

Power Simulation for Categorical Data Using the RANTBL Function

Peter H. Van Ness, Theodore R. Holford, Joel A. Dubin,
Program On Aging, Yale University School of Medicine, New Haven, CT

Abstract

Increases in computing power have enabled Monte Carlo simulations to become increasingly common in contemporary statistical practice. Some applications require specialized statistical software while others can be implemented with general multi-purpose statistical programs. Simulations of the power of statistical tests—the probability of rejecting the null hypothesis when it is false—are among those applications that can be implemented in programs like SAS[®]. This article will demonstrate how the SAS[®] program, and in particular, its RANTBL function can be used to simulate the power of many statistical tests for categorical data. Furthermore, the advantages and disadvantages will be described of simulating random variables from a probability distribution defined by a table of values rather than by the parameters of a probability mass function.

Code will be provided for simulating the power of the Kendall Tau-b correlation coefficient. The reason for simulating the power of the Kendall statistic is that there is no widely used power/ sample size formula for accomplishing this task. Code will also be given for simulating the power of the main predictor of a logistic regression model with a single covariate. For logistic regression models there are such formulas but they do not satisfactorily take into account the impact of covariates and this can be done in the simulation approach. Finally, guidance will be offered regarding the circumstances in which using the noncentral chi-square approach to power calculations might be easier to implement than Monte Carol simulations.

Introduction

Simulation has become a prominent part of contemporary statistical research. In *Elements of Computational Statistics* (2002) James Gentle reports that almost half of the articles then appearing in the *Journal of the American Statistical Association* included simulation studies as part of the reported research (Gentle, 2002: 338). Simulation can be used for various purposes and a recent book by Fan and colleagues describes many simulation applications that can be implemented in the SAS[®] system (Fan, Felsovalyi, Sivo et al., 2003). A common and fairly simple application is the use of simulation for determining the power of a statistical test. This article will describe how the SAS[®] system—and specifically, the RANTBL function—can be used to simulate the power of statistical tests for categorical data. Although the Fan text mentions the RANTBL function, it does not illustrate its use in power simulation and so this article will make a new and useful addition to the literature on this topic.

Paper 207-30

There are three major ways to determine the power of a statistical test: 1) to estimate it with a power/ sample size formula; 2) to calculate it with a noncentral probability distribution; and 3) to simulate it with Monte Carlo techniques that generate and transform pseudo-random numbers. No single approach is best for all circumstances. Certainly the most common and convenient approach is the use of power/ sample size formulas provided by software dedicated to this purpose. The PASS[®] (*PASS User's Guide*, 2002) and nQuery Advisor[®] (Elashoff, 2002) software packages are specialized programs for this purpose. SAS/STAT[®] now includes two procedures (POWER and GLMPOWER) that estimate power for a variety of statistical tests (*SAS/STAT[®] User's Guide*, 2003). As extensive as these software resources have become they do not provide formulas for all statistical tests and so recourse to the other two options is often necessary.

Background

The power of a statistical test is the probability of rejecting the null hypothesis when it is false. It can be determined via simulation in a very straightforward way. A large number of samples of random variables are simulated according to the relevant probability distribution and then the statistical test of interest is performed for each sample. Those tests that achieve a desired level of statistical significance, e.g., 0.05, are identified and the proportion of statistically significant tests among the total number of tests equals the test's power.

Simulating samples of random variables from a specific probability distribution is a two-step process. First, random (or more precisely, pseudorandom) numbers are generated. Most random number generators, including the one used by SAS[®], start with a seed or initial value (X_n) and then generate a sequence of random numbers by insertion of this value and other constants into the following recursive function:

$$X_{n+1} = (aX_n + c) \text{ modulo } m \text{ where } n, a, c, \text{ and } m \text{ are integers } \geq 0.$$

This expression means that $(aX_n + c)$ is divided by m and the remainder is said to equal X_{n+1} (Ross, 2001). It is called a "congruential generator" because the above equation yields a congruence relationship (modulo m) for any value of n over the sequence of generated numbers (Fishman and Moore, 1982). A generator of this sort is called "multiplicative" when the constant c is set to zero. The SAS[®] RANUNI function uses a multiplicative generator with a equal to 397204094 and m equal to $2^{31}-1$. Random numbers that can be generated by this method have a uniform distribution over the open interval (0,1).

