Paper SAS1983-2018

Power Analysis for Generalized Linear Models Using the New CUSTOM Statement in PROC POWER

John Castelloe, SAS Institute Inc.

Abstract

The CUSTOM statement that was added to the POWER procedure in SAS/STAT® 14.2 extends the scope of supported data analyses to include generalized linear models and other extensions of existing capabilities. It works in concert with an exemplary data set and the SAS/STAT procedure that you plan to use for the eventual data analysis. This paper explains the method and demonstrates it for a variety of data analyses, including Poisson regression, logistic regression, and zero-inflated models. It also discusses how you can use CUSTOM statement options to refine the method for sample size inflation/deflation or for extra covariates.

Prologue: The CUSTOM Statement in PROC POWER Comes In Where the LOGISTIC Statement Leaves Off

You're an industrial statistician working in a plant that rolls ingots into steel. Having an ingot that is improperly prepared for rolling is an expensive mistake, so the engineers know the importance of including your group in efforts to design experiments that maximize the probability that ingots are ready to be rolled. In fact, you recently used the LOGISTIC statement in PROC POWER in SAS/STAT software to make sure that such an experiment would have sufficient power to detect a certain odds ratio for ingot readiness. Your power analysis handled a lot of complexity. For example, it not only enabled you to study the effect of heating on ingot readiness, with different sampling rates for different levels of heating, but it also handled the mass of the ingot as a normally distributed covariate. For more information about the PROC POWER syntax required for an analysis like this, see “Example 92.9: Binary Logistic Regression with Independent Predictors” (Chapter 92, *SAS/STAT User's Guide*).

Now the plant wants to consider getting ingots from multiple suppliers, and the heating profile and mass distribution for the ingots produced by each supplier will be different. In other words, the Supplier covariate will be *correlated* with both the Heat predictor of interest and the Mass covariate. Because you came through on the previous experiment, the boss figures you'll deliver on this little complication, too. The problem is, the LOGISTIC statement in PROC POWER can't handle nominal predictors with more than two levels, and its support for correlated predictors isn't adequate for this situation. Never fear, though: the CUSTOM statement comes to the rescue!

Introduction

Statistical power analysis determines the ability of a study to detect a meaningful effect size. On the flip side, power analysis also finds the sample size that is required to provide a desired power for an effect of scientific interest. Proper planning reduces the risk of conducting a study that will not produce useful results and determines the most sensitive design for the resources available. Power analysis is now integral to the health and behavioral sciences, and its use is steadily increasing wherever empirical studies are performed.

Before SAS/STAT 14.2, the GLMPOWER and POWER procedures enabled you to conduct power analyses for two cases of generalized linear models: normal linear models (PROC GLMPOWER) and binary logistic regression (PROC POWER with the LOGISTIC statement). The scope of the LOGISTIC statement in PROC POWER is limited to single-degree-of-freedom tests, and classification variables with more than two levels are not supported. It has a rudimentary adjustment for correlation between the predictor of interest and the other covariates but otherwise assumes that all predictors are independent of one another.

In SAS/STAT 14.2, a new CUSTOM statement was added to PROC POWER that expands its scope to include generalized linear models that have nominal, count, or ordinal responses with arbitrary numbers of levels. Important special cases include logistic, Poisson, geometric, and negative binomial regression; proportional odds models; and zero-inflated models. The beauty of the approach is that it combines two tasks that are probably already familiar to you:

- creating a hypothetical data set
- writing SAS$^{®}$ code for data analysis in a SAS/STAT procedure such as PROC LOGISTIC or PROC GENMOD

You can use the CUSTOM statement to compute not only conditional power or sample size for a specific instance of the design matrix, but also unconditional power or sample size that accommodates whatever joint predictor distribution you specify. The features in the new CUSTOM statement can help you design studies for a wide variety of applications, such as industrial designs, clinical trials, agricultural variety studies, and advertising campaigns.

The CUSTOM statement also performs power analyis for extensions of existing analyses that involve the chi-square, *F*, *t*, normal, or correlation coefficient distribution. You can use these extensions to account for sample size inflation that is caused by correlated predictors or extra covariates, and for sample size deflation that is caused by correlation between the covariates and the response.

The following section provides an overview of power analysis for generalized linear models and explains the features in the new CUSTOM statement in the POWER procedure. The remainder of the paper consists of an extended example that covers power analysis for a logistic regression study with correlated predictors. The principles behind the method can easily be used in other applications of generalized linear models, such as Poisson regression, proportional odds models, and zero-inflated models.

Overview of Generalized Linear Models with Nominal, Count, or Ordinal Response

Generalized linear models extend the theory and methods of linear models to data that have nonnormal responses. Before this theory was developed, modeling of nonnormal data typically relied on transformations of the data, and the transformations were chosen to improve symmetry, homogeneity of variance, or normality. Such transformations must be performed with care because they also have implications for the error structure of the model, Also, back-transforming estimates or predicted values can introduce bias. The theory of generalized linear models originated with Nelder and Wedderburn (1972) and Wedderburn (1974) and was subsequently made popular by McCullagh and Nelder (1989).

Generalized linear models also apply a transformation, known as the *link function*, but it is applied to a deterministic component, the mean of the data. Furthermore, generalized linear models take the distribution of the data into account, rather than assuming that a transformation of the data leads to normally distributed data to which standard linear modeling techniques can be applied.

To put this generalization in place requires a slightly more sophisticated model setup than that required in linear models for normal data:

- The *systematic* component is a linear predictor similar to that in linear models, $\eta = \mathbf{x}'\boldsymbol{\beta}$. The linear predictor is a linear function in the parameters. In contrast to the linear model, η does not represent the mean function of the data.
- The *link function* $g(\cdot)$ relates the linear predictor to the mean, $g(\mu) = \eta$.
- The *random component* of a generalized linear model is the distribution of the data, which is assumed to be a member of the exponential family of distributions.

For more information about generalized linear models and their relationship to specific regression methods, see "Generalized Linear Models" (Chapter 3, *SAS/STAT User's Guide*) and "Generalized Linear Regression" (Chapter 4, *SAS/STAT User's Guide*).

