

# Package ‘ODS’

November 19, 2018

**Type** Package

**Title** Statistical Methods for Outcome-Dependent Sampling Designs

**Version** 0.2.0

**Author** Yinghao Pan [aut, cre],  
Haibo Zhou [aut],  
Mark Weaver [aut],  
Guoyou Qin [aut],  
Jianwen Cai [aut]

**Maintainer** Yinghao Pan <ypan8@uncc.edu>

## Description

Outcome-dependent sampling (ODS) schemes are cost-effective ways to enhance study efficiency. In ODS designs, one observes the exposure/covariates with a probability that depends on the outcome variable. Popular ODS designs include case-control for binary outcome, case-cohort for time-to-event outcome, and continuous outcome ODS design (Zhou et al. 2002) <doi: 10.1111/j.0006-341X.2002.00413.x>. Because ODS data has biased sampling nature, standard statistical analysis such as linear regression will lead to biased estimates of the population parameters. This package implements four statistical methods related to ODS designs: (1) An empirical likelihood method analyzing the primary continuous outcome with respect to exposure variables in continuous ODS design (Zhou et al., 2002). (2) A partial linear model analyzing the primary outcome in continuous ODS design (Zhou, Qin and Longnecker, 2011) <doi: 10.1111/j.1541-0420.2010.01500.x>. (3) Analyze a secondary outcome in continuous ODS design (Pan et al. 2018) <doi: 10.1002/sim.7672>. (4) An estimated likelihood method analyzing a secondary outcome in case-cohort data (Pan et al. 2017) <doi: 10.1111/biom.12838>.

**Depends** R (>= 3.5.0)

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.0

**Imports** cubature (>= 1.4-1), survival (>= 2.42-3), utils, stats

**URL** <https://github.com/Yinghao-Pan/ODS>

**BugReports** <https://github.com/Yinghao-Pan/ODS/issues>

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-11-19 17:50:03 UTC

## R topics documented:

Bfct . . . . .	2
casecohort_data_secondary . . . . .	3
Estimate_PLMODS . . . . .	4
gev_ODS . . . . .	6
logspace . . . . .	8
odsmle . . . . .	9
ods_data . . . . .	10
ods_data_secondary . . . . .	11
quantileknots . . . . .	12
se.spmle . . . . .	12
secondary_casecohort . . . . .	14
secondary_ODS . . . . .	16
<b>Index</b>	<b>18</b>

---

Bfct	<i>power basis functions of a spline of given degree</i>
------	--

---

### Description

Bfct returns the power basis functions of a spline of given degree.

### Usage

```
Bfct(x, degree, knots, der)
```

### Arguments

x	n by 1 matrix of the independent variable
degree	the order of spline
knots	the knots location
der	the der-order derivative. The default is 0

**Value**

$n$  by  $(1+\text{degree}+n\text{knots})$  matrix corresponding to the truncated power spline basis with knots and specified degree.

**Examples**

```
library(ODS)

x <- matrix(c(1,2,3,4,5),ncol=1)
degree <- 2
knots <- c(1,3,4)

Bfct(x, degree, knots)
```

---

casecohort\_data\_secondary

*Data example for the secondary analysis in case-cohort design*

---

**Description**

Data example for the secondary analysis in case-cohort design

**Usage**

```
casecohort_data_secondary
```

**Format**

A data frame with 1000 rows and 15 columns:

**subj\_ind** An indicator variable for each subject: 1 = SRS, 2 = supplemental cases, 0 = NVsample

**T** observation time for failure outcome

**Delta** event indicator

**Y2** a continuous secondary outcome

**X** expensive exposure

**Z11** first covariate in the linear regression model

**Z12** second covariate in the linear regression model

**Z13** third covariate in the linear regression model

**Z14** fourth covariate in the linear regression model

**Z21** first covariate in the Cox model

**Z22** second covariate in the Cox model

**Z23** third covariate in the Cox model

**Z31** first covariate that is related to the conditional distribution of X given other covariates

**Z32** second covariate that is related to the conditional distribution

**Z33** thid covariate that is related to the conditional distribution

**Source**

A simulated data set

---

Estimate_PLMODS	<i>Partial linear model for ODS data</i>
-----------------	--

---

**Description**

Estimate\_PLMODS computes the estimate of parameters in a partial linear model in the setting of outcome-dependent sampling. See details in Zhou, Qin and Longnecker (2011).

