_Vector;

run effect (id_vector, es_matrix, nn_matrix, es, one_n, two_n, type, w, noeffects, studyid);
run parm (ES_Matrix, NN_Matrix, covp);

* determines the number of between study variances and covariances needed;
%let j = %EVAL (((&noeffect**2 - &noeffect)/2)+&noeffect);


* +------------------------------------------------------------------------+
   Initialize values of between subjects variance components
* +------------------------------------------------------------------------+;
y= {.};
create temp from y[colname={"covp"}];
        do i = 1 to &j;
         y = .5;
         append from y;
        end;
Close temp;
quit;

*Merges between and within variances and covariances;
data parm3;
 set temp parm2;

*organizes variances and covariances;
proc transpose data= parm3 out=parm4 prefix=covp;
 var covp;
data parm4;
 set parm4 (drop = _name_);
run;

*determines needed number of parmeters;
proc sql noprint;
select count(*)
into :noparms
from parm3;
quit;

*makes j the starting point for fixed parmeters;
%let j = %EVAL(&j+1);
```

4

```
ODS SELECT NONE;
/* Proc Mixed procedure*/
proc mixed method=REML scoring = 20 cl data= a2 covtest ;
class studyid type;
model es=type/ noint s cl;
random type/ subject=studyid type=un ;
repeated type/ subject=studyid
group=studyid type=un;
ods output solutionF = c5
(keep = estimate stderr lower upper);
ods output covparms = c5a (keep = covparm group estimate);
ods output convergencestatus = c6 (keep = reason status);
parms / pdata=parm4 eqcons= &j to &noparms;
weight w;
run;
ods select all;

data c5b (keep = variance);
 set c5a;
 if group = ' ' then do;
          long = length(covparm);
          if long = 7 then do;
                    first_num = substr(covparm,4,1);
                    sec_num = substr (covparm,6,1);
                    if first_num = sec_num then do;
                             Variance = estimate;
                    end;
          end;
          if long = 9 then do;
                    first_num = substr(covparm,4,2);
                    sec_num = substr (covparm,7,2);
                    if first_num = sec_num then do;
                             variance = estimate;
                    end;
          end;
   end;
   if variance < 0 then delete;
run;


data c5;
 merge c5 c5b;
run;

proc iml;
* +------------------------------------------------------------------------------------------------------------+
   Subroutine to calculate weighted mean effect sizes,
   standard errors, and confidence intervals for mean of both DVs under
   either fixed effects or random effects models.
   Inputs to the subroutine are
     d1_vec - column vector of effect sizes (d1)
     var_d1 - column vector of estimation errors for d1(FIXED EFFECTS variance)
     tau2_d1 - scalar estimate of RANDOM EFFECTS variance for d1(set to zero for fixed effects analysis)
   Outputs are
     d1_mean = weighted mean d1 value   Mean ES of d1
            d1_resum_wt = scalar, sum of the d1 weights
     d1_SE = standard error of d1
     d1upper95, d1lower95 = endpoints of 95% CI
   +------------------------------------------------------------------------------------------------------------+;
start mean_d(d1_vec,var_d1,tau2_d1,d1_mean,d1_resum_wt,d1_SE,d1upper95,d1lower95);
```

```
* calculate RANDOM EFFECTS weighted mean effect size;
  k1 = nrow(d1_vec);
  d1_mean = 0;
  d1_resum_wt = 0;
  do i = 1 to k1;
        if d1_vec[i,1] ^=. then do;
        d1_mean = d1_mean + d1_vec[i,1]/(var_d1[i,1]+tau2_d1);
        d1_resum_wt = d1_resum_wt + (var_d1[i,1]+tau2_d1)##-1;
        end;
  end;
  d1_mean = d1_mean/d1_resum_wt; *****Random Effects Mean D1 ES;
  d1_SE = SQRT(d1_resum_wt##-1);
  d1upper95 = d1_mean + 1.96#d1_SE;
  d1lower95 = d1_mean - 1.96#d1_SE;
finish;
* +-------------------------------------------------------------------------+
  Subroutine to calculate the Q test
  of homogeneity.
  Inputs to the subroutine are
    d1_vec - column vector of effect sizes (d1)
    n_vec  - matrix (k X 2) of sample sizes
          corresponding to each effect size
  Outputs are
    d1_QQ = the obtained value of Q for d1
    d1_plus = weighted mean d1 value
    d1_prob_qq1 = chi-square probability associated with QQ
    var_d1 = column vector of variances of d1 effect sizes
  +-------------------------------------------------------------------------+;
start calcq(d1_vec,n_vec,qq_d1,d1_plus,d1_prob_qq1,var_d1);

