Package ‘MAPLES’

May 20, 2011

Type Package

Title Smoothed age profile estimation

Version 1.0

Date 2011-05-20

Author Roberto Impicciatore

Maintainer Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

Description MAPLES is a general method for the estimation of age profiles that uses standard micro-level demographic survey data. The aim is to estimate smoothed age profiles and relative risks for time-fixed and time-varying covariates.

Depends mgcv

License GPL (version 2 or later)

URL `http://www.demographic-research.org/Volumes/Vol24/29/`

LazyData yes

R topics documented:

`MAPLES-package` *Smoothed age profile estimation.*

Description

MAPLES is a general method for the estimation of age profiles that uses standard micro-level demographic survey data. The aim is to estimate smoothed age profiles and relative risks for time-fixed and time-varying covariates.

Details

Package:	MAPLES
Type:	Package
Version:	1.0
Date:	2011-05-20
License:	GPL-2
LazyLoad:	yes
LazyData:	yes

Main functions in the package:
- epdata: prepares episode data for event history analysis;
- splitter: creates a time-varying factor variable;
- ageprofile: computes smoothed transition rates by respondent's age (age profiles);
- plotap: plots age profiles.

Utilities:
- tabx: prints uni- and bivariate frequency table with marginal distributions;
- tabm: print regression estimates for previously fitted linear and logit regression models;
- mkdate: computes dates in continuous years or CMC (century month code);
- listvar: shows variables in a dataframe.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

References

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

Examples

```
# creates an episode-data structure relating to the transition
# childless-->first child
ep1<-with(demogr,epdata(start=dbirth, event=dch1, rcensor=dint,
          birth=dbirth,id=id,
          addvar=subset(demogr,select=c(-id,-dbirth))))

# creates a new episode-data structure with a time-varying factor
```

```
# variable relating to the status "never married" (not_marr) or
# "ever married" (marr)
ep2<-splitter(ep1,split=ep1$d1marr,tvar.lev=c("not_marr","marr"),
              tvar.name="mar")

# Estimates age profiles for the transition to the first birth
# according to the following factors:
# sex (respondent'sex w/2 levels: 'Male', 'Female');
# edu ('Level of education w/3 levels: 'low_sec','upp_sec', 'tert');
# mar (ever married w/2 levels: 'not_marr', 'marr')

ch1.ap<-ageprofile(formula=~sex+edu+mar, epdata=ep2,
                   tr.name="First child", agelimits=c(15,50))

# Plot age profiles in three different graphs
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("Male","Female"),title='first child by sex')
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("low_sec","upp_sec","tert"),title='first child by education')
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("not_marr","marr"),title='first child by marital status',
        ylim=0.4)
```

`ageprofile` *Computes smoothed transition rates by respondent's age (age profiles).*

Description

Computes age profiles for a specific transition between two states according to a set of time-fixed or time-varying covariates. It needs a micro-level episode-data structure, i.e. a longitudinal dataset containing the following variables:

start : starting date of observation;
Tstop : ending date of observation;
Status : is 0 if right censored; 1 if event occurred; 2 if left censored; 3 if interval censored. Status is equal to 1 if and only if the date of the event precedes the date of right censoring;
Agestart: age at the starting date (Start);
Agestop: age at the ending date (Stop)
An episode-data structure can be obtained through the command epdata.

Usage

```
ageprofile(formula, epdata, tr.name = "Transition", win,
           method = "car", agelimits = c(0, 100),
           outfile = FALSE, tails = c(FALSE, FALSE),
           subset = TRUE, weight,
           sig.eff = TRUE, sig.lev = 0.05)
```

Arguments

`formula`	a formula object specifying the ~ operator and factors variables separated by '+' operators. Response variable on the left is not required. The expression