**Usage**

```
Estimate_PLMODS(X, Y, Z, n_f, eta00, q_s, Cpt, mu_Y, sig_Y, degree, nknots,
  tol, iter)
```

**Arguments**

X	n by 1 matrix of the observed exposure variable
Y	n by 1 matrix of the observed outcome variable
Z	n by p matrix of the other covariates
n_f	$n_f = c(n_0, n_1, n_2)$ , where $n_0$ is the SRS sample size, $n_1$ is the size of the supplemental sample chosen from $(-\infty, \mu_Y - a \cdot \text{sig}_Y)$ , $n_2$ is the size of the supplemental sample chosen from $(\mu_Y + a \cdot \text{sig}_Y, +\infty)$ .
eta00	a column matrix. $\text{eta00} = (\theta^T \pi^T v^T \text{sig0\_sq})^T$ where $\theta = (\alpha^T, \gamma^T)^T$ . We refer to Zhou, Qin and Longnecker (2011) for details of these notations.
q_s	smoothing parameter
Cpt	cut point a
mu_Y	mean of Y in the population
sig_Y	standard deviation of Y in the population
degree	degree of the truncated power spline basis, default value is 2
nknots	number of knots of the truncated power spline basis, default value is 10
tol	convergence criteria, the default value is 1e-6
iter	maximum iteration number, the default value is 30

## Details

We assume that in the population, the primary outcome variable  $Y$  follows the following partial linear model:

$$E(Y|X, Z) = g(X) + Z^T\gamma$$

where  $X$  is the expensive exposure,  $Z$  are other covariates. In ODS design, a simple random sample is taken from the full cohort, then two supplemental samples are taken from two tails of  $Y$ , i.e.  $(-\text{Infty}, \mu_Y - a \cdot \text{sig}_Y)$  and  $(\mu_Y + a \cdot \text{sig}_Y, +\text{Infty})$ . Because ODS data has biased sampling nature, naive regression analysis will yield biased estimates of the population parameters. Zhou, Qin and Longnecker (2011) describes a semiparametric empirical likelihood estimator for estimating the parameters in the partial linear model.

## Value

Parameter estimates and standard errors for the partial linear model:

$$E(Y|X, Z) = g(X) + Z^T\gamma$$

where the unknown smooth function  $g$  is approximated by a spline function with fixed knots. The results contain the following components:

alpha	spline coefficient
gam	other linear regression coefficients
std_gam	standard error of gam
cov_mtxa	covariance matrix of alpha
step	numbers of iteration required to acheive convergence
pi0	estimated notation pi
v0	estimated notation vtheta
sig0_sq0	estimated variance of error

## Examples

```
library(ODS)
# take the example data from the ODS package
# please see the documentation for details about the data set ods_data

nknots = 10
degree = 2

# get the initial value of the parameters from standard linear regression based on SRS data #
dataSRS = ods_data[1:200,]
YS = dataSRS[,1]
XS = dataSRS[,2]
ZS = dataSRS[,3:5]

knots = quantileknots(XS, nknots, 0)
# the power basis spline function
MS = Bfct(as.matrix(XS), degree, knots)
```

```

DS = cbind(MS, ZS)
theta00 = as.numeric(lm(YS ~ DS -1)$coefficients)
sig0_sq00 = var(YS - DS %*% theta00)
pi00 = c(0.15, 0.15)
v00 = c(0, 0)
eta00 = matrix(c(theta00, pi00, v00, sig0_sq00), ncol=1)
mu_Y = mean(YS)
sig_Y = sd(YS)

Y = matrix(ods_data[,1])
X = matrix(ods_data[,2])
Z = matrix(ods_data[,3:5], nrow=400)

# In this ODS data, the supplemental samples are taken from (-Infty, mu_Y-a*sig_Y) #
# and (mu_Y+a*sig_Y, +Infty), where a=1 #
n_f = c(200, 100, 100)
Cpt = 1

# GCV selection to find the optimal smoothing parameter #
q_s1 = logspace(-6, 7, 10)
gcv1 = rep(0, 10)

for (j in 1:10) {
  result = Estimate_PLMODS(X,Y,Z,n_f,eta00,q_s1[j],Cpt,mu_Y,sig_Y)
  etajj = matrix(c(result$alpha, result$gam, result$pi0, result$v0, result$sig0_sq0), ncol=1)
  gcv1[j] = gcv_ODS(X,Y,Z,n_f,etajj,q_s1[j],Cpt,mu_Y,sig_Y)
}

b = which(gcv1 == min(gcv1))
q_s = q_s1[b]
q_s

# Estimation of the partial linear model in the setting of outcome-dependent sampling #
result = Estimate_PLMODS(X, Y, Z, n_f, eta00, q_s, Cpt, mu_Y, sig_Y)
result