* calculate variance for each effect size;
  d1_k = nrow(d1_vec);
  var_d1=J(d1_k,1,0);
  do i = 1 to d1_k;
        if d1_vec[i,1] ^= . then do;
        var_d1[i,1] = ((n_vec[i,1]+n_vec[i,2])/(n_vec[i,1]#n_vec[i,2])) +
                ((d1_vec[i,1]##2)/(2#(n_vec[i,1]+n_vec[i,2]))));
        end;
  end;
  * calculate weighted mean effect size;
  d1_plus = 0;
  d1_sum_wt = 0;
  do i = 1 to d1_k;
        if d1_vec[i,1] ^= . then do;
        d1_plus = d1_plus + d1_vec[i,1]/var_d1[i,1];
        d1_sum_wt = d1_sum_wt + var_d1[i,1]##-1;
        end;
  end;
  d1_plus = d1_plus/d1_sum_wt;

* calculate Q for d1;
  QQ_d1 = 0;
  k_count = 0;
  do i = 1 to d1_k;
        if d1_vec[i,1] ^= . then do;
        QQ_d1 = QQ_d1 + ((d1_vec[i,1] - d1_plus)##2/var_d1[i,1]);
                k_count = k_count + 1;
        end;
  end;
  d1_prob_qq1 = 1 - PROBCHI(QQ_d1,k_count-1);
finish;
```

6

```
* +-------------------------------------------------------------------------+
   Subroutine for Hedges Robust method

   Inputs to the subroutine are
          d_matrix
          d_matrix has column 1 for study ID
          n_matrix
   Outputs are
          mean_ES
          lower_CI
          upper_CI
          tau2
  +-------------------------------------------------------------------------+;
start hedges (d_matrix, N_matrix, mean_ES, SE_ES, lower_CI, upper_CI, tau2);
 k = NROW(d_matrix);
 j = NCOL(d_matrix);
 W = J(k,j-1,0); * Matrix of fixed effects weights;
  do i = 1 to k;
          do dv = 2 to j;
                   if d_matrix[i,dv] = . then d_matrix[i,dv] = 999;
          end;
  end;
 * Create matrix of fixed effects weights;
  do i = 1 to k;
      do dv = 2 to j;
          if d_matrix[i,dv] ^= 999 then do;
          First_N = (dv - 1) + (dv - 2);
          var_d = ((N_matrix[i,First_N]+N_matrix[i,First_N + 1])/(N_matrix[i,First_N]#N_matrix[i,First_N + 1])) +
((d_matrix[i,dv]##2)/(2#(N_matrix[i,First_N]+N_matrix[i,First_N + 1])));
          W[i,dv-1] = 1/var_d;
          end;
          if d_matrix[i,dv] = 999 then W[i,dv-1] = 0;
      end;
  end;
 * Estimate WLS parameters (equation 3);
 Xj = J(j-1,j-1,0);
 do change = 1 to j-1;
          Xj[change,change] = 1;
          Xj[change,1] = 1;
 end;
  do i = 1 to k;
          Wj = diag(w[i,]);
          Tj = d_matrix[i,2:j];
          Tj = Tj`;
          if i = 1 then do;
                   denom = Xj`*Wj*Xj;
                   numer = Xj`*Wj*Tj;
          end;
          if i > 1 then do;
                   denom = denom + (Xj`*Wj*Xj);
                   numer = numer + (Xj`*Wj*Tj);
          end;
  end;
 beta = INV(denom)*numer;

 * Compute random effects variance (equation 15);
 do i = 1 to k;
          Wj = diag(w[i,]);
          Tj = d_matrix[i,2:j];
          Tj = Tj`;
          if i = 1 then do;
                   term1 = Tj`*Wj*Tj;
```

7

