POWERLIB: SAS/IML Software for Computing Power in Multivariate Linear Models, Version 2.2

Jacqueline L. Johnson
University of North Carolina at Chapel Hill

Yueh-Yun Chi
University of Florida

Keith E. Muller
University of Florida

Authors of previous versions:

Matthew J. Gribbin
MedImmune, LLC.

Matthew J. Gurka
University of West Virginia

Lynette L Keyes-Elstein
Rho, Inc.

Bercedis L. Peterson
Duke University

Sean L. Simpson
Wake Forest University

James C. Slaughter
Vanderbilt University

Abstract

The **POWERLIB** SAS/IML software provides convenient power calculations for a wide range of multivariate linear models with Gaussian errors. The software includes the Box, Geisser-Greenhouse, Huynh-Feldt, and uncorrected tests in the "univariate" approach to repeated measures (UNIREP), the Hotelling Lawley Trace, Pillai-Bartlett Trace, and Wilks Lambda tests in "multivariate" approach (MULTIREP), as well as a limited but useful range of mixed models. The familiar univariate linear model with Gaussian errors is an important special case. Power can be computed for all tests based on known or estimated covariance. For estimated covariance, the software provides confidence limits for the resulting estimated power. For UNIREP tests with known covariance, the software can compute power in the context of an internal pilot design. Power for a UNIREP test in the context of high dimension, low sample size data can be computed assuming known or estimated covariance. All power and confidence limits values can be output to a SAS dataset, which can be used to produce plots and tables for manuscripts.

Keywords: power, multivariate linear models, mixed models, Gaussian errors, SAS, internal pilots, high dimension, low sample size.

1. Description of POWERLIB

POWERLIB is a suite of SAS/IML (SAS Institute 2013b) modules which computes statistical power for hypothesis tests in a wide variety of univariate, multivariate, and repeated measures linear models with Gaussian errors and fixed predictors. This paper describes version 2.2 of **POWERLIB**. This version was developed using SAS version 9.3 (SAS Institute 2013a) and can be run on both Windows and UNIX systems. The code includes matrix names longer than eight characters, a naming convention added in version 7; hence, the modules will not run in SAS versions 6.12 or earlier.

1.1. Available models and hypothesis tests

POWERLIB computes power for the four tests commonly used for the "univariate" approach to repeated measures: Box, Geisser-Greenhouse, Huynh-Feldt, and uncorrected. The four tests and associated methods will be collectively referred to as the UNIREP approach.

The program also computes power for the three most popular multivariate test statistics: Hotelling Lawley Trace, Pillai-Bartlett Trace, and Wilks Lambda. In this manual, these three tests and related methods are collectively referred to as the MULTIREP approach.

Power for both UNIREP and MULTIREP tests can be computed assuming known or estimated covariance. Power for UNIREP tests with known covariance can be computed within the context of an interal pilot study. Power for UNIREP tests with either known or estimated covariance can be computed for high dimension, low sample size (HDLSS) data via a new modified version of the Huynh-Felt test (CHI MULLER REF).

The UNIREP approach is equivalent to a restricted class of linear mixed models with Gaussian errors that meet the following restrictions (Gurka, Coffey, and Muller 2007):

1. no missing or mistimed observations (all subjects have the same number of observations at the same within-subject levels),
2. factorial within-subject design,
3. common between-subject design for all responses,
4. homogeneity of covariance parameters for all subjects, and
5. compound symmetric covariance structure.

The increasing popularity of using mixed models for data analysis naturally motivates the need for reliable power analysis based on mixed models. Unfortunately, as Verbeke and Molenbergs (2000) noted, very little is known about non-null distributions in mixed models. Consequently, there is no known software that dependably calculates power for general mixed models. Use of the methods proposed by Gurka *et al.* (2007) within **POWERLIB** allows for power analysis for a restricted class of linear mixed models. Though the required restrictions are not often met in practice, a power analysis based on this restricted class of mixed models may provide reasonable guidelines for sample size, even in general scenarios (e.g, missing data). Therefore, **POWERLIB** is a valuable tool for researchers who need to plan studies in which mixed models will be fit to the collected data.

The familiar univariate linear model with Gaussian errors is a special case of the multivariate and mixed model formulation. The program's output and syntax simplifies for univariate model power.

1.2. Model and hypothesis notation

Power computations in **POWERLIB** are derived within the framework of the general linear multivariate model (GLMM). Muller and Stewart (2006) and Timm (2002) provided detailed discussion of basic theory and practice for all models treated by **POWERLIB**, with the former focusing more on theory and the latter more on practice. For N independent sampling units, p responses, and q predictors, the GLMM may be stated as:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{B} + \boldsymbol{E} \ ,$$

with $\boldsymbol{Y}$ $(N \times p)$ containing information on random responses such as repeated measures, $\boldsymbol{X}$ $(N \times q)$ a fixed design matrix containing predictors, $\boldsymbol{B}$ $(q \times p)$ containing unknown parameters, and $\boldsymbol{E}$ $(N \times p)$ containing random errors. Rows of $\boldsymbol{Y}$, $\boldsymbol{X}$, and $\boldsymbol{E}$ correspond to independent sampling units (such as subjects), columns of $\boldsymbol{Y}$, $\boldsymbol{B}$, and $\boldsymbol{E}$ to levels of the multivariate response (often time), and columns of $\boldsymbol{X}$ and rows of $\boldsymbol{B}$ to predictors. With $r = \text{rank}(\boldsymbol{X})$, the methods used in this program assume $N > r$, $\text{row}_i\left(\boldsymbol{E}\right) \sim N_p\left(\boldsymbol{0}, \boldsymbol{\Sigma}\right)$, $i \in \{1, 2, \ldots, N\}$, all rows independent, and no missing data in $\boldsymbol{Y}$ or $\boldsymbol{X}$.

In the context of data analysis, values in $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ are estimated. In the context of power analysis, $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ are assumed to be known constants; however, in practice, estimates of $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ from a previous study are often used. The software can compute point estimates and confidence limits for power in order to reflect the uncertainty in the estimation of $\boldsymbol{\Sigma}$ and, in some cases, also in $\boldsymbol{B}$.

The corresponding general linear hypothesis (GLH) involves $\boldsymbol{\Theta} = \boldsymbol{C}\boldsymbol{B}\boldsymbol{U}$, with $\boldsymbol{C}$ an $a \times q$ matrix of known constants defining "between-subject" contrasts, and $\boldsymbol{U}$ a $p \times b$ matrix of known constants defining "within-subject" contrasts. The power program requires $\text{rank}(\boldsymbol{C}) = a \leq q$ and $\text{rank}(\boldsymbol{U}) = b \leq p$. The GLH is

$$\begin{aligned} H_0 &: \boldsymbol{\Theta} = \boldsymbol{\Theta}_0 \\ H_1 &: \boldsymbol{\Theta} \neq \boldsymbol{\Theta}_0 \ . \end{aligned}$$

Most often $\boldsymbol{\Theta}_0 = \boldsymbol{0}$.

1.3. Statistical theory

Detailed knowledge of the statistical theory behind power computations in multivariate linear models is not required to use this software; however, for more sophisticated users, **POWERLIB** provides options for choosing distributional approximations. Defaults have been chosen to reflect methods the authors believe to be the best available and should not be altered without explicit rationale. Muller, Lavange, Ramey, and Ramey (1992) gave a review of the theory behind the power methods implemented. Muller and Benignus (1992) provided a brief introduction to the most basic ideas of power in the context of toxicology, while O'Brien and Muller (1993) provided a lengthy tutorial in linear models power. Additionally, sections 2.9 - 2.12 give several references for various distributional approximations for UNIREP and MULTIREP tests, their confidence limits, application to internal pilot studies, and use with high dimension, low sample size data.

1.4. Why use this software?

Commercial software for computing power in linear models with Gaussian errors is available, most notably **NQuery** (Statistical Solutions 2013), **PASS** (NCSS 2013), and `PROC GLMPOWER` (SAS Institute 2013c). O'Brien (2003), Heitjan (2013), Hedeker, Gibbons, and Waternaux

(1999) and Spybrook, Raudenbush, Liu, Congdon, and Martinez (2011) also provide additional useful free software. **POWERLIB** currently has the following advantages over these and other products:

1. **POWERLIB** implements the power approximations described in Muller, Edwards, Simpson, and Taylor (2007), which has considerably better test size accuracy in some cases.
   (a) `PROC GLMPOWER` from SAS does not compute power for multivariate Gaussian models, only univariate.
   (b) **PASS** implements older multivariate power approximations from Muller and Barton (1989).
2. **POWERLIB** is easy to embed in other SAS/IML code for use in simulations.
3. **POWERLIB** is the only software with the following features:
   (a) It computes point estimates for power based on estimated covariance.
   (b) It computes confidence limits for power in Gaussian linear models.
   (c) It computes power within the context of an internal pilot design.
   (d) It computes power for high dimension, low sample size data.

2. How to use POWERLIB

2.1. Execution

The `POWERLIB22.IML` file included in the distribution contains several modules and is the only file required to run **POWERLIB**. The `POWER` module performs all power calculations. All other modules included in the `POWERLIB22.IML` file, except for some independent utility modules discussed later in this paper, are called by the `POWER` module and are transparent to the user.

With basic knowledge of SAS/IML, **POWERLIB** is simple to use via the general program skeleton given in Table 1. The first two program statements are always required. They initialize IML, ask for extra symbol space, and make the power modules available for use. The `RUN POWER` statement executes the `POWER` module and is also required. Note that the directory listed in the `%INCLUDE` statement is the directory where `POWERLIB22.IML` has been copied; this, most likely, must be modified from the skeleton, found in Table 1.

The software can also be run from within another module by using a `CALL` statement. This feature is particularly helpful when running simulations. Table 2 illustrates how to define a user module that calls the power software.

