

Package ‘surveyNG’

April 19, 2009

Title experimental revision of survey package

Description Complex survey samples – database interface, sparse matrices.

Version 0.3

Author Thomas Lumley

Maintainer Thomas Lumley <tlumley@u.washington.edu>

License LGPL

Depends R (>= 2.2.0), RSQLite, Matrix, survival, KernSmooth, survey

Suggests MASS, hexbin

URL <http://faculty.washington.edu/tlumley/survey/>

Repository CRAN

Date/Publication 2007-08-04 07:50:14

R topics documented:

sqlhexbin	2
sqlmodelmatrix	3
sqlocpoly	4
sqlsubst	5
sqlsurvey	5
sqlvar	7
svylm	7
svymean.sqlsurvey	8
svyquantile.sqlsurvey	9

Index	11
--------------	-----------

`sqlhexbin`*Hexagonal binning*

Description

Draws a scatterplot in which points are replaced by hexagons indicating the population weight in regions of the plot. Requires the "hexbin" package from Bioconductor

Usage

```
sqlhexbin(formula, design, xlab = NULL, ylab = NULL, ..., chunksize = 5000)
```

Arguments

<code>formula</code>	Formula specifying the two variables to plot
<code>design</code>	<code>sqlsurvey</code> object
<code>xlab</code>	x-axis label
<code>ylab</code>	y-axis label
<code>...</code>	Other arguments to <code>gplot.hexbin</code>
<code>chunksize</code>	Number of observations to fetch at a time

Details

Unlike most functions for `sqlsurvey` objects, this fetches all observations for the specified variables, and so would be slow over a network link.

Value

A object of S4 class `hexbin`, invisibly.

References

Carr, D. B. et al. (1987) Scatterplot Matrix Techniques for Large N. JASA 83, 398, 424-436.

See Also

[sqlsurvey](#), [sqlocpoly](#)

Examples

```
## Not run:
sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",
  data=system.file("apiclus1.db",package="surveyNG"),
  table.name="clus1", key="snum")

sqlhexbin(api00~api99, design=sqclus1,style="centroid")-
```

```
close(sqclus1)
## End(Not run)
```

```
sqlmodelmatrix      Create model matrix in a SQL table
```

Description

Given a formula and a `sqlsurvey` design object, create SQL tables with a model frame and model matrix.

Usage

```
sqlmodelmatrix(formula, design, fullrank = TRUE)
```

Arguments

<code>formula</code>	model formula
<code>design</code>	<code>sqlsurvey</code> object
<code>fullrank</code>	if FALSE return a matrix of indicators for any factor variables, not a full-rank design matrix

Details

Currently, no transformations are allowed in the formula, factors will be coded with treatment contrasts, and interactions are allowed only between factor variables.

The database tables created by this function will be removed if the R object is deleted and garbage collected.

Value

An object of class `sqlmodelmatrix`

See Also

`link{open.sqlmodelmatrix}` for reconnecting saved objects

Examples

```
sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",
  data=system.file("apiclus1.db",package="surveyNG"),
  table.name="clus1", key="snum")
mm <- sqlmodelmatrix(api00~stype*comp_imp+ell+api99, sqclus1)
dbGetQuery(sqclus1$conn,
  sqlsubst("select * from %%mm%% limit 2", list(mm=mm$table)))
dbListTables(sqclus1$conn)
rm(mm)
gc()
dbListTables(sqclus1$conn)
close(sqclus1)
```

sqlocpoly

Smoothing

Description

This function performs weighted density estimation and smoothing, using the binned local polynomial smoothers in the `KernSmooth` package.

Usage

```
sqlocpoly(formula, design, bandwidth, M = 401)
```

Arguments

<code>formula</code>	One-sided formula for density estimation, two-sided for smoothing
<code>design</code>	<code>sqlsurvey</code> object
<code>bandwidth</code>	Smoothing bandwidth
<code>M</code>	Number of grid points

Details

The binning is performed in SQL, so for large data sets the computation time is roughly linear in `M` and in the number of observations. `M` should be a few times larger than `range(x)/bandwidth`, but can often be reduced from the default value.