```
                        term2 = Tj`*Wj*Xj;
                        term3 = Xj`*Wj*Xj;
                        term4 = Xj`*Wj*Tj;
                end;
                if i > 1 then do;
                        term1 = term1 + (Tj`*Wj*Tj);
                        term2 = term2 + (Tj`*Wj*Xj);
                        term3 = term3 + (Xj`*Wj*Xj);
                        term4 = term4 + (Xj`*Wj*Tj);
                end;
        end;
        QE = term1 - (term2*INV(term3)*term4);

        Ji = J(j-1,j-1,1);
        do i = 1 to k;
                Wj = diag(w[i,]);
                Wij = 0; * scalar value: mean weight;
                n_wt = 0;
                do wt = 1 to j-1;
                        if W[i,wt] ^= 0 then do;
                                wij = wij + W[i,wt];
                                n_wt = n_wt + 1;
                        end;
                end;
                wij = wij/n_wt;
                Tj = d_matrix[i,2:j];
                Tj = Tj`;
                if i = 1 then do;
                        term1 = Xj`*Wj*Xj;
                        term2 = (0.5*Wij)*Xj`*Xj;
                        term3 = (0.5*Wij)*(Xj`*Ji*Xj - Xj`*Xj);
                        term4 = 2*Wij;
                        term5 = Wij*Wij*Xj`*Ji*Xj;
                end;
                if i > 1 then do;
                        term1 = term1 + (Xj`*Wj*Xj);
                        term2 = term2 + ((0.5*Wij)*Xj`*Xj);
                        term3 = term3 + ((0.5*Wij)*(Xj`*Ji*Xj - Xj`*Xj));
                        term4 = term4 + (2*Wij);
                        term5 = term5 + (Wij*Wij*Xj`*Ji*Xj);
                end;
        end;
        term1 = INV(term1);

        do rhoj = 5 to 5;
                rho = rhoj/10;
                numer = QE - k + TRACE(term1*term2) + rho*TRACE(term1*term3);
                denom = term4-TRACE(term1*term5);
                tau2 = numer/denom;
        end;
        if tau2 < 0 then tau2 = 0;

        * Redefine weights, including tau2 (equation 13);
        omega2 = 0; * variance component from equation 18;
        do i = 1 to k;
                do dv = 2 to j;
                        if W[i,dv-1] ^= 0 then do;
                                var_d = 1/W[i,dv-1];
                                w[i,dv-1] = 1/(var_d + tau2 + omega2);
                        end;
                end;
        end;
```

8

```
* Estimate WLS parameters with RE weights (equation 3);
do i = 1 to k;
        Wj = diag(w[i,]);
        Tj = d_matrix[i,2:j];
        Tj = Tj`;
        if i = 1 then do;
                denom = Xj`*Wj*Xj;
                numer = Xj`*Wj*Tj;
        end;
        if i > 1 then do;
                denom = denom + (Xj`*Wj*Xj);
                numer = numer + (Xj`*Wj*Tj);
        end;
end;
beta = INV(denom)*numer;

* Estimate robust standard errors (equation 6 and 7);
do i = 1 to k;
        Wj = diag(w[i,]);
        Tj = d_matrix[i,2:j];
        Tj = Tj`;
        ej = Tj - Xj*beta;
        if i = 1 then do;
                term1 = Xj`*Wj*Xj;
                term2 = Xj`*Wj*(ej*ej`)*Wj*Xj;
        end;
        if i > 1 then do;
                term1 = term1 + (Xj`*Wj*Xj);
                term2 = term2 + (Xj`*Wj*(ej*ej`)*Wj*Xj);
        end;
end;
VR = INV(term1)*term2*INV(term1);

* Compute mean effect sizes and 95% CI using t-distribution;
df = k - (j-1);
upperT = TINV(.975, df);
lowerT = TINV(.025, df);
mean_ES = J(j-1,1,0);
SE_ES = J(j-1,1,0);
lower_CI = J(j-1,1,0);
upper_CI = J(j-1,1,0);
do dv = 1 to j-1;
        if dv = 1 then do;
                mean_es[1,1] = beta[1,1];
                SR = SQRT((k#VR[1,1])/(k-(j-1))); *Robust SE;
                SE_ES[1,1] = SR;
                upper_CI[1,1] = mean_es[1,1] + upperT#SR;
                lower_CI[1,1] = mean_es[1,1] + lowerT#SR;
        end;
        if dv > 1 then do;
                mean_es[dv,1] = beta[1,1] + beta[dv,1];
                SR = SQRT((k#(VR[1,1] + VR[dv,dv] + 2#VR[1,dv]))/(k-(j-1))); *Robust SE;
                SE_ES[dv,1] = SR;
                upper_CI[dv,1] = mean_es[dv,1] + upperT#SR;
                lower_CI[dv,1] = mean_es[dv,1] + lowerT#SR;
        end;
end;
finish;
```

