

# Package ‘StratSel’

March 24, 2025

**Type** Package

**Title** Strategic Selection Estimator

**Version** 1.4

**Description** Provides functions to estimate a strategic selection estimator. A strategic selection estimator is an agent error model in which the two random components are not assumed to be orthogonal. In addition this package provides generic functions to print and plot objects of its class as well as the necessary functions to create tables for LaTeX. There is also a function to create dyadic data sets.

**License** GPL ( $\geq 2$ )

**Depends** R ( $\geq 3.5.0$ ), MASS, memisc, Formula, mnormt, pbivnorm

**LazyData** TRUE

**NeedsCompilation** no

**Author** Lucas Leemann [aut, cre]

**Maintainer** Lucas Leemann <lleemann@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-03-24 20:40:02 UTC

## Contents

StratSel-package . . . . .	2
data.fake . . . . .	3
fetch.rho.b . . . . .	4
fetch.rho.v . . . . .	5
gen.Startval . . . . .	7
getSummary . . . . .	8
logLik . . . . .	9
logLikStrat . . . . .	9
logLikStratSel . . . . .	11
makeDyadic . . . . .	13
plot . . . . .	14
predict . . . . .	15
print . . . . .	16

print.summary . . . . .	16
setStratSelDefault . . . . .	17
StratSel . . . . .	17
summary . . . . .	19
vcov . . . . .	19
war1800 . . . . .	20
<b>Index</b>	<b>22</b>

---

StratSel-package	<i>This package allows to estimate strategic selection models.</i>
------------------	--

---

## Description

This package provides functionality to estimate, summarize, plot, predict, and export strategic selection estimates. It allows researchers to incorporate the strategic nature of the DGP while not constraining the errors to be orthogonal. By relaxing the assumptions, this estimator becomes a blend of an agent error model and a Heckman selection model.

## Details

Package: StratSel  
 Type: Package  
 Version: 1.4  
 Date: 2025-03-24  
 License: GPL (>= 2)

## Author(s)

Lucas Leemann <lleemann@gmail.com>

## References

Lucas Leemann. 2014. "Strategy and Sample Selection - A Strategic Selection Estimator", *Political Analysis* 22: 374-397.

## See Also

games

## Examples

```
# replicate the example from Leemann (2014):
library(memisc)
data(war1800)
## Not run: out1 <- StratSel(Y ~ s_wt_re1 + revis1 | dem1 + mixed1 | balanc +
  dem2 + mixed2, data=war1800, corr=TRUE)
## End(Not run)
out2 <- StratSel(Y ~ s_wt_re1 + revis1 | dem1 + mixed1 | balanc +
  dem2 + mixed2, data=war1800, corr=FALSE)
setStratSelDefault()
## Not run: z <- mtable(out1,out2)
# toLatex(z) for a LaTeX output or just regular table:
```

---

data.fake

*Fake Data for Illustration*

---

## Description

This data is just for illustration. The code to generate it is:

```
set.seed(124)
n <- 1000
x24 <- cbind(rnorm(n), rnorm(n))
error <- rmnorm(n, c(0,0), matrix(c(1,0.6,0.6,1),2,2))
e24 <- error[,2]
y24.latent <- x24*%c(1,1) + e24
y2 <- rep(NA,n)
y2[y24.latent>0] <- 1
y2[y24.latent<0] <- 0
mod2 <- glm(y2 ~ x24, family=binomial(link=probit))
p24 <- pnorm(predict(mod2))
x11 <- cbind(rnorm(n, sd=0.2), rnorm(n, sd=0.2))
x14 <- cbind(x24[,2], rnorm(n))
e14 <- error[,1]
y14.latent <- x14*%c(2,1) * p24 - x11*%c(1,1) + e14
y1 <- rep(NA,n)
y1[y14.latent>0] <- 1
y1[y14.latent<0] <- 0
Y <- rep(NA,n)
Y[y1==0] <- 1
Y[y1==1&y2==0] <- 3
Y[y1==1&y2==1] <- 4
colnames(x11) <- c("var A", "var B")
colnames(x14) <- c("var C", "var D")
colnames(x24) <- c("var E", "var C")
data.fake <- data.frame(Y, x11, x14, x24)
```

**Usage**

```
data(data.fake)
```

**Format**

A data frame with 1000 observations on the following 7 variables.

