AgeStrucNe User Manual

July 29, 2019

Version: 1.0

Authors: Ted Cosart and Brian Hand

Contact: agestrucne@gmail.com

List of Figures

0.1 Adding an interface 26

0.2 Simulation interface, Load/Run section. 27

0.3 Simulation interface, Configuraton info. 28

0.4 Simulation interface, population section 28

0.5 Simulation interface, genome section 31

0.6 Simulation interface, simulation section 33

0.7 Nb/Ne estimation interface, Load/Run section 40

0.8 Nb/Ne estimation interface, genepop files loaded section 42

0.9 Nb/Ne estimations interface, parameters section 43

0.10 Nb/Ne estimations interface, pop subsampling parameters 46

0.11 Nb/Ne estimations interface, loci subsampling parameters 47

0.12 The LDNe Ne/Nb boxplot interface. 50

0.13 The Nb/Ne estimation regression interface with cycle number as the x-axis variable. 52

List of Tables

0.1 Short parameter names as found in configuration files, and their descriptions. Note that a few of our provided configuration files include male and female gamma A and B distribution vectors (noted in this table) which results in a fitness calulation during offpring creation, a feature currently not documented in our manual. 38

0.2 Nb or Ne estimation output values 49

AgeStrucNe

List of Figures **1**

List of Tables **1**

AgeStrucNe **2**
0.1 Introduction 4
0.2 Installation 4
0.2.1 README file 5
0.3 License 12
0.3.1 License file 12
0.4 Starting the program 25
0.5 Loading interfaces 25
0.6 Running a simulation 25
0.6.1 Load an interface 25
0.6.2 Load a configuration file. 26
0.6.3 Adjust simulation parameters 26
0.6.4 Start the simulation 26
0.6.5 Running a simulation from a terminal 26
0.7 Simulation input 27
0.7.1 The Load/Run section 27
0.7.1.1 Processes 27
0.7.1.2 Configuration File 27
0.7.1.3 Output directory 27
0.7.1.4 Output files base name 27
0.7.2 Configuration Info section 27
0.7.2.1 Configuration file name 28
0.7.2.2 Model name 28
0.7.3 The Population section 28
0.7.3.1 N0 (Newborns) 28
0.7.3.2 Nb/Nc 29
0.7.3.3 Nb/Ne 29
0.7.3.4 Nb 29
0.7.3.5 True Nb variance allowed 29
0.7.3.6 Ages 30
0.7.3.7 Female and Male Survival 30
0.7.3.8 Female and Male Fecundity 30
0.7.3.9 Force Skip 30
0.7.3.10 Proportional Litter Sizes 30

0.7.3.11 Monogamous 31
0.7.3.12 Probability of male birth 31
0.7.3.13 Population size 31
0.7.4 The Genome section 31
0.7.4.1 Use Loci File 31
0.7.4.2 Mutation frequency 32
0.7.4.3 Number of chromosomes 32
0.7.4.4 Number of Microsatellites 32
0.7.4.5 Number of SNPs 33
0.7.4.6 Starting Msat allele total 33
0.7.5 The Simulation section 33
0.7.5.1 Cull method 33
0.7.5.2 Filter recorded pops by heterozygosity 34
0.7.5.3 Het filter parameters 34
0.7.5.4 SNP het initialization 35
0.7.5.5 Msat het initialization 35
0.7.5.6 Nb and census adjustment 35
0.7.5.7 Replicates 36
0.7.5.8 Skip breeding probability 36
0.7.5.9 Cycles of burn-in 36
0.7.5.10 Start recording at cycle 36
0.7.5.11 Stop recording at cycle 36
0.7.5.12 Tolerance tries 36
0.8 Manually editing configuration files 37
0.9 Simulation output 37
0.10 Running an Nb or Ne Estimation 39
0.10.1 Running an Nb or Ne estimation from the command line 40
0.11 Nb/Ne Estimations input 41
0.11.1 The Load/Run section 41
0.11.1.1 Total processes 41
0.11.1.2 Load genepop files button 41
0.11.1.3 Select output directory 41
0.11.1.4 Output files base name 41
0.11.2 The Genepop Files Loaded section 41
0.11.3 The Parameters section 41
0.11.3.1 Use Chrom Loci File 42
0.11.3.2 Minimum allele frequency 42
0.11.3.3 Monagamy 42
0.11.3.4 Nb bias adjustment check box 42
0.11.3.5 Nb/Ne ratio 42
0.11.3.6 Pop sampling replicates 44
0.11.3.7 Loci sampling replicates. 44
0.11.3.8 Pop sampling scheme 44
0.11.3.9 Loci sampling scheme 45
0.11.4 Pop sampling parameters section 46
0.11.4.1 Pop number start 46
0.11.4.2 Pop number end 46
0.11.4.3 Indiv min per pop 46
0.11.4.4 Indiv max per pop 47
0.11.4.5 Scheme-specific parameters 47

