Paper CC-20

Macros for Two-Sample Hypothesis Tests

Jinson J. Erinjeri, D.K. Shifflet and Associates Ltd., McLean, VA

ABSTRACT

Statistical Hypothesis Testing is performed to determine whether enough statistical evidence exists to conclude that a hypothesis about a parameter is supported by the data. This paper deals with macro codes for Two-Sample hypothesis testing for means and proportions which are the commonly used statistical tests across all industries. The application was mainly developed to determine whether significant differences (mean/proportion) existed between important travel variables from year to year in the raw data. In this paper, generalized macros for Two-Sample hypothesis tests are presented.

INTRODUCTION

Statistical Hypothesis testing is a quantitative procedure for making inferences about a population by detecting whether enough statistical evidence exists to conclude that a hypothesis about a parameter is supported by the data. PROC TTEST and PROC SEQDESIGN are some of the few options where one can carry out two sample tests. However, if one wants to suppress all the information these outputs generate and make it simple, say yes/no to the end user - it is best to develop a macro. The main advantage will be to perform multiple two-sample tests across subjects in one run. In this paper, the mentioned advantages were incorporated to develop macros for hypothesis testing of two-sample (independent) tests, mainly for:

1. Test of Two Means
2. Test of Two Proportions

The t-distribution and normal distribution was used for test of means and proportions respectively. It is important to note that once the macro code is deciphered, it is easy to incorporate changes based on requirements. The developed macro is best explained with the help of an example. In this case, the travel related data is used to illustrate the macros.

The snapshot of travel data is shown in Figure1 with each record showing travel related information by a respondent.

Obs	nmmonth	resp	sdays	fstate	s_wt	tmonth	ttran	spurp	ifus	tpdays	tpers	Myear4	Year4
1	2	591	1	RI	0.275	1	4	12	1	1.025	0.549	2007	2007
2	2	639	0	GA	9.564	1	5	13	1	51.142	57.38	2007	2007
3	2	1790	3	FL	1.199	1	1	2	1	4.879	1.199	2007	2007
4	2	1790	1	MI	2.048	1	1	4	1	3.805	2.048	2007	2007
5	2	5053	5	FL	0.997	1	4	8	1	6.34	0.997	2007	2007
6	2	5053	5	IN	1.323	1	4	8	1	8.354	1.323	2007	2007
7	2	5053	5	IN	1.323	1	4	8	1	8.354	1.323	2007	2007
8	2	5053	5	IN	1.323	1	4	8	1	8.354	1.323	2007	2007
9	2	5107	0	AZ	10.80	1	4	1	1	25.156	32.41	2007	2007