For example, one of the most commonly used generalized linear regression models is the logistic model for binary or binomial data. Suppose that Y denotes a binary outcome variable that takes the values 1 and 0 with the probabilities π and $1 - \pi$, respectively. The probability π is also referred to as the "success probability," supposing that the coding $Y = 1$ corresponds to a success in a Bernoulli experiment. The success probability is also the mean of Y, and one of the aims of logistic regression analysis is to study how regressor variables affect the outcome probabilities or functions thereof, such as odds ratios.

The logistic regression model for π is defined by the linear predictor $\eta = \mathbf{x}'\boldsymbol{\beta}$ and the logit link function:

$$\text{logit}(\Pr(Y = 0)) = \log\left(\frac{\pi}{1 - \pi}\right) = \mathbf{x}'\boldsymbol{\beta}$$

The inversely linked linear predictor function in this model is

$$\Pr(Y = 0) = \frac{1}{1 + \exp(-\eta)}$$

The dichotomous logistic regression model can be extended to multinomial (polychotomous) data. You can fit two classes of models for multinomial data by using procedures in SAS/STAT software: models for ordinal data that rely on cumulative link functions, and models for nominal (unordered) outcomes that rely on generalized logits.

Many SAS/STAT procedures can fit generalized linear models. Two procedures that are especially useful for common analyses that involve nominal, ordinal, and count data are PROC LOGISTIC and PROC GENMOD.

Logistic Regression and Proportional Odds Models: The LOGISTIC Procedure

The LOGISTIC procedure fits logistic regression models and estimates parameters by maximum likelihood. The procedure fits the usual logistic regression model for binary data in addition to models that have the cumulative link function for ordinal data (such as the proportional odds model) and the generalized logit model for nominal data. PROC LOGISTIC offers a number of variable selection methods and can perform conditional and exact conditional logistic regression analysis.

Generalized Linear Models: The GENMOD Procedure

The GENMOD procedure is a generalized linear modeling procedure that estimates parameters by maximum likelihood. It uses CLASS and MODEL statements to form the statistical model and can fit models to binary and ordinal outcomes. PROC GENMOD does not fit generalized logit models for nominal outcomes. However, it can solve generalized estimating equations (GEE) to model correlated data and can perform a Bayesian analysis.

Overview of Power and Sample Size

This section reviews the concepts and terminology that you encounter in power analysis, including a clarification of prospective versus retrospective analyses and a breakdown of the components of a power analysis for a generalized linear model.

Power Analysis Concepts

Power analysis is what you use to get a “Goldilocks solution” for resource usage and study design, improving your chances of obtaining conclusive results with maximum efficiency. Power analysis is most effective when performed at the study planning stage, and therefore it encourages early collaboration between researcher and statistician. It also focuses attention on effect sizes and variability in the underlying scientific process, concepts that both researcher and statistician should consider carefully at this stage. Muller and Benignus (1992) and O’Brien and Muller (1993) cover these and related concepts. These references also provide a good general introduction to power analysis.

A power analysis involves many factors, such as the research objective, design, data analysis method, power, sample size, Type I error, variability, and effect size. By performing a power analysis, you can learn about the relationships among these factors, optimizing those that are under your control and exploring the implications of those that are not.

Power Analysis Terminology

In statistical hypothesis testing, you usually express the belief that some effect exists in a population by specifying an alternative hypothesis, H_1. You state a null hypothesis, H_0, as the assertion that the effect does *not* exist and attempt to gather evidence to reject H_0 in favor of H_1. You gather evidence in the form of sample data, and you perform a statistical test to assess H_0. If H_0 is rejected but there really is *no* effect, this is called a *Type I error*. The probability of a Type I error is usually designated as “alpha” or α, and statistical tests are designed to ensure that α is suitably small (for example, less than 0.05).

If there really is an effect in the population but H_0 is *not* rejected in the statistical test, then that’s a *Type II error*. The probability of a Type II error is usually designated as “beta” or β. The probability $1 - \beta$ of avoiding a Type II error—that is, of correctly rejecting H_0 and achieving statistical significance—is called the *power*. (Note, however, that another, more technical definition of power is the probability of rejecting H_0 for any given set of circumstances, even those that correspond to H_0 being true.)

An important goal in study planning is to ensure an acceptably high level of power. Sample size plays a prominent role in power computations, because the focus is often on determining a sufficient sample size to achieve a certain power, or conversely, on assessing the power for a range of different sample sizes. For this reason, terms like *power analysis*, *sample size analysis*, and *power computations* are often used interchangeably to refer to the investigation of relationships among power, sample size, and other factors involved in study planning.

Prospective versus Retrospective Power Analysis

It is crucial to distinguish between *prospective* and *retrospective* power analyses. As the names suggest, a prospective power analysis looks ahead to a future study, whereas a retrospective power analysis attempts to characterize a completed study. Sometimes the distinction is a bit fuzzy: for example, a retrospective analysis of a recently completed study can become a prospective analysis if it leads to the planning of a new study to address the same research objectives but with improved resource allocation.

Although a retrospective analysis is the most convenient type of power analysis to perform, it is often uninformative or misleading, especially when power is computed for the observed effect size. (For more information, see Lenth 2001.)

Power analysis is most effective when performed as part of study planning, and this paper considers only prospective power analysis.

Conditional versus Unconditional Power Analysis

A power analysis is *unconditional* with respect to the predictors if it fully accounts for their joint distribution. It is *conditional* on the predictors if it is based on a particular set of values—that is, a *fixed realization* of their joint distribution. Unconditional power is more desirable because it accounts entirely for uncertainty due to predictors, whereas conditional power loses accuracy as the predictors deviate from the fixed values that are assumed in the power analysis. A fully controlled experimental design automatically meets the criteria for an unconditional power analysis. But if any of the predictors is continuous, or discrete with an infinite or unreasonably large number of possible values, then it is impossible to conduct an unconditional power analysis without a power formula that accounts fully for the joint predictor distribution. Such a power formula is rarely available in practice for generalized linear models.

