# Package ‘asypow’

June 26, 2015

**Title** Calculate Power Utilizing Asymptotic Likelihood Ratio Methods

**Version** 2015.6.25

**Date** 2015-06-25

**Author** S original by Barry W. Brown, James Lovato and Kathy Russel.
R port by Kjetil B Halvorsen <kjetil1001@gmail.com>

**Description** A set of routines written in the S language
that calculate power and related quantities utilizing asymptotic
likelihood ratio methods.

**Maintainer** Kjetil B Halvorsen <kjetil1001@gmail.com>

**Depends** R (>= 2.2.0), stats

**LazyLoad** TRUE

**License** ACM | file LICENSE

**NeedsCompilation** no

**License_restricts_use** yes

**Repository** CRAN

**Date/Publication** 2015-06-26 10:55:16

## R topics documented:

---

`asypow.n` *Asymptotic Sample Size*

---

## Description

Calculates the sample size required to obtain the desired power for a test via likelihood ratio methods.

## Usage

```
asypow.n(asypow.obj, power, significance)
```

## Arguments

| | |
|---|---|
| `asypow.obj` | The object returned from asypow.noncent. |
| `power` | The desired power of the test. |
| `significance` | The desired significance level of the test. |

## Value

Returns the sample size needed to achieve specified power at the specified significance level.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

`asypow.noncent`, `asypow.sig`, `asypow.power`

## Examples

```
# Three Sample Poisson Example :
#    Three independent Poisson processes produce events at
#    mean rates of 1, 2 and 3 per day. For how many days
#    must the processes be observed to have an 80% chance
#    of detecting that the means are different at an
#    overall significance level of 0.05?
# Step 1 : Find the information matrix
pois.mean <- c(1,2,3)
info.pois <- info.poisson.kgroup(pois.mean, group.size=3)
# Step 2: Create the constraints matrix
constraints <- matrix(c(2,1,2,2,2,3),ncol=3,byrow=TRUE)
# Step 3: Find the noncentrality parameter and
```

```
#        degrees of freedom for the test
poisson.object <- asypow.noncent(pois.mean, info.pois, constraints)
# Step 4: Compute sample size needed for
#        0.8 power with significance level 0.05
n.pois <- asypow.n(poisson.object, 0.8, 0.05)
# Step 5: Divide the sample size by 3 (the number of processes)
#        to get the number of days required.
n.days <- n.pois/3
print(n.days)
```

---

`asypow.noncent` *Asymptotic Noncentrality Parameter*

---

## Description

Given an information matrix, alternative hypothesis parameter values, and constraints that create the null hypothesis from the alternative, calculates noncentrality parameter, degrees of freedom and parameter value estimates under the null hypothesis.

## Usage

```
asypow.noncent(theta.ha, info.mat, constraints,
            nobs.ell=1, get.ho=TRUE)
```

## Arguments

`theta.ha` Array of parameter values under the alternative hypothesis.

`info.mat` The information matrix, the second derivate matrix of the expected log likelihood under the alternative hypothesis. The negative of the hessian matrix.

`constraints` The constraints which set the null hypothesis from the alternative hypothesis. They are in matrix form. CONSTRAINT[,1] is 1 for setting parameter to a value 2 for equality of two parameters CONSTRAINT[,2] is case on CONSTRAINT[,1] (1) Index of parameter to set to value (2) Index of one of two parameters to be set equal CONSTRAINT[,3] is case on CONSTRAINT[,1] (1) Value to which parameter is set (2) Index of other of two parameters to be set equal

`nobs.ell` The number of observations used in computing the information matrix. That is, info.mat is that for nobs.ell observations. Default is 1, which is the correct value for all of the 'info.' routines supplied here.

`get.ho` If TRUE, estimates of the parameter values under the null hypothesis are calculated and returned, otherwise not. Default is TRUE.

## Value

Returns a list including

`w` The noncentrality parameter for 1 observation.