```

---

gcv\_ODS

*Generalized cross-validation for ODS data*


---

## Description

gcv\_ODS calculates the generalized cross-validation (GCV) for selecting the smoothing parameter in the setting of outcome-dependent sampling. The details can be seen in Zhou, Qin and Longnecker (2011) and its supplementary materials.

## Usage

```
gcv_ODS(X, Y, Z, n_f, eta, q_s, Cpt, mu_Y, sig_Y, degree, nknots)
```

**Arguments**

X	n by 1 matrix of the observed exposure variable
Y	n by 1 matrix of the observed outcome variable
Z	n by p matrix of the other covariates
n_f	$n_f = c(n_0, n_1, n_2)$ , where $n_0$ is the SRS sample size, $n_1$ is the size of the supplemental sample chosen from $(-\infty, \mu_Y - a \cdot \text{sig}_Y)$ , $n_2$ is the size of the supplemental sample chosen from $(\mu_Y + a \cdot \text{sig}_Y, +\infty)$ .
eta	a column matrix. $\eta = (\theta^T \pi^T v^T \text{sig}_0^2)^T$ where $\theta = (\alpha^T, \gamma^T)^T$ . We refer to Zhou, Qin and Longnecker (2011) for details of these notations.
q_s	smoothing parameter
Cpt	cut point a
mu_Y	mean of Y in the population
sig_Y	standard deviation of Y in the population
degree	degree of the truncated power spline basis, default value is 2
nknots	number of knots of the truncated power spline basis, default value is 10

**Value**

the value of generalized cross-validation score

**Examples**

```
library(ODS)
# take the example data from the ODS package
# please see the documentation for details about the data set ods_data

nknots = 10
degree = 2

# get the initial value of the parameters from standard linear regression based on SRS data #
dataSRS = ods_data[1:200,]
YS = dataSRS[,1]
XS = dataSRS[,2]
ZS = dataSRS[,3:5]

knots = quantileknots(XS, nknots, 0)
# the power basis spline function
MS = Bfct(as.matrix(XS), degree, knots)
DS = cbind(MS, ZS)
theta00 = as.numeric(lm(YS ~ DS -1)$coefficients)
sig0_sq00 = var(YS - DS %*% theta00)
pi00 = c(0.15, 0.15)
v00 = c(0, 0)
eta00 = matrix(c(theta00, pi00, v00, sig0_sq00), ncol=1)
mu_Y = mean(YS)
```

```

sig_Y = sd(YS)

Y = matrix(ods_data[,1])
X = matrix(ods_data[,2])
Z = matrix(ods_data[,3:5], nrow=400)

# In this ODS data, the supplemental samples are taken from (-Inf*sig_Y, mu_Y-a*sig_Y) #
# and (mu_Y+a*sig_Y, +Inf*sig_Y), where a=1 #
n_f = c(200, 100, 100)
Cpt = 1

# GCV selection to find the optimal smoothing parameter #
q_s1 = logspace(-6, 7, 10)
gcv1 = rep(0, 10)

for (j in 1:10) {

  result = Estimate_PLMODS(X,Y,Z,n_f,eta00,q_s1[j],Cpt,mu_Y,sig_Y)
  etajj = matrix(c(result$alpha, result$gam, result$pi0, result$v0, result$sig0_sq0), ncol=1)
  gcv1[j] = gcv_ODS(X,Y,Z,n_f,etajj,q_s1[j],Cpt,mu_Y,sig_Y)
}

b = which(gcv1 == min(gcv1))
q_s = q_s1[b]

q_s

# Estimation of the partial linear model in the setting of outcome-dependent sampling #
result = Estimate_PLMODS(X, Y, Z, n_f, eta00, q_s, Cpt, mu_Y, sig_Y)
result