Value

An object of class `svysmooth`

See Also

[sqlsurvey](#)

Examples

```
sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",
  data=system.file("apiclus1.db",package="surveyNG"),
  table.name="clus1", key="snum")

dens <- sqlocpoly(~api00, sqclus1, bandwidth=50)
smooth <- sqlocpoly(api00~api99, sqclus1, bandwidth=50)

dens
plot(dens)
plot(smooth)

close(sqclus1)
```

sqlsubst	<i>Substitute variables into a text string</i>
----------	--

Description

When the string contains a name bracketed on both signs by `%%`, substitute the element of `values` with the same name. If the element of `values` is a vector then concatenate with comma separators first.

Usage

```
sqlsubst(strings, values)
```

Arguments

<code>strings</code>	Vector of character strings
<code>values</code>	A list of vectors of values to substitute

Value

Substituted vector of strings

Examples

```
sqlsubst("get %%vars%% from %%table%%",
  list(vars=c("name", "age", "sex"), table="survey_data"))
```

sqlsurvey	<i>Survey design based on SQL database</i>
-----------	--

Description

Specifies a survey data set based on a connection to a SQL (currently SQLite) database. The data may be in an existing database or in a data frame that will be written out to a database. If data are in an existing database they should be in a single table in denormalized form.

Usage

```
sqlsurvey(id, strata = NULL, weights = NULL, fpc = "0", data,
  table.name = basename(tempfile("_tbl_")), key = "row_names")
## S3 method for class 'sqlsurvey':
close(con, tidy=TRUE, ...)
## S3 method for class 'sqlsurvey':
open(con, db=NULL, ...)
## S3 method for class 'sqlmodelmatrix':
open(con, design, ...)
```

Arguments

<code>id</code>	vector of strings with names of sampling unit identifiers
<code>strata</code>	vector of strings with names of stratum identifies
<code>weights</code>	string with name of weighting variable
<code>fpc</code>	string with name of population size variable (variable may be zero for sampling with replacement or infinite population size)
<code>data</code>	Either a data frame or the name of a SQLite database file
<code>table.name</code>	Name for the data table containing the survey data and design variables
<code>key</code>	name of a variable that can be used as a unique key
<code>con</code>	survey object or survey model matrix object to be disconnected from or reconnected to the database
<code>tidy</code>	Clear any pending results?
<code>db</code>	An existing connection (optional, allows multiple objects to be reconnected to the same database connection)
<code>design</code>	A <code>sqlsurvey</code> object with an active database connection
<code>...</code>	For future expansion

Value

`sqlsurvey` returns an object of class `sqlsurvey`

Examples

```
library(survey)
data(api)
sqclus2<-sqlsurvey(id=c("dnum", "snum"), fpc=c("fpc1", "fpc2"),
  weights="pw", data=apiclus2)
sqclus2
svymean(~api99+api00, design=sqclus2)

sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",
  data=system.file("apiclus1.db", package="surveyNG"),
  table.name="clus1", key="snum")

sqclus1
svymean(~api99+api00, design=sqclus1)
close(sqclus1)
close(sqclus2)
```

`sqlvar`*Horvitz-Thompson estimator for multistage sampling*

Description

Computes the estimated variance matrix for a population total estimate under stratified multistage cluster sampling with or without replacement. The variables must be in a separate table from the main survey data table; they will typically be estimating functions. The table must have the same key variable as the main data table, which will be used for an inner join.

Usage

```
sqlvar(U, utable, design)
```

Arguments

<code>U</code>	Vector of strings for variable names
<code>utable</code>	String with name of data table for variables
<code>design</code>	<code>sqlsurvey</code> object

Value

A matrix

See Also

[sqlsurvey](#)

`svylm`*Linear regression models*

Description

Fits a linear regression model to a `sqlsurvey` object using SQL code.

