Package 'ICODS'

June 5, 2019

Type Package

Title Data Analysis for ODS and Case-Cohort Designs with Interval-Censoring

Version 1.0

Date 2019-6-05

Author Shannon T. Holloway, Qingning Zhou, Jianwen Cai, Haibo Zhou

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

Description Sieve semiparametric likelihood methods for analyzing interval-censored failure time data from an outcome-dependent sampling (ODS) design and from a case-cohort design.
Zhou, Q., Cai, J., and Zhou, H. (2018) <doi:10.1111/biom.12744>;
Zhou, Q., Zhou, H., and Cai, J. (2017) <doi:10.1093/biomet/asw067>.

License GPL-2

Depends methods, stats, MASS

NeedsCompilation no

Repository CRAN

Encoding UTF-8

RoxygenNote 6.1.1

Collate 'myOptim.R' 'minObj.R' 'baseInfo.R' 'CaseCohort_Obj.R' 'CaseCohort_gr.R' 'CaseCohort_fn.R' 'class_ICODS.R' 'CaseCohort_class.R' 'bernstein.R' 'CaseCohortIC.R' 'CaseCohort_data.R' 'ODSDesign_Obj.R' 'ODSDesign_class.R' 'ODSDesignIC.R' 'ODSDesign_data.R' 'ODSDesign_fn.R' 'ODSDesign_gr.R' 'testInputData.R'

Date/Publication 2019-06-05 13:00:06 UTC

R topics documented:

`CaseCohortIC` *Case-Cohort Studies with Interval-Censored Failure Time Data*

Description

Provides a sieve semiparametric likelihood approach under the proportional hazards model for analyzing data from a case-cohort design with failure times subject to interval-censoring. The likelihood function is constructed using inverse probability weighting, and the sieves are built with Bernstein polynomials. A weighted bootstrap procedure is implemented for variance estimation.

Usage

```
CaseCohortIC(U, V, del1, del2, xi, z, sp, mVal, B, beta = NULL,
  maxit = 10L, verbose = TRUE, ...)
```

Arguments

`U`	numeric vector (n); examination time. See Details for further information.
`V`	numeric vector (n); examination time. See Details for further information.
`del1`	integer vector (n); indicator of a left-censored observation I(T<=U). See Details for further information.
`del2`	integer vector (n); indicator of an interval-censored observation I(U<T<=V). See Details for further information.
`xi`	integer vector (n); indicating membership of the case-cohort sample.
`z`	matrix (nxp); covariates.
`sp`	numeric (1); sampling probability 0 < sp < 1.
`mVal`	integer vector (m); one or more options for the degree of the Bernstein polynomials. If more than one option provided, the value resulting in the lowest AIC is selected. The results returned are for only that m-value.
`B`	integer (1); number of bootstrap samples used to calculate the variance estimate.
`beta`	numeric vector (p); initial values for beta. If NULL, initial guess set to 0.5 for each of the p parameters.
`maxit`	integer(1); maximum number of calls to optimization method.
`verbose`	logical; TRUE generates progress screen prints.
`...`	optional inputs to "control" of function optim().

Details

The implementation uses stats::optim() to minimize the likelihood. The hard-coded method is "BFGS". Users are able to make changes to the 'control' input of optim() by passing named inputs through the ellipses. If a call to optim() returns convergence = 1, i.e., optim() reached its internal maximum number of iterations before convergence was attained, the software automatically repeats the call to optim() with input variable par set to the last parameter values. This procedure is repeated at most maxit times.

Input parameters U, V, del1, and del2 are defined as follows. Suppose there are K follow-up examinations at times TE = (T1, T2, ..., TK), and the failure time is denoted as TF. For left-censored data, the failure occurs prior to the first follow-up examination (TF < T1); therefore, define U = T1, V = tau, and (del1,del2)=(1,0). For right-censored data, the failure has not yet occurred at the last follow-up examination (TF > TK); therefore, define U = 0, V = TK, and (del1,del2)=(0,0). For interval-censored data, the failure occurs between two follow-up examinations, e.g. T2 < TF < T3; therefore, define U and V to be the two consecutive follow-up examination times bracketing the failure time TF and (del1,del2)=(0,1).

Value

an object of class CaseCohort (inheriting from ICODS) containing

`optim`	a list of the results returned by optim().
`beta`	the estimated beta parameters.
`se`	the standard error of the estimated beta parameters.
`pValue`	the p-value of the estimated beta parameters.
`m`	the selected degree of the Bernstein polynomials.
`AIC`	the AIC value for the selected degree of the Bernstein polynomials.

