**scobit — Skewed logistic regression**

**Description**

scobit fits a maximum-likelihood skewed logit model.

**Quick start**

Skewed logistic regression of binary variable y on x1 and x2

```
scobit y x1 x2
```

Report results as odds ratios

```
scobit y x1 x2, or
```

With robust standard errors

```
scobit y x1 x2, vce(robust)
```

As above, and display coefficients and std. err. with two digits to the right of the decimal

```
scobit y x1 x2, vce(robust) cformat(%8.2f)
```

As above, and also display \( p \)-values with two digits to the right of the decimal

```
scobit y x1 x2, vce(robust) cformat(%8.2f) pformat(%5.2f)
```

**Menu**

Statistics > Binary outcomes > Skewed logistic regression
## Syntax

```
scobit depvar [ indepvars ] [ if ] [ in ] [ weight ] [ , options ]
```

<table>
<thead>
<tr>
<th>options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Model</strong></td>
<td></td>
</tr>
<tr>
<td><code>noconstant</code></td>
<td>suppress constant term</td>
</tr>
<tr>
<td><code>offset(varname)</code></td>
<td>include <code>varname</code> in model with coefficient constrained to 1</td>
</tr>
<tr>
<td><code>asis</code></td>
<td>retain perfect predictor variables</td>
</tr>
<tr>
<td><code>constraints(constraints)</code></td>
<td>apply specified linear constraints</td>
</tr>
<tr>
<td><strong>SE/Robust</strong></td>
<td></td>
</tr>
<tr>
<td><code>vce(vcetype)</code></td>
<td><code>vcetype</code> may be <code>oim</code>, <code>robust</code>, <code>cluster clustvar</code>, <code>opg</code>, <code>bootstrap</code>,</td>
</tr>
<tr>
<td></td>
<td>or <code>jackknife</code></td>
</tr>
<tr>
<td><strong>Reporting</strong></td>
<td></td>
</tr>
<tr>
<td><code>level(#)</code></td>
<td>set confidence level; default is <code>level(95)</code></td>
</tr>
<tr>
<td>or <code>nocnsreport</code></td>
<td>report odds ratios</td>
</tr>
<tr>
<td><code>display_options</code></td>
<td>do not display constraints</td>
</tr>
<tr>
<td><strong>Maximization</strong></td>
<td></td>
</tr>
<tr>
<td><code>maximize_options</code></td>
<td>control the maximization process</td>
</tr>
<tr>
<td><code>collinear</code></td>
<td>keep collinear variables</td>
</tr>
<tr>
<td><code>coeflegend</code></td>
<td>display legend instead of statistics</td>
</tr>
</tbody>
</table>

`indepvars` may contain factor variables; see [U] 11.4.3 Factor variables.

Weights are not allowed with the bootstrap prefix; see [R] bootstrap.

vce() and weights are not allowed with the svy prefix; see [SVY] svy.

fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.

collinear and coeflegend do not appear in the dialog box.

See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

The `Model` options include:

- `noconstant`, `offset(varname)`, `constraints(constraints)`; see [R] Estimation options.

- `asis` forces retention of perfect predictor variables and their associated perfectly predicted observations and may produce instabilities in maximization; see [R] probit.
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

level(#) allows for the estimation of confidence intervals at specified levels; see [R] Estimation options.

display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nolabels, fvwrap(#), fvwrapon(style), cformat(%fmt), pformat(%fmt), sformat(%fmt), and nolstretch; see [R] Estimation options.

maximize_options: difficult, technique(algorithm_spec), iterate(#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(#), ltolerance(#), nrtolerance(#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique(bhhh) resets the default vcetype to vce(opg).

The following options are available with scobit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

Remarks and examples

Remarks are presented under the following headings:

Skewed logistic model
Robust standard errors

Skewed logistic model

scobit fits maximum likelihood models with dichotomous dependent variables coded as 0/1 (or, more precisely, coded as 0 and not 0).