Y A numeric vector with values 1,3, and 4 depending on which outcome occurred.

var.A A numeric vector mimicking an explanatory variable as part of X11.

var.B A numeric vector mimicking an explanatory variable as part of X11.

var.C A numeric vector mimicking an explanatory variable as part of X14 and of X24.

var.D A numeric vector mimicking an explanatory variable as part of X14.

var.E A numeric vector mimicking an explanatory variable as part of X24.

var.C.1 A numeric vector mimicking an explanatory variable as part of X14 and of X24. Identical to var.C.

**Source**

Can be independently re-created by anybody.

**Examples**

```
data(data.fake)
summary(data.fake)
## Not run: out1 <- StratSel(Y ~ var.A + var.B | var.C + var.D |
var.E + var.C, data=data.fake, corr=TRUE)
## End(Not run)
## Not run: summary(out1)
# True parameters are 1 or 2 except the three constant terms (which are 0).
# The correlation parameter was set to +0.6.
```

---

 fetch.rho.b

---

*Function to transform  $f(\rho)$  back to  $\rho$* 


---

**Description**

The model has a correlation parameter which is estimated and theoretically bound between -1 and +1. To ensure that the estimated parameters are within the theoretical bounds a transformation is necessary. The chosen transformation is:

$$f(\rho) : \rho = \frac{2}{(1 - \exp(-\theta))} - 1$$

Whereas  $\rho$  is the actual correlation coefficient and  $\theta$  is the parameter we estimate in the model. This parametrization has been worked into the likelihood function and ensures that  $\rho$  will be between  $-1$  and  $+1$ .

**Usage**

```
fetch.rho.b(b)
```

**Arguments**

**b** The vector of estimated coefficients ( $\beta$ ) whereas the last element is the transformed correlation coefficient (here:  $\theta$ ).

**Details**

This function is for internal use but documented as a regular function to enable any user to assess the estimator and its functionality.

**Value**

The function returns the correct estimate for  $\rho$ .

**Note**

We want to estimate  $\rho$  but because it is theoretically bound, we estimate  $\theta$  which is not bound can range from  $-\infty$  to  $+\infty$ .

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**See Also**

[StratSel](#)

**Examples**

```
test <- c(1,1,-2.35)
fetch.rho.b(test)
```

---

fetch.rho.v

*Function to transform var( $\theta$ ) back to var( $\rho$ )*

---

**Description**

The model has a correlation parameter which is estimated and theoretically bound between -1 and +1. To ensure that the estimated parameters are within the theoretical bounds a transformation is necessary. The chosen transformation is:

$$f(\rho) : \rho = \frac{2}{(1 - \exp(-\theta))} - 1$$

Whereas  $\rho$  is the actual correlation coefficient and  $\theta$  is the parameter we estimate in the model. This parametrization has been worked into the likelihood function and ensures that  $\rho$  will be between  $-1$  and  $+1$ .

The variance covariance matrix thus contains entries based on  $\theta$  but not  $\rho$ . Hence, this function takes the variance of the transformed correlation parameter ( $\theta$ ) and produces the value correct for  $\rho$ .

To create the correct measure of  $var(\rho)$  this function simulates 1,000  $\theta$ 's and then transforms them to  $\rho$ 's. The variance of these  $\rho$ 's is then reported. Note, this means that the variance-covariance returned by `StratSel` is only correct for all diagonals and off-diagonals for the parameters ( $\beta$ ) but for the correlation coefficient only the variance is correct. Given that there is no reason to use the full variance-covariance for post-estimation commands this is not a problem.

### Usage

```
fetch.rho.v(v, b)
```

### Arguments

v	Variance-covariance matrix based on the regular parameters ( $\beta$ ) and $\theta$ .
b	Coefficient vector, first $n$ elements are for the regular parameter estimates and the last element is $\theta$ .

### Details

This function is for internal use but documented as a regular function to enable any user to assess the estimator and its functionality.

### Value

Returns the correct variance estimate for the estimate of the correlation coefficient  $\rho$ .

