%POWER: A SIMPLE MACRO FOR POWER AND SAMPLE SIZE CALCULATIONS

Kristin R. Latour

Abstract

Statisticians often desire to investigate the power of statistical effect tests, or to estimate the sample size required to provide a significant effect test. This paper describes the macro POWER which provides the following power related measures:

- Power of a test
- Adjusted power and confidence limits
- Least significant number

The POWER macro accepts a PROC GLM OUTSTAT data set as input. Users can supply an unlimited number of significance level, sample size, standard deviation, and effect size values. The calculations listed above are performed for all combinations of these parameters and output to a SAS® data set. The macro handles prospective as well as retrospective power analyses.

Background

JMP® software provides power calculations for all parameter tests, effect tests, contrasts, and custom tests from a linear model. The effect test power calculations are based on O'Brien (1989). However these are currently unavailable in SAS software. The POWER macro, originally written to test the power calculations provided by JMP software, utilizes the PROC GLM OUTSTAT data set to provide the same effect test power calculations available in JMP software.

The POWER Macro

When fitting linear models, statisticians often desire to know the power of getting a significant result. Moreover, the statistician would like to assess the affect of varying sample sizes, root mean square error, significance level of the test, and possibly the effect size. The POWER macro, which appears in the appendix, allows the user to investigate power related measures in this manner.

The POWER macro user must first run PROC GLM storing the results in an OUTSTAT data set. After running the GLM, the user calls POWER specifying the name of the OUTSTAT data set, the effect for which power calculations will be calculated, and the desired calculations, significance levels, sample sizes, standard deviations, and effect size values.

POWER extracts the sums of squares for the specified effect, the mean square error from the OUTSTAT data set and calculates the total original size and effect size delta. Before performing any power calculations, POWER prepends the original sample size, standard deviation, and effect size to the lists of user specified values. The requested power statistics are then computed for all possible and applicable combinations of sample size, standard error, significance level, and effect size. POWER outputs these calculations to a user specified data set.

Calculated Values

The POWER macro computes the following power related measures:

- Power, the probability of getting a significant test result.
- Least Significant Number (LSN), the sample size required to produce a significant test for a sample with the same profile (sums of squares, mean square error, etc.) as the current sample.
- Power for the LSN, the power of the effect test when the sample size is equal to the LSN. This power is always greater or equal to 50%.
- Adjusted Power (ADJPOW), for retrospective analyses the adjusted power removes bias in the noncentrality parameter. The macro calculates adjusted power only for the original delta.
- Confidence Interval for Adjusted Power (POWCI), the confidence interval in terms of the confidence interval around the noncentrality parameter estimate. The macro calculates confidence intervals only for the original delta.

Computational Formulas

Power is computed as

$$Power = 1 - F\left\{F_{Crit}, df_{Hyp}, n - df_{Model} - 1, \lambda\right\}$$

where

$$F_{Crit} = F^{-1}\left\{1-\alpha, df_{Hyp}, n - df_{Model} - 1\right\},$$

$$\lambda = \frac{n\delta^2}{\sigma^2} = \frac{SS(Hyp)}{\sigma^2} \approx \frac{SS(Hyp)}{MSE},$$

and delta, the effect size, is

$$\delta^2 = \frac{SS(Hyp)}{n}.$$

The LSN is calculated by solving the following equation iteratively for *n*

$$\alpha = 1 - F\left\{\frac{n\delta^2}{df_{Hyp}\sigma^2}, df_{Hyp}, n - df_{Model} - 1\right\}.$$