References

Zhou, Q., Zhou, H., and Cai, J. (2017). Case-cohort studies with interval-censored failure time data. Biometrika, 104(1): 17–29. <doi:10.1093/biomet/asw067>

Examples

```
data(ccData)

result <- CaseCohortIC(U = ccData$U,
                       V = ccData$V,
                       del1 = ccData$del1,
                       del2 = ccData$del2,
                       xi = ccData$xi,
                       z = ccData$z,
                       sp = 0.2,
                       mVal = 1L,
                       B = 10L,
                       beta = NULL,
                       maxit = 10L,
```

```
                        verbose = TRUE)

print(result)
mVal(result)
estimate(result)
optimObj(result)
minAIC(result)
summary(result)
```

---

ccData *Toy Example for Case-Cohort Design with Interval-Censored Data*

Description

This data set gives a simple toy example of case-cohort design with interval-censored data. It was generated following the simulation study used to illustrate the method in the original manuscript referenced below. This dataset is not meaningful and is intended for demonstration purposes only.

Usage

```
data(ccData)
```

Format

A data.frame containing 500 observations with 6 columns:

U examination time.

V examination time.

del1 indicator of a left-censored observation I(T<=U).

del2 indicator of an interval-censored observation I(U<T<=V).

xi indicating membership of the case-cohort sample.

z covariates.

See Details for further information.

Details

The data can be understood as follow. There are K follow-up examinations at times TE = (T1, T2, ..., TK), and the failure time is denoted as TF. For left-censored data, the failure occurred prior to the first follow-up examination (TF < T1); therefore, U = T1, V = tau, and (del1,del2)=(1,0). For right-censored data, the failure had not yet occurred at the last follow-up examination (TF > TK); therefore, U = 0, V = TK, and (del1,del2)=(0,0). For interval-censored data, the failure occurred between two follow-up examinations, e.g. T2 < TF < T3; therefore, U and V to be the two consecutive follow-up examination times bracketing the failure time TF and (del1,del2)=(0,1).

References

Zhou, Q., Zhou, H., and Cai, J. (2017). Case-cohort studies with interval-censored failure time data. Biometrika, 104(1): 17–29. <doi:10.1093/biomet/asw067>

`estimate` *Retrieve the Estimated Beta Parameters*

Description

Retrieves the estimated beta parameters for the m value that minimizes the AIC.

Usage

```
estimate(object, ...)
```

Arguments

`object`	An object of class ICODS
`...`	ignored

Value

A matrix containing the estimated parameter value, the standard error, and the p-value.

Examples

```
data(odsData)

resultODS <- ODSDesignIC(U = odsData$U,
                         V = odsData$V,
                         del1 = odsData$del1,
                         del2 = odsData$del2,
                         z = odsData$z,
                         mVal = 1L,
                         ind = odsData$ind,
                         a1 = 0.43,
                         a2 = 0.45,
                         beta = NULL,
                         maxit = 10L,
                         verbose = TRUE)

estimate(resultODS)

data(ccData)

resultCC <- CaseCohortIC(U = ccData$U,
                         V = ccData$V,
                         del1 = ccData$del1,
```

```
                          del2 = ccData$del2,
                          xi = ccData$xi,
                          z = ccData$z,
                          sp = 0.2,
                          mVal = 1L,
                          B = 10L,
                          beta = NULL,
                          maxit = 10L,
                          verbose = TRUE)

estimate(resultCC)
```

`minAIC` *Retrieve the Minimum AIC*

Description

Retrieves the minimum AIC.

Usage

```
minAIC(object, ...)
```

Arguments

`object`	An object of class ICODS
`...`	ignored

Value

numeric

Examples

```
data(odsData)

resultODS <- ODSDesignIC(U = odsData$U,
                          V = odsData$V,
                          del1 = odsData$del1,
                          del2 = odsData$del2,
                          z = odsData$z,
                          mVal = 1L,
                          ind = odsData$ind,
                          a1 = 0.43,
                          a2 = 0.45,
                          beta = NULL,
                          maxit = 10L,
                          verbose = TRUE)
```

```
minAIC(resultODS)

data(ccData)

resultCC <- CaseCohortIC(U = ccData$U,
                         V = ccData$V,
                         del1 = ccData$del1,
                         del2 = ccData$del2,
                         xi = ccData$xi,
                         z = ccData$z,
                         sp = 0.2,
                         mVal = 1L,
                         B = 10L,
                         beta = NULL,
                         maxit = 10L,
                         verbose = TRUE)

minAIC(resultCC)
```

`mVal` *Retrieve Degree of Optimal Bernstein Polynomial*

Description

Retrieves the degree of the Bernstein polynomial basis provided as input that minimizes the AIC.