### Author(s)

Lucas Leemann <lleemann@gmail.com>

### See Also

[StratSel](#)

### Examples

```
fetch.rho.v(matrix(c(1,0,0,1),2,2),c(0,0))  
fetch.rho.v(matrix(c(1,0,0,2),2,2),c(0,0))
```

---

gen.Startval                      *Generates good starting values for a strategic selection model*

---

### Description

The function creates good starting values based on the supplied data and model which are to be estimated. To do so, the function runs two probit models, whereas the first one is just on the lower node of the game tree (see StratSel). It then creates predicted probabilities ( $p_{24}$ ) to estimate a second probit at the first node whereas the variables which are part of  $X_{14}$  are weighted by  $p_{24}$ .

### Usage

```
gen.Startval(Startval, user.supplied.startval, corr, ys, xs11, xs14, xs24,
dim.x11, dim.x14, dim.x24)
```

### Arguments

Startval	Optional. A vector of user supplied starting values.
user.supplied.startval	Logical. If TRUE this function just returns the vector Startval. If the estimated model does not restrict the correlation (so corr=TRUE) then a 0 is attached to the vector Startval.
corr	Logical. Indicates whether the estimated agent error model assumes orthogonal errors (=FALSE) or whether the correlation is estimated (=TRUE).
ys	Vector. The outcome variable which is supplied by the user to <a href="#">StratSel</a> .
xs11	Matrix. Explanatory variables for player 1 and measuring utility from outcome 1.
xs14	Matrix. Explanatory variables for player 1 and measuring utility from outcome 4.
xs24	Matrix. Explanatory variables for player 2 and measuring utility from outcome 4.
dim.x11	Vector. Has two elements for the dimension of X11.
dim.x14	Vector. Has two elements for the dimension of X14.
dim.x24	Vector. Has two elements for the dimension of X24.

### Details

This function is for internal use but documented as a regular function to enable any user to assess the estimator and its functionality.

### Value

Vector. Has length of the number of parameters to be estimated.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**See Also**

[StratSel](#)

---

getSummary

*getSummary Method for extending mtable()*

---

**Description**

This function extends the `mtable()` to report strategic selection models (`StratSel`). Together with `setStratSelDefault` and the `mtable` command from the `memisc` package users can create multi-model tables and export them to LaTeX.

**Usage**

```
## S3 method for class 'StratSel'  
getSummary(obj, alpha = 0.05, ...)
```

**Arguments**

<code>obj</code>	An object of class <code>StratSel</code> .
<code>alpha</code>	Significance level.
<code>...</code>	additional arguments

**Value**

Returns a list of objects to be fed to `mtable`. Do not use this command directly. The command `mtable` will automatically call this function for an object of the `StratSel` class.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**References**

Elff, Martin. (2013). *memisc: Tools for Management of Survey Data, Graphics, Programming, Statistics, and Simulation* R package version 0.96-7.

**See Also**

[toLatex](#), [mtable](#)



**Examples**

```

data(data.fake)
out1 <- StratSel(Y ~ var.A | var.D | var.E , data=data.fake, corr=FALSE)
out2 <- StratSel(Y ~ var.A | var.C | var.E, data=data.fake, corr=FALSE)
mtable(out1,out2)

```

logLik

*Function to Extract Log-Likelihood from Objects of Class StratSel***Description**

Generic logLik function for objects of class StratSel.

**Usage**

```

## S3 method for class 'StratSel'
logLik(object, ...)

```

**Arguments**

object	An object of class StratSel
...	additional arguments.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

logLikStrat

*Log-Likelihood Function of an Agent Error Model***Description**

This function calculates the log-likelihood value for an agent error model (belongs to the general class of quantal response models). The underlying formal structure is



and shows a game where there are two players which move sequentially. Player 1 decides to move left or right and if she does move right player 2 gets to move. The final outcome in this case depends on the move of player 2.

**Usage**

```
logLikStrat(x11, x14, x24, y, beta)
```

**Arguments**

x11	A vector or a matrix containing the explanatory variables used to parametrize u11.
x14	A vector or a matrix containing the explanatory variables used to parametrize u14.
x24	A vector or a matrix containing the explanatory variables used to parametrize u24.
y	Vector. Outcome variable which can take values 1, 3, and 4 depending on which outcome occurred.
beta	Vector. Coefficients of the model.

**Details**

This function provides the likelihood of an agent error model (Signorino, 2003). Note, that to derive it one assumes that the two errors are independent. Further, as with probit and logit models, one needs to assume an error variance to achieve identification. Signorino uses  $\sqrt{2}$  while logLikStrat uses 1. Hence, the numeric results will differ, but all relevant statistics (predicted probabilities, z-values, ...) will be identical. Finally, u13 and u23 are set to 0 to achieve identification.