2.2. Inputs overview

User inputs to the **POWERLIB** modules are made through 28 global matrices listed in Tables 3 and 4, grouped by application. Sections 2.4–2.13 discuss how to use these matrices in the order the groups are presented in Tables 3 and 4. Because these matrices are global, use of these

```
PROC IML SYMSIZE=2000;
%INCLUDE "..\IML\POWERLIB22.IML"/NOSOURCE2;

...programming statements to
assign global matrices that
describe model and choose options...

RUN POWER;
QUIT;
```

Table 1: Basic program skeleton.

```
START user_defined_module ( parm1, parm2, ... parmn)
        GLOBAL (ESSENCEX, SIGMA, BETA, C, U, THETA0, REPN, BETASCAL, SIGSCAL,
                RHOSCAL, ALPHA, ROUND, TOLERANCE, TOLERANC, UCDF, UMETHOD,
                MMETHOD, IP_PLAN, N_IP, RANK_IP, LIMCASE, CLTYPE, N_EST,
                RANK_EST, ALPHA_CL, ALPHA_CU, SIGTYPE DSNAME, OPT_ON,
                OPT_OFF);

...your code here...
...programming statements to assign POWERLIB global matrices...

CALL POWER(_HOLDPOWER, _HOLDPOWERLBL, _POWERWARN, _POWERWARNLBL);

...some more of your code here...

FINISH user_defined_module;
```

Table 2: Calling **POWERLIB** from within a module.

matrix names for reasons other than their intended purpose in **POWERLIB** will result in an error. When calling **POWERLIB** from within a module, the 28 matrices, plus the matricies `TOLERANC` and `LIMCASE`, must be listed as global matrices in the module definition, as it appears in the `START` statement in Table 2. In previous versions of **POWERLIB**, the current input matrix `TOLERANCE` was spelled as `TOLERANC`, and the current input matrix `CLTYPE` was instead named `LIMCASE`. We retained `TOLERANC` and `LIMCASE` in the `CALL` statement so that version 2.2 is compatible with programs written with previous versions of **POWERLIB**.

2.3. Outputs overview

POWERLIB produces four output matrices:

`_HOLDPOWER` `_HOLDPOWERLBL` `_POWERWARN` `_POWERWARNLBL`.

Unlike the input matrices listed in Tables 3 and 4, these are **not** globally defined. As such, these four matrices must be listed if **POWERLIB** is executed with a `CALL` statement. As stated before, use of these four matrix names for reasons other than their intended purpose in **POWERLIB** may result in an error.

All power computations are saved to the matrix `_HOLDPOWER` with labels given in the vector `_HOLDPOWERLBL`. Section 2.7 discusses options for power computations as well as elements to include in `_HOLDPOWER`.

By default, `_HOLDPOWER` is sent to the output window. Section 2.7 shows how to save the output matrix to a SAS dataset, which can be used, among other things, for graphing or simulation purposes. By default, several model matrices and warnings are sent to the output window prior to printing the output matrix. Section 2.7 discusses options that control matrix printing and warning notification.

Information about any numerical difficulties and approximation accuracies is stored in the global output matrix `_POWERWARN` with labels given in the vector `_POWERWARNLBL`. These matrices are further documented in Section 2.14. Typical users will not need to inspect `_POWERWARN`.

2.4. Required matrices

Computing power for a GLH with fixed predictors requires knowing seven variables: $\boldsymbol{\Sigma}$, $\boldsymbol{X}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{U}$, α, and $\boldsymbol{\Theta}_0$. The user must always specify $\boldsymbol{\Sigma}$, $\boldsymbol{X}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ via the matrices `SIGMA`, `ESSENCEX`, `BETA`, and `C`, respectively. The power program assumes default values of $\boldsymbol{U} = \boldsymbol{I}_p$ (which corresponds to a MANOVA hypothesis with multivariate responses), $\boldsymbol{\Theta}_0 = \mathbf{0}$, and $\alpha = 0.05$, specified with the matrices `U`, `THETA0`, and `ALPHA`, respectively.

Note that for univariate models, `SIGMA` is the variance, not the standard deviation, because $\boldsymbol{\Sigma} = \sigma^2$ is 1×1. **POWERLIB** gains substantial advantages by treating the univariate case as a special case of the multivariate because it requires the same variable name (`SIGMA`) for both settings, creating a potential source of confusion.

Input checks

If a required input has not been given by the user, or if any input matrix does not conform to the dimensions specified in Tables 3 and 4, **POWERLIB** will stop running. **POWERLIB**

Matrix Name	Description	Size	Default
Matrices to Specify the Model			
ESSENCEX	Fixed effects design matrix	Varies	Required
BETA	Fixed effects matrix $\boldsymbol{B}$	$q \times p$	Required
C	Matrix $\boldsymbol{C}$ in GLH	$a \times q$	Required
U	Matrix $\boldsymbol{U}$ in GLH	$p \times b$, $b \leq p$	$\boldsymbol{I}_p$
SIGMA	Covariance matrix	$p \times p$	Required
SIGTYPE	Scalar describing whether SIGMA is known or estimated (=0 if known and =1 if estimated)	1×1	0
THETA0	Matrix $\boldsymbol{\Theta}_0$	$a \times b$	$\mathbf{0}$
ALPHA	Type I error rates	1 row or col	0.05
REPN	List specifying # of times to duplicate each row of ESSENCEX	1 row or col	1
BETASCAL	List of multipliers for BETA	1 row or col	1
RHOSCAL	List of multipliers for correlation matrix RHO, created from SIGMA	1 row or col	1
SIGSCAL	List of multipliers for SIGMA	1 row or col	1
Matrices to Specify Confidence Limits			
CLTYPE	Type of confidence interval to include in power calculations	1×1	-1
RANK_EST	Scalar giving design matrix rank in the analysis that yielded $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$ estimates	1×1	None
N_EST	# of observations in the analysis that yielded $\boldsymbol{\Sigma}$ and/or $\boldsymbol{B}$ estimates	1×1	None
ALPHA_CL	Scalar specifying lower tail confidence limit probability	1×1	0.025
ALPHA_CU	Scalar specifying upper tail confidence limit probability	1×1	0.025

Table 3: **POWERLIB** input matrices to specify the model and confidence limits.

Matrix Name	Description	Size	Default
Matrices to Compute Power for an Internal Pilots Design			
`IP_PLAN`	Scalar describing whether power is computed within the context of planning an internal pilot (=0 if no and =1 if yes)	1×1	0
`N_IP`	Number of observations planned in the future study in an internal pilot	1×1	None
`RANK_IP`	Rank of the design matrix in the future study in an internal pilot	1×1	None
Matrices Specifying Distributional Choices			
`MMETHOD`	List specifying HLT, PBT, WLK power approximation	3×1, 1×3, 1×1	$\begin{bmatrix} 4 & 2 & 2 \end{bmatrix}'$
`UCDF`	List specifying CDF approximation for UNIREP (UN, HF, CM, GG, BOX)	5×1, 1×5, 1×1	$\begin{bmatrix} 2 & 2 & 2 & 2 & 2 \end{bmatrix}'$
`UMETHOD`	List specifying method for approximate $\mathrm{E}\,\tilde{\epsilon}$ and $\mathrm{E}\,\widehat{\epsilon}$	3×1, 1×3, 1×1	$\begin{bmatrix} 2 & 2 \end{bmatrix}'$
Matrices Specifying Precision of Output and Computations			
`ROUND`	Scalar specifying how many decimal places to `ROUND` power values	1×1	3
`TOLERANCE`	Scalar specifying what the software considers numeric zero	1×1	1×10^{-12}
Matrices Specifying Options Turned On and Off			
`OPT_ON`	User options to turn on	1 row or col	See Tables 7 and 8
`OPT_OFF`	User options to turn off	1 row or col	See Tables 7 and 8
Matrix Specifying Output Dataset			
`DSNAME`	Specifies output SAS file name and location	1×2, 2×1, 1×3, 3×1	`{WORK DODFAULT WORK}`

Table 4: Additional **POWERLIB** input matrices.

also checks for other possible errors in the user input matrices, ensuring that they contain plausible values.

Specifying the design matrix $\boldsymbol{X}$

The essence matrix (Helms 1988) contains one and only one copy of each unique row of the original matrix. We use this notation for $\boldsymbol{X}$ to ease the selection of sample sizes for which to compute power. For example, the following shows the $\boldsymbol{X}$ matrix for an independent groups t test, with 10 observations, denoted as $\boldsymbol{X}_1$, in terms of its essence matrix:

$$\begin{aligned}\boldsymbol{X}_1 &= \begin{bmatrix}\mathbf{1}_{10} & \mathbf{0}_{10} \\ \mathbf{0}_{10} & \mathbf{1}_{10}\end{bmatrix} \\ &= \begin{bmatrix}1 & 0 \\ 0 & 1\end{bmatrix} \otimes \mathbf{1}_{10} \\ &= \mathrm{Es}\left(\boldsymbol{X}_1\right) \otimes \mathbf{1}_{10}.\end{aligned}$$

$\mathbf{1}_{10}$ denotes a 10×1 vector of 1's and $\mathbf{0}_{10}$ a 10×1 vector of 0's. The $\otimes$ symbol denotes the Kronecker multiplication operator, where $\boldsymbol{A} \otimes \boldsymbol{B} = \{a_{ij} \cdot \boldsymbol{B}\}$ for any matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. See Muller and Fetterman (2002) for an introduction to ANOVA coding in one- and two-way designs.

The program assumes that $\boldsymbol{X}$ is specified in terms of its essence matrix and row replication factor. The values are specified with the matrices `ESSENCEX` and `REPN`, respectively. Above, $\boldsymbol{X}_1$ is specified with:

```
ESSENCEX = I(2);
REPN     = 10;
```

The $\boldsymbol{X}$ matrix can also always be given as an essence matrix equal to the entire matrix and replication of one. In this way, $\boldsymbol{X}_1$ can be specified with:

```
ESSENCEX = I(2) @ J(10,1,1);
REPN     = 1;
```

Above, $\boldsymbol{X}_1$ has equal cell sizes. The program can tolerate unequal cell sizes through no special coding. As an example, consider:

$$\boldsymbol{X}_2 = \begin{bmatrix}\mathbf{1}_{10} & \mathbf{0}_{10} & \mathbf{0}_{10} \\ \mathbf{0}_{15} & \mathbf{1}_{15} & \mathbf{0}_{15} \\ \mathbf{0}_{20} & \mathbf{0}_{20} & \mathbf{1}_{20}\end{bmatrix} .$$

One way to specify the design matrix $\boldsymbol{X}_2$ is:

```
ESSENCEX = {1 0 0, 1 0 0,
            0 1 0, 0 1 0, 0 1 0,
            0 0 1, 0 0 1, 0 0 1, 0 0 1};
REPN     =  5;
```

Alternately, as mentioned previously, $\boldsymbol{X}$ can be specified with `ESSENCEX` as the entire $\boldsymbol{X}$ matrix and `REPN = 1`. In this representation, $\boldsymbol{X}_2$ is coded as:

```
ONE      = {1 0 0};
TWO      = {0 1 0};
THREE    = {0 0 1};
ESSENCEX = REPEAT(ONE, 10, 1) // REPEAT(TWO, 15, 1) // REPEAT(THREE, 20, 1);
REPN     = 1;
```

Finally, **POWERLIB** allows $\boldsymbol{X}$ to have fractional cell sizes, e.g, `REPN = DO(1 TO 5 BY .5)`′, by specifying the `FRACREPN` option. Section 2.7 discusses how to designate this and other options available with **POWERLIB**.