0.11.5 Loci sampling parameters section 47
0.11.5.1 Loci number start 47
0.11.5.2 Loci number end 47
0.11.5.3 Min Loci count 47
0.11.5.4 Max Loci count 48
0.11.5.5 Scheme specific parameters 48
0.12 Nb estimation output 48
0.12.1 Messages file 48
0.12.2 Estimates table 48
0.13 Visualization Interfaces 48
0.13.1 Boxplot Interface 48
0.13.1.1 Grouping. 48
0.13.1.2 Filtering. 50
0.13.1.3 Y-axis field. 51
0.13.1.4 Y-axis value limits. 51
0.13.1.5 Axis and tick label font size adjustment 51
0.13.1.6 The plot. 51
0.13.2 Regression Interface 51
0.13.2.1 Filtering. 51
0.13.2.2 Y-axis field 52
0.13.2.3 X axis variable 52
0.13.2.4 Min cycle and Max cycle text boxes 53
0.13.2.5 Axis and tick label font size adjustment 53
0.13.2.6 The plot 53
0.13.2.7 The regression stats text box 53

Bibliography **55**

0.1 Introduction

The AgeStrucNe GUI interface offers a user interface to allow easy access to simuPOP-based simulations [1] and the LDNe-based Nb and Ne estimations, using version 2 of the LDNe program [3]. The program integrates this functionality as implemented in Tiago Antao's python program, AgeStrucureNe, available at `https://github.com/tiagoantao/AgeStructureNe.git`.

We also offer an interface for plotting Nb and Ne estimations, and regressions based on the estimations. Our program offers a separate interface for each of three functions: population simulation, Nb and Ne estimation, and estimate visualization. The genepop file output from a simulation can be loaded into an Nb estimation interface, and in turn, the output from an Nb estimation can be loaded into a visualization interface. The Nb estimation interface can also use any genepop file for input.

With program questions, please contact us through the program's email account at agestrucne@gmail.com.

0.2 Installation

Details about our program's dependancies and installation are in our github project's README.md file, inserted here for convenience, and also available

at https://github.com/popgengui/agestrucne/blob/data/README.md:

0.2.1 README file

```
I. overview
-----------

    For more details about the program, see the manual.pdf file distributed
    with the program. Briefly, our program is a GUI front end that incorporates the
    simulation and LDNe based population genetics functions provided by Tiago
    Antao's python program at https://github.com/tiagoantao/agestructurene.git,
    enhanced by multiple methods for population and loci subsampling when performing
    LD-based Ne and Nb estimations from genepop file inputs. It also provides
    plotting facilities to show estimate distributions and regression lines.

    The program uses multi-processing to allow an arbitrary number of
    simultaneous simulations, LD-based Nb and Ne estimations, and plotting
    interfaces.  Further, within simulations, it can run simulation replicates
    simultaneously, and, within Ne or Nb estimation sessions, it allows
    simultaneously running genepop-file population sections.

    The core functionality for simulation is provided by the python simuPOP
    package (Peng, B. & Kimmel, M. simuPOP: a forward-time population genetics
    simulation environment. Bioinformatics 21, 36863687, 2005). The LDNe
    estimation by version 2 of the LDNe program (Waples, R. S. & Do, C.  LDNe: a
    program for estimating effective population size from data on linkage
    disequilibrium. Molecular Ecology Resources 8, 753756, 2008).

    Please direct questions/issues to our program's email account, agestrucne@gmail.com

II. current compatible python versions
--------------------------------------

    1.  python, 64-bit, version 3.5 or greater, on 64-bit operating systems,
        including Windows, Mas OS (OS X) and linux distributions.  If you don't
        already have python 3 installed on your computer, we strongly recommend
        using the Anaconda 3 python distribution (see
        https://www.anaconda.com/distribution/),
        because it's "conda" installer supplies a pre-compiled version of SimuPop,
        which can be difficult to install through pip and setuptools, especially
        on the Windows OS.


III. quickest installation procedures
-------------------------------------

    1. Find out if python 3 (version 3.5 or greater) is installed on your computer

         i. From a Mac OS (OSX) or Linux terminal
```

a. type this command:

python3 --version

b. if the command is not recognized, type:

python --version

c. if the version is 3.4 or more, you have a version compatible with agestrucne

ii. In Windows, a python distribution is not always accessible from a command prompt. If you're not sure whether you have python3 installed you'll need to search your application list (e.g. using the windows key).

2. If python 3 (version 3.5 or greater) is not installed on your computer, we strongly recommend installing the Anaconda3 python3 distribution, in order to access the pre-compiled simuPOP package.

i. Download and install the current 64-bit Anaconda3 distro for your OS from the Anaconda download page:

https://www.anaconda.com/distribution/

ii. Open the Anaconda Prompt window and type these commands in this order:

```
conda config --add channels conda-forge
conda install simupop
pip install agestrucne
```

3. If python 3 (3.5 or greater) is already on your computer

i. If your python 3 distribution is Anaconda3, then follow (2) above, step (ii). Otherwise, type this command (here we assume that your non-Anaconda installation has supplied an alias for pip, "pip3", to distinguish it from the pip installed for a python2 distribution):

pip3 install agestrucne

a. Note for Linux users: to install into your default python location you may need to issue the command as a super user:

sudo pip3 install agestrucne

and supply the sudo password.