Power for the LSN is calculated as

$$Power_{LSN} = 1 - F\left\{F_{Crit}, df_{Hyp}, n - df_{Model} - 1, \lambda_{LSN}\right\}$$

where

$$\lambda_{LSN} = \frac{\delta^2 LSN}{\sigma^2}.$$

The adjusted power is calculated as

$$Power_{Adj} = 1 - F\left\{F_{Crit}, df_{Hyp}, n - df_{Model} - 1, \lambda_{Adj}\right\}$$

where

$$\lambda_{Adj} = Max\left[0, \left\{\frac{\lambda\left(\left(N - df_{Model} - 1\right) - 2\right)}{N - df_{Model} - 1} - df_{Hyp}\right\}\right]$$

or

$$\lambda_{Adj} = Max\left[0, \left\{\frac{\lambda df_{Error}}{df_{Error}} - df_{Hyp}\right\}\right].$$

Confidence limits for adjusted power are constructed according to Dwass (1955) as

$$Lower\ Limit = df_{Hyp}\left\{Max\left[0, \sqrt{F_{Sample}} - \sqrt{F_{Crit}}\right]\right\}^2$$

and

$$Upper\ Limit = df_{Hyp}\left(\sqrt{F_{Sample}} + \sqrt{F_{Crit}}\right)^2.$$

Note that for all of the above calculations *n* varies with user specified values, while *N* is the original sample size and does not vary.

Examples

The following sections provide examples of retrospective and prospective power analyses.

A Retrospective Power Analysis

This data comes from an experiment on leprosy (Snedecor and Cochran 1967). The pre– and post–treatment leprosy scores, x and y respectively, are recorded for patients on three drugs. Table 1 displays the data.

Table 1: Data for Retrospective Power Analysis

Drug	Data										
A	X	11	8	5	14	19	6	10	6	11	3
	Y	6	0	2	8	11	4	13	1	8	0
B	X	6	6	7	8	18	8	19	8	5	15
	Y	0	2	3	1	18	4	14	9	1	9
C	X	16	13	1	9	21	16	12	12	7	12
	Y	13	10	18	5	23	12	5	16	1	20

To examine the power for testing the drug effect on the response Y in the presence of the covariate X., submit the following SAS statements:

```
PROC GLM DATA=DRUGS OUTSTAT=DRUGOUT;
    CLASS DRUG;
    MODEL Y = DRUG X;
RUN;

    %POWER(  DATA      =  DRUGOUT,
             OUT       =  DRUGPOW,
             EFFECT    =  DRUG,
             CALCS     =  POWER ADJPOW POWCI LSN,
             SS        =  SS3,
             ALPHA     =  0.05 0.01,
             N         =  60);
```

Table 2 displays the resulting output.

Table 2: Retrospective Power Analysis

Type I Error Rate	Sample Size	Root MSE	Effect Size	Power
0.01	30	4.01	1.5116	0.17573
0.01	60	4.01	1.5116	0.47671
0.05	30	4.01	1.5116	0.39681
0.05	60	4.01	1.5116	0.72205

Adjusted Power	CI: Lower Limit	CI: Upper Limit	Least Significant Number	Power for N=LSN
0.06645	0.01	0.97761	70	0.57050
0.30237	0.01	0.98257	N	N
0.19951	0.05	0.98207	46	0.59037
0.54998	0.05	0.98477	N	N

Prospective Power Analysis

Suppose you want to estimate the sample size needed and power for testing the effect of a variable with several levels. To use POWER you will need estimates of the group means and the variance. Create a data set as is in Table 3 giving proposed sample sizes.

Table 3: Data for Prospective Power Analysis

Group	Mean	Count
A	40	5
B	45	10
C	35	10

To get the GLM OUTSTAT data set required by POWER, issue the SAS statements below. Note the use of the FREQ statement.

```
PROC GLM DATA=PROSPECT OUTSTAT=PROSPOUT;
        CLASS GROUP;
        FREQ COUNT;
        MODEL MEAN=GROUP;
RUN;
```

Then call POWER supplying your estimates of the variance with the SIGMA parameter. The results appear in Table 4.

```
%POWER(   DATA=PROSPOUT,
          OUT       =  PROSPOW,
          EFFECT    =  GROUP,
          CALCS     =  POWER LSN,
          ALPHA     =  0.01 0.05,
          SIGMA     =  4.0 8.0,
          DELTA     =  2.0 5.0);
```