	~1 implies that only the baseline (age profile for the whole sample) will be provided. Note that interaction terms are not recognized by the formula: ~v1:v2 or ~v1*v2 gives the same results as ~v1+v2.
`epdata`	a dataframe containing episode-data prepared by epdata command optionally with time-varying factor variables created by splitter command.
`tr.name`	a string containing the name of the considered transition
`win`	a matrix with two columns containing the initial and final calendar date specifiyng a restricted window of observation. Only events and exposure times referring to this window will be considered in the analysis. If win argument is not specified, the whole episode is considered. For example, let us consider an episode data structure coming from a retrospective survey held on January 1, 2010. Since we want to compute transition rates according to behaviours experienced in the last 10 years, we can restrict our window ob observation to the decade January 1, 2000 - January 1, 2010. Thus, the win argument would be a matrix with two columns, the first containing the exact data 2000 and the second the exact date 2010. Window of observation may be also limited to specific ages or events. For example, win=cbind(date_at_birth+15,date_at_birth+20) restricts observations within the age group 15-19 (completed) years of age and win=cbind(date_at_marriage,date_at_the_interview) restricts observatins between marriage and the interview. Note that in the win matrix no missing data are allowed.
`method`	specifies the type of rates to compute. There are three alternatives: 'cpr' for Cohort Period Rates 'car' for Cohort Age Rates
`agelimits`	a couple of values indicating the lower and the upper limit of age interval to be considered. It can be useful to restrict age interval when the transition is never or almost never experienced outside a specific age interval. For example, the transition rates for the first child birth may be limited to the age interval (15-50).
`outfile`	if TRUE writes the output on a file named 'trname.txt' where 'trname' is the string specified in the tr.name argument.
`tails`	a vector of two logical elements indicating respectively if the left and the right tails must be flattened. This option may be useful if we can assume that the transition rate is approximately zero at the borders of the considered age interval.
`subset`	an optional vector specifying a subset of observations to be used in the estimation process.
`weight`	an optional vector containing (post-sampling) weights.
`sig.eff`	if TRUE the age profile for a specific subgroup defined by factor levels is fixed as identical to the baseline if the relative risk (for aech specific age-subinterval) is not statistically significant at the level specified by sig.lev. In other words, the relative risk in a specific age subinterval is zero if the pvalue is higher than sig.lev (see details).
`sig.lev`	specifies the maximum level significance under which the relative risk in a specific age-subinterval is non-zero (if sig.eff=TRUE). The default value is 0.05.

Details

p-values for the null hypothesis that the corresponding parameter is zero is calculated with reference to the t distribution with the estimated residual degrees of freedom for the model fit if the dispersion parameter has been estimated, and the standard normal if not.

Value

Gives a list of objects:

`profiles`	a matrix containing the smoothed age profiles for the whole sample (baseline) and for each factor level considered
`unsmoothed`	a matrix containing the unsmoothed transition rates, i.e. the ratio between occurences and exposures for each age and factor level
`knot`	a matrix with the two knots (in column) for each factor level (rows)
`event`	a matrix with the number of event occured in the three age sub-intervals defined by knots (columns) for each factor level (rows)
`rrisk`	a matrix with the estimated relative risks in the three age sub-intervals (columns) for each factor level (rows)
`se`	a matrix with the standard error related to the relative risk computed in the three age sub-intervals (columns) for each factor level (rows)
`pvalue`	a matrix with the pvalues related to the hypothesis that the relative risk in each age sub-interval is different from zero (columns) for each factor level (rows)
`factors`	a vector of names containing the factors specified in the formula argument
`trname`	the name of the transition in analysisi as specified in the tr.name argument.
`ANOVAtest`	pvalues anova test for each factor covariate
`factors`	vector of string containing the names of factor covariates as specified in the formula argument.
`method`	the method used for the computaion of rates as speficied in the method argument.
`agelimits`	the age interval as specified in the agelimits argument.
`tails`	tails argument.

Note

In order to have an unambiguosly output, it is strongly raccomended to label each level of factor variables and to avoid using the same label for different factors.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

References

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

See Also

`epdata`, `splitter`, `plotap`

Examples

```
# creates an episode-data structure relating to the transition
# childless-->first child
ep1<-with(demogr,epdata(start=dbirth, event=dch1, rcensor=dint,
        birth=dbirth,id=id,
        addvar=subset(demogr,select=c(-id,-dbirth))))

# creates a new episode-data structure with a time-varying factor
# variable relating to the status "never married" (not_marr) or
# "ever married"(marr)
ep2<-splitter(ep1,split=ep1$d1marr,tvar.lev=c("not_marr","marr"),
              tvar.name="mar")

# Estimates age profiles for the transition to the first birth
# according to the following factors:
# sex (respondent'sex w/2 levels: 'Male', 'Female');
# edu ('Level of education w/3 levels: 'low_sec','upp_sec', 'tert');
# mar (ever married w/2 levels: 'not_marr', 'marr')
ch1.ap<-ageprofile(formula=~sex+edu+mar, epdata=ep2,
                   tr.name="First child", agelimits=c(15,50))

# The estimates are obtained under the hypothesis of independence among
# factors. We can relax this hp by considering the interaction between
# factors. The following commands add the interaction between sex and edu.
ep2$inter<-ep2$sex:ep2$edu
ch1.ap<-ageprofile(formula=~sex+edu+mar+inter, epdata=ep2,
                   tr.name="First child", agelimits=c(15,50))
```

`demogr.RData` *Longitudinal data on marriage and first child birth.*

Description

Longitudinal data on marriage and first child birth. Individuals interwieved in March 2004.