ii. If the install fails with a message indicating

```
            that simuPOP could not be installed,
            this likely means your computer needs some compiler-related
            packages.  Please see the simuPOP installation information
            for installing simuPOP using pip.  See item (6), at
            http://simupop.sourceforge.net/Main/Download

         iii. Note that installing the program using a non-Anaconda3
              python distribution requires that simupop is compiled,
              which can take a long time, on slower systems more than
              an hour. Further, simuPOP compile errors may not appear
              until late in the compilation steps.

    4. If you have installation problems, see the section following,
       which list a few that we encountered.

IV. Notes about known installation problems
-------------------------------------------
    1. Mac OS (OS X)

        a. Using the Homebrew version of python3, the GUI backend
           has a bug that crashes the program when the mouse scroll
           wheel is used.  Our successful Mac installs were under
           the Anaconda3 distro and the distribution available at
           https://python.org.  Under the latter the simuPOP
           package took a long time compiling but was successful.

        b. Using the Mac OS installation of python3 available at
           python.org, the pip install agestrucne fails, unless
           the SSL/TLS cerificate validation command is executed
           after the installation.  Please see the installation
           messages and the notes for Mac users at
           https://www.python.org/download

    2. Windows

       a. It can be very difficult to install simuPOP under Windows
          with a non-Anaconda distrubution, because only
          Anaconda's conda repository has a pre-compiled
          package.  For details please see the documentation at
          http://simupop.sourceforge.net/Main/Download

       b. Under the Anaconda3 distribution, the commandline
          scripts pgdriveneestimation.py and
          pgdrivesimulation.py are not treated as executables.
          To use them you must run them as arguments to the python
          executable, using the full path to their location inside
          the Anaconda3 installation.  For example, if your installation
          path is, C:\Users\myname\Anaconda3, then, you can execute
          one of these command line scripts in an Anaconda
          promt window with this command:
```

```
           python C:\Users\myname\Anaconda3\Scripts <scriptname> <myarguments>

V. getting the manual and the example configuration files
----------------------------------------------------------

    1. If you have the "git" program, you can open a terminal,
       "cd" into a suitable directory, and clone the data branch with,
       the command:

           git clone -b data https://github.com/popgengui/agestrucne

    2. Instead of using git, you can download a zip archive of the
       files at https://github.com/popgengui/agestrucne/tree/data.
       You will see a green button on the right side of
       the screen, offering a download.

    3. The files provided include the program manual, this README.md
       file, and simulation configuration files that will get you
       started in the program pipeline (see the manual for details on how
       to load and edit the configuration files).

VI. notes on OS distributions
-----------------------------
    1. Linux. The program has been run on Linux (ubuntu 16.04).

    2. Mac OS (OS X). The program has been run successfully on Mac OS (OS X),
       v10.13 (High Sierra).

    3. Windows 10 and 8.1 (64-bit).  Note that on Windows, a persistent problem
       we have not solved is the inability of the program to remove files for
       some cleanup operations when processes do not finish (through error or
       user-cancellation). Thus you may have to remove output and temporary files
       manually when a run is cancelled or fails.

VII. dependencies
-----------------
    The following python packages, except pip and setuptools (which are provided by
    almost all python distribution),  are automatically installed when you use
    the installation methods outlined in section III, as well as the
    setup.py method (section VIII, below).

    1. pip and setuptools, the python package installation modules, included
       in nearly all recent python distributions.  If your
       distribution does not include them, please see

       https://packaging.python.org/tutorials/installing-packages/.

    2.  SimuPOP, a python package, hosted at
        http://SimuPop.sourceforge.net. Our python package scripts
```

```
        are designed to install simuPOP automatically.  However,
        If your python 3 distribution is not Anaconda3, the installer
        will will have to compile the simuPOP package, which can take
        a long time, and, in some cases can require extra steps.
        See our installation instructions and notes in sections III
        and IV. For troubleshooting failed simuPOP package compilation,
        the best source is the information at

             http://simupop.sourceforge.net/Main/Download

    3. Below are the other required python packages, which should be
       automatically installed when you use the pip installer, or the
       "setup.py install" command (see section III and VIII).
       These can also be installed one at a time through pip with the
       command "pip install <package>", or, if you use
       the Anaconda3 distribution of python, "conda install <package>".

        i.    numpy

        ii.   matplotlib

        iii.  scipy

        iv.   future

        v.    psutil

        vi.   natsort

VIII. alternative installation methods
--------------------------------------

    1. Alternatives to the quickest installation methods (see above, section III).

        A. Installing without the "pip" installer:

            ii. Clone (using the git program) or download the zip archive from
                our master repository at, https://github.com/popgengui/agestrucne.

            iii. Use the setup.py file.

                 a. If under Widows, with an Anaconda3 python distrubution, open
                    an Anaconda Prompt window, otherwise open a terminal.
                 b. use the "cd" command to move to the "agestrucne" directory
                    containing the unzipped files, in particular looking for the
                    "setup.py" file.  Type the following command, leading with "sudo"

                 c. If your python3 is an Anaconda3 distribution get the pre-compiled
                    simupop version with these commands:
                         conda config --add channels conda-forge
```

```
            conda install simupop

      d. Install the program and dependencies using
         the setup.py:

             python setup.py install

         Note for Linux users, you may have to precede this
         command with "sudo" and then provide your password.

B. Using our program without installing it inside the python package directory.

   i.  Use pip to install the dependancy packages (see the dependancies
       section above)

   ii. Download the master branch of our github repository.

       a. clone command:
          git clone -b master https://github.com/popgengui/agestrucne

       b. Or get the zip archive at https://github.com/popgengui/agestrucne.

   iii. Run the program directly using a terminal from the downloaded directories
        using the applicable one of these methods:

       a. In Windows, with an Anaconda distribution of python, open an
          Anaconda prompt and use cd, to move into the outermost "agestrucne"
          folder in the github repository you downloaded.
          Then, type these commands:

           set PYTHONPATH=%PYTHONPATH%;%cd%
           set NEPATH=%cd%\agestrucne

          Now, you can cd into any folder you'd like and execute the
          program with:

           python %NEPATH%\negui.py

          You can also invoke the console-based
          modules, pgdrivesimulation.py and pgdriveneestimation.py using

           python %NEPATH%\<module name>

       b. In Windows with a non-Anaconda python distribution, use a DOS command
          prompt and use the same procedure as above in (a).  However, just
          typing "python" may fail if your distrubution does not add python
          to your console's environmental variables.  If so,
          when you issue a python command, you may need to fully type the
          path to your python executable, where ever your disribution places
          it during the installation.
```

```
        c. In linux or OSX, you can open a terminal and cd into the outermost
           "agestrucne" directory of the programs github directories.
           Then you can type the following command:

            export PYTHONPATH=$PYTHONPATH:$(pwd); NEPATH=$(pwd)/agestrucne

            This terminal will then be able to execute the program interface
            or the console-based modules from any directory by typing

            python $NEPATH/<module name>

            where <module name> is negui.py, pgdrivesimulation.py, or
            pgdriveneestimation.py

        d. Using this method PYTHONPATH and NEPATH variables will need to be
            set every time you open a new console to run the program. If you
            are conversant with setting environmental variables for your user
            environment, you can add the agestrucne path to the PYTHONPATH
            path, and NEPATH to your
            environment so that they are available automatically when you open
            a console windows.

IX. starting the program
---------------------------

    1. If you installed the Anaconda3 python distribution,
       then, at an Anaconda prompt, start the program with the command,
       "agestrucne."

        A. Note that when you open the program, the current directory of your
           terminal will determine where the file-loading dialog will be initially
           set, as you locate, for example, a configuration file to load into
           the simulation interface.

    2. In addition to "agestrucne,"  there are two python scripts that offer
       ways to run simulations and LDNe estimations from a terminal command prompt.

        A. Command "pgdrivesimulation.py" performs simulations from the
           terminal, as specified in the user manual.

        B. Command "pgdriveneestimator.py" performs LDNe estimations
           from the terminal, as specified in the user manual.

X. using the program
--------------------

    To run a simulation, calculate Nb or Ne estimates, or plot results, load one
    of the three interfaces by clicking "New" on the main menu.
```

```
For details about running the different interfaces, see the
user manual.
```