Figure 1. Snapshot of Travel Data Collected

TEST OF TWO MEANS

In most statistical testing, the standard deviation of population is unknown and therefore we have used t-distribution for the test of two means. For the above data in Figure1, the hypothesis is to find if there are significant differences between average length of stay (variable =sdays) at each of the destination states (variable=fstate) between years (variable=year4) in the raw data. The advantage of the macro is that we can do multiple hypotheses testing for as many subjects. In the example above we can test the difference between lengths of stay between various years for all states. The macro identifies all the possibilities and produces output as an excel file. The comments presented in the code below gives the description of the macro with partial output presented in Figure 2.

```
%macro mean2test(dataname=,v2cb=,v2cw=,v2fall=,los=);
/*Storing the categorical information of the variables (names and the number of
them)*/
      proc sql noprint;
```

```
            select distinct (&v2cb.)
            into : v2cb_nam separated by ' '
            from &dataname.;

            select count(distinct (&v2cb.))
            into : n_v2cb
            from &dataname.;

            select distinct(&v2fall.)
            into : s2test_nam separated by ' '
            from &dataname.;

            select count(distinct(&v2fall.))
            into : n_v2fall
            from &dataname.;
      quit;
/*Testing the mandatory requirements of variables for two-sample mean test*/
      data _null_;
            set &dataname. (firstobs=1 obs=1);
            v2cw_type=vtype(&v2cw.);
            if v2cw_type='C' then do;
            put "The variable &v2cw. considered for the two sample test is not
            numeric.";
            abort;
            end;
            else if &n_v2cb. in ('0','1') then do;
            put "The variable &v2cb. should have atleast two categories.
            Currently it has %left(&n_v2cb.).";
            abort;
            end;
      run;
/* splitting the data for the variable under test (variable=v2cb)*/
      data
            %do i=1 %to &n_v2cb.;
                  &v2cb.&i.
            %end;
            ;
            set &dataname.;
            %do j=1 %to &n_v2cb.;
                  if &v2cb.=%scan(&v2cb_nam.,&j.) then output &v2cb.&j.;

            %end;
      run;
/* obtaining the frequency for the variables under test(v2cb) and total sample
size by the subjects (variable=v2fall)*/
      %do k=1 %to &n_v2cb.;
      data _null_;
            set &v2cb.&k.;
            call symputx('temps',year4);
      run;
      proc means data=&v2cb.&k. nway;
            class &v2cb. &v2fall.;
            var &v2cw.;
            output out=data&k.(drop= _freq_ _type_) mean=m&temps.
            std=std&temps. n=tot&temps.;
      run;
      %end;

/*combining all the data sets obtained from the above Proc means output*/
      data combine1;
            merge
            %do l=1 %to &n_v2cb.;
```

```
            data&l
            %end;
            ;
            by &v2fall.;
      run;
/* determining all possible combinations taken two at a time for the variable
under test (variable=v2cb)*/
      data trial;
            array v v1-v2;
            array temp(&n_v2cb.) _temporary_(&v2cb_nam.);
            do i=1 to %eval(&n_v2cb.-1);
               do j=2 to &n_v2cb.;
                        v{1}=temp{i};
                        v{2}=temp{j};
                        output;
                  end;
            end;
      run;
      data trial1;
            set trial;
            if i-j>=0 then delete;
            intvars=cat(v1,v2);
            drop i j /*v1 v2*/;
      run;
/*Storing the all the possible combinations*/
      proc sql;
            select count(*)
                  into :nobs
            from trial1;
            select intvars
                  into :coms1-:coms%left(&nobs.)
            from trial1;
            select v1
                  into :a1-:a%left(&nobs.)
            from trial1;
            select v2
                  into :b1-:b%left(&nobs.)
            from trial1;
      quit;

      data final;
         set combine1;
            %do o=1 %to &nobs.;
/*calculating all the required stats to perform two sample t-test on means*/
/*los is Level of Significance*/
      sp&&coms&o.=sqrt((((tot&&a&o.-1)*std&&a&o.*std&&a&o.) + ((tot&&b&o.-
      1)*std&&b&o.*std&&b&o.))/(tot&&a&o.+ tot&&b&o. -2));
      se&&coms&o.=sp&&coms&o.*(sqrt(1/tot&&a&o.+ 1/tot&&b&o.));
      teststat&&coms&o.=(m&&a&o.-m&&b&o.)/(se&&coms&o.);
      pval&&coms&o.=2*(1-probt(ABS(teststat&&coms&o.),(tot&&a&o.+ tot&&b&o. -
      2)));
/* constructing confidence intervals*/
      LCL&&coms&o.=(m&&a&o.-m&&b&o.)-(se&&coms&o.*tinv(1-(&los./2),tot&&a&o.+
      tot&&b&o. -2));
      UCL&&coms&o.=(m&&a&o.-m&&b&o.)+(se&&coms&o.*tinv(1-(&los./2),tot&&a&o.+
      tot&&b&o. -2));

/*checking for significance with yes/no*/
      if (LCL&&coms&o.= . or UCL&&coms&o.= .) then significant&&coms&o.="mis";
      else if LCL&&coms&o.<=0 and  UCL&&coms&o.>=0 then
      significant&&coms&o.='no';
```

```
      else if ((LCL&&coms&o.>0 and  UCL&&coms&o.>0) or (LCL&&coms&o.<0 and
      UCL&&coms&o.<0)) then significant&&coms&o.='yes';
      %end;
      run;
/*Output the results to excel*/
      ods html file="test2mean.xls" rs=none style=minimal;
            proc print data=final noobs;
                  var &v2fall. significant: LCL: UCL: ;
            run;
      ods html close;

%mend mean2test;

%mean2test(dataname=rates,v2cb=year4,v2cw=sdays,v2fall=fstate,los=0.05);
```

fstate	Significant 20072008	significant 20072009	Significant 20082009	LCL 20072008	LCL 20072009	LCL 20082009	UCL 20072008	UCL 20072009	UCL 20082009
AK	yes	no	yes	3.50668	-0.62712	-4.22413	4.47505	0.96106	-3.42366
AL	no	no	no	-0.2091	-0.08616	-0.08825	0.23715	0.37415	0.34819
AR	yes	no	no	-0.50749	-0.30993	-0.05686	-0.0301	0.15478	0.43931
AZ	yes	no	no	-0.52331	-0.29885	-0.01483	-0.02873	0.20455	0.47257
CA	yes	no	yes	1.16971	-0.0725	-1.26716	1.28508	0.10244	-1.1577