```

---

logspace

---

*Generate logarithmically spaced vector*


---

### Description

logspace generates  $n$  logarithmically spaced points between  $10^{d1}$  and  $10^{d2}$ . The utility of this function is equivalent to logspace function in matlab.

### Usage

```
logspace(d1, d2, n)
```

### Arguments

d1	first bound
d2	second bound
n	number of points

**Value**

a vector of n logarithmically spaced points between  $10^{d1}$  and  $10^{d2}$ .

**Examples**

```
logspace(-6, 7, 30)
```

---

odsmle	<i>MSELE estimator for analyzing the primary outcome in ODS design</i>
--------	--

---

**Description**

odsmle provides a maximum semiparametric empirical likelihood estimator (MSELE) for analyzing the primary outcome Y with respect to expensive exposure and other covariates in ODS design (Zhou et al. 2002).

**Usage**

```
odsmle(Y, X, beta, sig, pis, a, rs.size, size, strat)
```

**Arguments**

Y	vector for the primary response
X	the design matrix with a column of 1's for the intercept
beta	starting parameter values for the regression coefficients that relate Y to X.
sig	starting parameter values for the error variance of the regression.
pis	starting parameter values for the stratum probabilities (the probability that Y belongs to certain stratum) e.g. pis = c(0.1, 0.8, 0.1).
a	vector of cutpoints for the primary response (e.g., a = c(-2.5,2))
rs.size	size of the SRS (simple random sample)
size	vector of the stratum sizes of the supplemental samples (e.g. size = c(50,0,50) represents that two supplemental samples each of size 50 are taken from the upper and lower tail of Y.)
strat	vector that indicates the stratum numbers (e.g. strat = c(1,2,3) represents that there are three stratum).

**Details**

We assume that in the population, the primary outcome variable Y follows the following model:

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

where X are the covariates, and epsilon has variance sig. In ODS design, a simple random sample is taken from the full cohort, then two supplemental samples are taken from two tails of Y, i.e.  $(-\infty, \mu_Y - a \cdot \text{sig}_Y)$  and  $(\mu_Y + a \cdot \text{sig}_Y, +\infty)$ . Because ODS data has biased sampling nature, naive regression analysis will yield biased estimates of the population parameters. Zhou et al. (2002) describes a semiparametric empirical likelihood estimator for estimating the parameters in the primary outcome model.

**Value**

A list which contains the parameter estimates for the primary response model:

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

where epsilon has variance sig. The list contains the following components:

beta	parameter estimates for beta
sig	estimates for sig
pis	estimates for the stratum probabilities
grad	gradient
hess	hessian
converge	whether the algorithm converges: True or False
i	Number of iterations

**Examples**

```
library(ODS)
# take the example data from the ODS package
# please see the documentation for details about the data set ods_data

Y <- ods_data[,1]
X <- cbind(rep(1,length(Y)), ods_data[,2:5])

# use the simple random sample to get an initial estimate of beta, sig #
# perform an ordinary least squares #
SRS <- ods_data[1:200,]
OLS.srs <- lm(SRS[,1] ~ SRS[,2:5])
OLS.srs.summary <- summary(OLS.srs)

beta <- coefficients(OLS.srs)
sig <- OLS.srs.summary$sigma^2
pis <- c(0.1,0.8,0.1)

# the cut points for this data is Y < 0.162, Y > 2.59.
a <- c(0.162,2.59)
rs.size <- 200
size <- c(100,0,100)
strat <- c(1,2,3)

odsmle(Y,X,beta,sig,pis,a,rs.size,size,strat)
```