**Value**

Returns a numeric value for the log-likelihood function evaluated for  $\beta$ .

**Note**

The log-likelihood function:

$$\ell\ell = \sum_{i=1}^n (\log(p_{i1}) \cdot I(Y_i = 1) + \log((1 - p_{i1})(1 - p_{i4})) \cdot I(Y_i = 3) + \log((1 - p_{i1})(p_{i4})) \cdot I(Y_i = 4))$$

whereas

$$p_{i24} = \Phi(x_{24} \cdot \beta_{24})$$

and

$$p_{i1} = \Phi(x_{11} \cdot \beta_{11} - p_{24}(x_{14} \cdot \beta_{14}))$$

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**References**

Curtis S. Signorino. 2003. "Structure and Uncertainty in Discrete Choice Models." *Political Analysis* 11:316–344.

**See Also**[StratSel](#)

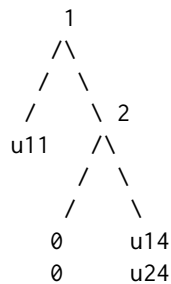

---

logLikStratSel	<i>Log-Likelihood Function of an Agent Error Model with Correlated Errors (strategic selection model)</i>
----------------	---

---

**Description**

This function calculates the log-likelihood value for an agent error model (belongs to the general class of quantal response models) with correlated errors. The underlying formal structure is



and shows a game where there are two players which move sequentially. Player 1 decides to move left or right and if she does move right player 2 gets to move. The final outcome in this case depends on the move of player 2.

**Usage**

```
logLikStratSel(x11, x14, x24, y, beta)
```

**Arguments**

x11	A vector or a matrix containing the explanatory variables used to parametrize u11.
x14	A vector or a matrix containing the explanatory variables used to parametrize u14.
x24	A vector or a matrix containing the explanatory variables used to parametrize u24.
y	Vector. Outcome variable which can take values 1, 3, and 4 depending on which outcome occurred.
beta	Vector. Coefficients of the model whereas the last element is the correlation coefficient $\rho$ . Note, that this parameter has been re-paramterized (see <i>details</i> ).

**Details**

This function provides the likelihood of an agent error model (Signorino, 2003) but in addition allows the random components to be correlated and hence can take selection into account. The correlation parameter is re-paramaterized (see *Note*). Further, as with probit and logit models, one needs to assume an error variance to achieve identification, here 1 is chosen as with a regular probit model. Finally, u13 and u23 are set to 0 to achieve identification.

**Value**

Returns a numeric value for the log-likelihood function evaluated for  $\beta$ .

**Note**

The notation  $\Phi_2(a; b; c)$  indicates a bivariate standard normal cumulative distribution evaluated at the values a, b whereas the two random variables have a correlation of c.

$$\begin{aligned} \ell\ell = & \sum_{i=1}^n \log \left( \Phi_2(p_{i4}(\mathbf{x}_{i14}\boldsymbol{\beta}_{14}) - \mathbf{x}_{i11}\boldsymbol{\beta}_{11}; \mathbf{x}_{i24}\boldsymbol{\beta}_{24}; -\rho)^{(1-I(y_i=1))(1-I(y_i=4))} \right) \\ & + \sum_{i=1}^n \log \left( \Phi_2(p_{i4}(\mathbf{x}_{i14}\boldsymbol{\beta}_{14}) - \mathbf{x}_{i11}\boldsymbol{\beta}_{11}; \mathbf{x}_{i24}\boldsymbol{\beta}_{24}; \rho)^{(1-I(y_i=1))I(y_i=4)} \right) \\ & + \sum_{i=1}^n \log (1 - \Phi(p_{i4}\mathbf{x}_{i14}\boldsymbol{\beta}_{14} - \mathbf{x}_{i11}\boldsymbol{\beta}_{11})) \end{aligned}$$

whereas

$$p_{i24} = \Phi(x_{i24} \cdot \beta_{24})$$

and

$$p_{i11} = \Phi(x_{i11} \cdot \beta_{11} - p_{i24}(x_{i14} \cdot \beta_{14}))$$

The re-parametrization is as follows:

$$\rho = \frac{2}{1 - \exp(-\theta)} - 1$$

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**References**

Lucas Leemann. 2014. "Strategy and Sample Selection - A Strategic Selection Estimator", *Political Analysis* 22: 374-397.

