Package ‘JMdesign’

January 12, 2022

Type Package

Title Joint Modeling of Longitudinal and Survival Data - Power Calculation

Version 1.2

Date 2022-1-12

Author Emil A. Cornea, Liddy M. Chen, Bahjat F. Qaqish, Haitao Chu, and Joseph G. Ibrahim

Maintainer Shannon T. Holloway <sthollow@ncsu.edu>

Description Performs power calculations for joint modeling of longitudinal and survival data with k-th order trajectories when the variance-covariance matrix, Sigma_theta, is unknown.

License GPL-2

Depends methods

NeedsCompilation no

Repository CRAN

Date/Publication 2022-01-12 22:42:44 UTC

R topics documented:

JMdesign-package .................................................. 1
powerLongSurv .................................................. 5
powerLongSurv-class .............................................. 8
show-methods .................................................... 9

Index 10

---

JMdesign-package Joint Modeling of Longitudinal and Survival Data - Power Calculation

Description

R function for power determination in joint modeling of longitudinal and survival data with k-th order trajectories and unknown variance-covariance matrix Sigma_theta.
The package contains the R-function `powerLongSurv` to perform power calculations for joint modeling of longitudinal and survival data when trajectories are of k-th order and the variance-covariance matrix Sigma_theta is unknown.

**Author(s)**

Emil A. Cornea, Liddy M. Chen, Bahjat F. Qaqish, Haitao Chu, and Joseph G. Ibrahim

Maintainer: Shannon T. Holloway <sthollow@ncsu.edu>

**References**


**See Also**

`powerLongSurv, powerLongSurv-class, show-methods`

**Examples**

```
## Example 1.
## **********
## Input elements of Sigma_theta in forumula 4.6;
SigmaTheta <- matrix(c(1.2, 0.0, 0.0, 0.0, 0.7, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.8), nrow=3, ncol=3)

N <- 200; # Total sample size;
nevents <- 140; # Number of events;
tmedian <- 0.7; # median survival;
meantf <- 1.4; # mean follow-up time;
beta <- 0.2; # Effect of the trajectory;
alp ha <- 0.05; # Type-I Error (2-sided);
sigmae_2 <- 0.09; # measurement error;

## schedule of measurement;
t <- c(0.4, 0.8, 1.2, 1.6, 2); # maximum 2 year follow-up;

## Input estimated proportion subjects with 2,3,4,5,6 measurements;
## This is \( \xi \) in formula 4.6;
The data is obtained from the simulated data for the calculation in table 2;
p <- c(0.3, 0.4, 0.15, 0.1, 0.05);

## Input the order of trajectories
ordtraj <- 1 # linear trajectories
```
## Call function

### Linear Trajectories

```r
pLSl <- powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj, beta, alpha=0.05)
```

```r
pLSl
show(pLSl)
unclass(pLSl)
```

### Constant Trajectories

```r
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj=0, beta, alpha=0.05)
```

### Quadratic Trajectories

```r
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj=2, beta, alpha=0.05)
```

### Example 2.

#### Input elements of Sigma_theta in formula 4.6;

```r
SigmaTheta <- matrix(c(1.2,0.0,0.0,0.0,0.7,0.0,0.0,0.0,0.8),nrow=3,ncol=3)
```

N <- 200; # Total sample size;
nevents <- 140; # Number of events;
tmedian <- 0.7; # median survival;
meantf <- 1.4; # mean follow-up time;
beta <- 0.2; # Effect of the trajectory;
alpha <- 0.05;# Type-I Error (2-sided);
sigmae_2 <- 0.09; # measurement error;

#### schedule of measurement;

t <- c(0.4, 0.8, 1.2, 1.6);

#### Input estimated proportion subjects with 2,3,4,5,6 measurements;

This is \( \xi \) in formula 4.6;

The data is obtained from the simulated data for the calculation in table 2;

```r
p <- c(0.3, 0.4, 0.2, 0.1);
```

#### Input the order of trajectories

```r
ordtraj <- 2 # quadratic trajectories
```

#### Call function

### Quadratic Trajectories

```r
pLSq <- powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj, beta, alpha=0.05)
```

```r
pLSq
show(pLSq)
unclass(pLSq)
```