Sequences of random numbers obtained in this way are then transformed so as to simulate a sample of random variables from other probability distributions. The inverse transformation method is one way to do this. Let U be a uniform random variable defined on the open interval (0,1). For any continuous distribution function F , if we define a second random variable Y by the following:

$$Y = F^{-1}(U) \text{ where } F^{-1}(x) \text{ equals the value } y \text{ for which } F(y) = x;$$

Paper 207-30

then the random variable Y has distribution function F . F is the probability density function of interest, for example, of an exponential or gamma random variable (Ross, 2001). The method can be applied to discrete probability mass functions—of binomial or Poisson random variables—with only slight modification.

The SAS[®] RANTBL function uses the inverse transformation method to simulate a discrete probability distribution, not by specifying the parameters, e.g., the mean and variance, of a probability mass function, but by having that function instantiate the values in a tabled distribution. Such random samples i would have the following probability mass function:

$$f(i) = \begin{cases} p_i & \text{for } i = 1, 2, \dots, n \\ 1 - \sum_{i=1}^n p_i & \text{for } i = n + 1 \text{ where } \sum_{i=1}^n p_i \leq 1 \end{cases}$$

where p is a vector of probabilities, such that $0 \leq p \leq 1$, and n is the largest integer such that $n \leq$ the size of p . This says that for any cell i of the table the function assigns to that cell some probability and for the last cell in the table it assigns the probability that is the difference between 1 and the sum of the other probabilities.

Kendall's Tau-b

Kendall's Tau-b is a correlation coefficient for ordered categorical data (Kendall, 1945). Unlike the Pearson correlation coefficient its usage does not assume a bivariate normal distribution for the two random variables being tested nor does it even assume that this joint distribution is continuous as does the Spearman rank correlation coefficient. It is a concordance statistic that explicitly takes into account the large number of tied values that can occur in ordered categorical data (Agresti, 2002: 68). It classifies each pair of items measured for random variables X and Y as either concordant, discordant, or tied. A pair of values is concordant if the item ranked higher for X is also ranked higher for Y , and discordant if the item ranked higher for X is ranked lower for Y . The pair is tied if the items have the same classification for X or for Y .

Let C equal the number of concordant pairs and D the number of discordant pairs. T_X is the number of pairs tied for X and T_Y the number tied for Y . Where n is the number of items and $\binom{n}{2}$ the number of pairs of items, then the Tau-b correlation coefficient is defined by the following formula:

$$\tau_b = \frac{C - D}{\left\{ \left[\binom{n}{2} - T_X \right] \left[\binom{n}{2} - T_Y \right] \right\}^{\frac{1}{2}}}$$

Paper 207-30

This correlation coefficient is applicable to the correlation of two binary variables even when the variables are both nominal. This is true because reversing the two categories changes the direction of the association but maintains its magnitude—something that is not true for nominal variables with more than two categories (Agresti, 1984: 167). Even though Kendall's Tau-b has wide application and is given comparable standing by SAS[®] with the Pearson and Spearman correlation statistics—all three are options to the CORR procedure—it is not widely used. A power/ sample size formula is not available for it in the major statistical packages cited above. Thus it is a statistic used in categorical data analysis for which power simulation might be very helpful.

Multiple Logistic Regression

Several power/ sample size formulas are available for logistic regressions models. One formulated by Hsieh and colleagues is particularly noteworthy (Hsieh, Bloch, and Larsen, 1998). It is used in the PASS software and it has a special adjustment to the formula that adjusts the sample size for the impact of covariates. Where ρ is the multiple correlation coefficient between the main predictor variable and other covariates, Hsieh and colleagues adjusts the sample size according to the following formula:

$$N_{\text{adjusted}} = \frac{N_{\text{unadjusted}}}{1-\rho^2}$$

Hosmer and Lemeshow suggest that this correction may be too conservative (Hosmer and Lemeshow, 2000: 345-46). One way to circumvent this controversy is to use a simulation approach to determining the power of logistic regression models. One can use either the p-values for a Wald statistic, a score statistic, or a likelihood ratio statistic in order to determine whether the main predictor of a given model qualifies as statistically significant according to some specified level of significance. In the code provided in Appendix B the likelihood ratio statistic yielded by the Type 3 option for the GENMOD MODEL statement was used for all significance tests. The form of the test is the usual one:

$$G = -2 \ln \left[\frac{(\text{likelihood without the variable(s) of interest})}{(\text{likelihood with the variable(s) of interest})} \right]$$

Under the null hypothesis this statistic follows the chi-square distribution with the number of degrees of freedom equal to the difference in the number of variables in the full and reduced models. Further mathematical details can be found in Hosmer and Lemeshow (Hosmer and Lemeshow, 2000: 11-17).

Program Commentary

The data for the Kendall Tau-b correlation analysis and simulation can be pictured as a two-by-two contingency table.