2.5. A simple power program – one power value from a two sample t test

Table 5 gives the complete code needed to compute power for a two-sample t test with ten observations per group and cell mean coding. The program computes one power value and is an example of the simplest program that can be written to call **POWERLIB**. Note that for univariate models, $\boldsymbol{\Sigma} = \sigma^2$, a variance, not a standard deviation.

2.6. Producing power for a range of scenarios

POWERLIB makes it easy to compute power for ranges of values of sample size, type I error, mean difference, variance, and correlation level among response variables, through the global matrices `REPN`, `ALPHA`, `BETASCAL`, `SIGSCAL`, and `RHOSCAL`, respectively.

`REPN` and `ALPHA` contain simple lists of desired sample size and significance levels. `BETASCAL` and `SIGSCAL` contain multipliers for the user-specified values of `BETA` and `SIGMA`, respectively. `BETASCAL` could be called `THETASCAL` because it also multiplies $\boldsymbol{\Theta}$ by the same amount. `RHOSCAL` contains multipliers for the model correlation matrix created internally from `SIGMA`.

`REPN`, `ALPHA`, `BETASCAL`, `SIGSCAL`, and `RHOSCAL` are each always a $1 \times n$ or $n \times 1$ vector. Each equals 1 by default, the equivalent of no change from the original model specification. When more than one of these matrices has been changed from the default, power is computed for all factorial combinations of inputs.

Table 6 generalizes the code in Table 5. The new inputs ask for:

1. 2 values of `REPN`,
2. 10 values of `BETASCAL`,
3. 3 values of `SIGSCAL`, and
4. 3 values of `ALPHA`.

The values lead to computing power for:

1. total sample sizes of 20 and 40,
2. mean differences of $\mu_1 - \mu_2 = 0$, $\mu_1 - \mu_2 = 0.1$, $\mu_1 - \mu_2 = 0.2, \ldots, \mu_1 - \mu_2 = 0.9$,

Program code:

```
PROC IML SYMSIZE=2000;
%INCLUDE "..\IML\POWERLIB22.IML"/NOSOURCE2;
ESSENCEX = I(2)  ;
REPN     = {10};
BETA     = {0 1}`;
SIGMA    = {1}; *=variance, because here covariance matrix is 1x1 ;
C        = {1 -1};
RUN POWER;
```

Output:

```
ALPHA   SIGSCAL   BETASCAL   TOTAL_N    POWER
 0.05         1          1        20    0.562
```

Table 5: The simplest power program.

```
PROC IML SYMSIZE=2000;
%INCLUDE "..\IML\POWERLIB2.IML"/NOSOURCE2;
BETA     = {0 1}`;
C        = {1 -1};
SIGMA    = {1}; *=variance, because covariance "matrix" is 1x1 ;
ESSENCEX = I(2);
REPN     = {10 20};
BETASCAL = DO(0,.9,.1);
SIGSCAL  = {.5 1 2};
ALPHA    = {.005 .01 .05};
RUN POWER;
```

Table 6: Producing power for a range of scenarios.

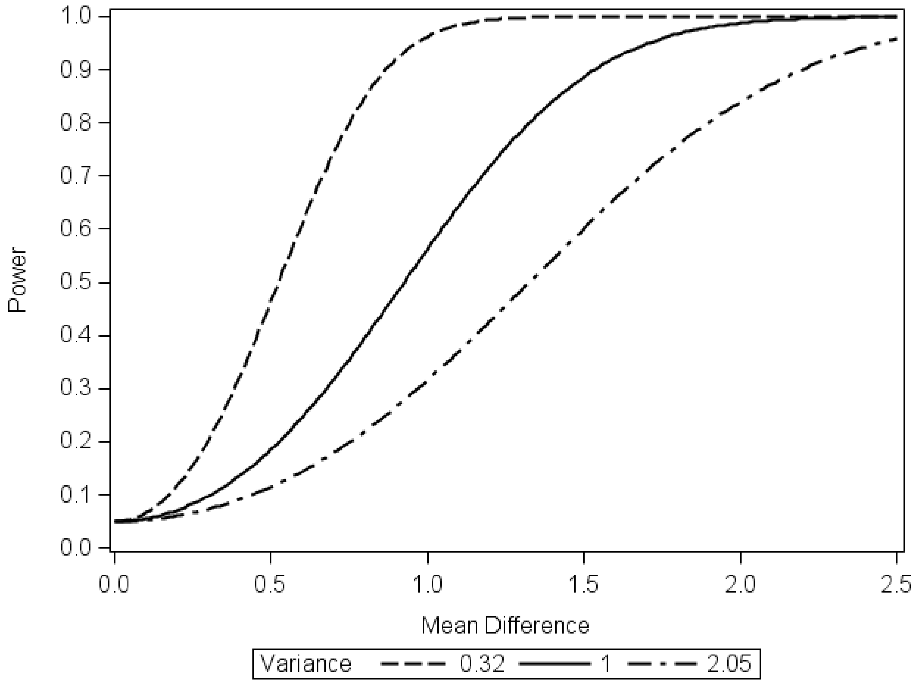

Figure 1: Power curve over several values of mean difference and sigma.

3. $\sigma^2 = 0.5$, $\sigma^2 = 1.0$, and $\sigma^2 = 2.0$, and
4. $\alpha = 0.005$, $\alpha = 0.01$, and $\alpha = 0.05$,

for a total of $2 \cdot 10 \cdot 3 \cdot 3$ computed power values.

The matrices described in this section are especially useful for producing power curves. Such graphical displays of power over a wide range of study design parameters are far more helpful than computation of point estimates. Figure 1 gives the output of Example 1 in the Examples folder of the software distribution. This program computes power over several values of mean difference (linear combination of elements of $\boldsymbol{B}$).

2.7. Power computation, printing, and output options

Overview

The next five sections describe the two global input matrices `OPT_ON` and `OPT_OFF`. Entries of these matrices are options which the user may specify to modify output produced by **POWERLIB**. These options allow the user to specify:

1. options that choose which hypothesis test statistic(s) are computed,
2. additional model specification options,
3. options that specify which columns are included in the final output matrix `_HOLDPOWER`,

4. options that specify which matrices are printed to the screen by default,
5. overall printing and warning notification options, and
6. options that control whether a SAS dataset is produced.

Tables 7 and 8 give a list of all possible entries (options) for `OPT_ON` and `OPT_OFF`. Assigning values to `OPT_ON` causes those options to be turned on; values in `OPT_OFF` are turned off. Both matrices must have only one row or have only one column. Order does not matter, nor does upper or lower case. The default selection of options corresponds to:

```
OPT_ON  = { GG HLT PBT WLK COLLAPSE ALPHA SIGSCAL BETASCAL TOTAL_N ESSENCEX
            BETA SIGMA RHO C U CBETAU WARN };
OPT_OFF = { UN HF CM BOX LTFR FRACREPN ORTHU UNIFORCE NONCENCL POWERCASE
            SIGTYPE RHOSCAL MAXRHOSQ CLTYPE N_EST RANK_EST ALPHA_CL ALPHA_CU
            UMETHOD MMETHOD FMETHOD UCDF IP_PLAN N_IP RANK_IP RANKX RANKC RANKU
            THETA0 NOPRINT DS };
```

By default, selecting certain options automatically turns on some other options. In particular,

> `THETA0` is on if `THETA0` is specified by the user,
>
> `SIGTYPE` is on if `SIGTYPE`=1,
>
> `CLTYPE`, `ALPHA_CL`, and `ALPHA_CU` are on if `CLTYPE`≥ 1, and
>
> `IP_PLAN` is on if `IP_PLAN`=1.

Options `GG`, `BOX`, `UN`, `HLT`, `PBT`, and `WLK` are disallowed when computing power for high dimension, low sample size studies with $B > (N - r)$. Since options `GG`, `HLT`, `PBT`, and `WLK` are on by default, the user must manually specify these options in `OPT_OFF` or the program will return an error. Details on computing power for high dimension, low sample size studies are given in section 2.10.

Choosing the hypothesis test statistic

POWERLIB can compute power for the Box, Geisser-Greenhouse, Huynh-Feldt, Chi-Muller, and uncorrected UNIREP tests, as well as the Hotelling Lawley Trace, Pillai-Bartlett Trace, and Wilks Lambda MULTIREP tests. To have power computed for each of these tests, specify the `BOX`, `GG`, `HF`, `CM`, `UN`, `HLT`, `PBT`, or `WLK` options, respectively, in `OPT_ON`. The three MULTIREP tests and the Geisser-Greenhouse UNIREP test are computed by default. If power for these tests is not desired, specify the corresponding option in `OPT_OFF`. Power for each separate test statistic requested is given a separate column in the output dataset.

When $b = 1$ (and $a \geq 1$), the UNIREP and MULTIREP tests coincide to give the familiar univariate linear model test. If the `COLLAPSE` option is on (as is the default), and if $b = 1$ (and $a \geq 1$), all MULTIREP and UNIREP powers would coincide, and, as a result, combine into one output column labeled `POWER`. If the collapse option is on, with $b > 1$ and $a = 1$, then all MULTIREP variable powers are combined into one column labeled `POWER_MULT`.

Option Name	Description	Default ON?
Options to Specify the Hypothesis Tests for Which Power is Computed		
UN	Compute power for UNIREP uncorrected	
HF	Compute power for UNIREP Huynh-Feldt	
GG	Compute power for UNIREP Geisser-Greenhouse	✓
CM	Compute power for UNIREP Chi-Muller	
BOX	Compute power for UNIREP Box	
HLT	Compute power for Hotelling-Lawley Trace	✓
PBT	Compute power for Pillai-Bartlett Trace	✓
WLK	Compute power for Wilks' Lambda	✓
COLLAPSE	$s = 1$ powers reduce to 1 column	✓
Model Specification Options		
LTFR	Allow use of less than full rank $\boldsymbol{X}$	
FRACREPN	Allow use of fractional REPN values	
ORTHU	Allow use of non-orthonormal $\boldsymbol{U}$	
UNIFORCE	Allow computation of power for a non-orthonormal $\boldsymbol{U}$ matrix	
NONCENCL	Compute confidence limits for noncentrality	
Options to Specify Which Inputs are Included in the Output Power Matrix		
POWERCASE	Include row #	
ALPHA	Include ALPHA	✓
SIGSCAL	Include SIGSCAL	✓
SIGTYPE	Include SIGTYPE	if SIGTYPE≥ 1
RHOSCAL	Include RHOSCAL	
BETASCAL	Include BETASCAL	✓
TOTAL_N	Include TOTAL_N	✓
MAXRHOSQ	Include max canonical correlation	
CLTYPE	Include CLTYPE	if CLTYPE≥ 1
N_EST	Include N_EST	
RANK_EST	Include RANK_EST	
ALPHA_CL	Include ALPHA_CL	if CLTYPE≥ 1
ALPHA_CU	Include ALPHA_CU	if CLTYPE≥ 1
UMETHOD	Include UMETHOD	
MMETHOD	Include MMETHOD	
FMETHOD	Include FMETHOD	
UCDF	Include UCDF	
IP_PLAN	Include IP_PLAN	
N_IP	Include N_IP	
RANK_IP	Include RANK_IP	

Table 7: Power computation options specified in OPT_ON or OPT_OFF.