`df` The degrees of freedom of the test

`theta.ho` Estimates of the parameter values under the null hypothesis.

### References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

### See Also

asypow.n, asypow.sig, asypow.power

### Examples

```
# Three Sample Poisson Example :
#    Three independent Poisson processes produce events at
#    mean rates of 1, 2 and 3 per day.
# Find the information matrix
pois.mean <- c(1,2,3)
info.pois <- info.poisson.kgroup(pois.mean,group.size=3)
# Create the constraints matrix
constraints <- matrix(c(2,1,2,2,2,3),ncol=3,byrow=TRUE)
# Calculate noncentrality parameter, degrees of freedom and parameter
# value estimates under the null hypothesis for the test.
poisson.object <- asypow.noncent(pois.mean,info.pois,constraints)
```

---

## asypow.power *Asymptotic Power*

---

### Description

Calculates the power of a test via likelihood ratio methods.

### Usage

```
asypow.power(asypow.obj, sample.size, significance)
```

### Arguments

| | |
|---|---|
| asypow.obj | The object returned from asypow.noncent. |
| sample.size | The sample size of the study. |
| significance | The significance level of the test. |

### Value

Returns the power of the test.

### References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

### See Also

asypow.noncent, asypow.n, asypow.sig

### Examples

```
# Single Group Binomial Example:
#    Consider testing the null hypothesis that the binomial
#    probability is p = .2 with a sample size of 47 and
#    signficance level of 0.05. What is the power of the
#    test if p is actually .4?
# Step 1: Find the information matrix
info.binom <- info.binomial.kgroup(.4)
# Step 2: Create the constraints matrix
constraints <- c(1, 1, .2)
# Step 3: Find the noncentrality parameter and
#         degrees of freedom for the test
binom.object <- asypow.noncent(.4, info.binom, constraints)
# Step 4: Compute the power of a test with
#         sample size of 47 and a significance level 0.05
power.binom <- asypow.power(binom.object, 47, 0.05)
print(power.binom)
```

---

## asypow.sig *Asymptotic Significance*

---

### Description

Calculates the significance level of a test via likelihood ratio methods.

### Usage

```
asypow.sig(asypow.obj, sample.size, power)
```

### Arguments

| | |
|---|---|
| asypow.obj | The object returned from asypow.noncent. |
| sample.size | The sample size of the test. |
| power | The power of the test. |

### Value

Returns the significance level of the test.

### References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

asypow.noncent, asypow.n, asypow.power

## Examples

```
# Single Group Binomial Example:
#    Consider testing the null hypothesis that the binomial
#    probability is p = .2 when the actual probability is .4.
#    What significance level corresponding to a sample
#    size of 47 and power of .8?
# Step 1: Find the information matrix
info.binom <- info.binomial.kgroup(.4)
# Step 2: Create the constraints matrix
constraints <- c(1, 1, .2)
# Step 3: Find the noncentrality parameter and
#         degrees of freedom for the test
binom.object <- asypow.noncent(.4, info.binom, constraints)
# Step 4: Compute the power of a test with
#         sample size of 47 and a significance level 0.05
sig.binom <- asypow.sig(binom.object, 47, 0.8)
print(sig.binom)
```

---

`info.binomial.design` *Expected Information Matrix for a Binomial Design*

---

## Description

Calculates the expected information matrix for a binomial design where the parameter $p$, probability of an event, depends on a covariate, $x$, through a logistic, $p = \exp(u)/(1 + \exp(u))$ p = exp(u)/(1+exp(u)), or complementary log, $p = 1-\exp(-\exp(u))$ p = 1 - exp(-exp(u)), model. The variable $u$ is either a linear, $u = a+bx$, or quadratic, $u = a+bx+cx^2$, function of the covariate $x$.