### Constant Trajectories

```r
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2,
```

```r
## Quadratic Trajectories

```r
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj=2, beta, alpha=0.05)
```
powerLongSurv


# Linear Trajectories

```
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, 
ordtraj=1, beta, alpha=0.05)
```

---

**powerLongSurv**  
*Power calculation in joint modeling of longitudinal and survival data - k-th Order Trajectories and Unknown Sigma*

---

**Description**

Compute the power in joint modeling of longitudinal and survival data when the variance-covariance matrix Sigma_Theta is unknown and the trajectories are order k.

The function computes power for a one-sided test, either

\[ H_0 : \beta = 0 \quad \text{and} \quad H_{1A} : \beta > 0 \]

or

\[ H_0 : \beta = 0 \quad \text{and} \quad H_{1B} : \beta < 0 \]

with Type I error $\alpha$. The choice of the alternative is determined by the sign of $\beta$. Negative values for $\beta$ indicate that the alternative hypothesis is $H_{1B}$, while $\beta \geq 0$ indicates that it is $H_{1A}$.

It creates a `powerLongSurv` object.

**Usage**

```
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, 
ordtraj, beta = 0, alpha = 0.05, tol = 1.5e-8)
```

**Arguments**

- **N**: numeric specifying the total sample size; minimum 20.
- **nevents**: numeric specifying the number of events; at least 20 and at most N.
- **tmedian**: numeric specifying the median survival time; positive
- **meantf**: numeric specifying the mean follow-up time; positive and no greater than max(t).
- **p**: numeric vector of estimated subject proportions with 2,3,... measurements, respectively, zero proportions allowed.
- **t**: numeric vector of measurement times, distinct positive components; same length as p.
- **SigmaTheta**: numeric matrix specifying the covariance matrix Sigma_Theta
- **sigmae_2**: numeric specifying the measurement error; positive.
- **ordtraj**: integer specifying the order of trajectories, must be less the order of Sigma_Theta
- **beta**: numeric specifying the effect of the trajectory; default value 0.
- **alpha**: numeric, strictly between 0.0 and 1.0, specifying the Type-I Error (2-sided), default value 0.05.
- **tol**: numeric, For floating point objects x and y, if $|x-y| \leq tol$, $x=y$. Passed to R function all.equal.
Details

The function `powerLongSurv` is used to calculate the power in joint modeling of longitudinal and survival data.

Value

An object of S4 class `powerLongSurv`, which has the following 12 components:

- `title`: character string
- `subtitle`: character string
- `t`: numeric vector
- `p`: numeric vector
- `N`: integer
- `nevents`: integer
- `censr`: numeric
- `tmedian`: numeric
- `meantf`: numeric
- `SigmaTheta`: numeric matrix
- `ordtraj`: integer
- `BSigma`: numeric matrix
- `beta`: numeric
- `alpha`: numeric
- `power`: numeric

Author(s)

Emil A. Cornea, Liddy M. Chen, Bahjat F. Qaqish, Haitao Chu, and Joseph G. Ibrahim

Maintainer: Shannon T. Holloway <sthollow@ncsu.edu>

References


See Also

`powerLongSurv-class`, `show-methods`
Examples

## Example 1.
## **********
## Input elements of Sigma_theta in formula 4.6;
SigmaTheta <- matrix(c(1.2,0.0,0.0,0.0,0.7,0.0,0.0,0.0,0.8),nrow=3,ncol=3)

N <- 200; # Total sample size;
nevents <- 140; # Number of events;
tmedian <- 0.7; # median survival;
tmeanf <- 1.4; # mean follow-up time;
beta <- 0.2; # Effect of the trajectory;
alpha <- 0.05; # Type-I Error (2-sided);
sigmae_2 <- 0.09; # measurement error;