Even in cases where a fully unconditional power analyis is impossible to achieve, however, you can still perform an *approximately unconditional* power analysis by incorporating an approximation of the joint predictor distribution. Lyles, Lin, and Williamson (2007) provide a useful framework for conducting such an analysis for the generalized linear models under consideration in this paper.

Components of a Power Analysis

Power and sample size computations for generalized linear models present a somewhat greater level of complexity than simple hypothesis tests require. You need to perform a number of steps to gather the required information to perform these computations. After settling on a clear research question, you must (1) define the *study design*; (2) characterize the *joint predictor distribution* and the *response distribution*; (3) specify the *statistical tests* that will best address the research question; and (4) specify the *goal* as either a power or sample size computation. In hypothesis testing, you usually want to compute the powers for a range of sample sizes, or vice versa. All this work has strong parallels to ordinary data analysis.

Even when the research questions and study design seem straightforward, the ensuing sample size analysis for a generalized linear model can seem technically daunting. It is often helpful to break the process down into the aforementioned four components as follows:

- **Study design**

  What is the structure of the planned design? This must be clearly and completely specified. What groups, treatments, or factors will you assess, and what will be the relative sample sizes across their levels?

- **Predictor and response distributions**

  What are your beliefs about patterns in the data? What levels of signal do you suspect in these patterns, and against what background noise? Alternatively, what levels of signals are you interested in detecting?

- **Statistical tests**

How will you cast your model in statistical terms and conduct the eventual data analysis in order to embody the study design and test the effects central to your research question? Given that the primary approach discussed in this paper (the approach of Lyles, Lin, and Williamson 2007) accommodates Wald and likelihood ratio tests of effects and contrasts, what effects and contrasts do you plan to test? Do Wald or likelihood ratio tests suit your needs? What significance level will you use?

- **Goal**

  Finally, what do you need to determine in the power analysis? Most often you want to examine the statistical powers across various scenarios for the total sample size and for the relationship between the predictors and the response. Or you might want to find sample size values that provide given levels of power—say, 80%, 90%, or 95%. Are you interested in the presumed actual effect size or in minimal clinical significance or a cost/benefit ratio?

The “predictor and response distributions” component is crucial, and complicated. A key concept is that of an *exemplary data set*. This characterizes the design profiles, based on your plans or conjectures for the joint predictor distribution, as well as the response distribution for each design profile. It is also constructed to reflect relative allocation ratios of study factors and the probability of each response value for each design profile.

Usually you will conjecture the effects and variability inherent in the exemplary data set by making educated guesses of their true values. If instead you are interested in minimal clinical significance or a cost/benefit threshold, then you can specify values that produce an effect size representing this. However, this minimal effect size is often so small that it requires excessive resources to detect. Alternatively, you can consider a variety of realistic possibilities for the effect size by performing a sensitivity analysis: you can construct multiple exemplary data sets that capture competing views of predictor and response distributions. Your choice of strategy is ultimately determined by the goal of your power analysis.

Practical Generalized Linear Model Power Analysis

The framework in Lyles, Lin, and Williamson (2007) provides an effective strategy for using the CUSTOM statement to compute power or sample size for generalized linear models. There are alternative approaches to power analysis for generalized linear models, but they cover only a subset of response models, impose additional restrictions, require computationally challenging calculations, or rely on questionable approximations.

In contrast, the framework of Lyles, Lin, and Williamson (2007) has the very great advantage of condensing the approximation for unconditional power into a single computation that is basically the same regardless of the response distribution.

A summary of the process is as follows:

1. **Analysis plan**: Specify the SAS code that you would use to perform the data analysis.
2. **Exemplary data set**: Create an “exemplary data set” that resembles the data you expect to obtain.
3. **Exemplary analysis**: Run the analysis on the exemplary data set and extract the appropriate statistic for the test of interest.
4. **Primary noncentrality**: Divide the extracted statistic by the “effective sample size” of the exemplary data set to obtain the primary noncentrality value.
5. **Power computation**: Run PROC POWER with the CUSTOM statement, specifying the DIST=CHISQUARE option and setting the PRIMNC= option to the value that you calculated in step 4.

Step-by-Step Guidelines for the Method

This section is intended as a roadmap for conducting the power analysis from start to finish, using recommendations from Lyles, Lin, and Williamson (2007) and other sources that have been validated with simulation results and tend to work well for most use cases in practice.

The following suggested guidelines will help you split up the power analysis into intuitive, manageable steps:

1. **Analysis plan**: Specify the SAS code that you would use to perform the data analysis.

   a) Make sure that your chosen analysis produces a Wald or likelihood statistic, because the method of Lyles, Lin, and Williamson (2007) uses power approximations for those two statistics in particular.

   b) If possible, choose an implementation of the data analysis that supports a weight variable. Otherwise, see the end of this section for a discussion of alternatives.

2. **Exemplary data set**: Create an "exemplary data set" that resembles the data you expect to obtain.

   a) In creating the exemplary data set, it is often helpful to address three different types of predictors separately as follows and then combine the results together factorially.

      i. For predictors that you can control by design, specify all possible design profiles. Incorporate relative allocation ratios by duplicating each design profile the appropriate number of times.

      ii. For each additional predictor that is independent of other predictors, generate a representative sample. If you know (or can reasonably conjecture) its distribution function F, then a recommended method is to generate quantiles on a probability grid by using the Blom (1958) adjustment: $F^{-1}((i-3.75)/(n+0.25))$ for i in $1,\ldots,n$, where n is the number of quantiles.

      iii. For a correlated set of predictors, generate a representative sample from its conjectured joint distribution. If feasible, generate quantiles as in step 2a(ii) but on a multidimensional probability grid. Another option is to simulate a set of values, in which case Wicklin (2013) is a useful reference.

   b) When you're adding response values to the exemplary data set, if the set of response values is reasonably small, then include all of them; otherwise, choose a subset that covers most of the probability mass. Incorporate this variable by combining its values factorially with the predictor value scenarios that you generated in step 2a(iii).

   c) For the response probabilities, your strategy depends on your goal for the power analysis, as discussed in the section "Components of a Power Analysis" on page 4. In most cases you will want to compute the response probabilities by using the planned data analysis model and conjectured parameter values that represent your best educated guess of the true probabilities.