Example 1

We have data on the make, weight, and mileage rating of 22 foreign and 52 domestic automobiles. We wish to fit a model explaining whether a car is foreign based on its mileage. Here is an overview of our data:
. use https://www.stata-press.com/data/r16/auto
(1978 Automobile Data)
. keep make mpg weight foreign
. describe
Contains data from https://www.stata-press.com/data/r16/auto.dta
obs: 74 1978 Automobile Data
vars: 4 13 Apr 2018 17:45
(_dta has notes)

<table>
<thead>
<tr>
<th>variable name</th>
<th>type</th>
<th>format</th>
<th>label</th>
<th>variable label</th>
</tr>
</thead>
<tbody>
<tr>
<td>make</td>
<td>str18</td>
<td>%-18s</td>
<td>Make and Model</td>
<td></td>
</tr>
<tr>
<td>mpg</td>
<td>int</td>
<td>%8.0g</td>
<td>Mileage (mpg)</td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>int</td>
<td>%8.0gc</td>
<td>Weight (lbs.)</td>
<td></td>
</tr>
<tr>
<td>foreign</td>
<td>byte</td>
<td>%8.0g</td>
<td>origin</td>
<td>Car type</td>
</tr>
</tbody>
</table>

Sorted by: foreign
Note: Dataset has changed since last saved.

. inspect foreign
Number of Observations

<table>
<thead>
<tr>
<th>Car type</th>
<th>Total</th>
<th>Integers</th>
<th>Nonintegers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#</td>
<td>Negative</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>#</td>
<td>Zero</td>
<td>52</td>
<td>52</td>
</tr>
<tr>
<td>#</td>
<td>Positive</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td># #</td>
<td>Total</td>
<td>74</td>
<td>74</td>
</tr>
<tr>
<td># #</td>
<td>Missing</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
<td>74</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(2 unique values)
foreign is labeled and all values are documented in the label.

The variable foreign takes on two unique values, 0 and 1. The value 0 denotes a domestic car, and 1 denotes a foreign car.

The model that we wish to fit is

\[
\Pr(\text{foreign} = 1) = F(\beta_0 + \beta_1 \text{mpg})
\]

where \( F(z) = 1 - 1/(1 + \exp(z))^\alpha \).

To fit this model, we type

. scobit foreign mpg

Fitting logistic model:
Iteration 0:  log likelihood = -45.03321
Iteration 1:  log likelihood = -39.380959
Iteration 2:  log likelihood = -39.288802
Iteration 3:  log likelihood = -39.28864
Iteration 4:  log likelihood = -39.28864

Fitting full model:
Iteration 0:  log likelihood = -39.28864
Iteration 1:  log likelihood = -39.286393
Iteration 2:  log likelihood = -39.284415
Iteration 3:  log likelihood = -39.284234
Iteration 4:  log likelihood = -39.284197
Iteration 5:  log likelihood = -39.284196
We find that cars yielding better gas mileage are less likely to be foreign. The likelihood-ratio test at the bottom of the output indicates that the model is not significantly different from a logit model. Therefore, we should use the more parsimonious model.

---

**Technical note**

Stata interprets a value of 0 as a negative outcome (failure) and treats all other values (except missing) as positive outcomes (successes). Thus if the dependent variable takes on the values 0 and 1, then 0 is interpreted as failure and 1 as success. If the dependent variable takes on the values 0, 1, and 2, then 0 is still interpreted as failure, but both 1 and 2 are treated as successes.

Formally, when we type `scobit y x`, Stata fits the model

$$Pr(y_j \neq 0 \mid x_j) = 1 - 1 \Big/ \left\{ 1 + \exp(x_j \beta) \right\}^\alpha$$

---

**Robust standard errors**

If you specify the `vce(robust)` option, `scobit` reports robust standard errors as described in [U 20.22 Obtaining robust variance estimates](https://www.stata.com/manuals/r220u_20.html). For the model of `foreign` on `mpg`, the robust calculation increases the standard error of the coefficient on `mpg` by around 25%

```
. scobit foreign mpg, vce(robust) nolog
```

Skewed logistic regression Number of obs = 74
Zero outcomes = 52
Log pseudolikelihood = -39.2842 Nonzero outcomes = 22

| foreign | Coef.  | Std. Err. | z    | P>|z| | [95% Conf. Interval] |
|---------|--------|-----------|------|------|---------------------|
| mpg     | .1813879 | .2407362  | 0.75 | 0.451 | -0.2904463 - .6532222 |
| _cons   | -4.274883 | 1.399305  | -3.06| 0.002 | -7.017471 -1.532295 |
| /lnalpha| -.4450405 | 3.879885 | -0.11 | 0.909 | -8.049476  7.159395 |
| alpha   | .6407983 | 2.486224 | .0003193 | 1286.133 |

LR test of alpha=1: chi2(1) = 0.01 Prob > chi2 = 0.9249

Note: Likelihood-ratio tests are recommended for inference with scobit models.
Without `vce(robust)`, the standard error for the coefficient on `mpg` was reported to be 0.241, with a resulting confidence interval of $[-0.29, 0.65]$.