0.3 License

Our license information follows, and is also available at `https://github.com/popgengui/agestrucne/blob/data/LICENSE`.

0.3.1 License file


```
GNU AFFERO GENERAL PUBLIC LICENSE
                       Version 3, 19 November 2007

 Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

                            Preamble

  The GNU Affero General Public License is a free, copyleft license for
software and other kinds of works, specifically designed to ensure
cooperation with the community in the case of network server software.

  The licenses for most software and other practical works are designed
to take away your freedom to share and change the works.  By contrast,
our General Public Licenses are intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users.

  When we speak of free software, we are referring to freedom, not
price.  Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

  Developers that use our General Public Licenses protect your rights
with two steps: (1) assert copyright on the software, and (2) offer
you this License which gives you legal permission to copy, distribute
and/or modify the software.

  A secondary benefit of defending all users' freedom is that
improvements made in alternate versions of the program, if they
receive widespread use, become available for other developers to
incorporate.  Many developers of free software are heartened and
encouraged by the resulting cooperation.  However, in the case of
software used on network servers, this result may fail to come about.
The GNU General Public License permits making a modified version and
letting the public access it on a server without ever releasing its
```

source code to the public.

The GNU Affero General Public License is designed specifically to ensure that, in such cases, the modified source code becomes available to the community. It requires the operator of a network server to provide the source code of the modified version running there to the users of that server. Therefore, public use of a modified version, on a publicly accessible server, gives the public access to the source code of the modified version.