**See Also**

[StratSel](#)

---

`makeDyadic`*A Function To Create Dyadic Data Sets*

---

## Description

This function allows the user to create dyadic data sets which can be directed or undirected.

## Usage

```
makeDyadic(x, directed = FALSE, show.progress = 5)
```

## Arguments

<code>x</code>	The data matrix whereas the first variable is the country code and the second column has to be the time variable.
<code>directed</code>	Logical. If <code>directed=TRUE</code> the dyadic data set will be directed and have variables with amended names (creating <code>sen_varname</code> for sender and <code>rec_varname</code> for receiver).
<code>show.progress</code>	Logical. The process may take some time depending on the size of the supplied data matrix. This option allows users to receive feedback of how far along the process is at periodical steps. Default is set to 5.

## Details

This function was first written for Simon Collrad-Wexler and then later amended for Fabio Wasserfallen.

## Value

Returns a data frame with the dyadic data set.

## Author(s)

Lucas Leemann <lleemann@gmail.com>

## Examples

```
dataOrig <- matrix(c( rep(c(1:4),3), rep(1,4), rep(2,4), rep(3,4),  
  rnorm(4,1.5,0.1), rnorm(4,2.5,0.1), rnorm(4,3.5,0.1), rnorm(4,4.5,0.1),  
  rnorm(4,5.5,0.1), rnorm(4,6.5,0.1)),12,4)  
colnames(dataOrig) <- c("countryCODE", "Year", "Variable 1", "Variable 2")  
dataNew <- makeDyadic(dataOrig, directed=TRUE)
```

---

plot

*Plots a StratSel Object*


---

**Description**

Plots predicted probabilities for all three possible outcomes based on an object of class StratSel.

**Usage**

```
## S3 method for class 'StratSel'
plot(x, profile, x.move, x.range, uncertainty = FALSE,
      n.sim = 100, ci = 0.95, ylim, xlab, ylab1, ylab2, ylab3, plot.nr, ...)
```

**Arguments**

x	An object of class StratSel
profile	Vector. The values of all independent variables including the three constants.
x.move	Scalar. Indicates which variable is changing (and displayed on the x-axis).
x.range	Vector. A vector with two elements. The x which is moving will start at x.range[1] and end at x.range[2].
uncertainty	Logical. Indicates whether confidence bands should be displayed or not.
n.sim	Scalar. If uncertainty=TRUE the prediction uncertainty is generated with a simulation based on n.sim simulations. Default is 100. It is recommended to use 1000 simulations.
ci	Scalar. Indicates which confidence interval should be plotted, the default is 0.95.
ylim	Vector. A vector with two elements defining the range of the plotted y (predicted probability).
xlab	String. A label to be used for the x-axis. Will be recycled in all three plots.
ylab1	String. Label for the y-axis of the first plot (predicted probability of outcome 1).
ylab2	String. Label for the y-axis of the second plot (predicted probability of not outcome 1).
ylab3	String. Label for the y-axis of the third plot (predicted probability of outcome 4).
plot.nr	Vector. If one does not want to plot all three outcomes, one can use this vector to indicate which plot(s) should be shown.
...	Further arguments to be supplied to plot.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**Examples**

```

data(data.fake)
# Running just an agent error model (note: corr=FALSE) with \code{var.C} being
#part of both actors' utilities
out1 <- StratSel(Y ~ var.A + var.B | var.C + var.D | var.E + var.C, data=data.fake, corr=FALSE)
par(mfrow=c(3,1))
plot(out1, profile=c(1,0.2,-0.2,1,0.2,-0.2,1,0.1,-0.3),
     x.move=c(5,9),x.range=c(-15,15), ci=0.7, uncertainty=TRUE)

```

---

predict

*Prediction Function for Objects of the StratSel Class*


---

**Description**

Prediction function for objects of the StratSel class. Provides either predictions for all observations in a model or for a specified profile. In addition, the function will either predict an outcome or three probabilities (indicating the probability for each outcome).

