Package 'FHDI'

June 11, 2019

Version 1.3.2

Date 2019-06-10

Title Fractional Hot Deck and Fully Efficient Fractional Imputation

Author Jongho Im [aut],
Inho Cho [aut, cre],
Jaekwang Kim [aut]

Maintainer Inho Cho <icho@iastate.edu>

Depends R (>= 3.4.0)

Description Impute general multivariate missing data with the fractional hot deck imputation based on Jaekwang Kim (2011) <doi:10.1093/biomet/asq073>.

License GPL (>= 2)

URL https://www.r-project.org,
https://sites.google.com/view/jaekwangkim/software

BugReports https://sites.google.com/site/ichoddcse2017/home/type-of-trainings/r-package-fhdi

NeedsCompilation yes

Repository CRAN

Date/Publication 2019-06-11 16:30:03 UTC

R topics documented:

FHDI-package *Fractional Hot Deck Imputation*

Description

Perform fractional hot deck imputation Perform fully efficient fractional imputation

Details

FHDI_Driver(daty, datr=NULL, datz=NULL, s_op_imputation="FEFI", i_op_variance=0, M = 5, k = 5, w = NULL, id = NULL, s_op_merge="fixed", categorical=NULL);

Author(s)

Author: Jongho Im [aut], Inho Cho [aut, cre], Jaekwang Kim [aut] <icho@iastate.edu>

References

Im, J., Cho, I.H. and Kim, J.K. (2018). FHDI: An **R** Package for Fractional Hot-Deck Imputation. *The R Journal*. 10(1), pp. 140-154; Im, J., Kim, J.K. and Fuller, W.A. (2015). Two-phase sampling approach to fractional hot deck imputation, *Proceeding of the Survey Research Methods Section*, Americal Statistical Association, Seattle, WA.

See Also

FHDI_CellMake and FHDI_CellProb

Examples

```
### Toy Example ###
# y : trivariate variables
# r : indicator corresponding to missingness in y

set.seed(1345)
n=100
rho=0.5
e1=rnorm(n,0,1)
e2=rnorm(n,0,1)
e3=rgamma(n,1,1)
e4=rnorm(n,0,sd=sqrt(3/2))

y1=1+e1
y2=2+rho*e1+sqrt(1-rho^2)*e2
y3=y1+e3
y4=-1+0.5*y3+e4

r1=rbinom(n,1,prob=0.6)
r2=rbinom(n,1,prob=0.7)
r3=rbinom(n,1,prob=0.8)
```

```
r4=rbinom(n,1,prob=0.9)

y1[r1==0]=NA
y2[r2==0]=NA
y3[r3==0]=NA
y4[r4==0]=NA

daty=cbind(y1,y2,y3,y4)

result_FEFI=FHDI_Driver(daty, s_op_imputation="FEFI", k=3)
result_FHDI=FHDI_Driver(daty, s_op_imputation="FHDI", M=5, k=3)
names(result_FEFI)
names(result_FHDI)
```

`FHDI_CellMake` *Imputation cell creation*

Description

Perform a categorization procedure on the continuous raw data and then create imputation cells through a built-in merge algorithm.

Usage

```
FHDI_CellMake(daty, datr=NULL, k=5, w=NULL, id=NULL, s_op_merge="fixed", categorical=NULL)
```

Arguments

`daty`	raw data matrix (nrow_y, ncol_y) containing missing values. Each row must have at least one observed value, and no completely missing (blank) rows are allowed.
`datr`	response indicator matrix with the same dimensions as daty. Each response is recorded with 0 for missing value and 1 for observed value. If NULL, automatically filled with 1 or 0 according to daty.
`k`	the number of total categories per variable. Default = 5. The maximum is 35 since 9 integers (1-9) and 26 alphabet letters (a-z) are used. When a scalar value is given, all variables will have the same number of categories, while when a vector is given, i.e. k(ncol_y), each variable may have different number of categories.
`w`	samping weight for each row of daty. Default = 1.0 if NULL. When a scalar value is given, all rows will have the same weight, while when a vector is given, i.e. w(nrow_y), each row may have a different sampling weight.
`id`	index for each row. Default = 1:nrow_y if NULL.
`s_op_merge`	option for random cell make. Default = "fixed" using the same seed number; "rand" using a purely random seed number.

`categorical` (FHDI Version >1.3) index vector indicating non-collapsible categorical variables. Default = zero vector of size ncol_y. For instance, when categorical=c(1,0), the first variable (i.e., 1st column) is considered strictly non-collapsible categorical, and thus no automatic cell-collapse will take place while the second variable (i.e., 2nd column) is considered as continuous or collapsible categorical variable.

Details

This function creates imputation cells with the given number of category k. If the input value k is given a scalar, the same number of category is applied into all variables for initial discretization. Imputation cells are created to assign at least two donors on each missing unit. The donors have the same cell values with the observed parts of the missing unit.