An older license, called the Affero General Public License and published by Affero, was designed to accomplish similar goals. This is a different license, not a version of the Affero GPL, but Affero has released a new version of the Affero GPL which permits relicensing under this license.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU Affero General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that

same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code;

keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular

product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by

this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a

publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Remote Network Interaction; Use with the GNU General Public License.

Notwithstanding any other provision of this License, if you modify the Program, your modified version must prominently offer all users interacting with it remotely through a computer network (if your version supports such interaction) an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge, through some standard or customary means of facilitating copying of software. This Corresponding Source shall include the Corresponding Source for any work covered by version 3 of the GNU General Public License that is incorporated pursuant to the following paragraph.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the work with which it is combined will remain governed by version 3 of the GNU General Public License.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU Affero General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU Affero General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU Affero General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU Affero General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY

APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

> <one line to give the program's name and a brief idea of what it does.>
> Copyright (C) <year> <name of author>
>
> This program is free software: you can redistribute it and/or modify it under the terms of the GNU Affero General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

```
    This program is distributed in the hope that it will be useful,
    but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU Affero General Public License for more details.

    You should have received a copy of the GNU Affero General Public License
    along with this program.  If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

  If your software can interact with users remotely through a computer
network, you should also make sure that it provides a way for users to
get its source.  For example, if your program is a web application, its
interface could display a "Source" link that leads users to an archive
of the code.  There are many ways you could offer source, and different
solutions will be better for different programs; see section 13 for the
specific requirements.

  You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU AGPL, see
<http://www.gnu.org/licenses/>.
```

.

0.4 Starting the program

The program is launched at a terminal using a python 2.7, 3.5, or 3.6 executable, invoking the "agestrucne" executable. Please see the README.md file for instructions on installing the program.

Once you have downloaded the "data" branch of our program's github repository, you can, for example, open a terminal, change your current directory to the agestrucne folder supplied by the download, and load a configuration file into the Simulation interface (Section 0.5).

0.5 Loading interfaces

To load one or more of the three interfaces for performing simulations, Nb/Ne estimations, or plotting programs, from the main menu click on the New menu (Figure 0.1). You can load any number of interfaces and run them simultaneously, though caution is warranted, since a computer can be taxed to a standstill by the over-allocation of CPUs and/or memory.

0.6 Running a simulation

0.6.1 Load an interface

Use the add menu (Figure0.1) to load a new simulation interface. Steps for preparing the interface to run a simulation follow.

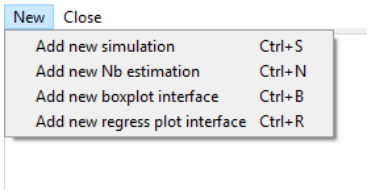


Figure 0.1: Adding an interface

0.6.2 Load a configuration file.

The initial simulation interface requires the user to load a configuration file (Figure 0.2). If you have not already downloaded our collection of configuration files, you can get them at our programs github repository, downloading our data branch at `https://github.com/popgengui/agestrucne/tree/data`. The files are inside the "configuration_files" subdirectory, inside the main program directory, "ageststrucne." You can also load your own configuration (see our provided configuration files for a formatted example and also see the section Manually configuration files). Note that you can also open these files and change the parameters manually, if you prefer it to setting them in the interface

0.6.3 Adjust simulation parameters

With a configuration file loaded (Section 0.6.2) you can change the values in the editable controls. These are detailed below in the Simulation Input section.

0.6.4 Start the simulation

Clicking "run simulation," (Figure 0.2) starts the simulation with the loaded parameters. The buttons text changes to say cancel simulation, and next to it a new label notes that a simulation is in progress. While the simulation is in progress, the parameter controls are disabled.

0.6.5 Running a simulation from a terminal

The module pgdrivesimulation.py provides a command line interface to run a simulation. At a Linux or DOS terminal, you can invoke the command with the form,

```
<python> <pgdrivesimulation.py> <options>
```

Where ⟨python⟩ is either the python3 or python2.7 executable, ⟨pgdrivesimulation.py⟩ is the module name, which may also include the full path the to the module (see the program installation instructions in our README.md file, distributed in our github repostory's main directory), and ⟨options⟩ is a list of parameters specified using the option flags. You can see the list of option flags by invoking the command with out any arguments. To see the details for each argument, execute:

```
<python> <pgdrivesimulation.py>  -h
```

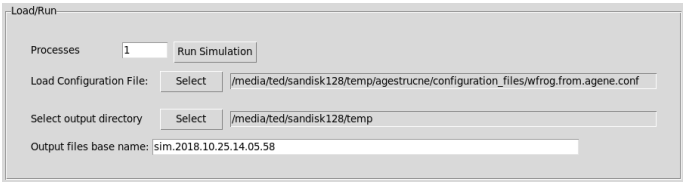


Figure 0.2: Simulation interface, Load/Run section.

Note that there are both required and optional arguments. The latter are only a select number of those offered in the GUI interface (Section 0.7).

0.7 Simulation input

The simulation interface is divided into controls inside sub-frames, based on category.

0.7.1 The Load/Run section

This section offers parameters related to input and output files (Figure 0.2).

