General Methods for Power Calculation

Paul A. Thompson, Ph.D. [1]

[1]Director, Methodology and Data Analysis Center
Professor, Department of Pediatrics, Department of Internal Medicine
Sanford School of Medicine of the University of South Dakota

May 19, 2014

General Approach to Power Analysis

- Power: An introduction
- Technical details
- General power methods
- 3 Examples
- Conclusion

What is statistical power?

"The power of a statistical test is the probability that it will yield statistically significant results."

Jacob Cohen

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- `SAS: PROC POWER, GLMPOWER`: Packages
- `PASS,G*Power`: Standalone programs
- `R` methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

Power methods lag behind methods development General approaches are needed which are less specific

Power availability

- Programs widely available
- SAS: PROC POWER, GLMPOWER: Packages
- PASS,G*Power: Standalone programs
- R methods
- Multiple websites online

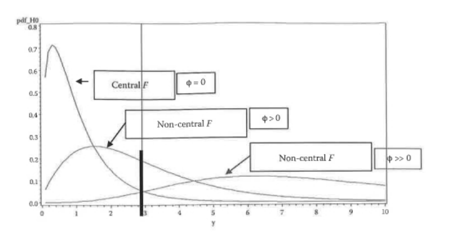
Power methods lag behind methods development General approaches are needed which are less specific

The non-centrality parameter φ

General concept

- Central F, H_0 true
- $\varphi = 0$
- Non-central F, H_0 false
- Deviation from H_0
- $\varphi > 0$

Distributional view



Stroup, 2013

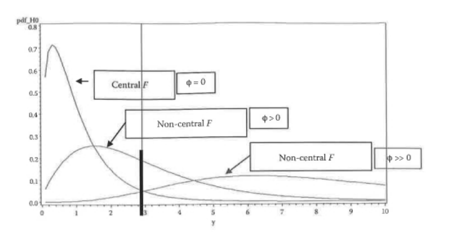
$$\varphi = (\mathbf{K}'\beta - \psi)' \left[\mathbf{K}' (\mathbf{X}'\mathbf{W}\mathbf{X})^{-} \mathbf{K}\right] (\mathbf{K}'\beta - \psi)$$
$$= F \times \text{rank}(K) = F \times \text{df}_n$$

The non-centrality parameter φ

General concept

- Central F, H_0 true
- $\varphi = 0$
- Non-central F, H_0 false
- Deviation from H_0
- $\varphi > 0$

Distributional view



Stroup, 2013

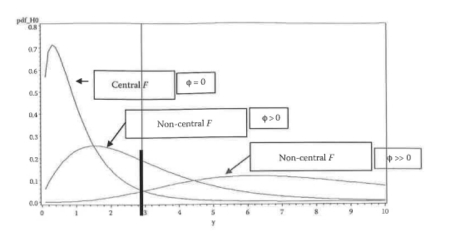
$$
\begin{aligned}
\varphi &= \left(\mathbf{K}'\beta - \psi\right)' \left[\mathbf{K}' \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-} \mathbf{K}\right] \left(\mathbf{K}'\beta - \psi\right) \\
&= F \times \text{rank}(K) = F \times \text{df}_n
\end{aligned}
$$

The non-centrality parameter φ

General concept

- Central F, H_0 true
- $\varphi = 0$
- Non-central F, H_0 false
- Deviation from H_0
- $\varphi > 0$

Distributional view



Stroup, 2013

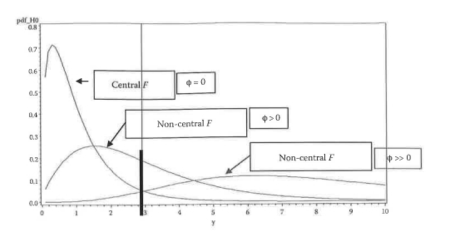
$$
\begin{aligned}
\varphi &= (\mathbf{K}'\beta - \psi)' \left[\mathbf{K}' \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-} \mathbf{K}\right] (\mathbf{K}'\beta - \psi) \\
&= F \times \text{rank}(K) = F \times \text{df}_n
\end{aligned}
$$