Value

`data`	matrix of raw data (nrow_y, ncol_y) attached with id and weights, w.
`cell`	categorized matrix of y. A real value is categorized into 1~k categories with 0 meaning missing value.
`cell.resp`	unique patterns of respondents (donots) that are fully observed.
`cell.non.resp`	unique patterns of nonrespondents that have at least one missing item.
`w`	reprint of the sampling weights "w" initially defined by the user.
`s_op_merge`	reprint of the option "s_op_merge" initially defined by the user.

Author(s)

Dr. Im, Jong Ho <jonghoim@iastate.edu> Dr. Cho, In Ho <icho@iastate.edu> Dr. Kim, Jae Kwang <jkim@iastate.edu>

References

Im, J., Cho, I.H. and Kim, J.K. (2018). FHDI: An **R** Package for Fractional Hot-Deck Imputation. *The R Journal*. 10(1), pp. 140-154; Im, J., Kim, J.K. and Fuller, W.A. (2015). Two-phase sampling approach to fractional hot deck imputation, *Proceeding of the Survey Research Methods Section*, Americal Statistical Association, Seattle, WA.

Examples

```
### Toy Example ###
# y : trivariate variables
# r : indicator corresponding to missingness in y

set.seed(1345)
n=100
rho=0.5
e1=rnorm(n,0,1)
e2=rnorm(n,0,1)
e3=rgamma(n,1,1)
e4=rnorm(n,0,sd=sqrt(3/2))
```

```
y1=1+e1
y2=2+rho*e1+sqrt(1-rho^2)*e2
y3=y1+e3
y4=-1+0.5*y3+e4

r1=rbinom(n,1,prob=0.6)
r2=rbinom(n,1,prob=0.7)
r3=rbinom(n,1,prob=0.8)
r4=rbinom(n,1,prob=0.9)

y1[r1==0]=NA
y2[r2==0]=NA
y3[r3==0]=NA
y4[r4==0]=NA

daty=cbind(y1,y2,y3,y4)

result_CM=FHDI_CellMake(daty, s_op_merge="fixed",k=3)
names(result_CM)
```

`FHDI_CellProb` *Joint cell probabilities for multivariate incomplete categorical data*

Description

Calculate the joint cell probabilities for multivariate missing data using the expectation maximization algorithm.

Usage

```
FHDI_CellProb(datz, w=NULL, id=NULL)
```

Arguments

`datz`	multivariate incomplete categorical data.
`w`	samping weight. Default = 1.0 if NULL. a scalar or w(nrow_y).
`id`	index for each unit. Default = 1:nrow_y if NULL.

Details

The joint cell probabilities are estimated using EM by weighting method. The algorithm computes the maximum likelihood estimates of the joint cell probabilities under missing at random assumption.

Value

`cellpr`	table of the joint cell probability. name of cell is linked to the user-defined categories in "k": e.g., name "325" denotes 3rd, 2nd, 5th categories for three variables, respectively, whereas "a1c" denotes 10th, 1st, 12th categories.
`w`	reprint of the sampling weights "w" initially defined by the user.

Author(s)

Dr. Im, Jongho <jonghoim@iastate.edu> Dr. Cho, Inho <icho@iastate.edu> Dr. Kim, Jaekwang <jkim@iastate.edu>

References

Im, J., Cho, I.H. and Kim, J.K. (2018). FHDI: An **R** Package for Fractional Hot-Deck Imputation. *The R Journal*. 10(1), pp. 140-154; Im, J., Kim, J.K. and Fuller, W.A. (2015). Two-phase sampling approach to fractional hot deck imputation, *Proceeding of the Survey Research Methods Section*, Americal Statistical Association, Seattle, WA.; Ibrahim, J.G. (1990). Incomplete data in generalized linear models. *Journal of the American Statistical Assocation* **85**, 765-769.

Examples

```
### Toy Example ###
# y : trivariate variables
# r : indicator corresponding to missingness in y

set.seed(1345)
n=100
rho=0.5
e1=rnorm(n,0,1)
e2=rnorm(n,0,1)
e3=rgamma(n,1,1)
e4=rnorm(n,0,sd=sqrt(3/2))

y1=1+e1
y2=2+rho*e1+sqrt(1-rho^2)*e2
y3=y1+e3
y4=-1+0.5*y3+e4

r1=rbinom(n,1,prob=0.6)
r2=rbinom(n,1,prob=0.7)
r3=rbinom(n,1,prob=0.8)
r4=rbinom(n,1,prob=0.9)

y1[r1==0]=NA
y2[r2==0]=NA
y3[r3==0]=NA
y4[r4==0]=NA

daty=cbind(y1,y2,y3,y4)

result_CM=FHDI_CellMake(daty, k=5, s_op_merge="fixed")
datz=result_CM$cell
result_CP=FHDI_CellProb(datz)
names(result_CP)
```

FHDI_Driver *Fractional Hot Deck Imputation*

Description

Fully efficient fractional imputation (FEFI) or fractional hot deck imputation (FHDI) is implemented to fill in missing values in a incomplete data.