## Usage

```
info.binomial.design(model="linear", link="logistic", theta,
                     xpoints, natx=1, group.size=1)
```

## Arguments

| | |
|---|---|
| `theta` | Matrix of parameters for the linear combination of the covariate x. Each row represents a group so theta[i,] is c(a,b) if model = "linear" or theta[i,] is c(a,b,c) if model = "quadratic". |
| `xpoints` | Matrix of covariate values for each group. If there is only 1 group or all groups have the same covariate value, xpoints should be a vector; otherwise, the number of rows in xpoints must equal the number of rows in theta. |
| `model` | One of {"linear", "quadratic"}. Specifies the function of the covariate x that will be used. Linear indicates, $u = a+bx$, and quadratic indicates, $u = a+bx+cx^2$. Only enough to ensure a unique match need be supplied. |

`link` One of {"logistic", "complementary log"}. Specifies the link between the linear or quadratic combination of the covariate $x$ and the parameter of the Binomial model, $p$. Logistic indicates $p = \exp(u)/(1 + \exp(u))$, and complementary log indicates, $p = 1 - \exp(-\exp(u))$. Only enough to ensure a unique match need be supplied.

`natx` Needed only if there are unequal sample sizes used at any of the values in xpoints. At covariate value xpoint[i,j] there are natx[i,j] observations. If specified, the dimensions of natx should be the same as the dimensions of xpoints.

`group.size` Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified relative sizes. If this value is specified, it should be a vector whose length is the same as the number of rows in theta.

## Value

The information matrix for one observation for this design.

If model = "linear" and there are $k$ groups, the information matrix is a square $(2k) \times (2k)$ matrix which is indexed by the parameters (a,b) for group 1, then (a,b) for group 2, etc.

If model = "quadratic", the information matrix is a square $(3k) \times (3k)$ matrix which is indexed by the parameter (a,b,c) for group 1, then (a,b,c) for group 2, etc.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

`info.poisson.design`, `info.ordinal.design`, `info.expsurv.design`

## Examples

```
# Find the information matrix for a 2 group
# logistic binomial Design with a quadratic
# combination of covariate x and different
# sample sizes at each point
abc <- rbind(c(1.2, .9, .3),c(0.33, .21, .05))
covar <- c(1, 2, 3,  4,5)
sample.size <- rbind(c(10,11,12,10,9), c(8,7,10,8,9))
info.binom <- info.binomial.design(model="quadratic", link="logistic",
                                 theta = abc, xpoints = covar,
                                 natx=sample.size)
print(info.binom)
```

info.binomial.kgroup *Expected Information Matrix for Single or Multiple Group Binomial*

## Description

Calculates expected information matrix for a single observation for single or multiple group binomial distribution.

The natural null hypothesis for a single group is that that the probability is some specified value. For multiple groups, the natural null hypothesis is that the group probabilities are the same.

## Usage

```
info.binomial.kgroup(p, group.size=1)
```

## Arguments

| | |
|---|---|
| p | Scalar or vector of probability values. The i'th component is the (alternative hypothesis or true) probability of an event in the i'th group. |
| group.size | Needed only if there are several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified relative sizes. If this value is specified, it should be a vector whose length is the same as p. |

## Value

Expected information matrix for a single observation. The matrix is square with each dimension the number of groups.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

info.poisson.kgroup, info.ordinal.kgroup, info.expsurv.kgroup

## Examples

```
# Find the information matrix for a 2 sample binomial with
# probability of events .2 and .4 and sample sizes 10 and 11
info.binom <- info.binomial.kgroup(c(.2,.4), c(10,11))
print(info.binom)
```

## info.expsurv.design — *Expected Information Matrix for a Clinical Trial with Exponential Survival Design*

### Description

Calculates expected information matrix for a clinical trial with exponential survival.

The clinical trial will accrue subjects over a time period $L$. Each subject will enter the study at a random time between 0 and $L$, so the subject's follow up time, $U$, will be uniformly distributed between 0 and $L$. A subject with follow up time $U$, can die at a time $t$ between 0 and $U$, or the subject can be withdrawn alive at time $U$. The density of time to death is exponential distribution with hazard, $w$.