0.7.1.1 Processes

Valid values are between 1 and the total number of available (logical) cores in your computer. Multiple processes are only useful if you have set the Replicates parameter (see the Simulation subframe details below) to a value greater than one.

0.7.1.2 Configuration File

Press the "Select" button next to the label, "Load Configuration File" to load a configuration file into the interface. We have included configuration files for many species. These can be found in the "configuration_files" subdirectory inside the main program folder.

0.7.1.3 Output directory

Press the select button next to the label, Select output directory, select a folder for the output files written by the simulation

0.7.1.4 Output files base name

You can type in a base name for the simulation output files. The simulation will prepend this to the *.genepop, *.conf, *_age_totals.tsv and *_nb_values.tsv output files (Section 0.9).

0.7.2 Configuration Info section

This parameter group simply shows you the input file information and has no settable parameters (Figure 0.3).

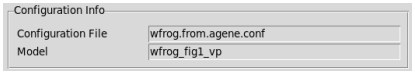


Figure 0.3: Simulation interface, Configuraton info.

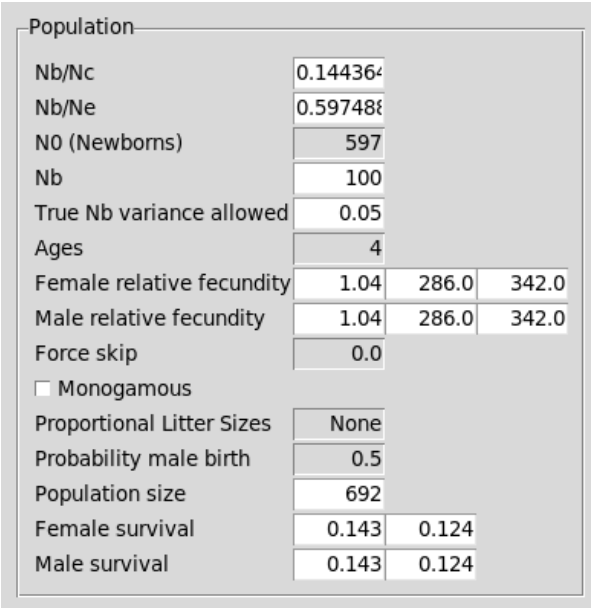


Figure 0.4: Simulation interface, population section

0.7.2.1 Configuration file name

This gives the file name of the loaded configuration file.

0.7.2.2 Model name

This gives the name of the model parameterized by the configuration file. In our example configuration files, the model name is usually a species common name. This parameter was used when the program depended on separate life table and configuration files. Currently the configuration files contain all of the life table information needed for the simulations, so that the model name can be any string you find suitable, to indicate, for example, that a given configuration file has parameters similar to another with the same name.

0.7.3 The Population section

offers many parameter settings that characterize the populations size and fecundity (Figure 0.4).

0.7.3.1 N0 (Newborns)

. This gives the number of newborns added at each simulated reproductive cycle. This value is not editable directly, but is calculated using several values,

all of which are editable. These including Nb, Nb/Nc , Female, Male Survival, and the probability of male birth. The N0 is recalculated whenever any of these values changes, using the following procedure:

1. Assign an Nc value, as Nb divided by Nb/Nc.
2. Assign a current_male_proportion equal to the Probability of male birth.
3. Assign a current_female_proportion equal to 1 the Probability of male birth.
4. Assign a cumulative_proportion=1.
5. For each age value age_val giving a male and female survival rate:
   a Update, current_male_proportion=current_male_proportion x male_survival at age_val.
   b Update, current_female=current_female x female_survival at age_val.
   c Update cumulative_proportion =cumulative_proportion + current_male_proportion.
   d Update cumulative_proportion=cumulative_proportion + current_female_proportion.
6. Set N0=Nc/cumulative_proportion, rounding it to the nearest integer.

0.7.3.2 Nb/Nc

This is the effective number of breeders in one reproductive cycle divided by the census size.

0.7.3.3 Nb/Ne

This is the ratio of the effective number of breeders in one reproductive cycle to the effective population size per generation. This value is not used in the simulation itself, but is written to the output genepop file, and can be used in the Nb estimation interface to make a bias correction in the LDNe estimation of Nb (see section 0.11.3.5).

0.7.3.4 Nb

This is the target effective number of breeders in the simulated population.

0.7.3.5 True Nb variance allowed

This determines the threshold for allowable "true" Nb values (see section 0.7.3.4) for simulated populations calculated using the parentage analysis without parents (PWoP) procedure ([4]). For example, if the Nb is set at 100, and the variance allowed is set at 0.01, allowable Nb values would be in the range from 199-101 for simulated populations. For any n^{th} reproductive cycle after burn-in the program re-simulates the population from the $n-1^{th}$ until the simulated population has a PWoP-calculated Nb within the tolerance.

0.7.3.6 Ages

This value gives the number of age classes for the population to be simulated. Note that this is disabled, and that the length of the lists for Female, Male Fecundity and Female, Male Survival values (see below) are set to length Ages minus one for the former and Ages minus two for the latter. The age value and changes in these lists, therefore, need to be edited in a configuration file (Section 0.8).