Table 4: Prospective Power Analysis

Type I Error Rate	Sample Size	Root MSE	Effect Size	Power	LSN	Power at LSN
0.01	25	4	4.47	0.98	13	0.61
0.01	25	4	2.00	0.28	42	0.57
0.01	25	4	5.00	0.99	12	0.66
0.01	25	8	4.47	0.37	35	0.58
0.01	25	8	2.00	0.05	153	0.57
0.01	25	8	5.00	0.48	29	0.58
0.05	25	4	4.47	0.99	9	0.63
0.05	25	4	2.00	0.54	28	0.60
0.05	25	4	5.00	0.99	8	0.63
0.05	25	8	4.47	0.64	23	0.60
0.05	25	8	2.00	0.17	99	0.58
0.05	25	8	5.00	0.75	19	0.59

Acknowledgments

The author wishes to acknowledge SAS Institute Inc. and Pharmaceutical Product Development, Inc. on whose computers this software was developed.

SAS, and JMP are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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

Please address comments/questions to:
Kristin R. Latour
113 Mariposa Drive
Cary, NC 27513
Work Phone: (919) 544–6269
Home Phone: (919) 467–7289

References

Dwass, M. (1955), "A Note on Simultaneous Confidence Intervals," *Annals of Mathematical Statistics* 26, 146–147.

SAS Institute Inc., *JMP User's Guide*, *Version 2 of JMP*, Cary, NC: SAS Institute Inc., 1991

SAS Institute Inc., *SAS/STAT® User's Guide, Version 6, Fourth Edition, Volumes I and II,* Cary, NC: SAS Institute Inc., 1989

Snedecor, G.W. and Cochran, W.G. (1967), *Statistical Methods*, Ames, IA: Iowa State Press

Wright, S.Paul, and Ralph G. O'Brien (1988), "Power Analysis in an Enhanced GLM Procedure: What it Might Look Like," *SUGI 1988, Proceedings of the Thirteenth Annual Conference*, 1097–1102, Cary, NC: SAS Institute Inc.