The parameter $w$ depends on a covariate, $x$, via the exponentiation of a linear or quadratic function of $x$, $w = \exp(a + bx)$ or $w = \exp(a + bx + cx^2)$.

This model is both the proportional hazards model and the accelerated failure model for exponential survival.

### Usage

```
info.expsurv.design(model="linear", theta, L, xpoints,
                    natx=1, group.size=1)
```

### Arguments

`theta` — Matrix of parameters for the linear combination of the covariate x. Each row represents a group so theta[i,] = c(a,b) if model = "linear" or theta[i,] = c(a,b,c) if model = "quadratic".

`L` — The length of the clinical trial. If all groups have the same trial length, L should be a single number; otherwise, L should be a vector the same length as the number of rows in w where L[i] is the length of the clinical trial for the i'th group.

`xpoints` — Matrix of covariate values for each group. If there is only 1 group or all groups have the same covariate value, xpoints should be a vector; otherwise, the number of rows in xpoints must each the number of rows in theta.

`model` — One of {"linear", "quadratic"}. Specifies the function of the covariate x that will be used. Linear indicates, $u = a+bx$, and quadratic indicates, $u = a+bx+cx^2$. Only enough to ensure a unique match need be supplied.

`natx` — Needed only if there are unequal sample sizes used at any of the values in xpoints. At covariate value xpoint[i,j] there are natx[i,j] observations. If specified, the dimensions of natx should be the same as the dimensions of xpoints.

`group.size` — Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified relative sizes. If this value is specified, it should be a vector whose length is the same as the number of columns in theta.

## Value

The information matrix for one observation for this design.

If model = "linear" and there are k groups, the information matrix is a square $(2k) \times (2k)$ matrix which is indexed by the parameters (a,b) for group 1, then (a,b) for group 2, etc.

If model = "quadratic", the information matrix is a square $(3k) \times (3k)$ matrix which is indexed by the parameter (a,b,c) for group 1, then (a,b,c) for group 2, etc.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

`info.binomial.design`, `info.poisson.design`, `info.ordinal.design`

## Examples

```
# Find the information matrix for a clinical trial
# with hazard w(x) = -0.848 + 0.7*x which lasts
# three years and has 10 x values equally spaced
# between -3 and 3 with equal sample sizes.
ab <- c(-.848, .7)
covar <- seq(-3, 3, length=10)
LL <- 3
info.expsurv <- info.expsurv.design(theta = ab, L = LL, xpoints = covar)
print(info.expsurv)
```

---

`info.expsurv.kgroup` *Expected Information Matrix for a Single or Multiple Group Clinical Trial with Exponential Survival*

---

## Description

Calculates expected information matrix for a single observation for a single or multiple group clinical trial with exponential survival.

The clinical trial will accrue subjects over a time period $L$. Each subject will enter the study at a random time between 0 and $L$, so the subject's follow up time, $U$, will be uniformly distributed between 0 and $L$. A subject with follow up time $U$, can die at a time $t$ between 0 and $U$, or the subject can be withdrawn alive at time $U$. The density of time to death is exponential distribution with hazard, $w$.

## Usage

```
info.expsurv.kgroup(w, L, group.size=1)
```

### Arguments

| | |
|---|---|
| `w` | Scalar or vector of exponential rates (reciprocals of the mean survival time ) for the groups. The i'th component is the (alternative hypothesis or true) rate of the i'th group. |
| `L` | The length of the clinical trial. If all groups have the same trial length, L should be a single number; otherwise, L should be a vector the same length as w where L[i] is the length of the clinical trial for the i'th group. |
| `group.size` | Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified sample size. If this value is specified, it should be a vector of the same length as lambda. |

### Value

Expected information matrix for a single observation. The matrix is square with dimension equal to the number of groups.