Usage

```
demogr
```

Format

A data frame with 2017 observations on 8 variables.

[,1] id integer ID
[,2] weight numeric Individual weight
[,3] dbirth numeric respondent's date at birth
[,4] dint numeric date at the interview
[,5] d1marr numeric date at marriage
[,6] dch1 numeric date at first child birth
[,7] sex factor sex
[,8] edu factor level od efucation

Details

Individuals were interviewed in March 2004 retrospectively on family and fertility life trajectory.

id: individual identification number (ID)
weight: individual post-sampling weight (with mean = 1)
dbirth: respondent's date at birth in exact years since January 1, 1900
dint: date at the interview in exact years since January 1, 1900 (March 2004 for all respondents)
d1marr: date at marriage (if any) in exact years since January 1, 1900. If NA, the individual has never been married before the interview.
dch1: date at marriage (if any) in exact years since January 1, 1900. If NA, the individual is childless at the interview.
sex: respondent's sex (factor w/2 levels: "Male", "Female")
edu: respondent's level of education (factor w/3 levels: "low_sec","upp_sec","tert")

Dates in exact years have been computed considering the midpoint of a specific month. Thus, March 1995 means March 15, 1995 and the date in exact years is 95+2.5/12=95.21.

Source

Synthetic data based on the format of the Generation and Gender Survey (GGS) (Vikat et al. 2007).

References

Vikat A., Speder Z., Beets G., Billari F.C., Buhler C., Desesquelles A., Fokkema T., Hoem J., MacDonald A., Neyer G., Pailhe A., Pinnelli A., Solaz A. (2007). "Generations and Gender Survey (GGS): Toward a better understanding of relationship and processes in the life course", Demographic research, 17 (14): 389-440.

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

Examples

```
str(demogr)
```

`epdata` *Prepares episode data for event history analysis.*

Description

Starting from a micro-level dataset containing dates for some relevant events, it prepares a file with an episode-data format suitable for event history analysis.

Usage

```
epdata(start, event, lcensor, rcensor,
               subset = TRUE, birth, id, addvar)
```

Arguments

`start`	a vector of dates at which the unit starts to be at risk to experience the transition. It may be unknown in case of left-censoring. When the episode is no left censored, start coincides with the beginning of the episode.
`event`	a vector of containing the date of the event of interest. It may be unknown in case of right-censoring.
`lcensor`	an optional vector of left censoring dates.
`rcensor`	an optional vector of right censoring dates.
`subset`	a logical expression indicating elements or rows to keep. It specifies the subset of cases at risk of experiencing the event of interest. If TRUE, the whole sample will be considered.
`birth`	a vector containing the respondent date of birth. This information in useful in order to compute ages at various events (NA not allowed).
`id`	a vector of identification numbers. If id remains unspecified, a default id is implemented.
`addvar`	a dataframe containing additional variables to add in the resulting output.

Details

All the specified objects must have the same length (i.e. the number of cases must be the same). For any observation at least one between start and lcensor and one between event and rcensor must be known. In general, we have start<=lcensor<= event<=rcensor.

Value

A data.frame suitable for MAPLES and R survival package. Following the 'counting process' formulation of survival analysis, each record ha the following variables:

`Tstart`	episode starting date
`Tstop`	episode ending date
`status`	0: right censored: 1: event occurred: 2: left censored: 3: interval censored. Is equal to 1 if and only if the date of the event precedes the date of right censoring.
`Agestart`	Age at starting date (Tstart)
`Agestop`	Age at the ending date (Tstop)

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

References

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

See Also

`splitter`, `ageprofile`, `plotap`

Examples

```
# creates an episode-data structure relating
# to the transition childless-->first child
ep1<-epdata(start=demogr$dbirth,
            event=demogr$dch1,
            rcensor=demogr$dint,
            birth=demogr$dbirth,
            id=demogr$id,
            addvar=subset(demogr,select=c(-id,-dbirth)))

# or, using 'with'
ep1<-with(demogr,epdata(start=dbirth, event=dch1,
          rcensor=dint, birth=dbirth,id=id,
          addvar=subset(demogr,select=c(-id,-dbirth))))
```

`listvar` *Shows variables in a dataframe.*

Description

Shows variables in a dataframe and the related number of column.

Usage

```
listvar(df)
```

Arguments

`df`	is the dataframe.