---

ods\_data

*Data example for analyzing the primary response in ODS design*


---

**Description**

Data example for analyzing the primary response in ODS design (zhou et al. 2002)

**Usage**

ods\_data

**Format**

A matrix with 400 rows and 5 columns. The first 200 observations are from the simple random sample, while 2 supplemental samples each with size 100 are taken from one standard deviation above the mean and below the mean, i.e. ( $Y1 < 0.162$ ) and ( $Y1 > 2.59$ ).

**Y1** primary outcome for which the ODS sampling scheme is based on

**X** expensive exposure

**Z1** a simulated covariate

**Z2** a simulated covariate

**Z3** a simulated covariate

**Source**

A simulated data set

---

ods\_data\_secondary      *Data example for the secondary analysis in ODS design*

---

**Description**

Data example for the secondary analysis in ODS design

**Usage**

ods\_data\_secondary

**Format**

A matrix with 3000 rows and 7 columns:

**subj\_ind** An indicator variable for each subject: 1 = SRS, 2 = lowerODS, 3 = upperODS, 0 = NVsample

**Y1** primary outcome for which the ODS sampling scheme is based on

**Y2** a secondary outcome

**X** expensive exposure

**Z1** a simulated covariate

**Z2** a simulated covariate

**Z3** a simulated covariate

**Source**

A simulated data set

quantileknots                      *Create knots at sample quantiles*

---

### Description

quantileknots creates knots at sample quantiles

### Usage

```
quantileknots(x, nknots, boundstab)
```

### Arguments

x                      a vector. The knots are at sample quantiles of x.  
nknots                number of knots  
boundstab            parameter for boundary stability. The default is 0. If boundstab = 1, then nknots+2 knots are created and the first and last are deleted. This mitigates the extra variability of regression spline estimates near the boundaries.

### Value

a vector of knots at sample quantiles of x.

### Examples

```
library(ODS)  
  
x <- c(1, 2, 3, 4, 5)  
quantileknots(x, 3, 0)
```

---

se.spmle                      *standard error for MSELE estimator*

---

### Description

se.spmle calculates the standard error for MSELE estimator in Zhou et al. 2002

### Usage

```
se.spmle(y, x, beta, sig, pis, a, N.edf, rhos, strat, size.nc)
```

**Arguments**

y	vector of the primary response
x	the design matrix with a column of 1's for the intercept
beta	final estimates of the regression coefficients obtained from odsmle
sig	final estimate of the error variance obtained from odsmle
pis	final estimates of the stratum probabilities obtained from odsmle
a	vector of cutpoints for the primary response (e.g., a = c(-2.5,2))
N.edf	should be the size of the SRS (simple random sample)
rhos	which is size/pis, where size is a vector representing the stratum sizes of supplemental samples. e.g. size = c(100, 0, 100), and pis are the final estimates obtained from odsmle.
strat	vector that indicates the stratum numbers of supplemental samples, except that you should only list stratum with size > 0. (e.g. if the supplemental size is c(100, 0, 100), then the strat vector should be c(1,3))
size.nc	total size of the validation sample (SRS plus supplemental samples)

**Value**

A list which contains the standard error estimates for betas in the model :

$$Y = \beta_0 + \beta_1 X + \epsilon,$$

where epsilon has variance sig.