### References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

### See Also

`info.binomial.kgroup`, `info.poisson.kgroup`, `info.ordinal.kgroup`

### Examples

```
# Find the information matrix for a clinical trial of
# length 3 with hazard 1
info.expsurv <- info.expsurv.kgroup(1, 3)
print(info.expsurv)
```

---

## `info.mvlogistic` *Expected Information Matrix for a Multivariate Logistic Model*

---

### Description

Calculates the expected information matrix for a multivariate logistic model where the parameter $p$, probability of an event, depends on the covariates, $x = c(x[1], x[2], \ldots, x[n]$, through a logistic, $p = \exp(u)/(1 + \exp(u))$, model. The variable $u$ is a linear combination of the covariates via a set of coefficients, $\text{coef} = c(\text{coef}[1], \ldots, \text{coef}[n])$, $u = \sum_{i=1}^{n} \text{coef}[i]x[i]$.

The usual use of this routine is for tabulated data in which case the x's will all be 0 or 1 valued indicator variables.

## Usage

```
info.mvlogistic(coef, design, rss=1)
```

## Arguments

| | |
|---|---|
| coef | Vector of length p (number of covariates) giving coefficients of variables. |
| design | Matrix of dimension $n \times p$ each row of which gives values of covariates at one of the n design points.<br>Note: Most models will include a constant term and the column of design corresponding to this term will be identically 1. |
| rss | The relative sample size at each design point. The default is the same sample size at each design point. If changed from the default, rss should be a vector of length n. |

## Value

The information matrix for one observation for this design.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

info.mvloglin

## Examples

```
# Find the information matrix for a multivariate
# logistic design with variables x, y and z
#    Define coefficient matrix so that
#    u = 1 + .5*x + .7*y + .9*z
coef <- c(1, .5, .7, .9)
#    Define the design matrix so that there are 10 design points
intercept <- rep(1, 10)
x <- rnorm(10)
y <- rnorm(10)
z <- rnorm(10)
design <- cbind(intercept, x, y, z)
#    Use info.mvlogistic to find the information matrix for
#    this design
info.xyz <- info.mvlogistic(coef, design)
print(info.xyz)
```

`info.mvloglin` *Expected Information Matrix for a Multivariate Log-Linear Model*

## Description

Calculates the expected information matrix for a multivariate log-linear model where the parameter $p$, probability of an event, depends on the covariates, $x = c(x[1], \ldots, x[n])$, through an exponential, $p = \exp(u)$. The variable $u$ is a log-linear combination of the covariates via a set of coefficients, coef $= c(\text{coef}[1], \ldots, \text{coef}[n]$, $u = \sum_{i=1}^{n} \log(\text{coef}[i]x[i])$.

The usual use of this routine is for tabulated data in which case the x's will all be 0 or 1 valued indicator variables.

## Usage

```
info.mvloglin(coef, design, rss=1)
```

## Arguments

`coef` Vector of length p (number of covariates) giving coefficients of variables.

`design` Matrix of dimension $n \times p$ each row of which gives values of covariates at one of the n design points.

Note: Most models will include a constant term and the column of design corresponding to this term will be identically 1.

`rss` The relative sample size at each design point. The default is the same sample size at each design point. If changed from the default, rss should be a vector of length n.

## Value

The information matrix for one observation for this design.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

info.mvlogistic

## Examples

```
# Find the information matrix for a multivariate
# log-linear design with variables x, y and z
#    Define coefficient matrix so that
#    u = .1 + .2*x + .3*y + .3*z
coef <- c(.1, .2, .3, .4)
#    Define the design matrix so that there are 10 design points
```

```
intercept <- rep(1, 10)
x <- seq(.1, .2, length=10)
y <- seq(.25, .3, length=10)
z <- seq(.2, .3, length=10)
design <- cbind(intercept, x, y, z)
#    Use info.mvloglin to find the information matrix for
#    this design
info.xyz <- info.mvloglin(coef, design)
print(info.xyz)
```