3. **Exemplary analysis**: Run the analysis on the exemplary data set and extract the appropriate statistic for the test of interest.

   a) If your power analysis is based on the Wald test, then extract the Wald statistic for the test of interest.

   b) Alternatively, if your power analysis is based on the likelihood ratio chi-square test, then run the data analysis twice—once using the full model and then again using the reduced model (lacking the predictor or predictors of interest). Extract the –2LogL statistic from each analysis and compute their absolute difference.

4. **Primary noncentrality**: Divide the extracted statistic by the "effective sample size" of the exemplary data set to obtain the primary noncentrality value.

   - Because you are using the response probability as a weight variable in the analysis, the effective sample size is equal to the grand sum of the response probabilities over the entire exemplary data set. If all possible response values are included with every design profile, then this number is also equal to the number of data set rows for each response value.

5. **Power computation**: Run PROC POWER with the CUSTOM statement, specifying the DIST=CHISQUARE option and setting the PRIMNC= option to the value that you calculated in step 4.

The Lyles, Lin, and Williamson (2007) method relies on the weight variable mechanism to fully represent the response distribution. But if you can't find a suitable data analysis implementation that supports weight variables, you could instead perform another round of row duplication such that the relative occurrence ratios of the response values for each design profile are as close as possible to their relative response probabilities. However, this can be extremely difficult to achieve without disrupting the relative occurrence ratios of the design profiles. For anything beyond the simplest design matrices and response distributions, you will end up in a futile struggle to satisfy least common multiples of too many row counts.

Why the CUSTOM Statement Method Works

This section explains *why* you can use SAS/STAT procedures for generalized linear models to do the computational heavy lifting in the power analysis. The moral of the story is that the Wald and likelihood ratio test statistics that these procedures provide are good estimators of a key ingredient in the power calculation: the *primary noncentrality*.

First, let's define some notation. Let Y be the response variable with J possible values $(y_1, \ldots, y_J)$. Let N be the total sample size (not to be confused with the number of exemplary data set rows), and let $\mathbf{X}$ be the $N \times q$ design matrix with rows $(\mathbf{x}_1, \ldots, \mathbf{x}_q)$. The J possible response values could be either the entire support for Y or a chosen subset whose probability mass is sufficiently close to one for each possible $\mathbf{x}_i$. Let $g(\cdot)$ be a known link function, and let $\boldsymbol{\beta}$ be the vector of regression coefficients. The hypothesis test is represented as

$$\begin{aligned} H_0&: \mathbf{L}\boldsymbol{\beta} = \boldsymbol{\theta}_0 \\ H_A&: \mathbf{L}\boldsymbol{\beta} \neq \boldsymbol{\theta}_0 \end{aligned}$$

where $\mathbf{L}$ is a $k \times q$ test or contrast matrix with full row rank and $\boldsymbol{\theta}_0$ is a $k \times 1$ custom null constant vector.

The Wald and likelihood ratio test statistics are computed as

$$T_{\mathrm{W}} = \left(\mathbf{L}\hat{\boldsymbol{\beta}} - \boldsymbol{\theta}_0\right)' \left[\mathbf{L}\widehat{\mathrm{Var}}(\hat{\boldsymbol{\beta}})\mathbf{L}'\right]^{-1} \left(\mathbf{L}\hat{\boldsymbol{\beta}} - \boldsymbol{\theta}_0\right)$$

and

$$T_{\mathrm{LR}} = -2\left[l\left(\hat{\boldsymbol{\beta}}^{\star}\right) - l\left(\hat{\boldsymbol{\beta}}\right)\right]$$

where $\hat{\boldsymbol{\beta}}$ is the unrestricted maximum likelihood estimate (MLE) of $\boldsymbol{\beta}$, $\hat{\boldsymbol{\beta}}^{\star}$ is the corresponding restricted MLE under H_0, and $l(\cdot)$ denotes the log-likelihood function.

The distribution of T_{W} is well approximated by a noncentral chi-square distribution with k degrees of freedom and primary noncentrality

$$\lambda^{\star}_{\mathrm{W}} = \frac{1}{N}\left(\mathbf{L}\boldsymbol{\beta} - \boldsymbol{\theta}_0\right)' \left[\mathbf{L}\mathrm{Var}(\hat{\boldsymbol{\beta}})\mathbf{L}'\right]^{-1} \left(\mathbf{L}\boldsymbol{\beta} - \boldsymbol{\theta}_0\right)$$

Similarly, the distribution of T_{LR} is well approximated by a noncentral chi-square distribution with k degrees of freedom and primary noncentrality

$$\lambda^{\star}_{\mathrm{LR}} = \frac{1}{N}\left(-2\left[l^{\star}\left(\boldsymbol{\beta}\right) - l\left(\boldsymbol{\beta}\right)\right]\right)$$

The term "plug-in estimator" is often used to characterize statistics such as T_{W} and T_{LR} because they are equivalent to the parameters that they are estimating, except with unknown portions replaced by their estimates.

Letting $\lambda^{\star}$ denote either primary noncentrality, the power is computed as

$$\mathrm{power} = P\left(\chi^2(k, N\lambda^{\star}) \geq \chi^2_{1-\alpha}(k)\right)$$

The trick of dividing the statistic by the "effective sample size" of the exemplary data set and expressing power in terms of *primary* noncentrality isn't mentioned by Lyles, Lin, and Williamson (2007), but it enables a simple solution for the sample size N by inversion of the power formula. Thus you can solve for N directly without the need for iteration.

Table 1 summarizes how the CUSTOM statement syntax elements correspond to the notation discussed in this section for power analysis for generalized linear models based on the method of Lyles, Lin, and Williamson (2007).