The non-centrality parameter φ

General concept

- Central F, H_0 true
- $\varphi = 0$
- Non-central F, H_0 false
- Deviation from H_0
- $\varphi > 0$

Distributional view



Stroup, 2013

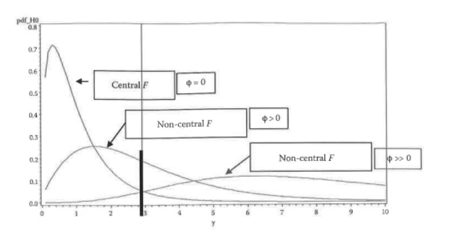
$$\begin{aligned} \varphi &= (\mathbf{K}'\beta - \psi)' \left[\mathbf{K}' \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-} \mathbf{K}\right] (\mathbf{K}'\beta - \psi) \\ &= F \times \text{rank}(K) = F \times \text{df}_n \end{aligned}$$

The non-centrality parameter φ

General concept

- Central F, H_0 true
- $\varphi = 0$
- Non-central F, H_0 false
- Deviation from H_0
- $\varphi > 0$

Distributional view



Stroup, 2013

$$\varphi = (\mathbf{K}'\beta - \psi)' \left[\mathbf{K}' \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-} \mathbf{K}\right] (\mathbf{K}'\beta - \psi)$$

$$= F \times \text{rank}(K) = F \times \text{df}_n$$

Using the non-centrality parameter

With non-centrality parameter, power is easy to calculate

Power is the "proportion under non-central F distribution that exceeds the critical value for the null hypothesis"

$$\text{Power} = 1 - \text{Fprob}(F, \text{df}_n, \text{df}_d, \omega) \tag{1}$$

This is a non-central F calculator, widely available

Using the non-centrality parameter

With non-centrality parameter, power is easy to calculate

Power is the "proportion under non-central F distribution that exceeds the critical value for the null hypothesis"

$$\text{Power} = 1 - \text{Fprob}(F, \text{df}_n, \text{df}_d, \omega) \quad (1)$$

This is a non-central F calculator, widely available

Using the non-centrality parameter

With non-centrality parameter, power is easy to calculate
Power is the "proportion under non-central F distribution that exceeds the critical value for the null hypothesis"

$$\text{Power} = 1 - \text{Fprob}(F, \text{df}_n, \text{df}_d, \omega) \tag{1}$$

This is a non-central F calculator, widely available

Using the non-centrality parameter

With non-centrality parameter, power is easy to calculate
Power is the "proportion under non-central F distribution that exceeds the critical value for the null hypothesis"

$$\text{Power} = 1 - \text{Fprob}(F, \text{df}_n, \text{df}_d, \omega) \tag{1}$$

This is a non-central F calculator, widely available

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- GLIMMIX in SAS
    - Standard LM
    - Generalized LM
    - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed GLIMMIX-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- GLIMMIX in SAS
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed GLIMMIX-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- `GLIMMIX` in `SAS`
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed `GLIMMIX`-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- GLIMMIX in SAS
    - Standard LM
    - Generalized LM
    - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed GLIMMIX-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- `GLIMMIX` in `SAS`
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed `GLIMMIX`-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- GLIMMIX in SAS
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed GLIMMIX-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- `GLIMMIX` in `SAS`
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed `GLIMMIX`-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- GLIMMIX in SAS
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed GLIMMIX-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- `GLIMMIX` in `SAS`
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed `GLIMMIX`-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- `GLIMMIX` in `SAS`
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed `GLIMMIX`-Power

Obtaining the appropriate F value

- Obtained from analysis tool
- General analysis tools are available
- `GLIMMIX` in SAS
  - Standard LM
  - Generalized LM
  - Multi-level models
- Muller, O'Brien, Stroup: GLM
- Stroup approach: Parameters in program
- Create exemplar dataset

Termed `GLIMMIX`-Power

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

Exemplar dataset

- Defined by problem
- Means and variances
- Covariances: Define RM models
- More difficult: Linear growth model

Key: Exemplar dataset is not random dataset
Structure defined by power problem
Structure reflects actual behavior

General method for power analysis: GLIMMIX-Power

The general approach:

- Define exemplar dataset
- Use appropriate computational method
- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
- Compute power $= 1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$

General method for power analysis: GLIMMIX-Power

The general approach:

- Define exemplar dataset
- Use appropriate computational method
- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
- Compute power $= 1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$

General method for power analysis: GLIMMIX-Power

The general approach:

- Define exemplar dataset
- Use appropriate computational method
- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
- Compute power $= 1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$

General method for power analysis: GLIMMIX-Power

The general approach:

- Define exemplar dataset
- Use appropriate computational method
- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
- Compute power $=1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$

General method for power analysis: GLIMMIX-Power

The general approach:

- Define exemplar dataset
- Use appropriate computational method
- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $F_c = \mathrm{Finv}(\mathrm{df}_n, \mathrm{df}_d, .05)$
- Compute power $= 1 - \mathrm{Fprob}(F_c, \mathrm{df}_n, \mathrm{df}_d, \varphi)$

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\text{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \text{df}_d$
  - Compute GLIMMIX-Power $= 1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$
  - Search to find n with power $>$ power$_c$

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\mathrm{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \mathrm{Finv}(\mathrm{df}_n, \mathrm{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \mathrm{df}_d$
  - Compute `GLIMMIX`-Power $=1 - \mathrm{Fprob}(F_c, \mathrm{df}_n, \mathrm{df}_d, \varphi)$
  - Search to find n with power $>$ power$_c$

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\text{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \text{df}_d$
  - Compute GLIMMIX-Power $= 1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$
  - Search to find n with power $>$ power_c

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi / \mathrm{df}_d$ (correct for sample size)
- Search:
    - Select n
    - Compute df_d from n
    - Compute $F_c = \mathrm{Finv}(\mathrm{df}_n, \mathrm{df}_d, .05)$
    - Compute $\varphi = \varphi_c * \mathrm{df}_d$
    - Compute GLIMMIX-Power $= 1 - \mathrm{Fprob}(F_c, \mathrm{df}_n, \mathrm{df}_d, \varphi)$
    - Search to find n with power $>$ power_c

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\text{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \text{df}_d$
  - Compute GLIMMIX-Power $=1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$
  - Search to find n with power $>$ power_c

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\mathrm{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \mathrm{Finv}(\mathrm{df}_n, \mathrm{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \mathrm{df}_d$
  - Compute GLIMMIX-Power $=1 - \mathrm{Fprob}(F_c, \mathrm{df}_n, \mathrm{df}_d, \varphi)$
  - Search to find n with power $>$ power_c

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\text{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \text{df}_d$
  - Compute GLIMMIX-Power $=1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$
  - Search to find n with power > power_c

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\text{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \text{df}_d$
  - Compute `GLIMMIX`-Power $= 1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$
  - Search to find n with power $>$ power_c

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\mathrm{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \mathrm{Finv}(\mathrm{df}_n, \mathrm{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \mathrm{df}_d$
  - Compute `GLIMMIX`-Power $=1 - \mathrm{Fprob}(F_c, \mathrm{df}_n, \mathrm{df}_d, \varphi)$
  - Search to find n with power $>$ power$_c$

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\mathrm{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \mathrm{Finv}(\mathrm{df}_n, \mathrm{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \mathrm{df}_d$
  - Compute GLIMMIX-Power $=1 - \mathrm{Fprob}(F_c, \mathrm{df}_n, \mathrm{df}_d, \varphi)$
  - Search to find n with power $>$ power$_c$

Search: Simple grid search

General method for sample size determination

The general approach:

- Compute $\varphi = F * rank(\mathbf{K})$
- Compute $\varphi_c = \varphi/\text{df}_d$ (correct for sample size)
- Search:
  - Select n
  - Compute df_d from n
  - Compute $F_c = \text{Finv}(\text{df}_n, \text{df}_d, .05)$
  - Compute $\varphi = \varphi_c * \text{df}_d$
  - Compute `GLIMMIX`-Power $=1 - \text{Fprob}(F_c, \text{df}_n, \text{df}_d, \varphi)$
  - Search to find n with power $>$ power_c