Usage

```
FHDI_Driver(daty, datr=NULL, datz=NULL, s_op_imputation="FEFI",
 i_op_variance=1, M=5, k=5, w=NULL, id=NULL,
 s_op_merge="fixed", categorical=NULL)
```

Arguments

daty	raw data matrix (nrow_y, ncol_y) containing missing values. Each row must have at least one observed value, and no completely missing (blank) rows are allowed.
datr	response indicator matrix with the same dimensions as daty. Each response is recorded with 0 for missing value and 1 for observed value. If NULL, automatically filled with 1 or 0 according to daty.
datz	imputation cell matrix. If daty is a set of continuous data, datz can be obtained using FHDI_CellMake.
s_op_imputation	"FEFI" for fully efficient fractional imputation or "FHDI" for fractional hot deck imputation.
i_op_variance	1: perform Jackknife variance estimation; 0: no variance estimation.
M	the number of donors for FHDI with default 5.
k	the number of total categories per variable. Default = 5. The maximum is 35 since 9 integers (1-9) and 26 alphabet letters (a-z) are used. When a scalar value is given, all variables will have the same number of categories, while when a vector is given, i.e. k(ncol_y), each variable may have different number of categories.
w	samping weight for each row of daty. Default = 1.0 if NULL. When a scalar value is given, all rows will have the same weight, while when a vector is given, i.e. w(nrow_y), each row may have a different sampling weight.
id	index for each row. Default = 1:nrow_y if NULL.
s_op_merge	option for random cell make. Default = "fixed" using the same seed number; "rand" using a purely random seed number.
categorical	(FHDI Version >1.3) index vector indicating non-collapsible categorical variables. Default = zero vector of size ncol_y. For instance, when categorical=c(1,0), the first variable (i.e., 1st column) is considered strictly non-collapsible categorical, and thus no automatic cell-collapse will take place while the second variable (i.e., 2nd column) is considered as continuous or collapsible categorical variable.

Details

In the FEFI method, all possible donors are assigned to each missing unit with the FEFI fractional weights. In the FHDI method, M (>1) donors are selected with the probability proportional to the FEFI fractional weights. Thus, the imputed values have equal fractional weights in general.

The jackknife replicated weights are produced as the default output. The replicated weights are presented by the product of replicated sampling weights and replicated fractional weights. Thus, the replicated weights can be directly used to compute the variance estimate of the estimators.

Value

`fimp.data`	imputation results with fractional weights in a form of matrix consisting of ID, donor id (FID), weight (WGT), fractional weight (FWGT), and fractionally imputed data.
`simp.data`	imputed data in the format of single imputation. The same shape as daty.
`imp.mean`	the mean estimates of each variable (first row) and the estimated standard error of each variable (second row). If input argument "i_op_variance=0" then this output is not produced.
`rep.weight`	replication fractional weights for variance estimation. If input argument "i_op_variance=0" then this output is not produced.
`M`	reprint of the number of donors M for FHDI defined by the user.
`s_op_imputation`	reprint of the option "s_op_imputation" initially defined by the user.
`i_op_merge`	reprint of the option "i_op_merge" initially defined by the user.

Author(s)

Dr. Im, Jongho <jonghoim@iastate.edu> Dr. Cho, Inho <icho@iastate.edu> Dr. Kim, Jaekwang <jkim@iastate.edu>

References

Im, J., Cho, I.H. and Kim, J.K. (2018). FHDI: An **R** Package for Fractional Hot-Deck Imputation. *The R Journal*. 10(1), pp. 140-154; Im, J., Kim, J.K. and Fuller, W.A. (2015). Two-phase sampling approach to fractional hot deck imputation, *Proceeding of the Survey Research Methods Section*, Americal Statistical Association, Seattle, WA.

Examples

```
### Toy Example ###
# y : trivariate variables
# r : indicator corresponding to missingness in y

set.seed(1345)
n=100
rho=0.5
e1=rnorm(n,0,1)
e2=rnorm(n,0,1)
e3=rgamma(n,1,1)
```

```
e4=rnorm(n,0,sd=sqrt(3/2))

y1=1+e1
y2=2+rho*e1+sqrt(1-rho^2)*e2
y3=y1+e3
y4=-1+0.5*y3+e4

r1=rbinom(n,1,prob=0.6)
r2=rbinom(n,1,prob=0.7)
r3=rbinom(n,1,prob=0.8)
r4=rbinom(n,1,prob=0.9)

y1[r1==0]=NA
y2[r2==0]=NA
y3[r3==0]=NA
y4[r4==0]=NA

daty=cbind(y1,y2,y3,y4)
result_FEFI=FHDI_Driver(daty, s_op_imputation="FEFI", k=3)
result_FHDI=FHDI_Driver(daty, s_op_imputation="FHDI", M=5, k=3)
names(result_FEFI)
names(result_FHDI)
```

Index