Specifying the `vce(cluster clustvar)` option relaxes the independence assumption required by the skewed logit estimator to being just independence between clusters. To demonstrate this, we will switch to a different dataset.

#### Example 2

We are studying the unionization of women in the United States and have a dataset with 26,200 observations on 4,434 women between 1970 and 1988. For our purposes, we will use the variables `age` (the women were 14–26 in 1968 and the data thus span the age range of 16–46), `grade` (years of schooling completed, ranging from 0 to 18), `not_smsa` (28% of the person-time was spent living outside an SMSA—standard metropolitan statistical area), `south` (41% of the person-time was in the South), and `year`. Each of these variables is included in the regression as a covariate along with the interaction between `south` and `year`. This interaction, along with the `south` and `year` variables, is specified in the `scobit` command using factor-variables notation, `south##c.year`. We also have variable `union`. Overall, 22% of the person-time is marked as time under union membership and 44% of these women have belonged to a union.

We fit the following model, ignoring that women are observed an average of 5.9 times each in these data:

```stata
. use https://www.stata-press.com/data/r16/union, clear
(NLS Women 14-24 in 1968)
. scobit union age grade not_smsa south##c.year, nrtol(1e-3)
```

| Coef. Std. Err. | z   | P>|z| | 95% Conf. Interval |
|-----------------|-----|-----|------------------|
| age             | .0185363 | .0043615 | 4.25 | 0.000 | .0099879 | .0270848 |
| grade           | .0452801 | .0057124 | 7.93 | 0.000 | .034084 | .0564761 |
| not_smsa        | -.1886826 | .0317801 | -5.94 | 0.000 | -.2509705 | -.1263947 |
| south           | -1.422372 | .3949301 | -3.60 | 0.000 | -2.196421 | -.6483233 |
| not_smsa##c.year| -.0133016 | .0049575 | -2.68 | 0.007 | -.0230181 | -.0035851 |

The reported standard errors in this model are probably meaningless. Women are observed repeatedly, so the observations are not independent. Looking at the coefficients, we find a large southern effect against unionization and a different time trend for the south. The `vce(cluster clustvar)` option provides a way to fit this model and obtains correct standard errors:
. scobit union age grade not_smsa south#c.year, vce(cluster id) nrtol(1e-3)
(output omitted)

Scobit — Skewed logistic regression

Number of obs = 26,200
Zero outcomes = 20,389
Log pseudolikelihood = -13540.61
Nonzero outcomes = 5,811

(Std. Err. adjusted for 4,434 clusters in idcode)

Robust

|         | Coef.  | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|---------|--------|-----------|-------|------|----------------------|
| union   |        |           |       |      |                      |
| age     | 0.0185363 | 0.0084867  | 2.18  | 0.029| 0.0019027 0.03517   |
| grade   | 0.0452801 | 0.0125765  | 3.60  | 0.000| 0.0206306 0.0699295 |
| not_smsa| -0.1886826 | 0.0642037  | -2.94 | 0.003| -0.3145194 -0.0628457|
| 1.south | -1.422372  | 0.5064933  | -2.81 | 0.005| -2.415081 -0.4296635 |
| year    | -0.0133016 | 0.0090622  | -1.47 | 0.142| -0.0310632 0.0044599|
| south#c.year 1 | 0.0105663 | 0.0063172  | 1.67  | 0.094| -0.0018153 0.0229478|
| _cons   | -10.3557 | 0.9411798  | -11.00| 0.000| -12.20038 -8.511025 |
| /lnalpha| 9.136018  | 0.7426176  | 12.30 | 0.000| 7.680514 10.59152  |
| alpha   | 9283.72  | 6894.254   | 2165.732 | 39795.99 |

Scobit, vce(cluster clustvar) is robust to assumptions about within-cluster correlation. That is, it inefficiently sums within cluster for the standard error calculation rather than attempting to exploit what might be assumed about the within-cluster correlation (as do the xtgee population-averaged models; see [XT] xtgee).

Technical note

The scobit model can be difficult to fit because of the functional form. Often, it requires many iterations, or the optimizer prints out warning and informative messages during the optimization. For example, without the nrtol(1e-3) option, the model using the union dataset will not converge. See [R] Maximize for details about the optimizer.