Value

a dataframe with two columns: VAR containing the names of the variables and COL with the number of related column in the original dataframe.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

Examples

```
listvar(demogr)
```

`mkdate` *Compute dates.*

Description

Computes dates in continuous years or CMC (Century Month Code) since January 1, 1900.

Usage

```
mkdate(year, month, cmc = FALSE)
```

Arguments

`year`	a vector containing the year of a date(XXXX format)
`month`	a vector containing the month of a date (XX format)
`cmc`	if TRUE date in CMC from January 1, 1900 is computed. Otherwise, date in continuos time since January 1, 1900 is computed.

Value

a vector containing dates in continuous time or CMC.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

References

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

Examples

```
dy<-c(1996,2001,2005,2003)
dm<-c(3,9,5,12)
mkdate(year=dy,month=dm)
mkdate(year=dy,month=dm, cmc=TRUE)
```

`plotap` *Plots age profiles.*

Description

Plots age profiles estimated through command ageprofile.

Usage

```
plotap(x, lev.labels, baseline = TRUE,  unsmoothed = FALSE,
        xlim, ylim, title)
```

Arguments

`x`	the resulting list given by the ageprofile command.
`lev.labels`	a vector of strings specifying the factor levels to consider. For example: lev.labels=c("Male","Female","low_sec","upp_sec","tert") specifies that 5 curves are drawn, one for each level specified. It is also possible to draw a combination of levels (under the hypothesis of independence between factors) by inserting the symbol "*" between two or more levels. For example: the string "Male*low_sec" draws a curve for the subgroups of men with a lower secondary level of education; the string "Male*low_sec*worker" draws a curve for the subgroup of lower educated men currently employed as a manual worker. Even though it is possible to draw any number of curves in one graph, we raccomend to consider no more than 5 levels (or combination of levels) in one graph.
`baseline`	if TRUE the baseline will be drawn.
`unsmoothed`	if TRUE the unsmoothed transition rates, i.e. the ratio between occurences and exposures for each age and factor level, are plotted as points in the graph.
`xlim`	a vector of two values defining the limits of X axis measured in years of age (default value: c(min(age), max(age))
`ylim`	a value defining the upper limit of the Y axis (transition rates). Given that transition rates are strictly positive, the lower limit is always 0 (default value: 2*max(baseline rates)).
`title`	title of the graph (default: transition name as stored in the ageprofile argument.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

References

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

See Also

`epdata`, `splitter`, `ageprofile`,

Examples

```
# creates an episode-data structure relating to the
# transition childless-->first child
ep1<-with(demogr,epdata(start=dbirth, event=dch1, rcensor=dint,
          birth=dbirth,id=id,
          addvar=subset(demogr,select=c(-id,-dbirth))))

# creates a new episode-data structure with a time-varying factor variable
# relating to the status "never married"(not_marr) or "ever married"(marr)
ep2<-splitter(ep1,split=ep1$d1marr,tvar.lev=c("not_marr","marr"),
              tvar.name="mar")

# Estimates age profiles for the transition to the first birth
# according to the following factors:
```

```
# sex (respondent'sex w/2 levels: 'Male', 'Female');
# edu ('Level of education w/3 levels: 'low_sec','upp_sec', 'tert');
# mar (ever married w/2 levels: 'not_marr', 'marr')

ch1.ap<-ageprofile(formula=~sex+edu+mar, epdata=ep2,
                   tr.name="First child", agelimits=c(15,50))

# Plot age profiles in three different graphs
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("Male","Female"),title='first child by sex')
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("low_sec","upp_sec","tert"),title='first child by education')
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("not_marr","marr"),title='first child by marital status',
        ylim=0.4)

# Plot age profiles for the combined effect of sex and level of education
# under the independence hypothesis
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("Female*low_sec","Female*upp_sec","Female*tert"),
        title='first child by education - women (indep hp)')
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
        lev=c("Male*low_sec","Male*upp_sec","Male*tert"),
        title='first child by education - men (indep hp)')

# The estimates are obtained under the hypothesis of independence among
# factors. We can relax this hp by considering the interaction between
# factors. The following commands add the interaction between sex and edu.
ep2$inter<-ep2$sex:ep2$edu
ch1.ap<-ageprofile(formula=~sex+edu+mar+inter, epdata=ep2,
        tr.name="First child", agelimits=c(15,50))

# Plot age profile for the interaction between sex and level of education
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
      lev=c("Female:low_sec","Female:upp_sec","Female:tert"),
      title='first child by education - women')
plotap(ch1.ap,base=TRUE, unsmoo=TRUE,
      lev=c("Male:low_sec","Male:upp_sec","Male:tert"),
      title='first child by education - men')
```

`splitter` *Creates a time-varying factor variables within a episode-data.*

Description

Splits episode in sub-episodes according to the dates specified as input. The resulting dataframe contains a new time-varying factor variable with different user-defined values in each subinterval.