Appendix: SAS Macro Code for %POWER

```
*****************************************************************
**      PROGRAM NAME: POWER.SAS                                **
**      AUTHOR: Kristin Latour                                 **
**      UPDATED: 12Mar92                                       **
**      PURPOSE: Calculates the following power                **
**      related measures:                                      **
**                                                             **
**        •  Power for an effect test                          **
**        •  Adjusted power and confidence limits              **
**        •  Least significant number                          **
**                                                             **
**      The macro is invoked with the following                **
**      statement:                                             **
**                                                             **
**      %power(DATA=outstat_data_set,                          **
**             OUT=output_data_set,                            **
**             EFFECT=effect_name,                             **
**             CALCS=calculations_to_report,                   **
**             SS=type_sums_of_squares_to_use,                 **
**             ALPHA=list_of_significance_levels,              **
**             N=list_of_sample_sizes,                         **
**             SIGMA=list_of_standards_deviations,             **
**             DELTA=list_of_effect_sizes)                     **
**                                                             **
**      The OUTSTAT_DATA_SET is created by the                 **
**      OUTSTAT option in PROC GLM. All                        **
**      calculations are output to OUTPUT_DATA_SET.            **
**      Calculations are done/reported for                     **
**      CALCULATIONS_TO_REPORT on                              **
**      EFFECT_NAME from OUTSTAT_DATA_SET                      **
**      with the corresponding                                 **
**      TYPE_SUMS_OF_SQUARES_TO_USE.                           **
**                                                             **
**      The keyword parameters, assigned in any                **
**      order, must be assigned with the format:               **
**                                                             **
**                 keyword=value                               **
**                                                             **
**      Alpha is assigned a default value of 0.05              **
**      when no other value is specified. User                 **
**      defined value lists for n, sigma, and delta            **
**      will have true values from the data                    **
**      prepended to them.                                     **
**                                                             **
**      Outputted values of .N indicate statistics             **
**      that were not calculated, values of .U                 **
**      indicate that the macro was unable to                  **
**      calculate the statistic.                               **
**                                                             **
****************************************************************;

%macro power   (data=,
                  out=,
               effect=,
                calcs=,
                ss=ss3,
            alpha=0.05,
                   n=,
               sigma=,
               delta=);

    %let calcs=%upcase(&calcs);
    %let ss=%upcase(&ss);

*   GET ERROR AND HYPOTHESIS DF AND
    SS FROM OUTSTAT DATA SET;
    data _null_;
      set &data(where=(_type_ in ("&ss",'ERROR')))
        end=lastobs;

      if _n_=1 then do;
        dfr=0;
        norig=0;
      end;

      norig+df;
      select(_source_);
        when ('ERROR') do;
          call symput('dfeorig',left(put(df,8.)));
          if ss gt 0 then
            call symput('sigorig', left(put(sqrt(ss/df),14.2)));
          else call symput('sigorig','');
        end; *when;
        when (upcase("&effect")) do;
          dfr+df;
          call symput('dfh',left(put(df,8.)));
          call symput('ssh',left(put(ss,14.2)));
          call symput('fsamp',left(put(f,14.2)));
        end; *when;
        otherwise do;
          dfr+df;
        end; *otherwise;
      end; *select;

      if lastobs then do;
        call symput('dfr',left(put(dfr,8.)));
        call symput('norig',left(put((norig+1),8.)));
      end; *if;
```

```
run;

*       PUT ORIGINAL DELTA INTO MACRO VARIABLE;
        data _null_;
           delta=sqrt(&ssh/&norig);
           call symput('delorig',left(put(delta,16.4)));
           stop;
        run;

*       PREPEND VALUES FOR N SIGMA DELTA THAT
        OCCUR IN DATA TO USER-SPECIFIED
        VALUES;
        %let n=&norig &n;
        %let sigma=&sigorig &sigma;
        %let delta=&delorig &delta;

        %put n=&n;
        %put sigma=&sigma;
        %put delta=&delta;
        %put alpha=&alpha;
        %put delorig=&delorig;
        %put sigorig=&sigorig;
        %put dfr=&dfr;
        %put dfh=&dfh;
        %put ssh=&ssh;
        %put fsamp=&fsamp;


*       MACRO TO ADD COMMAS AFTER VALUES