Option Name	Description	Default ON?
Options to Control Which Matrices are Printed		
ESSENCEX	Print ESSENCEX matrix	✓
RANKX	Print the rank of the $\boldsymbol{X}$ matrix	
BETA	Print $\boldsymbol{B}$ matrix	✓
SIGMA	Print $\boldsymbol{\Sigma}$ matrix	✓
RHO	Print RHO matrix	✓
C	Print $\boldsymbol{C}$ matrix	✓
RANKC	Print the rank of the $\boldsymbol{C}$ matrix	
U	Print $\boldsymbol{U}$ matrix	✓
RANKU	Print the rank of the $\boldsymbol{U}$ matrix	
THETA0	Print $\boldsymbol{\Theta}_0$ matrix	if THETA0$\neq \mathbf{0}$
CBETAU	Print $\boldsymbol{CBU}$ matrix	✓
Global Printing Options		
WARN	Print power program warnings	✓
NOPRINT	Suppress all printed output	
CMWARN	Print warning for large number of variables when requesting the Chi-Muller test	
Option to Create a Dataset Containing Power Calculations		
DS	Write _HOLDPOWER to a SAS file with variable names _HOLDPOWERLBL	

Table 8: Printing and dataset options specified in OPT_ON or OPT_OFF.

Additional model specification options

By default:

1. $\boldsymbol{X}$ must be full rank.
2. `REPN` must contain whole numbers.
3. $\boldsymbol{U}$ must be orthonormal.

The `LTFR` option allows choice of less than full rank $\boldsymbol{X}$. The `FRACREPN` option allows choice of fractional values in `REPN`. Specifying the `ORTHU` option allows the user to specify a non-orthonormal $\boldsymbol{U}$ matrix, which `POWERLIB` will then convert into an appropriate orthonormal matrix for use in power calculations. Note that user utility modules `UPOLY1`, `UPOLY2`, and `UPOLY3` (documented in section 2.15) may also be used to create orthonormal contrast matrices for 1–3 within or between factors. Option `UNIFORCE` is similar to `ORTHU` in that it allows specification of a non-orthonormal $\boldsymbol{U}$ matrix from the user, but in this case, `POWERLIB` performs no changes to the matrix and continues to compute power based on a non-orthonormal $\boldsymbol{U}$ matrix. Option `UNIFORCE` should be used with caution, and the user assumes responsbility that the results are the ones desired. Option `NONCENCL` allows computation of confidence limits for noncentrality when confidence limits are requested by the user.

Specifying columns included in the output matrix

Table 9 lists all possible columns included in the output matrix `_HOLDPOWER`. Most are not included by default. To add or remove columns, include the option of the same name from the third section of Table 7 in `OPT_ON` or `OPT_OFF`. Note that asking the program to compute confidence limits for power values adds the columns with suffixes `_L` and `_U` to the output matrix.

Options that control printing and warnings

POWERLIB prints several matrices to the screen by default after power computations are completed. The first section of Table 8 lists possible choices and defaults. User inspection of these matrices is important to verify that the model matrices specified to **POWERLIB** are those intended.

Specifying the `NOPRINT` option in `OPT_ON` ensures that no matrices are printed, including the final `_HOLDPOWER` output matrix. This option is especially useful when power values are output to a dataset and when many power values are computed in a simulation study.

By default, **POWERLIB** writes helpful warnings to the screen even when no fatal syntax error is present. To prevent the warnings from printing, specify the `WARN` option in `OPT_OFF`.

Output to a dataset

To create a dataset, the user must specify the name of the dataset in the input matrix `DSNAME`, as well as include the `DS` option in the matrix `OPT_ON`.

The user can name the data file by defining `DSNAME` as follows:

```
DSNAME = { libref membername };
```

Column Label	Description
ALPHA	Type I error, α value for power calculation
ALPHA_CL	α value for power lower confidence limit
ALPHA_CU	α value for power upper confidence limit
BETASCAL	Multiplier for $\boldsymbol{B}$
CLTYPE	Descriptor for type of confidence interval computed
EPSILON	Population value of ϵ for UNIREP test power
EXEPS_*test*	Approximate $\mathrm{E}\,\widehat{\epsilon}$ for UNIREP *test*
FMETHOD	F probability calculation method with COLLAPSE option
FMETHOD_*test*	F probability calculation method for *test*
FMETHOD_L	F probability calculation method for lower CL with COLLAPSE option
FMETHOD_*test*_L	F probability calculation method for lower CL for *test*
FMETHOD_U	F probability calculation method for upper CL with COLLAPSE option
FMETHOD_*test*_U	F probability calculation method for upper CL for *test*
IP_PLAN	Indicator for whether or not power is computed within the context of an internal pilot design
MAXRHOSQ	Maximum canonical correlation
N_EST	# of obs. which gave $\boldsymbol{\Sigma}$ and/or $\boldsymbol{B}$ estimates
N_IP	Number of observations planned in the future study in an internal pilot
NONCEN	Computed noncentrality with COLLAPSE option
NONCEN_*test*	Computed noncentrality for *test*
NONCEN_L	Lower confidence limit for noncentrality with COLLAPSE option
NONCEN_*test*_L	Lower confidence limit for noncentrality for *test*
NONCEN_U	Upper confidence limit for noncentrality with COLLAPSE option
NONCEN_*test*_U	Upper confidence limit for noncentrality for *test*
POWER	Computed power with COLLAPSE option
POWER_*test*	Computed power for *test*
POWER_L	Lower confidence limit for power with COLLAPSE option
POWER_*test*_L	Lower confidence limit for power for *test*
POWER_U	Upper confidence limit for power with COLLAPSE option
POWER_*test*_U	Upper confidence limit for power for *test*
POWERCASE	Row number of _HOLDPOWER matrix
RANK_EST	rank($\boldsymbol{X}$) in analysis providing $\boldsymbol{\Sigma}$ and/or $\boldsymbol{B}$ estimates
RANK_IP	Rank of the design matrix in the future study in an internal pilot
RHOSCAL	Multiplier for correlations from $\boldsymbol{\Sigma}$
SIGSCAL	Multiplier for $\boldsymbol{\Sigma}$
SIGTYPE	Indicator for whether $\boldsymbol{\Sigma}$ is known or estimated
TOTAL_N	Total sample size
UCDF_*test*	CDF approximation used for power for UNIREP *test*
UMETHOD_*test*	Method used for calculating $\mathrm{E}\,(\tilde{\epsilon})$ for UNIREP *test*

test $\in$ UN, HF, CM, GG, BOX, HLT, PBT, WLK

Table 9: All possible column labels for output matrix _HOLDPOWER.

For example, if `DSNAME = {IN1 MYDATA}`, the output file will be called `IN1.MYDATA`. Here `IN1` refers to a library defined by a `LIBNAME` statement.

If `DSNAME` is not defined and the `DS` option is also selected, or if "membername" already exists in the library specified by "libref," a default file name is used. The default file names are numbered and of the form `WORK.PWRDT###` (where `###` is a number). The program scans the library for the largest numbered data file and assigns the next number to the new data file. The maximum `###` is 999. If `PWRDT999` exists, then no more data files can be created. Note that the program uses the name `_PWRDTMP` as an intermediate file. If this file already exists in the specified library, then no files can be created. To use a library other than `WORK` as the default, define:

```
DSNAME = {libref membername defaultlib};
```

The software will not write over existing files. To continually write to the same file with multiple runs of the power software, the user must consciously delete the existing file. To delete a file `IN1.MYDATA`, for example, execute the statement:

```
CALL DELETE (IN, MYDATA);
```

prior to executing the software.

2.8. Computing power with estimated variance

Table 10 describes the full suite of cases for which **POWERLIB** can compute a point estimate and confidence limits for power given all possible combinations of known or estimated $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$.

Confidence Limits for Power

In power analysis, $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ are assumed known constants. In practice, however, estimates of $\boldsymbol{\Sigma}$ only, or of both $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ from a previous study, are often used. The randomness of these estimates creates randomness in resulting power values. Confidence intervals about power values, and confidence regions about power curves, greatly help in planning by accounting for the uncertainty due to using estimates. Of all current software, only **POWERLIB** allows computing confidence intervals for power in Gaussian linear models.

Taylor and Muller (1995, 1996) provided theory for calculation of exact confidence intervals for univariate linear models with Gaussian errors. Gribbin (2007) and Park (2007) developed approximate confidence intervals for UNIREP and MULTIREP tests, respectively. These methods are exact for univariate models and for the MULTIREP tests when $s = 1$. For confidence limits for multivariate tests, we assume the user has given the (unbiased) REML estimate for $\boldsymbol{\Sigma}$. Note that the current version of the program does not allow compensating for the bias of truncation discussed in Taylor and Muller (1996) and Muller and Pasour (1997).

The `CLTYPE`, `RANK_EST`, `N_EST`, `ALPHA_CL`, and `ALPHA_CU` input matrices specify how confidence limits are computed for all test powers.

`CLTYPE` specifies what type of confidence limits are computed and takes the following values:

$< 1 \Rightarrow$ No confidence limits are calculated; assumes $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ are known.

$\boldsymbol{\Sigma}$	$\boldsymbol{B}$	Univariate	UNIREP	MULTIREP
Power Point Estimate				
Known	Known	Yes	Yes	Yes
Estimated	Known	Yes[1]	Yes	Yes[1]
Estimated	Estimated	Yes[1]	Yes[2]	Yes[1]
Power Confidence Limits				
Known	Known	No[3]	No[3]	No[3]
Estimated	Known	Yes	Yes	Yes
Estimated	Estimated	Yes	No[4]	Yes

Table 10: **POWERLIB** computation of the power point estimate and confidence limits for combinations of known and estimated $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$.