Table 1 CUSTOM Statement Options for Generalized Linear Models

Option	Description	Symbol
DIST=CHISQUARE	Noncentral chi-square test statistic distribution	$\chi^2(k, N\lambda^{\star})$
PRIMNC=	Primary noncentrality	$\lambda^{\star}$
TESTDF=	Test degrees of freedom	k
ALPHA=	Significance level	α
NTOTAL=	Total sample size, or '.' to compute	N
POWER=	Desired power, or '.' to compute	power

Example: Logistic Regression with a Classification Variable and Correlated Predictors

Let's look back at the example in the Prologue, where you were concerned with creating an efficient study of steel processing, one that is as small as possible while still able to detect a given rise in the probability of producing a good product. The basic setup is shown in Table 2.

Table 2 Basic Setup for Logistic Regression Power Analysis

		Values
Study factors	Heating time	5, 10, 15, or 20 minutes
	Supplier	A, B, or C
	Ingot mass	Continuous
Outcome	Ingot not ready for rolling	Binary, 0 or 1

Furthermore, there are dependencies between the factors:

- There are twice as many ingots from supplier A as from either supplier B or C.
- Ingots are allocated to heating times differently for different suppliers: 2:3:3:2 for supplier A, 1:2:3:4 for supplier B, and 4:3:2:1 for supplier C.
- Ingot mass also varies for different suppliers, normally distributed with mean and standard deviation (in kg): (4, 2) for supplier A, (4.5, 2.2) for supplier B, and (3.9, 1.9) for supplier C.

The data analysis will be a Wald or likelihood ratio chi-square test of the main **Heat** effect in a logistic regression with a model that also includes the main **Supplier** and **Mass** effects.

For the power analysis, you also need to make some guesses about the underlying process that you are studying. Here, the odds ratio for a unit increase in **Mass** is assumed to be about 1.1, and the average probability of an ingot not being ready for rolling is assumed to be about 20%.

Finally, the study needs to include enough ingots to have a 95% chance of detecting a **Heat** odds ratio as small as 1.2 for a five-minute increase at a 0.05 significance level.

So that's the study that you're designing, for which you need to figure out the proper number of ingots to sample. As noted in the Prologue, it is primarily the dependencies between the factors that put this study beyond the range of the LOGISTIC statement in PROC POWER. However, because binary logistic regression fits within the framework of Lyles, Lin, and Williamson (2007), you can follow the steps in the section "Practical Generalized Linear Model Power Analysis" on page 5 and use the CUSTOM statement instead.

Step 1: Analysis Plan

You plan to use the following SAS code for the logistic regression:

```
proc logistic descending;
   class Supplier;
   weight PY;
   model Y = Supplier Heat Mass;
run;
```

Step 2: Exemplary Data

The next step is to create an initial exemplary data set that contains just the predictor variables. Conceptually, you can think of the exemplary data set starting out as a core of 4 rows—two for supplier A and one each for suppliers B and C—to satisfy their 2:1:1 allocation ratios. Then expand this to 40 rows by incorporating the **Heat** variable, making 10 copies of each original row to satisfy the allocation ratios of **Heat** for each **Supplier**. The count of 10 comes from the sum of each set of **Heat** allocation ratios: 2:3:3:2 for supplier A, 1:2:3:4 for supplier B, and 4:3:2:1 for supplier C. Finally, add the **Mass** variable to the mix by incorporating sets of any number of quantiles. You can use a large number of quantiles for better accuracy or a smaller number to spare computing resources; let's use 100 quantiles per sample. So you expand 40 rows to 4,000 by generating a sample of 100 quantiles for each row according to its supplier's **Mass** mean and standard deviation.

The following SAS code creates this initial exemplary data set of predictor variables, containing all possible design profiles. The number of quantiles for the **Mass** sample is set to 100, and the allocation ratios are incorporated by duplicating each profile the appropriate number of times.

```
%let nQ = 100;

data Exemplary;
   do Supplier = "A", "B", "C";

      * Mass distribution and heat allocation for each supplier;
      select (Supplier);
         when ('A') do; MeanStd = "4   2  "; Alloc = "4 6 6 4"; end;
         when ('B') do; MeanStd = "4.5 2.2"; Alloc = "1 2 3 4"; end;
         when ('C') do; MeanStd = "3.9 1.9"; Alloc = "4 3 2 1"; end;
         end;

      _meanMass = 1*scan(MeanStd,1,' ');
      _stdMass  = 1*scan(MeanStd,2,' ');

      do Heat = 5, 10, 15, 20;

         select (Heat);
            when ( 5) _alloc = scan(Alloc,1,' ');
            when (10) _alloc = scan(Alloc,2,' ');
            when (15) _alloc = scan(Alloc,3,' ');
            when (20) _alloc = scan(Alloc,4,' ');
            end;

         * Representative Mass sample: evenly spaced quantiles;
         do _iQ = 1 to &nQ;
            Mass = _meanMass + _stdMass *
                   quantile('normal', (_iQ-.375)/(&nQ+.25));
            do _jAlloc = 1 to _alloc;
               output;
            end;
         end;
      end;
   end;
run;
```

The next step is trivial for logistic regression. It's just a matter of splitting each row into two copies according to the two possible response values (1 and 0), as in the following SAS code:

```
data Exemplary;
   set Exemplary;
   do Y = 1, 0;
      output;
   end;
run;
```

For response probabilities, you can transform the conjectured odds ratios into logistic regression coefficients and then compute expected response values according to the logistic model equation. The coefficients are computed as

$$\Psi_0 = \log\left(\frac{\phi}{1-\phi}\right) - \boldsymbol{\Psi}'\boldsymbol{\mu}$$

$$\Psi_j = \frac{\log(\mathrm{OR}_j)}{U_j}$$

where

$\boldsymbol{\mu}$ = mean predictor vector
ϕ = response probability at predictor means
U_j = unit change for jth predictor
OR_j = odds ratio for jth predictor
Ψ_0 = intercept in full model
$\boldsymbol{\Psi}$ = nonintercept regression coefficients in full model