Figure 2. Partial Output of Two Sample Test of Means

TEST OF TWO PROPORTIONS

The test for two proportions macro is illustrated with the same example presented in Figure1. Let's assume that the hypothesis to be tested is that the share of travelers' transportation-mode to each state with respect to last year has remained the same. To perform multiple two-sample proportions tests across many subjects, it is best to develop a macro. The macro is presented below with partial output shown in Figure 3.

```
%macro prop2test(dataname=,v2cb=,v2cw=,v2fall=,los=);
/*Storing the categorical information of the variables (names and the number of
them)*/
      proc sql noprint;
            select distinct (&v2cb.)
                  into : v2cb_nam separated by ' '
                        from &dataname.;
            select count(distinct (&v2cb.))
                  into : n_v2cb
                        from &dataname.;
            select distinct (&v2cw.)
                  into : v2cw nam separated by ' '
                        from &dataname.;
            select count(distinct (&v2cw.))
                  into : n_v2cw
                        from &dataname.;
            select distinct(&v2fall.)
                  into : s2test_nam separated by ' '
                        from &dataname.;
            select count(distinct(&v2fall.))
                  into : n_v2fall
                        from &dataname.;
      quit;
/*Testing the mandatory requirements of variables for the two-sample proportion
test*/
      data _null_;
            set &dataname. (firstobs=1 obs=1);
            if &n_v2cw. in ('0','1') then do;
            put "The variable &v2cw. should have atleast two categories.
            Currently it has %left(&n_v2cw.).";
```

```
            abort;
            end;
            else if &n_v2cb. in ('0','1') then do;
            put "The variable &v2cb. should have atleast two categories.
            Currently it has %left(&n_v2cb.).";
            abort;
            end;
      run;
/* splitting the data for the variable under test (variable=v2cb)*/
      data
            %do i=1 %to &n_v2cb.;
                  &v2cb.&i.
            %end;
            ;
            set &dataname.;
            %do j=1 %to &n_v2cb.;
                  if &v2cb.=%scan(&v2cb_nam.,&j.) then output &v2cb.&j.;

            %end;
      run;
/* obtaining the frequency for each category of variable v2cb and total sample
size by variable v2fall */
            %do k=1 %to &n_v2cb.;
            proc sort data=&v2cb.&k.;by &v2cb. &v2fall.;run;
            proc freq data=&v2cb.&k.;
                  table fstate/out=freq&k.;
                  by &v2cb.;
            run;
/* obtaining the frequency for each category of variable v2cb and sample size by
variable v2cw */
            proc freq data=&v2cb.&k.;
                  table &v2cw.*fstate/out=freqa&k.;
                  by &v2cb.;
            run;
            data freqa&k.;
                  set freqa&k.;
            run;
            %end;
/*combining and rearranging all the above data sets*/
      data combine1;
            set
            %do l=1 %to &n_v2cb.;
            freq&l
            %end;
            ;
            drop percent;
      run;
      data combine2;
            set
            %do m=1 %to &n_v2cb.;
            freqa&m
            %end;
            ;
            drop percent;
      run;
      proc sort data=combine1;by &v2fall. ;
      proc transpose data=combine1 prefix=N out=combine1t;
            by &v2fall. ;
            var count;
            id &v2cb.;
      run;
      proc sort data=combine2;by &v2fall. &v2cw.;
```

```
      proc transpose data=combine2 prefix=ct out=combine2t;
            by &v2fall. &v2cw.;
            var count;
            id &v2cb.;
      run;
      proc sort data=combine1t;by &v2fall.;
      proc sort data=combine2t;by &v2fall.;

      data inorder;
            merge combine1t combine2t;
        by &v2fall.;
      run;
/* determinig all possible combinations taken two at a time for variable v2cb*/
data trial;
      array v v1-v2;
      array temp(&n_v2cb.) _temporary_(&v2cb_nam.);
    do i=1 to %eval(&n_v2cb.-1);
         do j=2 to &n_v2cb.;
                v{1}=temp{i};
                v{2}=temp{j};
                output;
            end;
      end;
run;
data trial1;
      set trial;
      if i-j>=0 then delete;
      intvars=cat(v1,v2);
      drop i j /*v1 v2*/;
run;
proc sql;
      select count(*)
            into :nobs
    from trial1;
      select intvars
            into :coms1-:coms%left(&nobs.)
      from trial1;
      select v1
            into :a1-:a%left(&nobs.)
      from trial1;
      select v2
            into :b1-:b%left(&nobs.)
      from trial1;
quit;
/*calculating all the required stats to perform two sample z-test on
proportions,los is level of significance*/
data final;
   set inorder;
      %do o=1 %to &n_v2cb.;
      phat%scan(&v2cb_nam.,&o)=ct%scan(&v2cb_nam.,&o.)/N%scan(&v2cb_nam.,&o.);
      %end;
      %do p=1 %to &nobs.;
            diff&&coms&p.=phat&&a&p.-phat&&b&p.;
            phat&&coms&p. = (ct&&a&p. + ct&&b&p.)/(n&&a&p. + n&&b&p.);
            sigdiff&&coms&p. = SQRT(phat&&coms&p.*(1-
            phat&&coms&p.)*(1/ct&&a&p. + 1/ct&&b&p.));
            TestStat&&coms&p. = (phat&&a&p.-phat&&b&p.)/sigdiff&&coms&p.;
            pval&&coms&p.  =  2*MIN ((1-Probnorm(Teststat&&coms&p. )),
            Probnorm(Teststat&&coms&p.));

            LCL&&coms&p.= diff&&coms&p. - (sigdiff&&coms&p.*probit(1-
            (&los./2)));
```

```
            UCL&&coms&p.= diff&&coms&p. + (sigdiff&&coms&p.*probit(1-
            (&los./2)));

            if (LCL&&coms&p.= . or UCL&&coms&p.= .) then
            significant&&coms&p.="mis";
            else if LCL&&coms&p.<=0 and  UCL&&coms&p.>=0 then
            significant&&coms&p.='no';
            else if ((LCL&&coms&p.>0 and  UCL&&coms&p.>0) or (LCL&&coms&p.<0 and
            UCL&&coms&p.<0)) then significant&&coms&p.='yes';

      %end;
run;
/* output to excel*/
ods html file="test2prop.xls" rs=none style=minimal;
      proc print data=final noobs;
            var &v2fall. &v2cw. significant: LCL: UCL:;
      run;
ods html close;

%mend prop2test;

%prop2test(dataname=rates,v2cb=year4,v2cw=ttran,v2fall=fstate,los=0.05);
```