[1] These combinations of $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$ can be treated appropriately by **POWERLIB**; however, the resulting power point estimate provided by **POWERLIB** will be equivalent to the power computed assuming known $\boldsymbol{\Sigma}$ and known $\boldsymbol{B}$. The estimate of power provided is a median unbiased estimator, and is either the REML or ML estimate, depending on whether a REML or ML estimate was input for $\boldsymbol{\Sigma}$. Taylor and Muller (1995, 1996) described how a mean unbiased estimator of noncentrality can lead to negative estimates, which are impossible values, and are therefore improper, so the median unbiased estimator is recommended. The user with estimated $\boldsymbol{\Sigma}$ and known or estimated $\boldsymbol{B}$ who wishes to compute power for a univariate or MULTIREP test should simply input these as $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$. No additional matrices or options are required to designate $\boldsymbol{\Sigma}$ or $\boldsymbol{B}$ as estimated. In particular, **POWERLIB** will return the same power value whether `SIGTYPE=1` or `SIGTYPE=0` is specified. Further, there is no explicit option in **POWERLIB** to specify $\boldsymbol{B}$ as estimated, if appropriate, in the computation of the power point estimate. (Note that the matrix `CLTYPE` provides a way to specify $\boldsymbol{B}$ as estimated only for the purpose of computing confidence limits for power.)

[2] This case can be implemented with **POWERLIB** similarly as described in footnote 1. The user with estimated $\boldsymbol{\Sigma}$ and estimated $\boldsymbol{B}$ who wishes to compute power for a UNIREP test should input these as $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$. For the UNIREP tests, the estimate of power computed by **POWERLIB** when $\boldsymbol{\Sigma}$ is estimated will be different than the power computed when $\boldsymbol{\Sigma}$ is known, so the user must also specify `SIGTYPE=1`.

[3] Computation of confidence limits in **POWERLIB** requires estimated $\boldsymbol{\Sigma}$.

[4] Theory for this case has been developed by the authors, but has not been published, so has intentionally not been implemented.

$= 1 \Rightarrow$ Confidence limits for $\boldsymbol{B}$ known and $\boldsymbol{\Sigma}$ estimated are calculated.

$= 2 \Rightarrow$ Confidence limits for $\boldsymbol{B}$ estimated and $\boldsymbol{\Sigma}$ estimated are calculated.

Confidence limits with $\boldsymbol{\Sigma}$ estimated and $\boldsymbol{B}$ known are available for all hypothesis tests available in **POWERLIB**. Currently, confidence limits where both $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ have been estimated are available for univariate linear models ($b = 1$) only. Future research is needed to develop theory for confidence limits for multivariate tests in this case. We caution the reader that only very narrow conditions make using an estimate of $\boldsymbol{B}$ a defensible choice (Lenth 2001).

If `CLTYPE` ≥ 1, then `RANK_EST` and `N_EST` are required inputs. `RANK_EST` describes the rank of the design matrix from the previous study from which $\boldsymbol{B}$ and $\boldsymbol{\Sigma}$ were obtained. `N_EST` gives the total sample size of the previous study.

`ALPHA_CL` and `ALPHA_CU` specify the desired lower and upper tail probabilities, respectively, for confidence limits computations. Both have default values equal to 0.025.

Table 11 gives an example of the additional code needed to create confidence limits for power. For the entire context of this code, see Example 6 in the Examples folder of the software distribution.

Figure 2, also produced by code in Example 6, shows an example of a useful plot for determining the impact of estimation of power parameters.

Computing the power point estimate for estimated variance

The input matrix `SIGTYPE`, a new addition to this version of **POWERLIB**, allows a point estimate for power to be computed based on estimated $\boldsymbol{\Sigma}$. This can be computed with or without computing associated confidence limits for power as described in the previous section.

Power based on estimated $\boldsymbol{\Sigma}$ is available for all tests. To request a point estimate for power based on estimated $\boldsymbol{\Sigma}$, the user must simply specify `SIGTYPE = 1` as an input matrix. As with specification of `CLTYPE` ≥ 1, `RANK_EST` and `N_EST` are required inputs when `SIGTYPE` $= 1$.

Example 8 illustrates computation of a power estimate for power based on estimated $\boldsymbol{\Sigma}$.

2.9. Computing power for internal pilot designs

The power of a test depends on specifying the population error variance $\boldsymbol{\Sigma}$, which is usually unknown. To get around this, most researchers will instead use the variance from a previous study and treat it as known. Doing this ignores the randomness introduced into the calculation of sample size by using estimated $\boldsymbol{\Sigma}$ and may lead to increased type I error rates.

One way to accurately estimate power for a future study given an estimated $\boldsymbol{\Sigma}$ from a previous study is to perform an internal pilot study. In this case, power and required sample size are estimated both at the beginning of the study and at a pre-specified point during the study. The interim power calculation allows use of $\boldsymbol{\Sigma}$ that has been estimated from the current study, so that the required study sample size can be increased or decreased based on its value.

Version 2.2 of **POWERLIB** implements theory described in Coffey and Muller (2003) in order to compute power at the time of the interim power calculation. Theory is available for computation of power for both univariate and UNIREP tests only and has not been developed for MULTIREP tests.

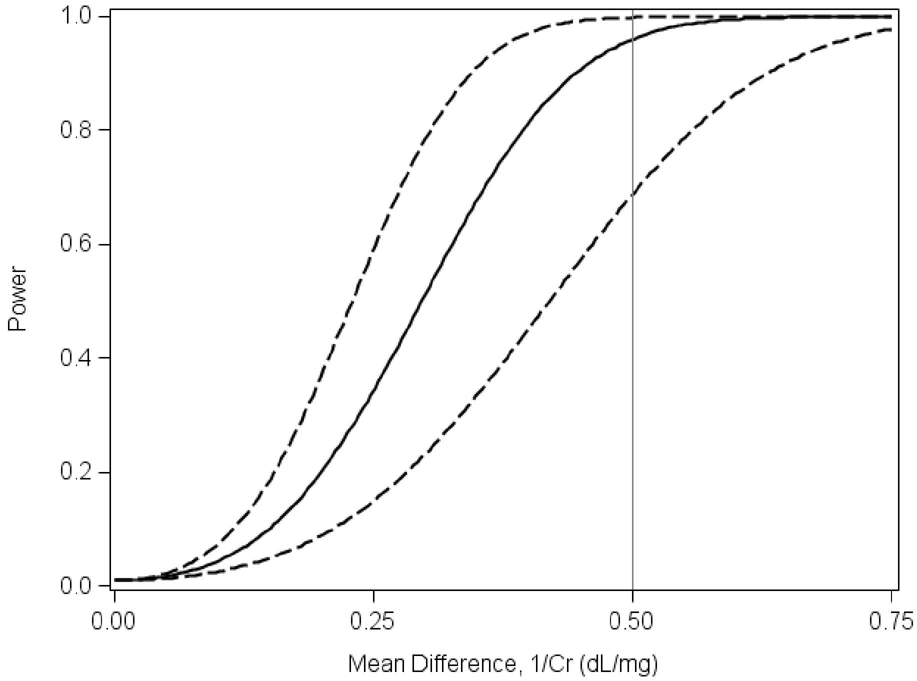

Figure 2: Plot of confidence limits.

```
CLTYPE   = 1;
N_EST    = 21;       *# Obs for variance estimate;
RANK_EST = 1;        *# model df for study giving variance estimate;
ALPHA_CL = 0.025;    *Lower confidence limit tail size;
ALPHA_CU = 0.025;    *Upper confidence limit tail size;
```

Table 11: Additional code to create confidence limits for power.

To compute power within the framework on an internal pilot design, the user should specify the input matrix `IP_PLAN=1`. The user must also specify the number of obervations and rank of the design matrix in the future study in input matricies `N_IP` and `RANK_IP`.

Computing power in the framework of an internal pilot treats $\mathbf{\Sigma}$ as known, and, as a result, disallows use of `SIGTYPE` and `CLTYPE` to specify use of estimated $\mathbf{\Sigma}$ for computation of power point estimates and confidence limits.

Table 12 summarizes the power computations available in **POWERLIB** for the context of internal pilot designs.

2.10. Computing power for high dimension, low sample size data

High dimension, low sample data occur when the number of subjects is less than the number of variables. Fields such as genetics, metabolomics, and proteomics often produce data with this property. Version 2.2 of **POWERLIB** now includes computation of an extension of the Huynh-Felt estimator proposed by Chi, Gribbin, Johnson, and Muller (2014) that provides accurate type I error and power computations for high dimension, low sample size data. Extensive simulations of group comparisons support the accuracy of the approximations even when the ratio of number of variables to sample size is large. Power can be computed for any multivariate model test. The test provides accurate error control when computing power assuming either fixed or estimated $\mathbf{\Sigma}$. Table 13 summarizes the power computations available in **POWERLIB** for high dimension, low sample size data.

The test proposed by Chi *et al.* (2014) can be requested by specifying the `CM` option in `OPT_ON`. While this test improves accuracy of type I error and power over other tests most greatly for high dimension, low sample size data, the test proposed by Chi *et al.* (2014) can be requested for any data, including data with large sample size compared to the number of variables. Since the estimator has been developed assuming both fixed and estimated $\mathbf{\Sigma}$, the software supports specifying `SIGTYPE=1` or `CLTYPE>=1` with the test, so that a point estimate based on estimated $\mathbf{\Sigma}$ or confidence limits for power can be produced.

Increasing the number of variables substantially may lead to a request for more computer memory than is available. Testing by the authors has shown that models with ≤ 3000 repeated measures should be computationally feasible on most computing systems. The option `CMWARN` is specified in `OPT_ON` by default and provides an error to the user if a model with > 3000 repeated measures is fit. This option can be turned off by specifying `CMWARN` in `OPT_OFF`, so that power is computed for models with > 3000 repeated measures. Turning off `CMWARN` should be done with caution; the user accepts responsibility for potential program failure due to insufficient memory.

2.11. Choosing power approximations

Overview

All powers are approximated by noncentral F probabilities. Approximations are used for all tests whenever $s = \min(a, b) > 1$, and for the UNIREP tests whenever $b > 1$ (whether or not $a = 1$). All MULTIREP tests' powers (and test sizes) coincide whenever $s = 1$, while MULTIREP and UNIREP powers (and test sizes) all coincide if $b = 1$.

Naturally, the default approximation methods have been chosen, given the current state

$\mathbf{\Sigma}$	$\boldsymbol{B}$	Univariate	UNIREP	MULTIREP
Power Point Estimate				
Known	Known	Yes[1]	Yes	No[2]
Estimated	Known	No[3]	No[3]	No[3]
Estimated	Estimated	No[3]	No[3]	No[3]
Power Confidence Limits				
Known	Known	No[4]	No[4]	No[4]
Estimated	Known	No[3]	No[3]	No[3]
Estimated	Estimated	No[3]	No[3]	No[3]

Table 12: **POWERLIB** computation of the power point estimate and confidence limits for combinations of known and estimated $\mathbf{\Sigma}$ and $\boldsymbol{B}$ for internal pilot studies.