		Variable 2	
		0	1
Variable 1	0	40	30
	1	10	20

In a preliminary step, the program printed in Appendix A first calculates the correlation coefficient using the group data format. The first step of the simulation itself assigns as values to a pair of macro variables the marginal row totals for variable 1. For a correlation coefficient it is not always helpful to distinguish the two variables as independent or dependent, as an exposure and an outcome; however, in order to establish continuity with the logistic regression program in Appendix B and to facilitate description of the simulation program variables, variable 1 will be considered an exposure and variable 2 an outcome. Thus 'var10' designates the unexposed level of variable 1 and 'nvar10' designates its marginal total. Since variable 2 is being thought of as a disease outcome, its levels may be distinguished as controls and cases. Thus 'contvar10' specifies the number of unexposed persons who do not have the disease. Analogous codings are used in the programs for the other key variables.

Step 2 defines input variables for the four cells of the contingency table. Its input values are provided by step 8 and equal the probability of an unexposed person being a control or being a case and the probability of an exposed person being a control or case. For instance, the probability of an unexposed person being a control is 0.5714 (70/40); further, note that the rows sum to 1. Step 2 creates two arrays that will be invoked in the application of the RANTBL function in steps 5 and 6. Before, step 3 defines a replicate variable and step 4 initializes the frequency tables generated in the following two steps. The RANTBL function in step 5 generates 1s and 2s for the 1-by-2 frequency table for the unexposed according to the probability assigned to the cells of this table by step 8. (A '1' signifies a control and '2' signifies a case.) Step 6 does the same for the exposed. The arrays invoked in these steps work in combination with the 'do' loop involving the macro variable for marginal totals in order to convert the numbers generated by the RANTBL function into tabulations of the frequency of the occurrence of those numbers. These latter numbers occupy the cells of the replicated tables. Step 7 creates a file of the replicated studies which can be printed and examined to ensure that they have approximately the same proportions as the original contingency table.

Paper 207-30

Step 9 creates a count variable that records the numbers associated with the cells in the contingency tables of the replicated studies. It is needed to run the correlation analysis for each of the replicated studies: this is done in step 10. The program concludes by creating a file of the hypothesis tests, assigning a significance level and a variable that corresponds to it, and then finally tabulating the proportion of times that the replicated studies rejects the null hypothesis at the specified level of significance.

For this data, the simulated power using 10,000 replications is approximately 0.61. A power curve can be created by systematically varying the proportions entered in step 8 (Zelterman, 2002). One might begin by testing to see that the Type I error is correctly simulated. By placing 0.5 in all four of the cells in step 8 one is indicating that the relationship between exposure and outcome is apparently random. Thus, one would expect the power to be equal to the assumed level of significance, in this case, 0.05. A simulation with 10,000 replications yields a power of 0.053.

The program in Appendix B for simulating the power of a binary main predictor with a single binary control variable in a logistic regression model is very similar to the one just described. Two differences are most important. The model run on replicated studies in step 10 is now a logistic regression model rather than a Kendall Tau-b correlation analysis. The generality of this SAS[®] approach to simulation is demonstrated by the fact that basically the same code can be used for any statistic that can be computed by a SAS[®] procedure that has FREQ and BY statements and that generates a p-value. The other main difference in the second program is that the frequency tables generated by steps 5 and 6 are larger, and assigned values in step 8, have four cells apiece rather than two. Additionally, determining these values is somewhat more complicated, requiring the calculation of predicted probabilities using the regression equation for the logistic model. Assumptions for the intercept and parameter estimates of the regression equation must be made.

If a covariate with more than two levels is used, or if more than one covariate is included in the multivariable model, then these tables would be yet larger. Continuous variables need to be categorized, though not necessarily dichotomized. This approach to power simulation of logistic regression models has very feasible generalizations.

Conclusion

In practice statisticians choose methods for determining statistical power on the basis of the availability of relevant power/ sample size formulas and software and the availability of relevant pilot data. It is certainly more convenient to use a commercial software package specifically designed for power calculations than to write one's own code for doing so in SAS[®] or some other programming language. In our experience with several such software programs, determining power in this way tends to be conservative, yielding less power than by simulation or the chi-square approach. For instance, the power results yielded by the noncentral chi-square and simulation program in Appendix B are very

Paper 207-30

similar but the power determined by the PASS software for the same logistic regression model is somewhat less.