The **Heat** and **Mass** means are weighted means according to the relative allocation ratios. The following DATA step computes these probabilities as values of a variable **PY**:

```
data Exemplary; set Exemplary;

   ORHeat    = 1.2;
   UnitsHeat = 5;
   CoeffHeat = log(ORHeat)/UnitsHeat;
   MeanHeat  = (5*9 + 10*11 + 15*11 + 20*9)/40;

   ORMass    = 1.1;
   UnitsMass = 1;
   CoeffMass = log(ORMass)/UnitsMass;
   MeanMass = (4*2 + 4.5 + 3.9)/4;

   ResponseProb = 0.2;
   Intercept = log(ResponseProb/(1-ResponseProb))
                - CoeffHeat*MeanHeat - CoeffMass*MeanMass;

   _pi = logistic (Intercept + Heat*CoeffHeat + Mass*CoeffMass);
   if (Y = 1) then PY =     _pi;
   else            PY = 1 - _pi;
   keep Supplier Heat Mass Y PY;
run;
```

The following SAS code shows a sample of the exemplary data set in Figure 1:

```
proc print data=Exemplary (obs=10);
run;
proc print data=Exemplary (firstobs=7991 obs=8000);
run;
```

Figure 1 First and Last 10 Observations of Exemplary Data Set

Obs	Supplier	Heat	Mass	Y	PY
1	A	5	-0.99718	1	0.10474
2	A	5	-0.99718	0	0.89526
3	A	5	-0.99718	1	0.10474
4	A	5	-0.99718	0	0.89526
5	A	5	-0.99718	1	0.10474
6	A	5	-0.99718	0	0.89526
7	A	5	-0.99718	1	0.10474
8	A	5	-0.99718	0	0.89526
9	A	5	-0.27841	1	0.11134
10	A	5	-0.27841	0	0.88866

Figure 1 *continued*

Obs	Supplier	Heat	Mass	Y	PY
7991	C	20	7.09874	1	0.30428
7992	C	20	7.09874	0	0.69572
7993	C	20	7.31450	1	0.30865
7994	C	20	7.31450	0	0.69135
7995	C	20	7.58617	1	0.31420
7996	C	20	7.58617	0	0.68580
7997	C	20	7.96449	1	0.32202
7998	C	20	7.96449	0	0.67798
7999	C	20	8.64732	1	0.33639
8000	C	20	8.64732	0	0.66361

Step 3: Exemplary Analysis

Use the following SAS code to run the analysis on the exemplary data set, fitting the full model with **PY** as the weight variable and using ODS to gather the relevant output:

```
proc logistic data=Exemplary descending;
   ods output parameterestimates=WaldAndDF fitstatistics=LogLFull;
   class Supplier;
   weight PY;
   model Y = Supplier Heat Mass;
run;
```

The following two DATA steps extract the Wald chi-square and the likelihood statistic for the test of **Heat**:

```
data WaldAndDF; set WaldAndDF;
   where Variable = "Heat";
   keep DF WaldChiSq;
run;

data LogLFull; set LogLFull;
   where Criterion = "-2 Log L";
   Neg2LogLFull = InterceptAndCovariates;
   keep Neg2LogLFull;
run;
```

A likelihood ratio test requires the likelihood statistics from both the full and reduced models. Thus, you need to run the analysis again on the reduced model and fetch the other likelihood statistic that is needed for the likelihood ratio test, as follows:

```
proc logistic data=Exemplary descending;
   ods output fitstatistics=LogLRed;
   class Supplier;
   weight PY;
   model Y = Supplier Mass;
run;

data LogLRed; set LogLRed;
   where Criterion = "-2 Log L";
   Neg2LogLRed = InterceptAndCovariates;
   keep Neg2LogLRed;
run;
```

Step 4: Primary Noncentrality

Now you've got enough information to compute the primary noncentralities that you'll use in the CUSTOM statement. Run the following SAS code to divide the extracted statistics by the effective sample size of the exemplary data set, $40 \times$ **nQ**, to obtain the primary noncentrality values for both the Wald and likelihood ratio tests:

```
data PrimNC;
   merge WaldAndDF LogLFull LogLRed;
   WaldPrimNC = WaldChiSq                   / (40 * &nQ);
   LRPrimNC   = -(Neg2LogLFull-Neg2LogLRed) / (40 * &nQ);
   keep DF WaldPrimNC LRPrimNC;
   call symputx ("DF"        , DF        );
   call symputx ("WaldPrimNC", WaldPrimNC);
   call symputx ("LRPrimNC"  , LRPrimNC  );
run;
```

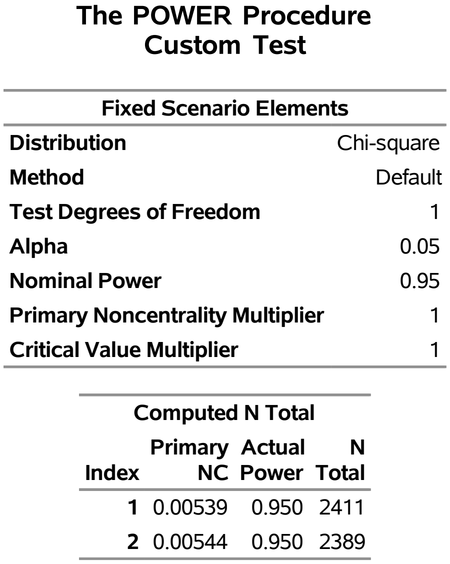
Step 5: Power Computation

Finally, specify the following SAS code to run PROC POWER with the CUSTOM statement to calculate the total number of ingots required for your experiment, specifying the DIST=CHISQUARE option and setting the PRIMNC= option to the values calculated in step 4:

```
proc power;
   custom
      dist   = chisquare
      primnc = &WaldPrimNC &LRPrimNC
      testdf = &DF
      ntotal = .
      alpha  = 0.05
      power  = 0.95;
run;
```

The results in Figure 2 show computed sample sizes of 2,411 ingots for the Wald test and 2,389 ingots for the likelihood ratio chi-square test.