[1] Theory for computing power for a univariate model test within the context of an internal pilot design is the same as for the usual case with fixed $\mathbf{\Sigma}$ and fixed $\boldsymbol{B}$. Therefore this case is treated appropriately by **POWERLIB** whether the user specifies `IP_PLAN=1` or `IP_PLAN=0`.

[2] Theory for this case has been developed by the authors, but has not been published, so has intentionally not been implemented.

[3] Computation of power for an internal pilot design requires known $\mathbf{\Sigma}$ and $\boldsymbol{B}$.

[4] Computation of confidence limits in **POWERLIB** requires estimated $\mathbf{\Sigma}$.

$\boldsymbol{\Sigma}$	$\boldsymbol{B}$	Univariate	UNIREP	MULTIREP
Power Point Estimate				
Known	Known	No[1]	Yes[2]	No[3]
Estimated	Known	No[1]	Yes[2]	No[3]
Estimated	Estimated	No[1]	Yes[2]	No[3]
Power Confidence Limits				
Known	Known	No[1,4]	No[4]	No[3,4]
Estimated	Known	No[1]	Yes[2]	No[3]
Estimated	Estimated	No[1]	No[5]	No[3]

Table 13: **POWERLIB** computation of the power point estimate and confidence limits for combinations of known and estimated $\boldsymbol{\Sigma}$ and $\boldsymbol{B}$ with high dimension, low sample size data.

[1] Any univariate test does not apply to high dimension, low sample size data, because high dimension data implies that data have repeated measures outcomes.

[2] The Geisser-Greenhouse, Box, and uncorrected UNIREP tests have poor properties when applied to high dimension, low sample size data, so they are disallowed. Only power for UNIREP Huynh-Feldt and Chi-Muller tests can be computed by **POWERLIB** for HDLSS data.

[3] Satisfactory theory has not been developed for power calculations for MULTIREP tests when applied to high dimension, low sample size data.

[4] Computation of confidence limits in **POWERLIB** requires estimated $\boldsymbol{\Sigma}$.

[5] Theory for this case has been developed by the authors, but has not been published, so has intentionally not been implemented.

of knowledge, as the best available option; hence, most users will never need to change the approximation methods from the default. The inclusion of the information allows comparisons to earlier versions of **POWERLIB** and other program output to improve future developments.

The next two sections discuss the input matrices `UMETHOD`, `UCDF`, and `MMETHOD` used to specify the distributional approximations used in computing power for the UNIREP and MULTIREP tests, respectively. Choices for these matrices are summarized in Table 14.

Choosing UNIREP approximations

The software allows the user to choose one of four approximations for the distribution of the UNIREP test statistic under the alternative via the values of `UCDF` shown in Table 14.

The default is `UCDF = {2, 2, 2, 2, 2}`, with all five UNIREP tests using the Muller *et al.* (2007) approximation for the distribution of the test statistic under the alternative. Exact results may be achieved, at the cost of computing time, for the uncorrected (UN) and Box tests by specifying `UCDF = {3, 2, 2, 2, 3}`.

`UMETHOD` specifies whether to use the Muller and Barton (1989) or Muller *et al.* (2007) approximations for $\mathrm{E}(\tilde{\epsilon})$ and $\mathrm{E}(\widehat{\epsilon})$. In turn, this option implies the approximate critical value used for the HF, CM, and GG tests. The default is `UMETHOD = {2,2,2}`.

Choosing MULTIREP approximations

For the MULTIREP tests, the Muller and Peterson (1984) approach requires specifying approximate degrees of freedom, which implies a critical value via the `FINV()` function, and an approximate noncentrality. By default, the program uses two moment approximations (Rao 1951; McKeon 1974; Muller 1998) for the null distributions, which imply degrees of freedom and critical values. Optionally, older and less accurate one moment methods are also available (Pillai 1954, 1955; Pillai and Samson 1959). By default, the MULTIREP tests use the Muller and Peterson (1984) noncentrality approximations. Each MULTIREP noncentrality may be multiplied by $N/[N - \mathrm{rank}(X)]$, as recommended by O'Brien and Shieh (1992). Using the O'Brien and Shieh (1992) multiplier gives slightly larger approximate powers. Especially for the Hotelling-Lawley test, the original Muller and Peterson (1984) noncentralities can be somewhat conservative in small samples.

Power approximations used for MULTIREP tests can be specified in the matrix `MMETHOD`. `MMETHOD` is a 3×1 vector whose elements correspond to one of four choices for the method used for the Hotelling Lawley Trace, Pillai-Bartlett Trace, and Wilks Lambda tests, respectively, as given in Table 14. The duplication of settings for Wilks' test is merely for programming convenience.

The default setting is `MMETHOD ={4,2,2}`. Use of the O'Brien and Shieh (1992) multiplier for all three multivariate tests may be chosen by setting `MMETHOD = {4,4,4}` or `MMETHOD = 4`, since `MMETHOD` can be specified as a 1×1 matrix if all the entries are the same.

2.12. F distribution probability calculations

In all but the most extreme cases, **POWERLIB**computes the probabilities from an F distribution, which is necessary for power computations, using the `SAS` supplied `CDF` function. If an evaluation of the extremeness of conditions indicates a likelihood of the `CDF` function failing, then **POWERLIB** computes F probabilities using the Tiku approximation (Kotz, Bal-

Choice for CDF approximation for all UNIREP tests		
`UCDF[1]`	$\rightarrow$	Choice for Uncorrected test
`UCDF[2]`	$\rightarrow$	Choice for Huynh-Feldt test
`UCDF[3]`	$\rightarrow$	Choice for Chi-Muller test
`UCDF[4]`	$\rightarrow$	Choice for Geisser-Greenhouse test
`UCDF[5]`	$\rightarrow$	Choice for Box test
`UCDF[J]` $= 1$	$\rightarrow$	Muller and Barton (1989) one moment approximation
$= 2$	$\rightarrow$	Muller *et al.* (2007) two moment approximation
$= 3$	$\rightarrow$	Exact via Davies (1980) method (Note: This may fail. If it does, a missing value is returned.)
$= 4$	$\rightarrow$	Exact via Davies (1980) Method. If it fails, use Muller *et al.* (2007) approximation.

Choice for expectation of epsilon for Huynh-Feldt, Chi-Muller, and Geisser-Greenhouse		
`UMETHOD[1]` $= 1$	$\rightarrow$	Muller and Barton (1989) approximation for HF $\mathrm{E}(\tilde{\epsilon})$
$= 2$	$\rightarrow$	Muller *et al.* (2007) approximation for HF $\mathrm{E}(\tilde{\epsilon})$
`UMETHOD[2]` $= 1$	$\rightarrow$	Muller and Barton (1989) approximation for CM $\mathrm{E}(\tilde{\epsilon})$
$= 2$	$\rightarrow$	Muller *et al.* (2007) approximation for CM $\mathrm{E}(\tilde{\epsilon})$
`UMETHOD[3]` $= 1$	$\rightarrow$	Muller and Barton (1989) approximation for GG $\mathrm{E}(\hat{\epsilon})$
$= 2$	$\rightarrow$	Muller *et al.* (2007) approximation for GG $\mathrm{E}(\hat{\epsilon})$

Choice of CDF approximation for Hotelling-Lawley Trace		
`MMETHOD[1]` $= 1$	$\rightarrow$	Pillai (1954, 1955) one moment null approximation
$= 2$	$\rightarrow$	McKeon (1974) two moment null approximation
$= 3$	$\rightarrow$	Pillai and Samson (1959) one moment null approximation + O'Brien and Shieh (1992) noncentrality multiplier
$= 4$	$\rightarrow$	McKeon (1974) two moment null approximation + O'Brien and Shieh (1992) noncentrality multiplier

Choices of CDF approximation for Pillai-Bartlett Trace		
`MMETHOD[2]` $= 1$	$\rightarrow$	Pillai (1954, 1955) one moment null approximation
$= 2$	$\rightarrow$	Muller (1998) two moment null approximation
$= 3$	$\rightarrow$	Pillai and Samson (1959) one moment null approximation + O'Brien and Shieh (1992) noncentrality multiplier
$= 4$	$\rightarrow$	Muller (1998) two moment null approximation + O'Brien and Shieh (1992) noncentrality multiplier

Choices of CDF approximation for Wilks' Lambda		
`MMETHOD[3]` $= 1$	$\rightarrow$	Rao (1951) two moment null approximation
$= 2$	$\rightarrow$	Rao (1951) two moment null approximation
$= 3$	$\rightarrow$	Rao (1951) two moment null approximation + O'Brien and Shieh (1992) noncentrality multiplier
$= 4$	$\rightarrow$	Rao (1951) two moment null approximation + O'Brien and Shieh (1992) noncentrality multiplier

Table 14: UNIREP and MULTIREP distributional approximation methods.

akrishnan, and Johnson 2000). In situations where the Tiku approximations fails or will be inaccurate, **POWERLIB** uses a Gaussian approximation via the `CDF` function.

The user can see which method has been used by specifying the `FMETHOD` option in the `OPT_ON` matrix. This includes the `FMETHOD` columns listed in Table 9 into the output matrix `_HOLDPOWER`. Note that these columns merely describe which methods have been used; the user cannot modify the method chosen.

Values of `FMETHOD` are as follows:

$= 1 \Rightarrow$ CDF function (no approximation),

$= 2 \Rightarrow$ Tiku approximation (best approximation),

$= 3 \Rightarrow$ Normal approximation, |Z-score|$<$ 6 (worst approximation),

$= 4 \Rightarrow$ Normal approximation, |Z-score|$>$ 6 (power is almost certainly zero or one), and

$= 5 \Rightarrow$ Power missing.

Difficulties with power calculations occur almost always when power approaches zero or one.

2.13. Numerical accuracy

The input matrices `ROUND` and `TOLERANCE` control the rounding of output values and the threshold for judging whether a numerical value is judged to be zero, respectively. The `TOLERANCE` matrix is included to provide a user with sophistication in computing methods with some flexibility when working with large models.

2.14. Error checking

Counts of certain numerical difficulties are stored in entries of the output vector `_POWERWARN`. Table 15 describes the elements of this matrix.