fstate	ttran	Significant 20072008	Significant 20072009	Significant 20082009	LCL 20072008	LCL 20072009	LCL 20082009	UCL 20072008	UCL 20072009	UCL 20082009
AL	1	no	no	no	-0.07093	-0.05213	-0.04242	0.05559	0.07754	0.08316
AL	2	no	mis	mis	-0.06579	.	.	0.06632	.	.
AL	3	no	no	no	-0.0683	-0.06517	-0.06211	0.06468	0.0703	0.07086
AL	4	no	yes	yes	-0.01774	0.04361	0.02979	0.04742	0.11468	0.09882
AL	5	no	yes	yes	-0.06385	-0.17805	-0.17319	0.06419	-0.03533	-0.04052
AL	1	no	no	no	-0.07093	-0.05213	-0.04242	0.05559	0.07754	0.08316

Figure 3. Partial Output of Two Sample Test of Proportions

CONCLUSION

The macros developed in this paper can be used to perform multiple-two-sample tests across many subjects thereby saving time. Also, the macros can be tailored to specific needs with minimal changes in the code. The changes can be related to rounding numbers, performing one tailed test, incorporating weights, considering equality of variances and so forth.

ACKNOWLEDGMENTS

The author would like to thank Nandini Nadkarni for her valuable input while reviewing this paper.

CONTACT INFORMATION

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

Jinson J. Erinjeri
D.K. Shifflet and Associates Ltd.
1750 Old Meadow Rd., Suite, 620
Mclean, VA 22102
Work Phone: 703-536-0924
Fax: 703-536-0580
E-mail: jerinjeri@dksa.com
Web: www.dksa.com