**Usage**

```

## S3 method for class 'StratSel'
predict(object, prob = FALSE, profile, ...)

```

**Arguments**

object	An object of class StratSel.
prob	Logical. If prob=TRUE the function will compute the predicted probabilities for each of the three possible outcomes. Default is prob=FALSE.
profile	Vector. A vector defining a specific profile for which the prediction is made.
...	...

**Value**

Either a matrix with dimension  $n \times m$ , where there are  $n$  observations in the original model and  $m$  is three (for the three possible outcomes) or it will be a vector with  $n$  elements indicating for each observation which the most likely outcome would be.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**Examples**

```

data(data.fake)
out1 <- StratSel(Y ~ var.A + var.B | var.C + var.D | var.E + var.C, data=data.fake, corr=FALSE)
predict(out1)
predict(out1, prob=TRUE)
predict(out1, profile=c(1,0.2,0.2,1,0.2,0.2,1,0.2,0.2))

```

print *Print Function for Objects of Class StratSel*

---

**Description**

Generic print function for objects of class StratSel.

**Usage**

```
## S3 method for class 'StratSel'  
print(x,...)
```

**Arguments**

x                   An object of class StratSel  
...                  additional arguments.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

---

print.summary *Function to Print the Summary Output of an Object of Class StratSel*

---

**Description**

Function to print the summary output of an object of class StratSel

**Usage**

```
## S3 method for class 'StratSel'  
print.summary(x, ...)
```

**Arguments**

x                   An object of class StratSel  
...                  additional arguments.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>



---

setStratSelDefault      *Function Changes Default Settings to Use mtable Command*

---

**Description**

Function changes default settings to use mtable command.

**Usage**

```
setStratSelDefault()
```

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

**References**

Elff, Martin. (2013). *memisc*: Tools for Management of Survey Data, Graphics, Programming, Statistics, and Simulation R package version 0.96-7.

**See Also**

See `link[memisc]{mtable}` table command in the memisc package.

---

StratSel      *Fitting Strategic Selection Models*

---

**Description**

This function estimates a strategic selection estimator. This function fits a strategic selection estimator which is based on an agent error model (belongs to the general class of quantal response models). The underlying formal structure is



and shows a game where there are two players which move sequentially. Player 1 decides to move left or right and if she does move right player 2 gets to move. The final outcome in this case depends on the move of player 2.

**Usage**

```
StratSel(formula, corr = TRUE, Startval, optim.method = "BFGS", data, ...)
```

**Arguments**

formula	The formula has the following form $Y \sim X11   X14   X24$ whereas $Y$ is the outcome variable taking values 1, 3, or 4 depending on which outcome resulted. The explanatory variables are supplied via $X11$ , $X14$ , and $X24$ depending on which part of the utility function they should explain.
corr	Logical. If <code>corr=TRUE</code> a strategic selection estimator is estimated. If <code>corr=FALSE</code> an ordinary agent error model is estimated as the command <code>egame12</code> in the <code>games</code> package does. Note, that the variance of the error terms is fixed differently and hence the coefficients will be different, but all resulting statistics (z-values, predicted probabilities, ...) are identical as the fixed variances are identifying assumptions. See <a href="#">logLikStratSel</a> for details.
Startval	Vector. Allows the user to specify starting values. If there is no user-supplied vector the function will generate starting values itself. It is strongly recommended to let the function determine the optimal starting values.
optim.method	Optimization method to be used by <code>optim</code> ; can be: "Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", or "Brent". See <code>optim</code> for details. The default is "BFGS".
data	an optional data frame, list or environment (or object coercible by <a href="#">as.data.frame</a> to a data frame) containing the variables in the model. If not found in <code>data</code> , the variables are taken from <code>environment(formula)</code> , typically the environment from which <code>StratSel</code> is called.
...	additional arguments.

**Value**

`StratSel` returns an object of class `StratSel` for which appropriate [plot](#), [print](#), [summary](#), and [predict](#) functions exist.

**Author(s)**

Lucas Leemann <[lleemann@gmail.com](mailto:lleemann@gmail.com)>

**References**

- Lucas Leemann. 2014. "Strategy and Sample Selection - A Strategic Selection Estimator", *Political Analysis* 22: 374-397.
- Curtis S. Signorino. 2003. "Structure and Uncertainty in Discrete Choice Models." *Political Analysis* 11:316-344.