2.15. User utilities

Four modules that are useful for creating $\boldsymbol{U}$ contrast matrices are included. These are:

`UMEAN   UPOLY1   UPOLY2   UPOLY3.`

`UMEAN` is a function module that generates a $p \times 1$ $\boldsymbol{U}$ matrix, which computes the average response, $\boldsymbol{U} = p^{-1}\begin{bmatrix}1 & 1 & \dots & 1\end{bmatrix}'$. For example, it could create the matrix:

$$\boldsymbol{U} = \begin{bmatrix}1/5 & 1/5 & 1/5 & 1/5 & 1/5\end{bmatrix}'.$$

It has one input, p, which indicates the size of matrix to create. For any arbitrary value of p, the user may execute:

```
U = UMEAN(p);
```

Element	Description
1	A Tiku approximation was used in calculating power.
2	A Z approximation was used in calculating power.
3	A Z approximation was used in calculating power and $\vert Z\vert > 6$, so that the power returned is exactly 0 or 1.
4	Power is missing because the `FINV` function returned a missing value.
5	The lower confidence limit on power is conservative.
6	A Tiku approximation was used in calculating the lower confidence limit on power.
7	A Z approximation was used in calculating the lower confidence limit on power.
8	A Z approximation was used in calculating the lower confidence limit on power and $\vert Z\vert > 6$, so that the power returned is exactly 0 or 1.
9	The lower confidence limit on power is missing because the `FINV` function returned a missing value.
10	The upper confidence limit on power is conservative.
11	A Tiku approximation was used in calculating the upper confidence limit on power.
12	A Z approximation was used in calculating the upper confidence limit on power.
13	A Z approximation was used in calculating the upper confidence limit on power and $\vert Z\vert > 6$, so that the power returned was exactly 0 or 1.
14	The upper confidence limit on power is missing because the `FINV` function returned a missing value.
15	Power is missing because because the noncentrality could not be computed.
16	Confidence limits are missing because power is missing.
17	The approximate expected value of estimated epsilon was truncated up to $1/b$.
18	The approximate expected value of estimated epsilon was truncated down to 1.
19	Power missing due to failure of Davies' algorithm.
20	Inputs give off-diagonal correlation = 1 in `RHO`.
21	(N - R) <= 5, so power approximations may be inaccurate, especially Huynh-Feldt.
22	Power values were rounded to 1 using the value contained in `ROUND` and should not be be reported as Power = 1. For example, if `ROUND` = 3 then report Power > 0.999.
23	Power is missing, because Uncorrected, Geisser-Greenhouse and Box tests are poorly behaved (super low power and test size) when B > N-R, i.e., HDLSS.

Table 15: Error messages in `_POWERWARN`.

For example, the previous matrix would be obtained with `U = UMEAN(5)`.

`UPOLY1`, `UPOLY2`, and `UPOLY3` each generate $\boldsymbol{U}$ contrast matrices with orthogonal polynomial coding for one, two, or three repeated factors, respectively, via the SAS `ORPOL` function.

`UPOLY1` takes two inputs: `VALUES` and `NAME`. Here, `VALUES` is a $k \times 1$ or $1 \times k$ vector that gives the k levels of the single repeated factor. `NAME` is a 1×1 character matrix describing the repeated factor. The module outputs two matrices: `U` and `ULBL`. Columns of `U` contain up to level $k-1$ polynomial contrasts; c contains labels for the order of the polynomial contrast each column represents. The `UPOLY1` module may be called using the following syntax:

```
CALL UPOLY1 (VALUES, NAME, U, ULBL);
```

As an example, the following code creates a $\boldsymbol{U}$ matrix with orthogonal polynomial contrasts for four levels, 1, 10, 100, and 1000, of the factor `DOSE`:

```
LEVELS = {1, 10, 100, 1000};
FACTOR = {DOSE};
CALL UPOLY1 (LEVELS, FACTOR, U, ULBL);
PRINT U [COLNAME = ULBL];
```

The columns of $\boldsymbol{U}$ (as described by `ULBL`) are the linear, quadratic, and cubic polynomial trends for `DOSE`. This produces the matrix:

$$
\begin{array}{cc}
 & \begin{array}{ccc} \texttt{DOSE1} & \texttt{DOSE2} & \texttt{DOSE3} \end{array} \\
\boldsymbol{U} = & \begin{bmatrix} -0.330 & 0.442 & -0.667 \\ -0.320 & 0.314 & 0.741 \\ -0.212 & -0.836 & -0.074 \\ 0.862 & 0.080 & 0.001 \end{bmatrix}.
\end{array}
$$

`UPOLY2` and `UPOLY3` generate $\boldsymbol{U}$ contrast matrices with orthogonal polynomial coding for two and three repeated factors, respectively. Modules `UPOLY2` and `UPOLY3` work the same way as the `UPOLY1` module, except that they require 4 or 6 input matrices, respectively, and produce 6 or 12 matrices, respectively, due to adding one or two more factors. These modules are called with the following syntax:

```
CALL UPOLY2 (VALUES1, NAME1, VALUES2, NAME2,
             U1, U1LBL,  U2, U2LBL, U12, U12LBL);
CALL UPOLY3 (VALUES1, NAME1, VALUES2, NAME2, VALUES3, NAME3,
             U1, U1LBL, U2, U2LBL, U3, U3LBL,
             U12, U12LBL, U13, U12LBL, U23, U23LBL, U123,U123LBL);
```

Here, `VALUES1`, `VALUES2`, and `VALUES3` give the levels for factors 1, 2, and 3, respectively, and `NAME1`, `NAME2`, `NAME3` describe factors 1, 2, and 3, respectively.

Also, `U1`, `U2`, and `U3` give the main effect contrasts for factors 1, 2, and 3, respectively, and `U12`, `U13`, and `U23` give the two way interaction contrasts for factor 1 with 2, 1 with 3, and 2 with 3, respectively. `U123` gives the three-way interaction contrasts for factor 1 with 2 and 3.

U1LBL, U2LBL, U3LBL, U12LBL, U13LBL, U23LBL, and U123LBL give column labels for matrices U1, U2, U3, U12, U13, U23, and U123, respectively.

The following example code gives all orthogonal polynomial trends for three factors, AGE, DRUG, and TIME, with levels {2, 4, 6}, {1, 2, 3}, and {10, 30, 60}:

```
LEVELS1 = {2, 4, 6};
NAME1   = "AGE";
LEVELS2 = {1, 2, 3};
NAME2   = "DRUG";
LEVELS3 = {10, 30, 60};
NAME3   = "TIME";
CALL UPOLY3 (LEVELS1, NAME1, LEVELS2, NAME2, LEVELS3, NAME3,
             U1, U1LBL, U2, U2LBL, U3,U3LBL,
             U12, U12LBL, U13, U13LBL, U23, U23LBL, U123,U123LBL);
PRINT U1   [COLNAME = U1LBL];  PRINT U2   [COLNAME = U2LBL];
PRINT U3   [COLNAME = U3LBL];  PRINT U12  [COLNAME = U12LBL];
PRINT U13  [COLNAME = U13LBL]; PRINT U23  [COLNAME = U23LBL];
PRINT U123 [COLNAME = U123LBL];
```

Again, numbers in the column labels describe the degree of the polynomial trend corresponding to that column. Labels with one variable indicate a main effect; labels with two variable names indicate a two-way interaction; labels with three variables indicate a three-way interaction. The following statements create a $\boldsymbol{U}$ orthogonal polynomial trends matrix including all trends for main effects, two-way interactions, and the three-way interaction:

```
U   = U1 || U2 || U3 || U12 || U13 || U23 || U123;
LBL = U1LBL || U2LBL || U3LBL || U12LBL || U13LBL || U23LBL || U123LBL;
```

3. Additional examples

The code, log, output listings, and required pre-existing datasets for all the following power programs are found in the ZIP file available for download with this paper. They can be run in an interactive or a batch environment. One change is needed for the user to run the programs. The folder where the **POWERLIB** files and folders have been copied must be specified in the macro variable ROOT with:

```
%LET ROOT = Your location here..;
```

This variable is used when bringing in the POWERLIB22.IML code as in the statement:

```
%INCLUDE "&ROOT.\Iml\POWERLIB22.IML"/NOSOURCE2;
```

Additionally, in order for the statement will run, the POWERLIB22.IML program is assumed to reside in a sub-folder of the ROOT directory named IML. Similarly, some programs assume that necessary, pre-existing datasets reside in a sub-folder of the ROOT directory named Data. The programs that create plots write all plots to a required existing `Examples` sub-folder of the `ROOT` directory. Programs creating three-dimensional plots will likely also require changing the `FILENAME` statement, as well as a few `GOPTIONS`, such as `DEVICE`, to tailor the output to the particular computer.

3.1. Power for a t test with overlay plot

Example 1 calculates power for a two-sample t test. The hypothesis tested is whether the two group means are equal. Power is computed for three values of σ^2 and several values of mean difference ($\boldsymbol{B}$). Powers for these values are then plotted on a power curve.

3.2. Power for a paired t test

Example 2 performs power calculations for a simple paired t test using a general linear hypothesis in a multivariate setting. The second section of code produces results equivalent to those produced by the first section; however, it uses difference scores to test the null hypothesis of no difference between group means.

3.3. Power for a t test with three-dimensional plot

Example 3 produces a three-dimensional graph that illustrates power trade offs among total sample size and the hypothesized difference between two group means in an independent groups t test.

3.4. Power for a test of an interaction term in a multivariate model

Example 4 performs a more complicated set of power calculations for a test of the hypothesis of no time by treatment interaction in a multivariate model.

3.5. Test in a multivariate model with two within factors

Example 5 illustrates use of `UPOLY2` for a design with two within- and no between-subject factors. It uses `SIGSCAL` and combines results from multiple runs of the power module. The results reproduce, except for some rounding differences, the predicted GG and HF powers in Table III in Coffey and Muller (2003), which used version 1 of **POWERLIB**, based on Muller and Barton (1989) methods. The example program also produces predicted Huynh-Feldt and Geisser-Greenhouse powers using the methods from Muller *et al.* (2007), which are the default in **POWERLIB** version 2.1. The new methods are far more accurate, especially for very small (near $1/b$) or very large (near 1) values of ϵ.

3.6. Confidence limits for a univariate model test

Example 6 produces three graphs showing two-sided or one-sided lower or upper confidence limits for power, which reflect uncertainty in power calculations due to use of estimated variance parameters. The program utilizes the power confidence limit calculations available in **POWERLIB** and replicates the figures seen in Taylor and Muller (1995).

3.7. Confidence limits for a UNIREP test in a multivariate model

Example 7 utilizes a dataset that contains cerebral vessel tortuosity measures for subjects in four regions of the brain. A set of power calculations is performed for the test of the hypothesis of no Gender-By-Region interaction with five age groups. Confidence limits are computed for these power values.