Search: Simple grid search

Alternative: Simulation

Simulation is often used

Determine empirical rejection rate

Problems with simulation

- Simulation datasets are NOT correct
- Long-run behavior:
  - Simulated parameter ests = model-defined
  - Long-run correct
  - Assumes symmetry in φ
- Not clear how well simulation works

Alternative: Simulation

Simulation is often used

Determine empirical rejection rate

Problems with simulation

- Simulation datasets are NOT correct
- Long-run behavior:
  - Simulated parameter ests = model-defined
  - Long-run correct
  - Assumes symmetry in φ
- Not clear how well simulation works

Alternative: Simulation

Simulation is often used

Determine empirical rejection rate

Problems with simulation

- Simulation datasets are NOT correct
- Long-run behavior:
  - Simulated parameter ests = model-defined
  - Long-run correct
  - Assumes symmetry in φ
- Not clear how well simulation works

Alternative: Simulation

Simulation is often used

Determine empirical rejection rate

Problems with simulation

- Simulation datasets are NOT correct
- Long-run behavior:
  - Simulated parameter ests = model-defined
  - Long-run correct
  - Assumes symmetry in φ
- Not clear how well simulation works

Alternative: Simulation

Simulation is often used

Determine empirical rejection rate

Problems with simulation

- Simulation datasets are NOT correct
- Long-run behavior:
  - Simulated parameter ests = model-defined
  - Long-run correct
  - Assumes symmetry in φ
- Not clear how well simulation works

Alternative: Simulation

Simulation is often used

Determine empirical rejection rate

Problems with simulation

- Simulation datasets are NOT correct
- Long-run behavior:
    - Simulated parameter ests = model-defined
    - Long-run correct
    - Assumes symmetry in φ
- Not clear how well simulation works

Two sample test: Basic model

Two sample test setup

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon \quad (2)$$

$$\mathbf{Y} \sim N(\mu, \sigma^2\mathbf{I}) \quad (3)$$

$$\text{Test statistic:} \quad \mathbf{c}\beta - \theta \quad (4)$$

Power: Well-defined closed-form

Two sample test: Basic model

Two sample test setup

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon \quad (2)$$

$$\mathbf{Y} \sim N(\mu, \sigma^2\mathbf{I}) \quad (3)$$

$$\text{Test statistic:} \quad \mathbf{c}\beta - \theta \quad (4)$$

Power: Well-defined closed-form

Two sample test: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset
- Group i: Impose μ_i and σ_i^2, n_i obs
- Compute F using analysis method
- Compute power and/or sample size as above

Two sample test: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset
- Group i: Impose μ_i and σ_i^2, n_i obs
- Compute F using analysis method
- Compute power and/or sample size as above

Two sample test: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset
- Group i: Impose μ_i and σ_i^2, n_i obs
- Compute F using analysis method
- Compute power and/or sample size as above

Two sample test: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset
- Group i: Impose μ_i and σ_i^2, n_i obs
- Compute F using analysis method
- Compute power and/or sample size as above

Two sample test: Comparisons

Three power methods compared

- Closed form (`PROC POWER`)
- `GLIMMIX`-Power
- Simulation: ES: (.25 .5 1), n_i: (10,20,30,50,75,100), 2000 reps, examine subsets

Two sample test: Comparisons

Three power methods compared

- Closed form (`PROC POWER`)
- `GLIMMIX`-Power
- Simulation: ES: (.25 .5 1), n_i: (10,20,30,50,75,100), 2000 reps, examine subsets

Two sample test: Comparisons

Three power methods compared

- Closed form (PROC POWER)
- GLIMMIX-Power
- Simulation: ES: (.25 .5 1), n_i: (10,20,30,50,75,100), 2000 reps, examine subsets

Two sample test: Results

				Simulation, q3, Power		
n1,n2	Power	Diff	N-GLIM	100	250	500
10	0.083	0	457	0.025	0.016	0.012
20	0.120	0	482	0.040	0.022	0.015
30	0.159	0	490	0.040	0.024	0.021
50	0.236	0	497	0.046	0.036	0.025
75	0.331	0	500	0.051	0.031	0.027
100	0.438	0	502	0.065	0.042	0.038