**Examples**

```
library(ODS)
# take the example data from the ODS package
# please see the documentation for details about the data set ods_data

Y <- ods_data[,1]
X <- cbind(rep(1,length(Y)), ods_data[,2:5])

# use the simple random sample to get an initial estimate of beta, sig #
# perform an ordinary least squares #
SRS <- ods_data[1:200,]
OLS.srs <- lm(SRS[,1] ~ SRS[,2:5])
OLS.srs.summary <- summary(OLS.srs)

beta <- coefficients(OLS.srs)
sig <- OLS.srs.summary$sigma^2
pis <- c(0.1,0.8,0.1)

# the cut points for this data is Y < 0.162, Y > 2.59.
a <- c(0.162,2.59)
rs.size <- 200
size <- c(100,0,100)
strat <- c(1,2,3)
```

```

# obtain the parameter estimates
ODS.model = odsmle(Y,X,beta,sig,pis,a,rs.size,size,strat)

# calculate the standard error estimate
y <- Y
x <- X
beta <- ODS.model$beta
sig <- ODS.model$sig
pis <- ODS.model$pis
a <- c(0.162,2.59)
N.edf <- rs.size
rhos <- size/pis
strat <- c(1,3)
size.nc <- length(y)

se = se.spmlle(y, x, beta, sig, pis, a, N.edf, rhos, strat, size.nc)

# summarize the result
ODS.tvalue <- ODS.model$beta / se
ODS.pvalue <- 2 * pt( - abs(ODS.tvalue), sum(rs.size, size)-2)

ODS.results <- cbind(ODS.model$beta, se, ODS.tvalue, ODS.pvalue)
dimnames(ODS.results)[[2]] <- c("Beta","SEbeta","tvalue","Pr(>|t|)")
row.names(ODS.results) <- c("(Intercept)","X","Z1","Z2","Z3")

ODS.results

```

---

secondary\_cascohort    *Secondary analysis in case-cohort data*

---

## Description

secondary\_cascohort performs the secondary analysis which describes the association between a continuous secondary outcome and the expensive exposure for case-cohort data.

## Usage

```
secondary_cascohort(SRS, CCH, NVsample, Z1.dim, Z2.dim, Z3.dim)
```

## Arguments

**SRS**                    A data frame for subjects in the simple random sample. The first column is T: observation time for time-to-event outcome. The second column is Delta: the event indicator. The third column is Y2: the continuous scale secondary outcome. The fourth column is X: the expensive exposure. Starting from the fifth column to the end are Z1, Z2 and Z3. Z1 is the set of covariates that are included in the linear regression model of the secondary outcome. Z2 is the set of covariates that are included in the Cox model (the proportional hazards model

which relates the primary failure time to covariates).  $Z_3$  is the set of covariates that are related to the conditional distribution of  $X$  given other covariates.

CCH	A data frame for subjects in the case-cohort sample. The case-cohort sample includes the simple random sample (SRS) and the supplemental cases. The data structure is the same as SRS.
NVsample	A data frame for subjects in the non-validation sample. Subjects in the non-validation sample don't have the expensive exposure $X$ measured. The data structure is the following: The first column is $T$ . The second column is $\Delta$ . The third column is $Y_2$ . Starting from the fourth column to the end are $Z_1$ , $Z_2$ and $Z_3$ .
Z1.dim	Dimension of $Z_1$ .
Z2.dim	Dimension of $Z_2$ .
Z3.dim	Dimension of $Z_3$ . Note here that the algorithm requires $Z_3$ to be discrete and not high-dimensional, because we use the SRS sample to estimate the conditional distribution of $X$ given other covariates.

### Value

A list which contains parameter estimates, estimated standard error for the primary outcome model:

$$\lambda(t) = \lambda_0(t) \exp \gamma_1 Y_2 + \gamma_2 X + \gamma_3 Z_2,$$

and the secondary outcome model:

$$Y_2 = \beta_0 + \beta_1 X + \beta_2 Z_1.$$

The list contains the following components:

gamma_paramEst	parameter estimates for gamma in the primary outcome model
gamma_stdErr	estimated standard error for gamma in the primary outcome model
beta_paramEst	parameter estimates for beta in the secondary outcome model
beta_stdErr	estimated standard error for beta in the secondary outcome model

### Examples

```
library(ODS)
# take the example data from the ODS package
# please see the documentation for details about the data set casecohort_data_secondary
data <- casecohort_data_secondary

# obtain SRS, CCH and NVsample from the original cohort data based on subj_ind
SRS <- data[data[,1]==1, 2:ncol(data)]
CCH <- data[data[,1]==1 | data[,1]==2, 2:ncol(data)]
NVsample <- data[data[,1]==0, 2:ncol(data)]

# delete the fourth column (columns for X) from the non-validation sample
NVsample <- NVsample[,-4]
```

```
Z1.dim <- 4
Z2.dim <- 3
Z3.dim <- 3
secondary_casecohort(SRS, CCH, NVsample, Z1.dim, Z2.dim, Z3.dim)
```

---

secondary\_ODS

*Secondary analysis in ODS design*


---

### Description

secondary\_ODS performs the secondary analysis which describes the association between a continuous scale secondary outcome and the expensive exposure for data obtained with ODS (outcome dependent sampling) design.