Figure 2 Sample Size for Logistic Regression Computed with CUSTOM Statement



```
                 The POWER Procedure
                     Custom Test

             Fixed Scenario Elements
Distribution                           Chi-square
Method                                    Default
Test Degrees of Freedom                         1
Alpha                                        0.05
Nominal Power                                0.95
Primary Noncentrality Multiplier                1
Critical Value Multiplier                       1

                Computed N Total
             Primary  Actual      N
       Index      NC   Power  Total
           1 0.00539   0.950   2411
           2 0.00544   0.950   2389
```

Steps for a Poisson Regression Alternative

Finally, to see how general and versatile this method of Lyles, Lin, and Williamson (2007) is when implemented with the CUSTOM statement in PROC POWER, suppose that instead of a binary outcome (whether or not the ingot was ready for rolling), the outcome of interest is the number of defects in the rolled ingot, with a Poisson distribution that depends on **Supplier**, **Heat**, and **Mass**. To perform power and sample size analysis for this new setup, all that you need to change is the following:

1. How the predicted response **PY** is computed in the exemplary data setup:

```
      _lambda_i = exp(Intercept + Heat*CoeffHeat + Mass*CoeffMass +
                      SupplierA*CoeffSupplierA + SupplierB*CoeffSupplierB);
      PY = (exp(-_lambda_i) * _lambda_i**Y)/gamma(Y+1);
```

2. How the relevant statistics are computed and extracted in the exemplary value analysis: instead of PROC LOGISTIC, use PROC GENMOD with the DIST=POISSON option, and retrieve the statistics from the ParameterEstimates and ModelFit tables.

Similar changes enable you to perform power analysis for any generalized linear model with a noncontinuous response.

Conclusion and Extensions

The process that is outlined in the section "Step-by-Step Guidelines for the Method" on page 5, based on the combination of the method of Lyles, Lin, and Williamson (2007) and the CUSTOM statement in PROC POWER, provides a practical and flexible strategy for conducting a power analysis for any Wald or likelihood ratio test for a generalized linear model that has nominal, count, or ordinal responses. The first three steps in the process (analysis plan, exemplary data set, and exemplary analysis) vary with the choice of link function and response distribution, but the last two steps (primary noncentrality and power computation) are independent of those choices and thus easily reproducible.

In case you want to perform a power analysis for a test other than Wald or likelihood ratio for a generalized linear model, there are other well-established power approximations that you can combine with other test statistic distributions that the CUSTOM statement supports. For example, you can use power approximations from O'Brien and Shieh (1992) or Muller and Peterson (1984) and the DIST=F option in the CUSTOM statement to perform power analyses for multivariate general linear models for normal data. The GLMPOWER procedure is based on those same approximations, but the CUSTOM statement in PROC POWER offers some additional tools that are not available in PROC GLMPOWER. For example:

- custom degrees-of-freedom values to accommodate, for example, additional covariates or alternative degrees-of-freedom approximations (MODELDF= option)
- sample size inflation or deflation due to correlated predictors or correlation between covariates and response, respectively (PNCMULT= option)

The CUSTOM statement also supports several other test statistic distributions: normal (DIST=NORMAL option), noncentral t (DIST=T option), and the correlation coefficient distribution for multivariate normal data (DIST=CORR option). For more information about the CUSTOM statement, see Chapter 92, "The POWER Procedure" (*SAS/STAT User's Guide*), in SAS Institute Inc. (2017).

You can find power analysis examples with SAS code for other types of generalized linear models based on the method of Lyles, Lin, and Williamson (2007) in the following publications:

- Poisson regression: Lyles, Lin, and Williamson (2007), section 3.3
- proportional odds model: Lyles, Lin, and Williamson (2007), section 3.6
- negative binomial regression: Lyles, Lin, and Williamson (2007), section 3.1
- zero-inflated Poisson regression (ZIP): Doyle (2009), pp. 363–376, and Williamson et al. (2007), section 3.1
- zero-inflated negative binomial regression (ZINB): Doyle (2009), pp. 363–376

For complete details about the syntax of PROC POWER, PROC LOGISTIC, PROC GENMOD, and other procedures that deal with generalized linear models, see SAS Institute Inc. (2017).

REFERENCES

Blom, G. (1958). *Statistical Estimates and Transformed Beta Variables*. New York: John Wiley & Sons.

Doyle, S. R. (2009). “Examples of Computing Power for Zero-Inflated and Overdispersed Count Data.” *Journal of Modern Applied Statistical Methods* 8:360–376.

Lenth, R. V. (2001). “Some Practical Guidelines for Effective Sample Size Determination.” *American Statistician* 55:187–193.

Lyles, R. H., Lin, H.-M., and Williamson, J. M. (2007). “A Practical Approach to Computing Power for Generalized Linear Models with Nominal, Count, or Ordinal Responses.” *Statistics in Medicine* 26:1632–1648.

McCullagh, P., and Nelder, J. A. (1989). *Generalized Linear Models*. 2nd ed. London: Chapman & Hall.

Muller, K. E., and Benignus, V. A. (1992). “Increasing Scientific Power with Statistical Power.” *Neurotoxicology and Teratology* 14:211–219.

Muller, K. E., and Peterson, B. L. (1984). “Practical Methods for Computing Power in Testing the Multivariate General Linear Hypothesis.” *Computational Statistics and Data Analysis* 2:143–158.

Nelder, J. A., and Wedderburn, R. W. M. (1972). “Generalized Linear Models.” *Journal of the Royal Statistical Society, Series A* 135:370–384.

O’Brien, R. G., and Muller, K. E. (1993). “Unified Power Analysis for *t*-Tests through Multivariate Hypotheses.” In *Applied Analysis of Variance in Behavioral Science*, edited by L. K. Edwards, 297–344. New York: Marcel Dekker.

O’Brien, R. G., and Shieh, G. (1992). “Pragmatic, Unifying Algorithm Gives Power Probabilities for Common *F* Tests of the Multivariate General Linear Hypothesis.” Poster presented at the American Statistical Association Meetings, Statistical Computing Section, Boston.