Diff: Compares `POWER` and `GLIMMIX`-based power

Simulation values can be discrepant, esp in small reps

Two sample test: Results

				Simulation, q3, Power		
n1,n2	Power	Diff	N-GLIM	100	250	500
10	0.083	0	457	0.025	0.016	0.012
20	0.120	0	482	0.040	0.022	0.015
30	0.159	0	490	0.040	0.024	0.021
50	0.236	0	497	0.046	0.036	0.025
75	0.331	0	500	0.051	0.031	0.027
100	0.438	0	502	0.065	0.042	0.038

Diff: Compares `POWER` and `GLIMMIX`-based power
Simulation values can be discrepant, esp in small reps

Multilevel RM: Basic model

Multilevel RM setup

$$\mathbf{Y} = \mathbf{X}\beta\epsilon \quad (5)$$

$$\mathbf{Y} \sim N(\mu, \mathbf{\Sigma}) \quad (6)$$

$$\text{Test statistic:} \quad \mathbf{c}\beta\mathbf{u} = \theta \quad (7)$$

$$(8)$$

pause Less common situation

Power: No well-defined approach

Multilevel RM: Basic model

Multilevel RM setup

$$\mathbf{Y} = \mathbf{X}\beta\epsilon \quad (5)$$

$$\mathbf{Y} \sim N(\mu, \mathbf{\Sigma}) \quad (6)$$

$$\text{Test statistic:} \quad \mathbf{c}\beta\mathbf{u} = \theta \quad (7)$$

$$(8)$$

pause Less common situation

Power: No well-defined approach

Multilevel RM: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset, n_i obs
- Orthogonalize each group, impose μ_{it} and Σ structure
- Compute F using analysis method, power and/or sample size

Multilevel RM: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset, n_i obs
- Orthogonalize each group, impose μ_{it} and Σ structure
- Compute F using analysis method, power and/or sample size

Multilevel RM: Exemplar dataset

Exemplar dataset construction process:

- Generate random dataset, n_i obs
- Orthogonalize each group, impose μ_{it} and Σ structure
- Compute F using analysis method, power and/or sample size

Multilevel RM: Comparisons

Three power methods compared

- Closed form (`PROC GLMPOWER`) (multivariate)
- `GLIMMIX`-Power
- Simulation: Similar to first, addition of covariance structure

Multilevel RM: Comparisons

Three power methods compared

- Closed form (`PROC GLMPOWER`) (multivariate)
- `GLIMMIX`-Power
- Simulation: Similar to first, addition of covariance structure

Multilevel RM: Comparisons

Three power methods compared

- Closed form (`PROC GLMPOWER`) (multivariate)
- `GLIMMIX`-Power
- Simulation: Similar to first, addition of covariance structure

Multilevel RM: Results

		Power Est		abs(Diff) - Sim	
den df	F	gp	SAS	vs gp	vs SAS
36	2.19	0.419	0.268	0.041	0.172
36	3.16	0.569	0.677	0.035	0.103
36	1.67	0.328	0.366	0.036	0.052
76	1.58	0.325	0.225	0.031	0.127
76	2.31	0.456	0.333	0.046	0.135
76	1.63	0.335	0.262	0.033	0.102

GLIMMIX-based power better than PROC GLMPOWER

Many other situations, GLIMMIX-based power better

Multilevel RM: Results

		Power Est		abs(Diff) - Sim	
den df	F	gp	SAS	vs gp	vs SAS
36	2.19	0.419	0.268	0.041	0.172
36	3.16	0.569	0.677	0.035	0.103
36	1.67	0.328	0.366	0.036	0.052
76	1.58	0.325	0.225	0.031	0.127
76	2.31	0.456	0.333	0.046	0.135
76	1.63	0.335	0.262	0.033	0.102