## schedule of measurement;
t <- c(0.4, 0.8, 1.2, 1.6, 2) ; # maximum 2 year follow-up;

## Input estimated proportion subjects with 2,3,4,5,6 measurements;
## This is \xi in formula 4.6;
## The data is obtained from the simulated data for the calculation in table 2;
p <- c(0.3, 0.4, 0.15, 0.1, 0.05);

## Input the order of trajectories
ordtraj <- 1 ## linear trajectories

## Call function
## Linear Trajectories
pLSl <- powerLongSurv(N, nevents, tmedian, tmeanf, p, t, SigmaTheta, 
sigmae_2, ordtraj, beta, alpha=0.05)

pLSl
show(pLSl)
unclass(pLSl)

## Constant Trajectories
powerLongSurv(N, nevents, tmedian, tmeanf, p, t, SigmaTheta, sigmae_2, 
ordtraj=0, beta, alpha=0.05)

## Quadratic Trajectories
powerLongSurv(N, nevents, tmedian, tmeanf, p, t, SigmaTheta, sigmae_2, 
ordtraj=2, beta, alpha=0.05)

## **************************************************************************
## Example 2.
## **********
## Input elements of Sigma_theta in formula 4.6;
SigmaTheta <- matrix(c(1.2,0.0,0.0,0.0,0.7,0.0,0.0,0.0,0.8),nrow=3,ncol=3)

N <- 200; # Total sample size;
nevents <- 140; # Number of events;
tmedian <- 0.7; # median survival;
tmeanf <- 1.4; # mean follow-up time;
beta <- 0.2; # Effect of the trajectory;
alpha <- 0.05; # Type-I Error (2-sided);
sigmae_2 <- 0.09; # measurement error;

## schedule of measurement;
t <- c(0.4, 0.8, 1.2, 1.6);

## Input estimated proportion subjects with 2,3,4,5,6 measurements;
## This is \( \xi \) in formula 4.6;
## The data is obtained from the simulated data for the calculation in table 2;
p <- c(0.3, 0.4, 0.2, 0.1);

## Input the order of trajectories
ordtraj <- 2 # quadratic trajectories

## Call function
## Quadratic Trajectories
pLSq <- powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj, beta, alpha = 0.05)
pLSq
show(pLSq)
unclass(pLSq)

## Constant Trajectories
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj=0, beta, alpha=0.05)

## Linear Trajectories
powerLongSurv(N, nevents, tmedian, meantf, p, t, SigmaTheta, sigmae_2, ordtraj=1, beta, alpha=0.05)

---

**powerLongSurv-class**  
Class "powerLongSurv"

**Description**

Class of objects like the output of function "powerLongSurv()".

**Objects from the Class**

Objects can be created by calls of the form `new("powerLongSurv", ...)`.

**Slots**

- title: Object of class "character"
- subtitle: Object of class "character"
- t: Object of class "vector"
- p: Object of class "vector"
- N: Object of class "integer"
show-methods

nevents: Object of class "integer"
censr: Object of class "numeric"
tmedian: Object of class "numeric"
meantf: Object of class "numeric"
SigmaTheta: Object of class "matrix"
ordtraj: Object of class "integer"
BSigma: Object of class "matrix"
beta: Object of class "numeric"
alPHA: Object of class "numeric"
power: Object of class "numeric"

Methods

show signature(object = "powerLongSurv")

Author(s)

Emil A. Cornea, Liddy M. Chen, Bahjat F. Qaqish, Haitao Chu, and Joseph G. Ibrahim
Maintainer: Shannon T. Holloway <sthollow@ncsu.edu>

See Also

powerLongSurv, show-methods

Examples

showClass("powerLongSurv")

Description

Methods for function show

Methods

signature(object = "powerLongSurv")
Index

* **classes**
  * powerLongSurv-class, 8
* **functions**
  * powerLongSurv, 5
* **methods**
  * show-methods, 9
* **package**
  * JMdesign-package, 1

JMdesign (JMdesign-package), 1
JMdesign-package, 1
JMdesign-powerLongSurv (powerLongSurv), 5
JMdesign-powerLongSurv-class (powerLongSurv-class), 8

powerLongSurv, 3, 5, 9
powerLongSurv-class, 8

show, powerLongSurv-method (show-methods), 9
show-methods, 9