        SPECIFIED FOR ALPHA, N, SIGMA, DELTA;
        %macro comma(string,varname);
           %local count word;
           %let count=1;
           %let word=%qscan(&string,&count,%str( ));
           %let &varname=&word;
           %do %while (&word ne);
                 %let count=%eval(&count+1);
                 %let word=%qscan(&string,&count,%str( ));
                 %if (&word ne) %then
                    %let &varname=&&&varname, &word;
           %end; * %do;
           %put &varname = &&&varname;
        %mend comma;


*       CREATE COMMA DELIMITTED LIST FOR USER-SPECIFIED
        VALUES OF ALPHA, N, SIGMA, AND DELTA;
        %let given=alpha n sigma delta;
        %do i=1 %to 4;
           %let current=%scan(&given,&i,%str( ));
           %comma(&&&current,&current);
        %end; *%do;


* PERFORM CALCULATIONS;
        data &out;
           do alpha=&alpha;
              do number=&n;
                 dfe=number-&dfr-1;
                 do sigma=&sigma;
                    do delta=&delta;

                 *  CALCULATE LAMBDA AND POWER;
                    lambda=(number*delta**2)/(sigma**2);
                    astar=1-alpha;
                    fcrit=finv(astar,&dfh,dfe);
                    if lambda>135 then power=1.0;
                    else power=
                       1-probf(fcrit,&dfh,dfe,lambda);

                    *  CALCULATE ADJUSTED LAMBDA AND
                       CI FOR ORIGINAL DELTA;
                       if delta=&delorig and
                       index("&calcs",'ADJPOW')>0 then do;
                          lambadj=max(0,(lambda*(&dfeorig-2)/
                             &dfeorig)-&dfh);
                          if lambadj>135 then adjpow=1.0;
                          else adjpow=1-probf(fcrit,&dfh,dfe,lamadj);
                       end; *if;
                       else if delta=&delorig then adjpow=.U;
                       else do;
                          adjpow=.N;
                       end; * else;


                    *  GET CI ON LAMBDA AND POWER;
                       if index("&calcs",'POWCI')>0 and
                          adjpow not in (.N,.U) then do;
                          lamlow=&dfh*(max(0,(sqrt(&fsamp)-
                             sqrt(fcrit))))**2;
                          if lamlow>135 then powlow=1.0;
                          else powlow=1-probf(fcrit,&dfh,dfe,lamlow);

                          lamup=&dfh*(sqrt(&fsamp)+sqrt(fcrit))**2;
                          if lamup>135 then powup=1.0;
                          else powup=1-probf(fcrit,&dfh,dfe,lamup);
                       end; *if;
                       else do;
                          powlow=.N;
                          powup=.N;
                       end; * else;

                    *  FIND LEAST SIGNIFICANT N;
                       if number=&norig and
                          index("&calcs",'LSN')>0 then do;
                          niter=&dfr+2;
                          lstar=(niter*delta**2)/(&dfh*sigma**2);

                          do until (diff<0.0000001);
                             niter=niter+1;
                             errn=niter-&dfr-1;
                             lstar=(niter*delta**2)/(&dfh*sigma**2);
                             diff=astar-probf(lstar,&dfh,errn);
                          end; * do;

                          lsn=niter;
                          lsndfe=lsn-&dfr-1;
                          lamblsn=(lsn*delta**2)/(sigma**2);
                          astar=1-alpha;
                          fcrit=finv(astar,&dfh,lsndfe);
                          if lamlsn>135 then powlsn=1.0;
                          else powlsn=1-probf(fcrit,&dfh,lsndfe,lamblsn);
                       end; *if;
                       else do;
                          lsn=.N;
                          powlsn=.N;
                       end; *else;

                       output;

                    end; * do delta;
                 end; * do sigma;
              end; * do number;
           end; * do alpha;

           label
              alpha='Type I Error Rate'
              number='Sample Size'
              sigma='Root Mean Square Error'
              delta='Effect Size'
              power='Power of Test'
```

```
        adjpow='Adjusted Power'
        powlow='Confidence Interval: Lower Limit'
        powup='Confidence Interval: Upper Limit'
        lsn='Least Significant Number'
        powlsn='Power when N=LSN';

        keep alpha number sigma delta
            %if %index(&calcs,POWER)>0 %then %str(power);
            %if %index(&calcs,ADJPOW)>0 %then %str(adjpow);
            %if %index(&calcs,POWCI)>0 %then
                %str(powlow powup);
            %if %index(&calcs,LSN)>0 %then %str(lsn powlsn);;
    run;

    %let ssnum=%substr(&ss,3,1);
*   PRINT RESULTS;
    proc print data=&out noobs label;
        var alpha number sigma delta
            %if %index(&calcs,POWER)>0 %then %str(power);
            %if %index(&calcs,ADJPOW)>0 %then %str(adjpow);
            %if %index(&calcs,POWCI)>0 %then
                %str(powlow powup);
            %if %index(&calcs,LSN)>0 %then %str(lsn powlsn);;
        title1 "Power Calculation for effect %upcase(&effect)";
        title2 "Type &ssnum Sums of Squares";
    run;
    title;

%mend power;
```