Usage

```
mVal(object, ...)
```

Arguments

`object`	An object of class ICODS
`...`	ignored

Value

an integer

Examples

```
data(odsData)

resultODS <- ODSDesignIC(U = odsData$U,
                         V = odsData$V,
                         del1 = odsData$del1,
```

```
                          del2 = odsData$del2,
                          z = odsData$z,
                          mVal = 1L,
                          ind = odsData$ind,
                          a1 = 0.43,
                          a2 = 0.45,
                          beta = NULL,
                          maxit = 10L,
                          verbose = TRUE)

mVal(resultODS)

data(ccData)

resultCC <- CaseCohortIC(U = ccData$U,
                          V = ccData$V,
                          del1 = ccData$del1,
                          del2 = ccData$del2,
                          xi = ccData$xi,
                          z = ccData$z,
                          sp = 0.2,
                          mVal = 1L,
                          B = 10L,
                          beta = NULL,
                          maxit = 10L,
                          verbose = TRUE)

mVal(resultCC)
```

`odsData` *Toy Example for ODS Design with Interval-Censored Data*

Description

This data set gives a simple toy example of ODS design with interval-censored data. It was generated following the simulation study used to illustrate the method in the original manuscript referenced below. This dataset is not meaningful and is intended for demonstration purposes only.

Usage

```
data(odsData)
```

Format

A data.frame containing 501 observations with 6 columns:

U examination time; see Details.

V examination time; see Details.

del1 indicator of a left-censored observation I(T<=U).

del2 indicator of an interval-censored observation I(U<T<=V).

z covariates.

ind indicating membership of the simple random sample (0), lower-tail supplemental sample (1), or upper-tail supplemental sample (2).

Details

The data can be understood as follow. There are K follow-up examinations at times TE = (T1, T2, ..., TK), and the failure time is denoted as TF. For left-censored data, the failure occurred prior to the first follow-up examination (TF < T1); therefore, U = T1, V = tau, and (del1,del2)=(1,0). For right-censored data, the failure had not yet occurred at the last follow-up examination (TF > TK); therefore, U = 0, V = TK, and (del1,del2)=(0,0). For interval-censored data, the failure occurred between two follow-up examinations, e.g. T2 < TF < T3; therefore, U and V to be the two consecutive follow-up examination times bracketing the failure time TF and (del1,del2)=(0,1).

References

Zhou, Q., Cai, J., and Zhou, H. (2018). Outcome-dependent sampling with interval-censored failure time data. Biometrics, 74(1): 58–67. <doi:10.1111/biom.12744>

`ODSDesignIC` *Outcome-Dependent Sampling with Interval-Censored Failure Time Data*

Description

Provides an outcome-dependent sampling (ODS) design with interval-censored failure time data, where the observed sample is enriched by selectively including certain more informative failure subjects. The method is a sieve semiparametric maximum empirical likelihood approach for fitting the proportional hazards model to data from the interval- censoring ODS design.

Usage

```
ODSDesignIC(U, V, del1, del2, z, mVal, ind, a1, a2, beta = NULL,
  maxit = 10L, verbose = TRUE, ...)
```

Arguments

`U`	numeric vector (n); examination time. See Details for further information.
`V`	numeric vector (n); examination time. See Details for further information.
`del1`	integer vector (n); indicator of a left-censored observation I(T<=U). See Details for further information.
`del2`	integer vector (n); indicator of an interval-censored observation I(U<T<=V). See Details for further information.
`z`	matrix (nxp); covariates.

mVal	integer vector (m); one or more options for the degree of the Bernstein polynomials. If more than one option provided, the value resulting in the lowest AIC is selected. The results returned are for only that m-value.
ind	integer vector (n); indicating membership of the simple random sample (0), lower-tail supplemental sample (1), or upper-tail supplemental sample (2).
a1	numeric (1); lower cut-off point for selecting ODS sample (0 < a1 < a2 < tau).
a2	numeric (1); upper cut-off point for selecting ODS sample (0 < a1 < a2 < tau).
beta	numeric vector (p); initial values for beta. If NULL, initial guess set to 0.5 for each of the p parameters.
maxit	integer(1); maximum number of calls to optimization method.
verbose	logical; TRUE generates progress screen prints.
...	optional inputs to "control" of function optim().