Usage

```
svylm(formula, design)
```

Arguments

<code>formula</code>	Model formula
<code>design</code>	<code>sqlsurvey</code> object

Value

Vector of regression coefficients with attribute "var" containing variance matrix.

See Also

[sqlmodelmatrix](#)

Examples

```
sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",
  data=system.file("apiclus1.db",package="surveyNG"),
  table.name="clus1", key="snum")
svylm(api00~api99+stype*comp_imp, design=sqclus1)
close(sqclus1)
```

svymean.sqlsurvey *Means and totals*

Description

Horvitz-Thompson estimator of means and totals for a database-backed survey object. Computes cell proportions and counts for factor variables.

Usage

```
## S3 method for class 'sqlsurvey':
svymean(x, design, na.rm = FALSE, byvar = NULL,
  se = FALSE, keep.estfun = FALSE, ...)
## S3 method for class 'sqlsurvey':
svyttotal(x, design, na.rm = FALSE, byvar = NULL,
  se = FALSE, keep.estfun = FALSE, ...)
```

Arguments

x	model formula
design	sqlsurvey object
na.rm	Remove missing values (treat as domain)
byvar	Formula with grouping variables
se	Compute variance-covariance matrix?
keep.estfun	Keep the SQL table of estimating functions after the standard errors are computed.
...	For future expansion

Value

A data frame with a column for each variable and a row for each group, with the covariance matrix in the "var" attribute.

See Also[sqlmodelmatrix](#)**Examples**

```

sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",
  data=system.file("apiclus1.db",package="surveyNG"),
  table.name="clus1", key="snum")

sqclus1
svymean(~api99+stype, design=sqclus1)
svytotal(~enroll+stype, design=subset(sqclus1, api99 > 500))
close(sqclus1)

```

svyquantile.sqlsurvey

Probability-weighted quantiles

Description

Estimates quantiles from data in a `sqlsurvey` object.

Usage

```
svyquantile.sqlsurvey(x, design, quantiles, build.index = FALSE, ...)
```

Arguments

<code>x</code>	Model formula specifying variables
<code>design</code>	<code>sqlsurvey</code> object
<code>quantiles</code>	Quantiles to be estimated
<code>build.index</code>	Build a temporary index (may increase speed)
<code>...</code>	For future expansion

Details

Currently this function does not compute confidence intervals, and requires `quantiles` to be a single number. Both limitations will be relaxed in future versions.

Value

A vector of quantiles.

See Also[sqlsurvey](#)

Examples

```
sqclus1<-sqlsurvey(id="dnum", fpc="fpc", weights="pw", strata="fpc",  
  data=system.file("apiclus1.db",package="surveyNG"),  
  table.name="clus1", key="snum")
```

```
sqclus1  
svyquantile(~api99+api00, design=sqclus1, quantiles=0.5)  
close(sqclus1)
```

Index

*Topic **hplot**

- sqlhexbin, 1
- sqlocpoly, 3

*Topic **survey**

- sqlhexbin, 1
- sqlmodelmatrix, 2
- sqlocpoly, 3
- sqlsubst, 4
- sqlsurvey, 5
- sqlvar, 6
- svylm, 7
- svymean.sqlsurvey, 8
- svyquantile.sqlsurvey, 9

close.sqlsurvey(*sqlsurvey*), 5

dim.sqlsurvey(*sqlsurvey*), 5

open.sqlmodelmatrix(*sqlsurvey*), 5

open.sqlsurvey(*sqlsurvey*), 5

sqlhexbin, 1

sqlmodelmatrix, 2, 7, 8

sqlocpoly, 2, 3

sqlsubst, 4

sqlsurvey, 2, 4, 5, 7, 9

sqlvar, 6

svylm, 7

svymean.sqlsurvey, 8

svyquantile.sqlsurvey, 9

svytotal.sqlsurvey
(*svymean.sqlsurvey*), 8