

## NAME

simplex-1991, simplex, simpfit – Nelder-Mead simplex minimization for parameter estimates with standard deviations

## SYNOPSIS

```
xxxxfit [ -s {a | c | l | n} ] [ -d {a | c | l | n} ] [ [ -i ] inputfile ] [ [ -o ] diskfile ]
```

## DESCRIPTION

**Simpfit** minimizes a function by adjusting the variable parameters according to the value returned by a user-coded subroutine. There are no restrictions on the form of the function, except that a value must be computable given the input parameters. Derivatives are not calculated. Singularities can be tolerated. Bounds tests can be incorporated simply.

The Nelder-Mead (1965) algorithm is a procedure for systematically shrinking or otherwise adjusting the shape of a polygon, the simplex. The simplex is constructed in parameter space, with vertices that are arbitrary except they describe a volume that includes the minimum point. Shrinking is carried out so that the function minimum remains within the simplex. Shrinking is continued until the vertices are so close together, satisfying an exit test, that the mean of the vertices defines with sufficient accuracy the parameter values of the function minimum. Sometimes the procedure can locate the minimum even if the starting simplex does not include it. If the function surface is sufficiently rough, the procedure fails through being trapped within a local minimum. The Nelder-Mead method appears to be more robust than gradient methods, but slower to converge when close to the minimum.

Standard deviations of the parameters are calculated by fitting a quadratic to the function surface at the minimum and then using the properties of the quadratic to calculate the variance-covariance matrix. The strategy is given by Nelder and Mead. The quadratic approximation at the current simplex minimum can be used to define a better estimate of the minimum and to construct a new simplex based upon it. Alternation between iterations of simplex shrinking and quadratic approximation can lead to faster approach to the best estimate of the minimum.

The Nelder-Mead algorithm can be used to fit a non-linear model to data. In this case, the function minimized is the least-squares test for quality of fit, the weighted sum of squares of the residuals. A model function explicitly relating a dependent variable to one or more independent variables and variable parameters is fit to a set of data points (observed values of dependent and independent variables). The simplex search locates the least squares minimum, giving estimates of the best-fit values of the parameters of the model function. Standard deviations of the parameters are estimated by quadratic approximation of the least squares surface near the minimum.

## CODING

Several routines have to be recoded for each new model function. Generally only a few lines have to be rewritten, given a working template file (see Section **FILES**, *linefit.c* or *ldhfit.c*). (1) **func()** calculates for a simplex vertex (for the set of parameter values passed to it): (a) the value of the dependent variable for each data point and (b) the least squares function (the weighted sum of residuals squared) over all data points. Bounds on parameters can be implemented easily, by a test that on violation returns an excessively large least squares function value (e.g., 1.E38). Code should be efficient; most of the minimization is spent in **func()**. (2) **fdatprint()** prints calculated and input values for the set of data points, may have to be recoded after change in the function or data. **fdatprint()** can be structured to call other functions, e.g., **fpointprint()**, to print additional customizable data-related output. (3) **fspecial()** controls a customizable and optional display printed after the standard display at the end of each iteration of the simplex minimization.

The minimizer knows nothing about the data, except that it should read data in if told to, call **func()** to give the test value (least-squares function) needed by the fitting procedure, and call **fdatprint()** or **fspecial()** to display results, when appropriate. You are on your own in designing these functions, beyond keeping the interface shown in the examples (templates).

## COMMAND LINE OPTIONS

**-s** {a | c | l | n} -- control output to terminal; default is all possible output ( **a** ):

**a** = ALL possible output; information tracking each iteration of the simplex minimization is

displayed.

**c** = summary output each CYCLE of perhaps 30 iterations; any output from a quadratic minimization is displayed at this point.

**l** = summary output on LAST cycle only;

**n** = NO output;

**-d** {a | c | l | n} -- control output to *diskfile*, with options as for control of terminal output; default is summary output ( **c** ):

[ **-i** ] *inputfile* -- specify input file; default is stdin.

[ **-o** ] *diskfile* -- specify output file; default is /dev/null, i.e., no output is saved unless requested.

## INPUT AND OUTPUT

The input file has the following structure:

one-line title, with no tabs or ctrl characters;

lines following give control variables, the starting simplex, optionally lower and upper bounds on the parameters, and the data array;

the order of entry is fixed by the order in **read\_data()**; see sample data file for structure, and also the description in *input.txt*;

the one-word descriptors must be present in the data file; the data format is otherwise free-form;

comments at the end of the file are not read by the program

Output is meant to be self-documenting. For details, please consult the documentation file *Output.txt* (see following), the source code, and the Nelder and Mead (1965) publication ( *nelder-mead-paper.pdf*; see following).

## FILES

*Simplex-1991/Line/linefit.c*

*Simplex-1991/Ldh/ldhfit.c*

Examples of coding for specific functions: *ldhfit.c* fits enzyme kinetic data for lactate dehydrogenase with an ordered ternary complex mechanism with abortive ternary complexes; *linefit.c* fits a set of manufactured data points with a straight line, for which comparison can be made to hand calculations of the best-fit parameters and standard deviations.

*Simplex-1991/Line/linefit.dat*

*Simplex-1991/Ldh/ldhfit.dat*

Sample input files, with typical values of the control variables.

There are several documentation files:

*Simplex-1991/Doc/Simplex.txt*

Description of the Nelder-Mead algorithm and of the quadratic fit.

*Simplex-1991/Doc/Output.txt*

Description of the full ( **a** option) output for the minimization.

*Simplex-1991/Doc/Input.txt*

Template input file, showing the style of data entry, with comments.

*Simplex-1991/Doc/Data\_redn.txt*

Comments on data reduction and a description of the model coded in *ldhfit.c*, for the lactate dehydrogenase reaction mechanism.

*nelder-mead-paper.pdf*

The classic Nelder-Mead paper describing the simplex algorithm used here for the fitting and the quadratic approximation for the region near the least-squares minimum.

*Simplex-1991/Lib*

Directory - sources for the simplex fit library.

*Simplex-1991/Ldh*

Directory - sources, executable, and test files for **ldhfit**.

*Simplex-1991/Line*

Directory - sources, executable, and test files for **linefit**.

#### **REFERENCE:**

A simplex method for function minimization.

J.A. Nelder and R. Mead (1965). *Computer J.* 7, 308.

#### **AUTHOR**

J. A. Rupley, Tucson, Arizona

rupley!local@megaron.arizona.edu

#### **BUGS**

There are limits, set in *simpdefs.h*, on the number of parameters, data points, and variables for each data point.