As illustrated in preliminary code for the logistic regression simulation in Appendix B, the noncentral chi-square approach is often very easy to implement retrospectively, i.e., when you have all the information you need to actually calculate the statistic of interest. The noncentral chi-square approach is also feasible when the exposure marginal totals are known but there is insufficient information to fit a logistic regression model; however, it becomes extremely complicated to use this approach for the case-control circumstance when covariates are added to the logistic regression model (Holford, 2002: 305-09). When there is a paucity of pilot data, the simulation approach is very effective because it allows for the explicit assumption of relevant items of information, e.g., the parameter estimates of a regression model needed for calculating the predicted probabilities of the exposure categories. The simulation approach is also easily adapted to the circumstance of a case-control study where one knows the column or outcome marginal totals rather than the row or exposure marginal totals. An application of Bayes' rule is needed in to get the requisite predicted probabilities for the combination of exposure categories from the logistic regression equation.

Finally, many power simulations for categorical data can be accomplished by simulating a random sample of binomial or other discrete random variables. SAS[®] provides functions RANBIN and RANPOI for this purpose but no function for multinomial distributions. As suggested by the logistic regression program in Appendix B, the RANTBL function can handle this distribution so long as the table values are available. Thus the RANTBL function provides a flexible and general framework for power simulation for a variety of statistic important in categorical data analysis.

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

Paper 207-30

Appendix A: Power Simulation Code for a Kendall's Tau-b Correlation Coefficient

```

*0) Original data step and analysis;
data correlation;
input var1 var2 count;
cards;

0 0 40
0 1 30
1 0 10
1 1 20
;
proc corr kendall;
  freq count;
  var var1 var2;
  ods output kendallcorr=tests;
run;

*1) Input marginal exposure totals as macro variables;
%let nvar10=70;
%let nvar11=30;

*2) Code data step input line and arrays for the RANTBL function;
data alldata;
input contvar10 casevar10
      contvar11 casevar11;
array var10_tab (2) cont0 case0;
array var11_tab (2) cont1 case1;

do rep=1 to 1000;
  do j=1 to 2;
    var10_tab(j)=0;
    var11_tab(j)=0;
  end;
  do i = 1 to &nvar10;
    k=rantbl(-1, contvar10, casevar10);
    var10_tab(k) = var10_tab(k) + 1;
  end;
  do i = 1 to &nvar11;
    k=rantbl(-1, contvar11, casevar11);
    var11_tab(k) = var11_tab(k) + 1;
  end;
  do j=1 to 2;
    var2 = j;
    var1 = mod(j-1,2);
    var10 = var10_tab(j);
    var11 = var11_tab(j);
    output;
  end;
end;

```

*3) Create replicates;

*4) Initialize frequency tables;

*5) Generate table for unexposed;

*6) Generate table for exposed;

*7) Create replicate studies file;

Paper 207-30

```

*8) Enter distribution of controls and cases for exposure categories;
cards;
0.5714 0.4286
0.333 0.667
run;

*9) Create the count variable needed for the regression model;
data newdata; set alldata;
array expvar [2] var10 var11;
if var1=0 and var2=1 then do a= 1 to 2;
  var1=a-1; count=expvar[a]; output;
end;
if var1=1 and var2=2 then do a= 1 to 2;
  var1=a-1; count=expvar[a]; output;
end;
keep var1 var2 count rep;
run;

*10) Run the correlation analysis for each replicate study;
proc corr data=newdata kendall;
  freq count;
  by rep;
  var var1 var2;
  ods output kendallcorr=tests;
  ods listing select tests;
run;

*11) Create file of exposure tests;
data subtest; set tests;
if variable='var1';

*12) Define the indicator variable of null hypothesis rejection;
reject = (pvar2 < 0.05);
run;

*13) Tabulate times the null hypothesis is rejected;
proc freq; tables reject; run;

```

Appendix B: Power Simulation Code for a Multiple Logistic Regression Model

```

*0) Original data step and analysis;
data temp1;
  input covariate predictor br count;
cards;
          0          0          0    32
          0          1          0     6
          1          0          0    36
          1          1          0    10
          0          0          1    97
          0          1          1    45
          1          0          1   129
          1          1          1    51
;
run;