0.7.3.7 Female and Male Survival

These are lists whose i^{th} value gives the probability of survival for an individual of the i^{th} age category.

0.7.3.8 Female and Male Fecundity

These are lists whose i^{th} item gives the probability of reproducing for individuals of the i^{th} age category.

0.7.3.9 Force Skip

This value gives a probability, for each non-zero value, f_a, in the female fecundity list, that during a given reproductive cycle r the value will be replaced with zero. Such replacement means that females belonging to the age class a, given by f_a, for cycle r, are infertile. This parameter is set (assigned a non-zero value) in only a few of the configuration files we copied from the AgeStructureNe program, and we have not made it editable in our interface. In your own custom configuration files you can set it to any value 0 through 100 (the value shown in the interface will be the files value divided by 100).

0.7.3.10 Proportional Litter Sizes

If not a "None" value, it will be a list of integers, affecting litter sizes. Note that we do not allow interface editing of these parameters, but note that, as above for the Force Skip setting, you can inter this parameter value in a configuration file. This should be a list, and can have one of 2 valid configurations:

a The list can have a single value l, and $l < 0$, then at each reproductive cycle the maximum possible number of offspring available to each reproducing female is given by $l * -1$.

b Otherwise, the list should have (positive) integers. In this case these integers proportionally allot litter sizes, as given by their indices in the list. In particular, at each reproductive cycle, as a female is chosen to mate:

   i An age, a, is chosen randomly.

   ii A female f_a is chosen randomly from the females of age a.

   iii A list index i (i.e. one of 1,2,3, ... n , where n is the number of items in the litter list), is selected by weighted probability, proportionally according to the ratio of each list value to the sum of the list values.

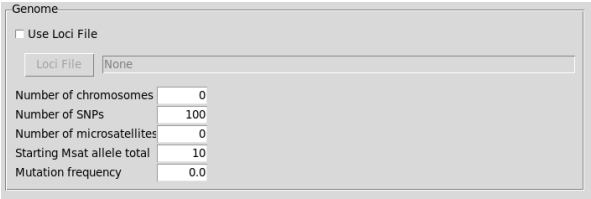


Figure 0.5: Simulation interface, genome section

iv Female f_a is then the mother of the next i offspring (i.e. the female selection steps are skipped for the next i pairings, since f_a is the female of the pair). Thus, she will parent the next i offspring, unless the j^{th} of her offspring assignments produces the maximum total offspring for the cycle (i.e. N0 is reached), and $j < i$.

0.7.3.11 Monogamous

When this box is checked, monogamy is enforced.

0.7.3.12 Probability of male birth

This value is used during reproductive cycles to determine the sex of new individuals. As noted above in the description of the N0 (see 0.7.3.1), it is used also in the N0 calculation, and so the latter is recalculated when this value is changed. When the Cull method (0.7.5.1) is set to equal_sex_ratio, this parameter is automatically set to 0.5, and its entry box is disabled.

0.7.3.13 Population size

This value determines the number of individuals that will be created in the simulations initial population. In subsequent cycles, the size will change according to the reproductive parameters, notably N0 (0.7.3.1).

0.7.4 The Genome section

Parameters in the Genome section determine the simulated individuals allelic content (Figure 0.5).

0.7.4.1 Use Loci File

Check this box and click on Loci File to locate a file with or comma-delimited fields, without spaces, that include (in order), loci_name, chromosome_name, and position.

You can also submit a 4-columnn file that adds a for each entry a 4th column that gives an allele frequency f, with $0.0 \leq f \leq 1.0$ such that the biallelic SNP loci will be initialized with frequencies f and $f - 1$

Note that the program expects a floating point or integer value for the position, and that each loci name occurs only once in the file. If multiple loci share the same position on the same chromosome, then, so that all will

be included in the simulation, the position of the i^{th} loci with the identical position p will be, $p + (i-1)(1^{-13})$.

With the entries in this file, the simulation's genome is initialized accordingly each loci a biallelic SNP postioned at the given value on the given chromosome, and, if fourth column values f are added, at frequencies f and $f-1$.

When the Use Loci file box is checked and a file is specified, the program ignores the values for the other parameters that specify the number of chromosomes (Section 0.7.4.3), number of snps (Section 0.7.4.5), number of microsats (Section 0.7.4.4), and starting msat allele totals (Section 0.7.4.6). Also, the simulation will generate a chromosome loci file (Section 0.9) that can be used in an LDNe2 run to limit the alleles used in the estimation to those from loci pairs, l_i, l_j, on chromosomes c_k, c_l, such that $c_k \neq c_l$ (Section 0.11.3.1).

0.7.4.2 Mutation frequency

If non-zero, this value is applied to microsatellites (not to SNPs). It will be used to set the simuPOP simulation StepwiseMutaters rate parameter.