Details

The implementation uses stats::optim() to minimize the likelihood. The hard-coded method is "BFGS". Users are able to make changes to the 'control' input of optim() by passing named inputs through the ellipses. If a call to optim() returns convergence = 1, i.e., optim() reached its internal maximum number of iterations before convergence was attained, the software automatically repeats the call to optim() with input variable par set to the last parameter values. This procedure is repeated at most maxit times.

Input parameters U, V, del1, and del2 are defined as follows. Suppose there are K follow-up examinations at times TE = (T1, T2, ..., TK), and the failure time is denoted as TF. For left-censored data, the failure occurred prior to the first follow-up examination (TF < T1); therefore, define U = T1, V = tau, and (del1,del2)=(1,0). For right-censored data, the failure had not yet occurred at the last follow-up examination (TF > TK); therefore, define U = 0, V = TK, and (del1,del2)=(0,0). For interval-censored data, the failure occurred between two follow-up examinations, e.g. T2 < TF < T3; therefore, define U and V to be the two consecutive follow-up examination times bracketing the failure time TF and (del1,del2)=(0,1).

Value

an object of class ODSDesign (inheriting from ICODS) containing

optim	a list of the results returned by optim().
beta	the estimated beta parameters.
se	the standard error of the estimated beta parameters.
pValue	the p-value of the estimated beta parameters.
m	the selected degree of the Bernstein polynomials.
AIC	the AIC value for the selected degree of the Bernstein polynomials.

References

Zhou, Q., Cai, J., and Zhou, H. (2018). Outcome-dependent sampling with interval-censored failure time data. Biometrics, 74(1): 58–67. <doi:10.1111/biom.12744>

Examples

```
data(odsData)

result <- ODSDesignIC(U = odsData$U,
                      V = odsData$V,
                      del1 = odsData$del1,
                      del2 = odsData$del2,
                      z = odsData$z,
                      mVal = 1L,
                      ind = odsData$ind,
                      a1 = 0.43,
                      a2 = 0.45,
                      beta = NULL,
                      maxit = 10L,
                      verbose = TRUE)

print(result)
mVal(result)
estimate(result)
optimObj(result)
minAIC(result)
summary(result)
```

---

optimObj *Retrieve the Optimization Results*

Description

Retrieves the final optimization results for the m value that minimizes the AIC.

Usage

```
optimObj(object, ...)
```

Arguments

object	An object of class ICODS
...	ignored

Value

the value object returned by stats::optim()

Examples

```
data(odsData)

resultODS <- ODSDesignIC(U = odsData$U,
                         V = odsData$V,
                         del1 = odsData$del1,
                         del2 = odsData$del2,
                         z = odsData$z,
                         mVal = 1L,
                         ind = odsData$ind,
                         a1 = 0.43,
                         a2 = 0.45,
                         beta = NULL,
                         maxit = 10L,
                         verbose = TRUE)

optimObj(resultODS)

data(ccData)

resultCC <- CaseCohortIC(U = ccData$U,
                         V = ccData$V,
                         del1 = ccData$del1,
                         del2 = ccData$del2,
                         xi = ccData$xi,
                         z = ccData$z,
                         sp = 0.2,
                         mVal = 1L,
                         B = 10L,
                         beta = NULL,
                         maxit = 10L,
                         verbose = TRUE)

optimObj(resultCC)
```

`summary` *Retrieve the Key Results*

Description

Retrieves the estimated beta parameters for the m value that minimizes the AIC; the m value; and the AIC value.

Arguments

`object`	An object of class ICODS
`...`	ignored

Value

A list containing

par	A matrix containing the estimated parameter value, the standard error, and the p-value.
m	The selected m value.
AIC	The AIC.

Examples

```
data(odsData)

resultODS <- ODSDesignIC(U = odsData$U,
                         V = odsData$V,
                         del1 = odsData$del1,
                         del2 = odsData$del2,
                         z = odsData$z,
                         mVal = 1L,
                         ind = odsData$ind,
                         a1 = 0.43,
                         a2 = 0.45,
                         beta = NULL,
                         maxit = 10L,
                         verbose = TRUE)

summary(resultODS)

data(ccData)

resultCC <- CaseCohortIC(U = ccData$U,
                         V = ccData$V,
                         del1 = ccData$del1,
                         del2 = ccData$del2,
                         xi = ccData$xi,
                         z = ccData$z,
                         sp = 0.2,
                         mVal = 1L,
                         B = 10L,
                         beta = NULL,
                         maxit = 10L,
                         verbose = TRUE)

summary(resultCC)
```

Index