---

`info.ordinal.design` *Expected Information Matrix for an Ordinal Design*

---

## Description

Calculates the expected information matrix for an ordinal design where the parameters $p_j$, probability of an event in category $j$ or less, depend on a covariate, $x$, through a logistic, $p_j = \exp(u_j)/(1+\exp(u_j)$, or complementary log, $p_j = 1 - \exp(-\exp(u_j))$, model. The variable $u_j$ is a linear, $u_j = a_j + bx$, or quadratic, $u_j = a_j + bx + cx^2$, function of the covariate $x$.

## Usage

```
info.ordinal.design(model="linear", link="logistic", theta,
                    xpoints, natx=1, group.size=1)
```

## Arguments

`theta` Matrix of parameters for the linear combination of the covariate x. Each row represents a group so if model = "linear"

$$\theta[i,] = c(a[1], a[2], a[3], \ldots, a[r-1], b)$$

where $r$ is the number of categories. If model = "quadratic"

$$\theta[i,] = c(a[1], a[2], \ldots, a[r-1], b, c)$$

theta[i,] = c(a[1],a[2],a[3],. . . ,a[r-1],b,c)

`xpoints` Matrix of covariate values for each group. If there is only 1 group or all groups have the same covariate value, xpoints should be a vector; otherwise, the number of rows in xpoints must equal the number of rows in theta.

`model` One of {"linear", "quadratic"}. Specifies the function of the covariate x that will be used. Linear indicates, $u_j = a_j + bx$, and quadratic indicates, $u_j = a_j + bx + cx^2, \quad j = 1, \ldots, r-1$. Only enough to ensure a unique match need be supplied.

`link` One of {"logistic", "complementary log"}. Specifies the link between the linear or quadratic combination of the covariate $x$ and the parameters of the ordinal model, $p_j$. Logistic indicates $p_j = \exp(u_j)/(1+\exp(u_j))$, and complementary log indicates, $p_j = 1 - \exp(-\exp(u_j)), \quad j = 1, \ldots, r-1$. Only enough to ensure a unique match need be supplied.

`natx` Needed only if there are unequal sample sizes used at any of the values in xpoints. At covariate value xpoint[i,j] there are natx[i,j] observations. If specified, the dimensions of natx should be the same as the dimensions of xpoints.

`group.size` Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified relative sizes. If this value is specified, it should be a vector whose length is the same as the number of rows in theta.

## Value

The information matrix for one observation for this design.

If model = "linear" and there are $r$ categories and $k$ groups, the information matrix is a square $(rk) \times (rk)$ matrix which is indexed by the parameters $a[1], a[2], \ldots, a[r-1], b$ for group 1, then $(a[1], a[2], \ldots, a[r-1], b)$ for group 2, etc.

If model = "quadratic", the information matrix is a square $((r+1)k) \times ((r+1)k)$ matrix which is indexed by the parameters $(a[1], a[2], \ldots, a[r-1], b, c)$ for group 1, then $(a[1], a[2], \ldots, a[r-1], b, c)$ for group 2, etc.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

`info.binomial.design`, `info.poisson.design`, `info.expsurv.design`

## Examples

```
# Find the information matrix for an ordinal design
# with one group and equal sample sizes.
# Assume 4 categories and use a logistic
# line and quadratic model. Let
# u[1] = 1 + 2.5*x
# u[2] = 2 + 2.5*x
# u[3] = 3 + 2.5*x
# Use values x = -3,0,3
theta <- c(1, 2, 3, 2.5)
covar <- c(-3, 0, 3)
info.ord <- info.ordinal.design(theta = theta, xpoints = covar)
print(info.ord)
```

info.ordinal.kgroup *Expected Information Matrix for Single or Multiple Group Ordinal Observations*

## Description

Calculates expected information matrix for a single observation for ordered outcomes in a single or multiple groups.