```
* +----------------------------------------------------------------------+
   Transfer data from regular SAS into IML and populate matrices
* +----------------------------------------------------------------------+;
use &data;
read all var {&ES} into ES_Matrix;
read all var {&NN} into NN_Matrix;
read all var {&ID} into ID_Vector;
d_matrix = ID_Vector||ES_Matrix; * Effect size matrix, augmented with study ID variable;


* +----------------------------------------------------------------------+
   Tally descriptive information (sample sizes, etc.) for output
* +----------------------------------------------------------------------+;
Total_Studies = NROW(ES_Matrix);
Total_Vars = NCOL(ES_Matrix);
Treatment_Counts = J(Total_Vars,2,0);
Control_Counts = J(Total_Vars,2,0);
Study_Counts = J(Total_Vars,1,0);
min_treatment = J(Total_Vars,1,1000000);
max_treatment = J(Total_vars,1,0);
min_control = J(Total_Vars,1,1000000);
max_control = J(Total_vars,1,0);
do i = 1 to total_studies;
        do j = 1 to total_vars;
                if ES_Matrix[i,j] ^= . then do;
                        study_counts[j,1] = study_counts[j,1] + 1;
                        treatment_counts[j,1] = treatment_counts[j,1] + NN_matrix[i,2#j-1];
                        if NN_matrix[i,2#j-1] < min_treatment[j,1] then min_treatment[j,1] = NN_matrix[i,2#j-1];
                        if NN_matrix[i,2#j-1] > max_treatment[j,1] then max_treatment[j,1] = NN_matrix[i,2#j-1];

                        control_counts[j,1] = control_counts[j,1] + NN_matrix[i,2#j];
                        if NN_matrix[i,2#j] < min_control[j,1] then min_control[j,1] = NN_matrix[i,2#j];
                        if NN_matrix[i,2#j] > max_control[j,1] then max_control[j,1] = NN_matrix[i,2#j];
                end;
        end;
end;
do j = 1 to total_vars;
        treatment_counts[j,2] = treatment_counts[j,1]/study_counts[j,1];
        control_counts[j,2] = control_counts[j,1]/study_counts[j,1];
end;
file print;
        put @1 'Estimates of Mean Effect Sizes' ///
                @1 'Total Number of Studies:' @40 total_studies 8. /
                @1 'Total Number of Variables:' @40 total_vars 8. //
                @1 'Sample Sizes in Corpus of Studies:' //
                @13 'N of      Treatment Group Observations      Control  Group  Observations' /
                @1 'Variable  Studies    Total  Min   Max   Mean      Total  Min   Max   Mean' /
                @1 '----------------------------------------------------------------------------' /;
do row = 1 to total_vars;
        var = row;
        studies = study_counts[row,1];
        t_total = treatment_counts[row,1];
        t_min = min_treatment[row,1];
        t_max = max_treatment[row,1];
        t_mean = treatment_counts[row,2];
        c_total = control_counts[row,1];
        c_min = min_control[row,1];
        c_max = max_control[row,1];
        c_mean = control_counts[row,2];
        file print;
                put @4 var 2. @13 studies 4. @24 t_total 5. @31 t_min 5. @38 t_max 5. @45 t_mean best5. @58
c_total 5. @65 c_min 5. @72 c_max 5. @79 c_mean best5.;
end;
```