**Examples**

```
# replicate the example from Leemann (2014):
data(war1800)
## Not run: out1 <- StratSel(Y ~ s_wt_re1 + revis1 | dem1 + mixed1 | balanc
+ dem2 + mixed2, data=war1800, corr=TRUE)
## End(Not run)
out2 <- StratSel(Y ~ s_wt_re1 + revis1 | dem1 + mixed1 | balanc
+ dem2 + mixed2, data=war1800, corr=FALSE)
```

summary

*Summary Function for StratSel Objects***Description**

Summary function for StratSel objects which displays a table of estimated coefficients and their standard errors.

**Usage**

```
## S3 method for class 'StratSel'
summary(object, ...)
```

**Arguments**

object	An object of class StratSel
...	...

**Note**

See [StratSel](#) help-file for an example.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

vcov

*Function to Extract Variance-Covariance from Objects of Class StratSel***Description**

Generic vcov function for objects of class StratSel.

**Usage**

```
## S3 method for class 'StratSel'
vcov(object, ...)
```

**Arguments**

object            An object of class StratSel  
...                additional arguments.

**Author(s)**

Lucas Leemann <lleemann@gmail.com>

---

war1800

*A Data Set for Illustrative Purposes*

---

**Description**

This is a subset (only some variables included) of the data set which is also included in the package `games`. The data set can also be used to replicate the example that is provided in Leemann (2014) illustrating the strategic selection estimator. It is a data set of militarized international disputes between 1816 and 1899.

**Usage**

```
data(war1800)
```

**Format**

A data frame with 313 observations on the following 10 variables.

esc a numeric vector  
war a numeric vector  
dem1 a numeric vector  
mixed1 a numeric vector  
dem2 a numeric vector  
mixed2 a numeric vector  
s\_wt\_re1 a numeric vector  
revis1 a numeric vector  
balanc a numeric vector  
Y a numeric vector

**Source**

This data set is taken from the package `games`.

## References

Daniel M. Jones, Stuart A. Bremer and J. David Singer. 1996. "Militarized Interstate Disputes, 1816-1992: Rationale, Coding Rules, and Empirical Patterns." *Conflict Management and Peace Science* 15(2): 163–213.

Lucas Leemann. 2014. "Strategy and Sample Selection - A Strategic Selection Estimator", *Political Analysis* 22: 374-397.

## Examples

```
data(war1800)
summary(war1800)
```

# Index

- \* **Creating dyadic datasets**
    - makeDyadic, [13](#)
  - \* **Data war1800**
    - war1800, [20](#)
  - \* **Illustration**
    - plot, [14](#)
  - \* **LaTeX**
    - getSummary, [8](#)
    - setStratSelDefault, [17](#)
  - \* **Log-Likelihood Function**
    - logLikStrat, [9](#)
    - logLikStratSel, [11](#)
  - \* **Make tables**
    - getSummary, [8](#)
  - \* **Prediction**
    - predict, [15](#)
  - \* **Producing tables**
    - setStratSelDefault, [17](#)
  - \* **Starting values**
    - gen.Startval, [7](#)
  - \* **Strategic Selection Estimator**
    - StratSel, [17](#)
  - \* **Strategic models**
    - StratSel-package, [2](#)
  - \* **correlation coefficient**
    - fetch.rho.b, [4](#)
    - fetch.rho.v, [5](#)
  - \* **datasets**
    - data.fake, [3](#)
  - \* **transformation**
    - fetch.rho.b, [4](#)
    - fetch.rho.v, [5](#)
- as.data.frame, [18](#)
- data.fake, [3](#)
- fetch.rho.b, [4](#)
- fetch.rho.v, [5](#)
- gen.Startval, [7](#)
- getSummary, [8](#)
- logLik, [9](#)
- logLikStrat, [9](#)
- logLikStratSel, [11](#), [18](#)
- makeDyadic, [13](#)
- mtable, [8](#)
- plot, [14](#), [18](#)
- predict, [15](#), [18](#)
- print, [16](#), [18](#)
- print.summary, [16](#)
- setStratSelDefault, [17](#)
- StratSel, [5–8](#), [11](#), [12](#), [17](#), [19](#)
- StratSel-package, [2](#)
- summary, [18](#), [19](#)
- toLatex, [8](#)
- vcov, [19](#)
- war1800, [20](#)