Technical note

The main reason for using scobit rather than logit is that the effects of the regressors on the probability of success are not constrained to be the largest when the probability is 0.5. Rather, the independent variables might show their largest impact when the probability of success is 0.3 or 0.6. This added flexibility results because the scobit function, unlike the logit function, can be skewed and is not constrained to be mirror symmetric about the 0.5 probability of success.

As Nagler (1994) pointed out, the point of maximum impact is constrained under the scobit model to fall within the interval \((0, 1 − e(−1))\) or approximately \((0, 0.63)\). Achen (2002) notes that if we believe the maximum impact to be outside that range, we can instead fit the “power logit” model by simply reversing the 0s and 1s of our outcome variable and fitting a scobit model on failure, rather than success. We would need to reverse the signs of the coefficients if we wanted to interpret them in terms of impact on success, or we could leave them as they are and interpret them in terms of impact on failure. The important thing to remember is that the scobit model, unlike the logit model, is not invariant to the choice of which result is assigned to success.
**scobit** stores the following in $e()$:

Scalars
- $e(N)$: number of observations
- $e(k)$: number of parameters
- $e(k_eq)$: number of equations in $e(b)$
- $e(k_aux)$: number of auxiliary parameters
- $e(k_dv)$: number of dependent variables
- $e(ll)$: log likelihood
- $e(ll_c)$: log likelihood, comparison model
- $e(N_f)$: number of failures (zero outcomes)
- $e(N_s)$: number of successes (nonzero outcomes)
- $e(alpha)$: alpha
- $e(N_clust)$: number of clusters
- $e(chi2_c)$: $\chi^2$ for comparison test
- $e(rank)$: rank of $e(V)$
- $e(ic)$: number of iterations
- $e(rc)$: return code
- $e(converged)$: 1 if converged, 0 otherwise

Macros
- $e(cmd)$: scobit
- $e(cmdline)$: command as typed
- $e(depvar)$: name of dependent variable
- $e(wtype)$: weight type
- $e(wexp)$: weight expression
- $e(title)$: title in estimation output
- $e(clustvar)$: name of cluster variable
- $e(offset)$: linear offset variable
- $e(chi2_ct)$: Wald or LR; type of model $\chi^2$ test corresponding to $e(chi2_c)$
- $e(vcetype)$: vcetype specified in $vce()$
- $e(vcetype)$: title used to label Std. Err.
- $e(opt)$: type of optimization
- $e(which)$: max or min; whether optimizer is to perform maximization or minimization
- $e(ml_method)$: type of ml method
- $e(user)$: name of likelihood-evaluator program
- $e(technique)$: maximization technique
- $e(properties)$: b V
- $e(predict)$: program used to implement predict
- $e(footnote)$: program used to implement the footnote display
- $e(asbalanced)$: factor variables fvset as asbalanced
- $e(asobserved)$: factor variables fvset as asobserved

Matrices
- $e(b)$: coefficient vector
- $e(Cns)$: constraints matrix
- $e(ilog)$: iteration log (up to 20 iterations)
- $e(gradient)$: gradient vector
- $e(V)$: variance–covariance matrix of the estimators
- $e(V_modelbased)$: model-based variance

Functions
- $e(sample)$: marks estimation sample

In addition to the above, the following is stored in $r()$:

Matrices
- $r(table)$: matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
Skewed logit analysis is an alternative to logit that relaxes the assumption that individuals with initial probability of 0.5 are most sensitive to changes in independent variables.

The log-likelihood function for skewed logit is

\[ \ln L = \sum_{j \in S} w_j \ln F(x_j b) + \sum_{j \notin S} w_j \ln \{1 - F(x_j b)\} \]

where \( S \) is the set of all observations \( j \) such that \( y_j \neq 0 \), \( F(z) = 1 - 1/\{1 + \exp(z)\}^{\alpha} \), and \( w_j \) denotes the optional weights. \( \ln L \) is maximized as described in [R] Maximise.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using \texttt{vce(robust)} and \texttt{vce(cluster clustvar)}, respectively. See [P] \_robust, particularly \textit{Maximum likelihood estimators} and \textit{Methods and formulas}.

\texttt{scobit} also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

References


Also see

[R] \texttt{scobit postestimation} — Postestimation tools for scobit
[R] \texttt{cloglog} — Complementary log-log regression
[R] \texttt{glm} — Generalized linear models
[R] \texttt{logistic} — Logistic regression, reporting odds ratios
[SVY] \texttt{svy estimation} — Estimation commands for survey data
[U] 20 Estimation and postestimation commands