```
run hedges (d_matrix,NN_matrix,Hmean_ES,H_SE,Hlower_CI,Hupper_CI,Htau2);
file print;
        put   @1 /// 'Meta-Analysis Results' /
                    @1 // 'Robust Variance Estimation Method' //
              @1 'Variable' @13 'Mean  d' @22 'Standard Error' @37'Lower Limit' @52 'Upper Limit' @67 'Variance' /;
do row = 1 to NROW(Hmean_ES);
        mean =    Hmean_ES[row,1];
        SE =      H_SE[row,1];
        Low95 =   Hlower_CI[row,1];
        Hi95 =    Hupper_CI[row,1];
        est_tau = Htau2[1,1];
        file print;
                if row = 1 then put @4 row 2. @12 mean 8.5 @25 SE 8.6 @38 Low95 8.6 @54 Hi95 8.6 @67
est_tau 8.6;
                if row > 1 then put @4 row 2. @12 mean 8.5 @25 SE 8.6 @38 Low95 8.6 @54 Hi95 8.6;
end;
*Regular RE Analysis;
do col = 1 to ncol(ES_Matrix);
        d1_vec = ES_Matrix[,col];
        n_vec = NN_Matrix[,2#col-1]||NN_Matrix[,2#col];
run calcq(d1_vec,n_vec,d1_qq,d1_plus,d1_prob_qq1,var_d1);
/* Compute estimates of Tau-squared;
    * Estimator #2:  Q based REVC Estimate*/
                k_count = 0;
                Pt1 = 0;
                Pt2 = 0;
                do i = 1 to NROW(ES_Matrix);
                        if d1_vec[i,1] ^= . then do;
                                Pt1 = Pt1 + var_d1[i,1]##-1;
                                Pt2 = Pt2 + var_d1[i,1]##-2;
                                k_count = k_count + 1;
                        end;
                end;
                CC_d1 = Pt1 - (Pt2/Pt1);
                Tau2_d1 = (d1_qq - (K_count - 1)) / cc_d1;  *****REVC;
                if tau2_d1 < 0 then tau2_d1 = 0;
*-----------------------------------------------------------
  Results for Random Effects Method
*-----------------------------------------------------------;
 * compute mean ds and CIs using estimator 2;
run mean_d(d1_vec,var_d1,tau2_d1,d1_mean,d1_resum_wt,d1_SE,d1upper95,d1lower95);
RE_results = RE_results//(d1_mean||d1_se||d1lower95||d1upper95||tau2_d1);
end; * End RE Analysis;
file print;
        put   @1 //'Univariate Random Effects Method' //
              @1 'Variable' @13 'Mean  d' @22 'Standard Error' @37'Lower Limit' @52 'Upper Limit' @67 'Variance' /;
do row = 1 to NROW(RE_results);
        mean =    RE_results[row,1];
        SE =      RE_results[row,2];
        Low95 =   RE_results[row,3];
        Hi95 =    RE_results[row,4];
        est_tau = RE_results[row,5];
        file print;
                put @4 row 2. @12 mean 8.5 @25 SE 8.6 @38 Low95 8.6 @54 Hi95 8.6 @67 est_tau 8.6;
end;
* Read ODS files from PROC MIXED;
 Use c6;
 read all var {Reason} into Converge_status;
 read all var {Status} into status;
 If status = 1 then do;
        file print;
        put @1 // 'PROC MIXED Analysis' //
```

```
           @1 'Analysis did not successfully execute: PROC MIXED did not converge with the sample' //
                       @1 'Message returned from PROC MIXED:   ' converge_status;
 end;
 if status = 0 then do;
   USE c5;
   READ ALL INTO mix;
   Nmix= NROW(mix);
   Est=mix[,1];
   se =mix[,2];
   up =mix[,4];
   lo =mix[,3];
   revar=mix[,5];
   Mixed_results = Est || se || lo || up || revar;
          file print;
            put   @1 //'PROC MIXED Analysis' //
            @1 'Variable' @13 'Mean  d' @22 'Standard Error' @37'Lower Limit' @52 'Upper Limit' @67 'Variance' /;
          do row = 1 to nmix;
                       mean =    Mixed_results[row,1];
                       SE =      Mixed_results[row,2];
                       Low95 =   Mixed_results[row,3];
                       Hi95 =    Mixed_results[row,4];
                       revar = Mixed_results[row,5];
                       file print;
                               put @4 row 2. @12 mean 8.5 @25 SE 8.6 @38 Low95 8.6 @54 Hi95 8.6 @67 revar 8.6;
          end;
   end;

quit;

%mend MV_meta;
```

INVOKING THE MACRO
The MV_META macro can be called using:

```
 %MV_Meta (data = <data set name>, ID = <study ID name>, ES = <Name of effect sizes>, NN = <Name of sample
sizes with 2 per effect size>); run;
```

Below is an example dataset that can be used with MV_META and how that example can be called for the macro.
The dataset can have multiple Cohen's D effect sizes per study and requires two sample sizes per effect size. The
dataset needs to be organized like the example below such that the first two sample sizes (n1 and n2) correspond to
the first effect size (es1) and this patter follows for each effect size.