Four graphical displays are produced:

1. three three-dimensional plots of power by sample size by mean difference, displayed on different axes,
2. a plot of the hypothesized gender-By-region interaction with a sample size of 100 and an approximate Geisser-Greenhouse power of 0.90,
3. a plot of Geisser-Greenhouse power curves for sample sizes of 20, 40, and 80 for the gender-By-region interaction, and
4. a plot with confidence limits for power with N=40 from the third plot.

3.8. Point estimate for power of a UNIREP test based on estimated variance

Example 8 illustrates use of the `SIGTYPE` input matrix to compute a point estimate for power assuming estimated $\boldsymbol{\Sigma}$. Powers computed assuming both fixed and estimated $\boldsymbol{\Sigma}$ are compared using the sample data from Example 7.

3.9. Power within the context of an internal pilot design

Example 9 illustrates use of the `IP_PLAN` input matrix to compute power when designing an internal pilot study. Powers computed in Example 5, not in an internal pilot design, are re-computed for use in planning an internal pilot study.

3.10. Power for high dimension, low sample size data

Example 10 performs power calculations for the UNIREP test statistic proposed by Chi *et al.* (2014) for high dimension, low sample size data. It reproduces the power calculations for a study of vitamin B6 deficiency described in Section 4 of Chi *et al.* (2014).

3.11. Illustrate use of the `UPOLY1` module

Example 11 demonstrates the use of the `UPOLY1` module when performing power calculations for a time by treatment interaction.

3.12. Illustrate use of the `UPOLY3` module

The first part of Example 12 demonstrates the direct creation of three-way contrast matrices of two types: orthonormal polynomials and pair-wise differences to a reference level. As long as cell mean coding is used for a factorial design (including the special case of a one-way design), the approach taken to create $\boldsymbol{U}$ matrices may be applied to $\boldsymbol{C}$ matrices, and vice versa, with the obvious change of transposing matrices. Although we recommend using UPOLY3, this example is intended to provide a basis for creating contrasts for more unusual designs. The second part of Example 12 demonstrates creation of the same contrast matrices using the UPOLY3 module.

4. Concluding remarks

For those familiar with SAS, **POWERLIB** substantially increases the power analysis capabilities related to univariate and multivariate linear models. With some simple manipulations of code in SAS/IML, one can calculate power for a wide range of tests and for a variety of approximation methods associated with them. At the same time, users who wish to use the best approximation methods available can simply use the default options in **POWERLIB**. **POWERLIB** offers cutting-edge power analysis capabilities, such as confidence intervals for power and power analysis for a special class of linear mixed models. For these reasons among others, **POWERLIB** is a capable tool for any SAS user who needs comprehensive sample size calculations when planning a study. Additionally, the ability to easily implement **POWERLIB** in simulation studies makes it a useful tool for those researching statistical power in general.

Acknowledgments

Over the past twenty years, this power software has evolved through multiple versions. Bercedis Peterson and Keith Muller wrote the first version of **POWERLIB** for PROC MATRIX in 1984. Lynette Keyes and Keith Muller updated the first version of **POWERLIB** for SAS/IML in 1992. Keyes also provided the first version of this manual, which was in turn based on the PROC MATRIX manual. Doug Taylor created code for confidence limits calculations which Keith Muller then used to create **POWERLIM** in 2001. Jacqueline Johnson, Keith Muller, James Slaughter, Matthew Gurka, Matthew Gribbin, and Sean Simpson programmed Version 2.1 in 2003. Programming of this version by the authors began in 2012.

References

Chi YY, Gribbin MJ, Johnson JL, Muller KE (2014). "Power Calculation for Overall Hypothesis Testing with High-Dimensional Commensurate Outcomes." *Statistics in Medicine*, **To appear**.

Coffey CS, Muller KE (2003). "Properties of Internal Pilots with the Univariate Approach to Repeated Measures." *Statistics in Medicine*, **22**(15), 2469–2485.

Davies RB (1980). "Algorithm AS 155: The Distribution of a Linear Combination of Chi-Square Random Variables." *Applied Statistics*, **29**, 323–333.

Gribbin MJ (2007). *Better Power Methods for the Univariate Approach to Repeated Measures.* Ph.D. thesis, University of North Carolina at Chapel Hill.

Gurka M, Coffey C, Muller K (2007). "Internal Pilots for a Class of Linear Mixed Models with Gaussian and Compound Symmetric Data." *Statistics in Medicine*, **26**(22), 4083–4099.

Hedeker D, Gibbons R, Waternaux C (1999). "Sample Size Estimation for Longitudinal Designs with Attrition: Comparing Time-Related Contrasts Between Groups." *Journal of Educational and Behavioral Statistics*, **24**, 70–93. URL http://tigger.uic.edu/~hedeker/ml.html.

Heitjan D (2013). *Power and Sample Size Functions*. URL http://www.cceb.upenn.edu/pages/heitjan/power/.

Helms RW (1988). "Definitions of Linear-Model Parameters and Hypotheses as Functions of E[Y]." *Communications in Statistics-Theory and Methods*, **17**(8), 2715–2723.

Kotz S, Balakrishnan N, Johnson N (2000). *Continuous Multivariate Distributions, Volume 1: Models and Applications, 2nd edition*. Wiley, New York.

Lenth RV (2001). "Some Practical Guidelines for Effective Sample Size Determination." *The American Statistician*, **55**, 187–193.

McKeon JJ (1974). "F-Approximations to Distribution of Hotellings-T2/3." *Biometrika*, **61**(2), 381–383.

Muller KE (1998). "A New F Approximation for the Pillai-Bartlett Trace Under H-0." *Journal of Computational and Graphical Statistics*, **7**(1), 131–137.

Muller KE, Barton CN (1989). "Approximate Power for Repeated-Measures ANOVA Lacking Sphericity." *Journal of the American Statistical Association*, **84**(406), 549–555.

Muller KE, Benignus VA (1992). "Increasing Scientific Power with Statistical Power." *Neurotoxicology and Teratology*, **14**(3), 211–219.

Muller KE, Edwards L, Simpson S, Taylor D (2007). "Statistical Tests with Accurate Size and Power for Balanced Linear Mixed Models." *Statistics in Medicine*, **26**(19), 3639–3660.

Muller KE, Fetterman B (2002). *Regression and ANOVA: an Integrated Approach Using SAS Software*. SAS Institute, Cary, NC.

Muller KE, Lavange LM, Ramey SL, Ramey CT (1992). "Power Calculations for General Linear Multivariate Models Including Repeated Measures Applications." *Journal of the American Statistical Association*, **87**(420), 1209–1226.

Muller KE, Pasour VB (1997). "Bias in Linear Model Power and Sample Size Due to Estimating Variance." *Communications in Statistics-Theory and Methods*, **26**(7), 1811–1811.

Muller KE, Peterson BL (1984). "Practical Methods for Computing Power in Testing the Multivariate General Linear Hypothesis." *Computational Statistics and Data Analysis*, **2**, 143–158.

Muller KE, Stewart PW (2006). *Linear Model Theory: Univariate, Multivariate, and Mixed Models*. Wiley, Hoboken, NJ.

NCSS (2013). *PASS, Power Analysis and Sample Size*. URL http://www.ncss.com/pass.html.

O'Brien R (2003). *UnifyPow*. URL http://www.bio.ri.ccf.org/power.html.

O'Brien R, Muller K (1993). "A Unified Approach to Statistical Power for T-Tests to Multivariate Models." In L Edwards (ed.), *Applied Analysis of Variance in Behavioral Sciences*, pp. 297–344. Marcel Dekker, New York.

O'Brien R, Shieh G (1992). "Pragmatic, Unifying Algorithm Gives Power Probabilities for Common Tests of the Multivariate General Linear Hypothesis." *Unpublished work*.

Park S (2007). *Accounting for Bias and Uncertainty in Power for Multivariate Gaussian Linear Models*. Ph.D. thesis, University of North Carolina at Chapel Hill.

Pillai KCS (1954). "On Some Distribution Problems in Multivariate Analysis." *University of North Carolina, Mimeo Series*, **88**.

Pillai KCS (1955). "Some New Test Criteria in Multivariate Analysis." *Annals of Mathematical Statistics*, **26**(1), 117–121.

Pillai KCS, Samson P (1959). "On Hotellings Generalization of T2." *Biometrika*, **46**(1-2), 160–168.

Rao C (1951). "An Asymptotic Expansion of the Distribution of Wilks' Criterion." *Bulletin of the Institute of International Statistics*, **33c**, 177–180.

SAS Institute (2013a). *The SAS System Version 9.3*. Cary, NC. URL http://support.sas.com/documentation/93/index.html.

SAS Institute (2013b). *SAS/IML Software Version 9.3*. Cary, NC. URL http://support.sas.com/documentation/cdl/en/imlug/64248/HTML/default/viewer.htm#titlepage.htm.

SAS Institute (2013c). *SAS/STAT Software Version 9.3*. Cary, NC. URL http://support.sas.com/documentation/cdl/en/statug/63962/HTML/default/viewer.htm#titlepage.htm.

Spybrook J, Raudenbush S, Liu X, Congdon R, Martinez A (2011). *Optimal Design Software for Multi-level and Longitudinal Research (Version 3.01) [Software]*. URL http://sitemaker.umich.edu/group-based/optimal_design_software.

Statistical Solutions (2013). *Nquery Advisor*. URL http://www.statsol.ie/.

Taylor DJ, Muller KE (1995). "Computing Confidence-Bounds for Power and Sample-Size of the General Linear Univariate Model." *The American Statistician*, **49**(1), 43–47.

Taylor DJ, Muller KE (1996). "Bias in Linear Model Power and Sample Size Calculation Due to Estimating Noncentrality." *Communications in Statistics-Theory and Methods*, **25**(7), 1595–1610.

Timm N (2002). *Applied Multivariate Analysis*. Springer-Verlag, New York, NY.

Verbeke G, Molenbergs G (2000). *Linear Mixed Models for Longitudinal Data*. Springer, New York.

Affiliation:

Jacqueline L. Johnson
Medical School Wing B, Room 331
CB#7160
University of North Carolina at Chapel Hill
Chapel Hill, NC 27599
E-mail: jacqueline_johnson@med.unc.edu

Yueh-Yun Chi
2004 Mowry Road, Room 5238
PO Box 117450
Gainesville, FL 32611
E-mail: yychi@ufl.edu

Keith E. Muller
2004 Mowry Road, Room 2246
PO Box 100177
Gainesville, FL 32610-0177
E-mail: KMuller@ufl.edu
URL: http://www.health-outcomes-policy.ufl.edu/muller/
URL: http://www.samplesizeshop.org