SAS Institute Inc. (2017). *SAS/STAT 14.3 User’s Guide*. Cary, NC: SAS Institute Inc. http://go.documentation.sas.com/?docsetId=statug&docsetTarget=titlepage.htm&docsetVersion=14.3&locale=en.

Wedderburn, R. W. M. (1974). “Quasi-likelihood Functions, Generalized Linear Models, and the Gauss-Newton Method.” *Biometrika* 61:439–447.

Wicklin, R. (2013). *Simulating Data with SAS*. Cary, NC: SAS Institute Inc.

Williamson, J. M., Lin, H.-M., Lyles, R. H., and Hightower, A. W. (2007). “Power Calculations for ZIP and ZINB Models.” *Journal of Data Science* 5:519–534.

Acknowledgments

The authors are grateful to Randy Tobias, Ed Huddleston, and Tim Arnold of the Advanced Analytics Division at SAS Institute, and to Jill Tao of the Technical Support Division at SAS, for their valuable assistance in the preparation of this paper.

Contact Information

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

John Castelloe
SAS Institute Inc.
SAS Campus Drive
Cary, NC 27513
919-531-5728
john.castelloe@sas.com

```
******************************************************************
* SGF '18 Paper 1983-2018 - Power Analysis for Generalized Linear *
* Models Using the New CUSTOM Statement in PROC POWER -          *
* Sample Code                                                    *
******************************************************************;

%let nQ = 100;

data Exemplary;
  do Supplier = "A", "B", "C";

    * Mass distribution and heat allocation for each supplier;
    select (Supplier);
      when ('A') do; MeanStd = "4   2  "; Alloc = "4 6 6 4"; end;
      when ('B') do; MeanStd = "4.5 2.2"; Alloc = "1 2 3 4"; end;
      when ('C') do; MeanStd = "3.9 1.9"; Alloc = "4 3 2 1"; end;
      end;

    _meanMass = 1*scan(MeanStd,1,' ');
    _stdMass  = 1*scan(MeanStd,2,' ');

    do Heat = 5, 10, 15, 20;

      select (Heat);
        when ( 5) _alloc = scan(Alloc,1,' ');
        when (10) _alloc = scan(Alloc,2,' ');
        when (15) _alloc = scan(Alloc,3,' ');
        when (20) _alloc = scan(Alloc,4,' ');
        end;

      * Representative Mass sample: evenly spaced quantiles;
      do _iQ = 1 to &nQ;
        Mass = _meanMass + _stdMass *
             quantile('normal', (_iQ-.375)/(&nQ+.25));
        do _jAlloc = 1 to _alloc;
          output;
        end;
      end;
    end;
  end;
run;

data MassHists;
  do Supplier = "A", "B", "C";
    select (Supplier);
      when ('A') do; MeanStd = "4   2  "; end;
      when ('B') do; MeanStd = "4.5 2.2"; end;
      when ('C') do; MeanStd = "3.9 1.9"; end;
      end;
    _meanMass = 1*scan(MeanStd,1,' ');
    _stdMass  = 1*scan(MeanStd,2,' ');
    do _iQ = 1 to &nQ;
      Mass = _meanMass + _stdMass *
```

```
                quantile('normal', (_iQ-.375)/(&nQ+.25));
         output;
      end;
   end;
run;
proc sgpanel data=MassHists;
   panelby Supplier / columns=1 uniscale=all;
   histogram Mass;
run;

data Exemplary;
   set Exemplary;
   do Y = 1, 0;
      output;
   end;
run;
data Exemplary; set Exemplary;

   ORHeat    = 1.2;
   UnitsHeat = 5;
   CoeffHeat = log(ORHeat)/UnitsHeat;
   MeanHeat  = (5*9 + 10*11 + 15*11 + 20*9)/40;

   ORMass    = 1.1;
   UnitsMass = 1;
   CoeffMass = log(ORMass)/UnitsMass;
   MeanMass = (4*2 + 4.5 + 3.9)/4;

   ResponseProb = 0.2;

   Intercept = log(ResponseProb/(1-ResponseProb))
                  - CoeffHeat*MeanHeat - CoeffMass*MeanMass;
   _pi = logistic (Intercept + Heat*CoeffHeat + Mass*CoeffMass);
   if (Y = 1) then PY =     _pi;
   else            PY = 1 - _pi;
   keep Supplier Heat Mass Y PY;
run;
proc print data=Exemplary (obs=10);
run;
proc print data=Exemplary (firstobs=7991 obs=8000);
run;
proc logistic data=Exemplary descending;
   ods output parameterestimates=WaldAndDF fitstatistics=LogLFull;
   class Supplier;
   weight PY;
   model Y = Supplier Heat Mass;
run;
data WaldAndDF; set WaldAndDF;
   where Variable = "Heat";
   keep DF WaldChiSq;
run;
data LogLFull; set LogLFull;
   where Criterion = "-2 Log L";
   Neg2LogLFull = InterceptAndCovariates;
   keep Neg2LogLFull;
run;
proc logistic data=Exemplary descending;
   ods output fitstatistics=LogLRed;
   class Supplier;
   weight PY;
   model Y = Supplier Mass;
run;
data LogLRed; set LogLRed;
   where Criterion = "-2 Log L";
   Neg2LogLRed = InterceptAndCovariates;
   keep Neg2LogLRed;
run;
data PrimNC;
   merge WaldAndDF LogLFull LogLRed;
   WaldPrimNC = WaldChiSq                   / 4000;
   LRPrimNC   = -(Neg2LogLFull-Neg2LogLRed) / 4000;
   keep DF WaldPrimNC LRPrimNC;
   call symputx ("DF"        , DF        );
   call symputx ("WaldPrimNC", WaldPrimNC);
   call symputx ("LRPrimNC"  , LRPrimNC  );
run;
proc power;
   custom
      dist   = chisquare
      primnc = &WaldPrimNC &LRPrimNC
      testdf = &DF
      ntotal = .
      alpha  = 0.05
      power  = 0.95;
run;
```