```
data a1;
 Input studyID es1 es2 es3 n1 n2 n3 n4 n5 n6;
cards;
1       1.45   0.12  -2.12  145   129   145   129   145   129
2       .      -1.4   2.45   72   129    72   129    72   129
3       1.5    1.1    2.14   71   129    71   129    71   129
4      -1.2    1.17  -2.1   225   193   225   193   225   193
5       0.55   0.93   2.02  192   684   192   684   192   684
;
run;
```

```
%MV_Meta (data = a1, ID = studyid, ES = es1 es2 es3, NN = n1 n2 n3 n4 n5 n6);
run;
```

OUTPUT FROM MACRO MV_META
The output consists of descriptive statistics about the meta-analysis including the number of studies, number of
variables, and sample sizes in corpus of studies by variable (see Figure 1). The information provided about the
sample sizes by variable are the number of effect sizes and the total, minimum, maximum, and mean number of
subjects. The meta-analytic results are displayed as three tables. The first table is the results from the RVE method.
This table includes the mean effect sizes, the standard error for each mean effect size, the 95% confidence interval

for each mean effect size, and estimated tau. The second table includes the results from the RE method. This table displays the mean effect sizes and their standard error, 95% confidence interval and variance. The final table is the result of the mixed method including the mean effect sizes and their standard errors, 95% confidence intervals and the variance for the random effects. If the mixed method fails to converge, the output displays the message from the MIXED procedure.

```
Estimates of Mean Effect Sizes

Total Number of Studies:                       5
Total Number of Variables:                     3

Sample Sizes in Corpus of Studies:

           N of      Treatment Group Observations     Control  Group  Observations
Variable   Studies     Total   Min    Max   Mean         Total  Min    Max    Mean
----------------------------------------------------------------------------------

    1         4          633    71     225   158.3        1135   129    684   283.8
    2         5          705    71     225   141          1264   129    684   252.8
    3         5          705    71     225   141          1264   129    684   252.8

Meta-Analysis Results

Robust Variance Estimation Method

Variable    Mean  d  Standard Error Lower Limit    Upper Limit    Variance

    1        0.57379     0.863668      -3.14227      4.289852     4.368509
    2        0.38494     0.683234      -2.55478      3.324660
    3        0.47602     1.497695      -5.96804      6.920086


Univariate Random Effects Method

Variable    Mean  d  Standard Error Lower Limit    Upper Limit    Variance

    1        0.57145     0.611083      -.626272      1.769174     1.477697
    2        0.38922     0.396874      -.388653      1.167094     0.770722
    3        0.47650     1.076462     -1.63336       2.586366     5.770548