Usage

```
splitter(epdata, split, tvar.name = "Tvar", tvar.lev)
```

Arguments

`epdata`	a dataframe containing an episode-data file (obtained through the epdata command)
`split`	an array containing the date(s) at one (or more) specific event(s) that causes the change in the resulting time-varying variable. The original episode is split at this date(s). If n vectors of dates have been specified, the original episode is splitted in n+1 subepisodes. For example, let us consider a matrix with three columns containing the dates at the first, second, and third child birth. The original episode may be split into (up to) four subepisodes according to non-NA specified dates (see details).
`tvar.name`	a string containing the name of the time-varying factor variable in the resulting dataframe.
`tvar.lev`	a vector of n+1 strings containing the level labels of the resulting time-varying variable (where n is the number of columns in the split argument).

Details

Multiple split dates refer to events that can be experienced sequentially (first birth, second birth, third birth, ecc or first marriage, second marriage, third marriage, etc.). This implies that
1. split dates must be strictly sequential: date1<=date2<=date3<=etc.
2. NA date cannot be followed by a non-NA date (e.g. for the j-th the sequence of dates "date1,date2, NA" is allowed and it means that event3 has not been experienced, whereas "NA, date1, date2" is not allowed because we do not have any information about the first change).
The number of rows in the split argument must be the same as the number of rows in the epdata argument.

Value

In the resulting dataframe each row is a subepisode obtained through the splitting procedure. The columns will be the same as the epdata argument plus one time-varying factor variable with levels specified through tvar.lev argument.
In order to create several time-varying variables on the same dataframe, splitter procedure should be applied repeatedly for each new time-varying variable. At any step the input dataframe is the resulting dataframe obtained in the previous step.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

References

Impicciatore R. and Billari F.C., (2011), MAPLES: A general method for the estimation of age profiles from standard demographic surveys (with an application to fertility), Demographic Research, 24(29):719-748 http://www.demographic-research.org/Volumes/Vol24/29/

See Also

`epdata`, `ageprofile`, `plotap`

Examples

```
# creates an episode-data structure relating to the transition
# childless-->first child
ep1<-with(demogr,epdata(start=dbirth, event=dch1, rcensor=dint,
          birth=dbirth,id=id,
          addvar=subset(demogr,select=c(-id,-dbirth))))

# creates a new episode-data structure with a time-varying factor
# variable relating to the status "never married" (not_marr) or
# "ever married" (marr)
ep2<-splitter(ep1,split=ep1$d1marr,tvar.lev=c("not_marr","marr"),
              tvar.name="mar")
```

`tabm` *Print regression estimates for previously fitted linear and logit regression models.*

Description

prints some basic information about a previously estimated linear regression models (Beta, pvalues, AIC, adjusted R square); for logit models, it also prints exp(Beta).

Usage

```
tabm(mod, pvalues = TRUE, digits = 3)
```

Arguments

`mod`	is an oject of class 'lm' or 'glm'.
`pvalues`	if TRUE pvalues will be printed out.
`digits`	specifies the number of digits in the output.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

See Also

`summary.lm`, `summary.glm`,

Examples

```
chi<-ifelse(!is.na(demogr$dch1),1,0)
logit.mod<- glm(chi ~ demogr$sex + demogr$edu,
                family=binomial(link = "logit"))
tabm(logit.mod)
# for more details
summary(logit.mod)
```

`tabx` *Prints univariate and bivariate frequency table.*

Description

Prints univariate or a bivariate frequency distribution table including marginal distribution and total number of cases.

Usage

```
tabx(x, y, prow = FALSE, pcol = FALSE, chisq=FALSE )
```

Arguments

`x`	a vector which can be interpred as factor.
`y`	an optional second vector which can be interpreted as factor (columns in the contingency table).
`prow`	if TRUE it adds a bivariate table containing row percentages.
`pcol`	if TRUE it adds a bivariate table containing column percentages.
`chisq`	gives the results for a Chi square test for independence of all factors.

Author(s)

Roberto Impicciatore ⟨roberto.impicciatore@unimi.it⟩

See Also

`table`, `summary.table`,

Examples

```
tabx(demogr$sex,demogr$edu,chisq=TRUE)
tabx(demogr$sex,demogr$edu,prow=TRUE,pcol=TRUE)
```

Index