### Usage

```
secondary_ODS(SRS, lowerODS, upperODS, NVsample, cutpoint, Z.dim)
```

### Arguments

SRS	A data matrix for subjects in the simple random sample. The first column is Y1: the primary outcome for which the ODS scheme is based on. The second column is Y2: a secondary outcome. The third column is X: the expensive exposure. Starting from the fourth column to the end is Z: other covariates.
lowerODS	A data matrix for supplemental samples taken from the lower tail of Y1 (eg. $Y1 < a$ ). The data structure is the same as SRS.
upperODS	A data matrix for supplemental samples taken from the upper tail of Y1 (eg. $Y1 > b$ ). The data structure is the same as SRS.
NVsample	A data matrix for subjects in the non-validation sample. Subjects in the non-validation sample don't have the expensive exposure X measured. The data structure is the same as SRS, but the third column (which represents X) has values NA.
cutpoint	A vector of length two that represents the cut off points for the ODS design. eg. <code>cutpoint &lt;- c(a,b)</code> . In the ODS design, a simple random sample is taken from the full cohort, then two supplemental samples are taken from $\{Y1 < a\}$ and $\{Y1 > b\}$ , respectively.
Z.dim	Dimension of the covariates Z.

### Value

A list which contains parameter estimates, estimated standard error for the primary outcome model:

$$Y_1 = \beta_0 + \beta_1 X + \beta_2 Z,$$

and the secondary outcome model:

$$Y_2 = \gamma_0 + \gamma_1 X + \gamma_2 Z.$$

The list contains the following components:

beta\_paramEst parameter estimates for beta in the primary outcome model  
beta\_stdErr estimated standard error for beta in the primary outcome model  
gamma\_paramEst parameter estimates for gamma in the secondary outcome model  
gamma\_stdErr estimated standard error for gamma in the secondary outcome model

### Examples

```
library(ODS)
# take the example data from the ODS package
# please see the documentation for details about the data set ods_data_secondary
data <- ods_data_secondary

# divide the original cohort data into SRS, lowerODS, upperODS and NVsample
SRS <- data[data[,1]==1,2:ncol(data)]
lowerODS <- data[data[,1]==2,2:ncol(data)]
upperODS <- data[data[,1]==3,2:ncol(data)]
NVsample <- data[data[,1]==0,2:ncol(data)]

# obtain the cut off points for ODS design. For this data, the ODS design
# uses mean plus and minus one standard deviation of Y1 as cut off points.
meanY1 <- mean(data[,2])
sdY1 <- sd(data[,2])
cutpoint <- c(meanY1-sdY1, meanY1+sdY1)

# the data matrix SRS has Y1, Y2, X and Z. Hence the dimension of Z is ncol(SRS)-3.
Z.dim <- ncol(SRS)-3

secondary_ODS(SRS, lowerODS, upperODS, NVsample, cutpoint, Z.dim)
```

# Index

## \*Topic **datasets**

- casecohort\_data\_secondary, [3](#)
- ods\_data, [10](#)
- ods\_data\_secondary, [11](#)
  
- Bfct, [2](#)
  
- casecohort\_data\_secondary, [3](#)
  
- Estimate\_PLMODS, [4](#)
  
- gcv\_ODS, [6](#)
  
- logspace, [8](#)
  
- ods\_data, [10](#)
- ods\_data\_secondary, [11](#)
- odsmle, [9](#)
  
- quantileknots, [12](#)
  
- se.spml, [12](#)
- secondary\_casecohort, [14](#)
- secondary\_ODS, [16](#)