The natural null hypothesis for a single group is that the probabilities of the outcomes is some specified set of values. For multiple groups, the natural null hypothesis is that the probabilities are the same.

## Usage

```
info.ordinal.kgroup(p, group.size=1)
```

## Arguments

p
: If there are $n$ ordered outcome categories and $k$ groups then $p$ will have dimension $k \times (n-1)$. If the number of groups is one, then $p$ can be a vector of length (n-1).

  Within a single group, $p_i$ is the probability that the outcome of a trial is some category $j$ with $j \leq i$. Consequently, the $p_i$ must be monotonely increasing with $k$. If there are $n$ categories, of necessity $p_n = 1$, so this redundant value is not included.

  Within a single group, the probability that the outcome is category 1 is $p_1$. For $2 \leq i \leq n$ the probability that the outcome is category $k$ is $p_i - p_{i-1}$. The probability of outcome $n$ is $1 - p_{n-1}$.

group.size
: Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified sample size. If this value is specified, it should be a vector whose length is the row dimension of p.

## Value

Expected information matrix for a single observation. The matrix is dimensioned $(k(n-1)) \times (k(n-1))$ ( k*(n-1) ) X ( k*(n-1) ).

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

info.binomial.kgroup, info.poisson.kgroup, info.expsurv.kgroup

### Examples

```
# Find the information matrix for a 2 group ordinal
# model with 4 categories.
p1 <- c(.1, .2, .3)  # Probabilities for group 1
p2 <- c(.2, .5, .7)  # Probabilities for group 2
p <- rbind(p1,p2)
ngrps <- c(.4, .6)  # Percentage of data in each group
info.ord <- info.ordinal.kgroup(p, ngrps)
print(info.ord)
```

---

## `info.poisson.design` *Expected Information Matrix for a Poisson Design*

---

### Description

Calculates the expected information matrix for a Poisson design where the parameter $\lambda$, the mean of the distribution, depends on a covariate, $x$, via the exponentiation of a linear or quadratic function of $x$, $\lambda = \exp(a + bx)$ or $\lambda = \exp(a + bx + cx^2)$.

### Usage

```
info.poisson.design(model="linear", theta, xpoints,
                          natx=1, group.size=1)
```

### Arguments

| | |
|---|---|
| `theta` | Matrix of parameters for the linear combination of the covariate x. Each row represents a group so theta[i,] = c(a,b) if model = "linear" or theta[i,] = c(a,b,c) if model = "quadratic". |
| `xpoints` | Matrix of covariate values for each group. If there is only 1 group or all groups have the same covariate value, xpoints should be a vector; otherwise, the number of rows in xpoints must each the number of rows in theta. |
| `model` | One of {"linear", "quadratic"}. Specifies the function of the covariate x that will be used. Linear indicates, $u = a+bx$, and quadratic indicates, $u = a+bx+cx^2$. Only enough to ensure a unique match need be supplied. |
| `natx` | Needed only if there are unequal sample sizes used at any of the values in xpoints. At covariate value xpoint[i,j] there are natx[i,j] observations. If specified, the dimensions of natx should be the same as the dimensions of xpoints. |
| `group.size` | Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified relative sizes. If this value is specified, it should be a vector whose length is the same as the number of columns in theta. |

### Value

The information matrix for one observation for this design.

If model = "linear" and there are k groups, the information matrix is a square $(2k) \times (2k)$ matrix which is indexed by the parameters (a,b) for group 1, then (a,b) for group 2, etc.

If model = "quadratic", the information matrix is a square $(3k) \times (3k)$ matrix which is indexed by the parameter (a,b,c) for group 1, then (a,b,c) for group 2, etc.