PROC MIXED Analysis

Variable    Mean  d  Standard Error Lower Limit    Upper Limit    Variance

    1        1.82961     1.357510     -1.15825       4.817472     9.141877
    2        0.38345     0.483463      -.680647      1.447541     1.168338
    3        0.47780     1.058837     -1.85269       2.808285     5.605028
```

Figure 1. Macro Output.

COMPARISON OF THE MULTIVARIATE META-ANALYSIS APPROACHES

Four different meta-analytic methods of dealing with dependent effect sizes were examined in a simulation study: a multivariate multi-level (Mixed) approach (Kalaian & Raudenbush, 1996); robust variance estimation (RVE) strategy (Hedges, Tipton, & Johnson, 2010); stochastically dependent effect (SDE) size approach (Gleser & Olkin, 2009); and traditional univariate random effects (RE) approach (Hedges & Olkin, 1985). Three of these methods are part of this macro: Mixed, RVE, and RE. Factors investigated in the simulation study included: number of studies included in the meta-analysis; population mean effect sizes; covariance between effect sizes; within study sample size; and population effect size variance (Owens, Gloudemans, & Kromery, 2011).

All approaches remained relatively unbiased, regardless of condition. All approaches tended to produce biased estimates as the population effect size increased with the worst bias values occurring when the within study sample size was the smallest. The SDE approach produced the narrowest confidence intervals and not surprisingly tended to undercover, regardless of condition. The other three approaches (RVE, RE, and Mixed) produced wider intervals and coverage close to the nominal .95 coverage rate.

Accuracy and precision increased for the RVE, RE, and Mixed approaches as the number of primary studies and within study sample size increased, and as the population effect size decreased. All approaches, with the exception of the SDE approach, produced promising results for meta-analysts with multivariate data. In addition, it is tenable to assume the SDE approach is only appropriate for use with dependent effect size data when the data truly conform to a fixed effects model.

CONCLUSIONS

The macro MV_META was created to calculate three different methods of meta-analyzing data with dependent effect sizes. The simulation study, based on the studied conditions, showed that the three models had similar results. This code has been tested with two, three, and four effect sizes along with missing values. The macro can be used with any meta-analysis that uses dependent effect sizes. This work can be expanded in many ways. For example, the macro can be used to study the effects of missing dependent effect sizes .This was not an included condition of the simulation study. Additionally, this macro only uses Cohen's D effect size. The code can be adjusted to allow for different types of effect sizes.

REFERENCES

Biggerstaff, B. J. & Tweedie, R. L. (1997). Incorporating variability in estimates of heterogeneity in the random effects model in meta-analysis. *Statistics in Medicine*, *16*, 753 – 768.

Cooper, H., & Hedges, L. V. (2009). Research synthesis as a scientific process. In Cooper, H., Hedges, L. V. & Valentine, J.C. (Eds.), *The handbook of research synthesis and meta-analysis* (pp. 3-16). New York, NY: Russell Sage Foundation.

Field, A. P. (2001). Meta-analysis of correlation coefficients: A Monte Carlo comparison of fixed- and random-effects methods. *Psychological Methods*, *6*, 161 – 180.

Friedman, L. (2000). Estimators of random effects variance components in meta-analysis. *Journal of Educational and Behavioral Statistics*, *25*, 1-12.

Hedges, L. V. (1994). Fixed effects models. In H. Cooper & L. V. Hedges (Eds.), *The handbook of research synthesis* (pp. 285-299). New York: Russell Sage Foundation.

Hedges, L.V., & Olkin, I. (1985). *Statistical methods for meta-analysis.* Orlando, FL: Academic Press.

Hedges, L. V., Tipton, E., and Johnson, M. C. (2010). Robust variance estimation in meta-regression with dependent effect size estimates. *Research Synthesis Methods.* 1(1): 39-65.

Hedges, L. V. & Vevea, J. L. (1998). Fixed- and random-effects models in meta-analysis. *Psychological Methods*, *3*, 486-504.

Gleser, L. J., & Olkin, I. (2009). Stochastically dependent effect sizes. In H. Cooper, L. V. Hedges, and J. C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis*, 2nd edition (pp. 357-376). New York: Russell Sage Foundation.

Kalaian, H. A., & Raudenbush, S. W. (1996). A multivariate linear model for meta-analysis. *Psychological Methods*, 1(3), 227-235.

Owens, C. M., Gloudemans, J. & Kromrey, J. D. (2011). *Meta-Analysis of multivariate outcomes: A Monte Carlo comparison of alternative strategies.* Paper presented at the Joint Statistical Meetings, Miami, FL.

Raudenbush, S. W. (1994). Random effects models. In H. Cooper & L. V. Hedges (Eds.), *The handbook of research synthesis* (pp. 301-321). New York: Russell Sage Foundation.

Shadish, W.R. & Haddock, D.M. (2007). Combining estimates of effect sizes. In H. Cooper, L. Hedges, & J.C. Valentine (Eds.), *The handbook of research synthesis and meta-analysis* (2nd ed., pp.557- 578). New York: Russell Sage Foundation.

CONTACT INFORMATION

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

Julie Gloudemans
1936 Bruce B Downs Blvd #343
Wesley Chapel, FL 33543
E-mail: Julie.gloudemans@gmail.com