```

Paper 207-30

```

proc genmod data=temp1 descending ;
    freq count;
    title2 'Logistic Regression Model with a Binary Covariate';
    model br = covariate predictor / link=logit dist=bin type3;
    ods output type3=tests;
run;

*00) Calculation of power by the noncentral chi-square method;
data power; set tests;
format chisq 6.4;
test = cinv(0.95,1);
power = 1-probchi(test,1, chisq);

proc print data=power;
run;

*1) Input marginal exposure totals as macro variables;
%let nexp00=129;
%let nexp01=51;
%let nexp10=165;
%let nexp11=61;

*2) Code data step input line and arrays for the RANTBL function;
data alldata;
input  contexp00 caseexp00
       contexp01 caseexp01
       contexp10 caseexp10
       contexp11 caseexp11;

array exp00_tab (2) cont00 case00;
array exp01_tab (2) cont01 case01;
array exp10_tab (2) cont10 case10;
array exp11_tab (2) cont11 case11;

do rep=1 to 1000;
do j=1 to 2;
    exp00_tab(j)=0;
    exp01_tab(j)=0;
    exp10_tab(j)=0;
    exp11_tab(j)=0;
end;
do i = 1 to &nexp00;
    k=rantbl(-1, contexp00, caseexp00);
    exp00_tab(k) = exp00_tab(k) + 1;
end;
do i = 1 to &nexp01;
    k=rantbl(-1, contexp01, caseexp01);
    exp01_tab(k) = exp01_tab(k) + 1;
end;
do i = 1 to &nexp10;
    k=rantbl(-1, contexp10, caseexp10);
    exp10_tab(k) = exp10_tab(k) + 1;
end;
do i = 1 to &nexp11;
    k=rantbl(-1, contexp11, caseexp11);
    exp11_tab(k) = exp11_tab(k) + 1;
end;
end;

```

Paper 207-30

```

do j=1 to 2;                                *7)Create replicate studies file;
    categ = j;
    exp00 = exp00_tab(j);
    exp01 = exp01_tab(j);
    exp10 = exp10_tab(j);
    exp11 = exp11_tab(j);
    output;
end;
end;

*8) Enter distribution of controls and cases for exposure categories;
cards;
0.23667 0.76333
0.14645 0.85355
0.22709 0.77291
0.13985 0.86015
run;

*9) Create the count variable needed for the regression model;
data newdata; set alldata;
array expvar [4] exp00 exp01 exp10 exp11;

if categ=1 then do a= 1 to 4;
covariate = int((a-1)/2);
predictor = mod(a-1,2);
count=expvar[a]; output;
end;
if categ=2 then do a= 1 to 4;
covariate = int((a-1)/2);
predictor = mod(a-1,2);
count=expvar[a]; output;
end;
keep categ count rep covariate predictor;
run;

*10) *Fit regression models for the replicate studies;
proc genmod data=newdata;
freq count;
model categ = covariate predictor / dist=binomial link=logit
type3 pred;
by rep;
ods output type3=tests obstats=stats;
ods listing select tests stats;
run;

*11) Create file of hypothesis tests;
data subtest; set tests;
if source='predictor';

*12) Define the indicator variable of null hypothesis rejection;
reject = (probchisq < 0.05);
run;

*13) Tabulate times the null hypothesis s rejected;
proc freq; tables reject; run;

```

Acknowledgments

The authors thank Heather G. Allore for her help and encouragement. This work was supported in part by NIH Grant # P30AG21342 and in part by NIH Grant # AG19769.

References

- Agresti, A. (1984). *Analysis of Ordinal Categorical Data*. John Wiley & Sons, New York.
- Agresti, A. (2002). *Categorical Data Analysis*. John Wiley & Sons, New York.
- Elashoff, J.D. (2002). *nQuery Advisor, 5.0*. Statistical Solutions, Los Angeles.
- Fan, X., Felsovalyi, A., Sivo, S.A., et al. (2003). *SAS[®] for Monte Carlo Studies: A Guide for Quantitative Researchers*. SAS Institute, Cary, NC.
- Fishman, G.S., and Moore, L.R. (1982). A statistical evaluation of multiplicative congruential generators with modulus ($2^{31} - 1$). *Journal of the American Statistical Association* **77**, 129-36.
- Gentle, J.E. (2002). *Elements of Computational Statistics*. Springer, New York.
- Holford, T.R. (2002). *Multivariate Methods in Epidemiology*. Oxford University Press, New York.
- Hosmer, D.W., and Lemeshow, S. (2000). *Applied Logistic Regression*. John Wiley & Sons, New York.
- Hsieh, F.Y., Bloch, D.A., and Larsen, M.D. (1998). A simple method of sample size calculation for linear and logistic regression. *Statistics in Medicine* **17**, 1623-34.
- Kendall, M.G. (1945). The treatment of ties in ranking problems. *Biometrika* **33**, 239-51.
- PASS User's Guide* (2002). NCSS, Kaysville, UT.
- Ross, S.M. (2001). *Simulation*. 3rd ed. Academic Press, San Diego.
- SAS/STAT[®] User's Guide* (2003). *SAS OnlineDoc: Version 9*. SAS Institute, Cary, N.C.
- Zelterman, D. (2002). *Advanced Log-Linear Models Using SAS[®]*. SAS Institute, Cary, NC.