### References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

### See Also

info.binomial.design, info.ordinal.design, info.expsurv.design

### Examples

```
# Find the information matrix for a 2 group
# logistic Poisson design with a quadratic
# combination of covariate x and different
# sample sizes at each point
abc <- rbind(c(1.2,.9,.3), c(0.33,.21,.05))
covar <- c(1, 2, 3, 4, 5)
sample.size <- rbind(c(10,11,12,10,9), c(8,7,10,8,9))
info.poiss <- info.poisson.design(model="quadratic",
                                  theta = abc, xpoints = covar,
                                  natx=sample.size)
print(info.poiss)
```

---

## info.poisson.kgroup *Expected Information Matrix for Single or Multiple Group Poisson*

---

### Description

Calculates expected information matrix for a single observation for single or multiple group Poisson distribution.

The natural null hypothesis for a single group is that that the mean is some specified value. For multiple groups, the natural null hypothesis is that the group means are the same.

### Usage

```
info.poisson.kgroup(lambda, group.size=1)
```

## Arguments

`lambda` Scalar or vector of Poisson means for the groups. The i'th component is the (alternative hypothesis or true) mean of the i'th group.

`group.size` Needed only if there are to be several groups with unequal sample sizes. The value of the i'th component is the relative sample size of the i'th group. The calculation made is for a single observation spread over the several groups in proportion to the specified sample size. If this value is specified, it should be a vector of the same length as lambda.

## Value

Expected information matrix for a single observation. The matrix is square with dimension equal to the number of groups.

## References

Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.

## See Also

`info.binomial.kgroup`, `info.ordinal.kgroup`, `info.expsurv.kgroup`

## Examples

```
# Find the information matrix for a 3 sample Poisson with
# means 1, 2 and 3 and equal sample sizes
info.pois <- info.poisson.kgroup(c(1,2,3))
print(info.pois)
```

---

`info.reparam` *Reparameterize Expected Information Matrix*

---

## Description

Calculates the expected information matrix after reparameterization of a model using the method of propagation of error.

## Usage

```
info.reparam(theta, info.mat, dg)
```

### Arguments

| | |
|---|---|
| theta | Matrix of parameters of the linear part of the model. Each row represents a group. This is under the original parameterization. |
| info.mat | The information matrix under the original parameterization. |
| dg | A function that computes the partial derivatives of g, the transformation function. Let g.i be the function which transforms the vector of old parameters, theta, into the i'th element of the new parameters. The function dg should take theta and return a matrix whose [i,j] element is the derivative of g.i with respect to theta[j] |

### Value

Returns the expected information matrix under the new parameterization.

### References

Bishop, Y.M., Fienberg, S.E., and Holland, P.W. (1975) *Discrete Multivariate analysis: Theory and Practice* MIT Press, Cambridge, Mass.
Cox, D.R. and Hinkley, D.V. (1974). *Theoretical Statistics* Chapman and Hall, London.
Tong, Y.L. (1990). *The Multivariate Normal Distribution* Springer-Verlag, New York.

### Examples

```
# A logistic model posits that the probability of response
# is a logtistic function of a + b*x.
# Consider the value of x that produces 50%
# response, x = -a/b. Since -a/b is not one of the parameters
# of the model, we must reparameterize to
# roe[1] = -a/b
# roe[2] = b
dg <- function(theta) {
# theta is a vector of length 2 containing c(a,b)
# dg <- [d{roe[1]}/d{a} d{roe[1]}/d{b}
#        d{roe[2]}/d{a} d{roe[2]}/d{b}]
  a <- theta[1]
  b <- theta[2]
  return(matrix(c(-1/b,a/b^2,0,1), nrow=2, ncol=2, byrow=TRUE))
}
# Let a = -0.9 and b = .7
theta <- c(-.9, .7)
# assign a set of covariate values
covar <- c(0.3, .9, 1.3, 2.5)
# Use info.binomial.design to calculate the information
# matrix under the original parameterization
info.orig <- info.binomial.design(model="linear", link="logistic",
                                  theta=theta, xpoints=covar)
# Get the information matrix of the reparameterized model
info.new <- info.reparam(theta, info.orig, dg)
print(info.new)
```

# Index