`GLIMMIX`-based power better than `PROC GLMPOWER`

Many other situations, `GLIMMIX`-based power better

RGCM: Basic model

$$Y_{gij} = a_{gi} + b_{gi}x_{gij} + e_{gij} \quad (9)$$

$$\text{RGCM} : Y_{ij} = a_i + b_i x_{ij} + e_{ij} \quad (10)$$

$$\text{RM} : Y_{ij} = F_{ij} + e_{ij} \quad (11)$$

Relationship: (means are obvious)

$$\sigma_t^2 = \sigma_a^2 + t^2\sigma_b^2 + \sigma_e^2 \quad (12)$$

$$\sigma_{tv} = \left(\sigma_a^2 + vt\sigma_b^2\right) / \left(\sigma_t \sigma_v\right) \quad (13)$$

- Impose μ_{it} and Σ structure, Eq 12-13
- Compute F using analysis method, power & sample size

RGCM: Basic model

$$Y_{gij} = a_{gi} + b_{gi} x_{gij} + e_{gij} \tag{9}$$

$$\text{RGCM} : Y_{ij} = a_i + b_i x_{ij} + e_{ij} \tag{10}$$

$$\text{RM} : Y_{ij} = F_{ij} + e_{ij} \tag{11}$$

Relationship: (means are obvious)

$$\sigma_t^2 = \sigma_a^2 + t^2 \sigma_b^2 + \sigma_e^2 \tag{12}$$

$$\sigma_{tv} = \left(\sigma_a^2 + vt\sigma_b^2\right) / \left(\sigma_t \sigma_v\right) \tag{13}$$

- Impose μ_{it} and Σ structure, Eq 12-13
- Compute F using analysis method, power & sample size

RGCM: Comparisons

Two power methods compared

- No closed form approach
- GLIMMIX-Power
- Simulation: Similar to previous, Δ slope

RGCM: Comparisons

Two power methods compared

- No closed form approach
- `GLIMMIX`-Power
- Simulation: Similar to previous, Δ slope

RGCM: Comparisons

Two power methods compared

- No closed form approach
- `GLIMMIX`-Power
- Simulation: Similar to previous, Δ slope

RGCM: Comparisons

Two power methods compared

- No closed form approach
- `GLIMMIX`-Power
- Simulation: Similar to previous, Δ slope

RGCM: Comparisons

Two power methods compared

- No closed form approach
- `GLIMMIX`-Power
- Simulation: Similar to previous, Δ slope

RCGM: Results

Same setup, different sample sizes

Test	n	Power	abs(diff)	S Size
Grp:V=6	10	0.145	0.039	194
	20	0.238	0.016	201
	30	0.340	0.077	196
	40	0.421	0.108	202
vis*grp	10	0.204	0.029	121
	20	0.349	0.015	127
	30	0.497	0.078	124
	40	0.599	0.054	129

Power increases with sample size appropriately, optimal n same
Occasional computational issues

RCGM: Results

Same setup, different sample sizes

Test	n	Power	abs(diff)	S Size
Grp:V=6	10	0.145	0.039	194
	20	0.238	0.016	201
	30	0.340	0.077	196
	40	0.421	0.108	202
vis*grp	10	0.204	0.029	121
	20	0.349	0.015	127
	30	0.497	0.078	124
	40	0.599	0.054	129

Power increases with sample size appropriately, optimal n same

Occasional computational issues

RCGM: Results

Same setup, different sample sizes

Test	n	Power	abs(diff)	S Size
Grp:V=6	10	0.145	0.039	194
	20	0.238	0.016	201
	30	0.340	0.077	196
	40	0.421	0.108	202
vis*grp	10	0.204	0.029	121
	20	0.349	0.015	127
	30	0.497	0.078	124
	40	0.599	0.054	129

Power increases with sample size appropriately, optimal n same
Occasional computational issues

Conclusions about GLIMMIX-Power

1. Method is much quicker
2. Values consistent with other approaches
3. Some unresolved computation issues
4. Extension to interim power natural

Conclusions about GLIMMIX-Power

1. Method is much quicker
2. Values consistent with other approaches
3. Some unresolved computation issues
4. Extension to interim power natural

Conclusions about GLIMMIX-Power

1. Method is much quicker
2. Values consistent with other approaches
3. Some unresolved computation issues
4. Extension to interim power natural

Conclusions about GLIMMIX-Power

1. Method is much quicker
2. Values consistent with other approaches
3. Some unresolved computation issues
4. Extension to interim power natural