0.7.4.3 Number of chromosomes

When set to zero, the simulation assigns each loci to a separate chromosome. When set to a positive and non-zero integer M, the simulation such that the N loci (SNPs plus microsats) are assigned evenly over the M chromosomes. Compliance with simuPOP's loci-adjacency implemenation means that the first N/M loci will be assigned to chromsome 0, the second N/M to chromosome 1, and so on.

When the number of chromosomes don't evenly divide the loci, the remainder, $r = N \ mod \ M$, is distrubuted such that $(N/M) + 1$ loci are assigned to chromosomes $0, 1, 2 \ldots r-1$.

Whenever this paramter's value is non-zero the simulation will generate a chromosome loci file (Section 0.9) that can be used in an LDNe2 run to limit the alleles used in the estmation to those of loci pairs, l_i, l_j, on chromosomes c_k, c_l, such that $c_k \neq c_l$ (Section 0.11.3.1).

While simuPOP accepts any numeric values for loci positions, when this parameter is non-zero, our program will assign, on each chromosome, loci positions spaced evenly using 100 as the total units on the chromorome, in order to emulate 100-centimorgan chromosomes.

Note that while you can run the simulation using both m microsats and s SNPs, as noted elsewhere (Section 0.7.4.4), the microsats will always be the first m, and SNPs the last s loci. If you subsequently want to evaluate LDNe for microsats only (or SNPs only), you may unintentionally impose high physical linkage in your loci sample for LDNe. For example, if you use few chromosomes and few microsats in a simulation, and then run LDNe2, limiting the loci to only the microsats (Section 0.11.5.1), while also using the chromosome-loci file output by the simulation (Section 0.11.3.1), you may be severly limiting the loci pairs available to calculate LDNe.

0.7.4.4 Number of Microsatellites

Microsatellites are simulated under diploidy. Note that in cases in which you specify both m microsatellites and s SNPs, in the output genepop file, the first

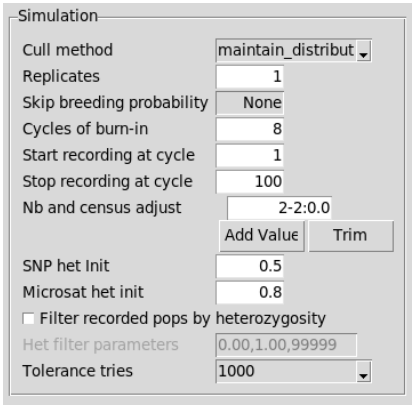


Figure 0.6: Simulation interface, simulation section

m loci are the microsatellites and the last s loci are the SNPs.

0.7.4.5 Number of SNPs

SNPs are simulated as bi-allelic, under diploidy. Frequencies are initialized, targeting the value for initital heterozygosity and using a truncated normal distribution. See Section 0.7.5.4 for details.

0.7.4.6 Starting Msat allele total

This value gives the initial number of microsatellite alleles a for each microsatellite in the initial population. For each microsatellite, the initial genotype frequencies are drawn from the Dirichlet distribution, using a heuristic that draws a distribution of n allele frequencies by trial, by adjusting the Dirichlet shape parameter until the expected heterozygosity is within a tolerance, currently set to 0.001, of the value for the parameter "Microsat het init" (Section 0.7.5.5).

0.7.5 The Simulation section

These parameters determine several per-cycle behaviours (Figure 0.6).

0.7.5.1 Cull method

Cull method indicates one of two possible per-cycle methods of removing individual from the population.

1. Survival rates. Individuals of age greater than zero are removed from the population by comparing a random number against the probability for that individuals age and sex, in the survival list (Section 0.7.3.7).

2. Maintain distribution. For each cohort with age $a, a > 0$, (i.e. excepting newborns), divide the individuals by sex, s, into two lists. From each list with t_s individuals, randomly cull n_s individuals from each list where

$$n_s = floor(p_s), \text{ where } p_s = (t_s(1 - survival_s[a]))$$

and $survival_s[a]$ is the survival rate (Section 0.7.3.7) for individuals of age a and sex s. One more individual from each is culled by the probability given by the fractional part of p_s.

0.7.5.2 Filter recorded pops by heterozygosity

When checked, the genepop file output will be restricted to the populations as filtered using the Heterozygosity filter parameters. See section 0.7.5.3.

0.7.5.3 Het filter parameters

If the Het filter checkbox is checked, apply a filter to each pop of the form m, x, t, where,

- Mean expected heterozygosity is calculated as

  $$mean(H_{L_1}, H_{L_2}, H_{L_3} \ldots H_{L_N})$$

  for N loci, and

  $$H_{L_i} = 1 - \sum_{j=1}^{n} freq(a_j)^2$$

  and $freq(a_j)$ is the frequency of the j^{th} of the n alleles of loci L_i.

- m is the minimum mean expected heterozygosity,

- x is the maximum mean het, and

- t is the total number of populations to record.

The output genepop file will then record only populations whose mean Het falls inside the range. Further, it will stop the simulation as soon as one of the following is met:

- t populations are recorded.

- The current population's mean Het is less than m and at least one of the following is true:

  - The simulation includes no microsatellites.
  - The mutation frequency is zero (Section 0.7.4.2).

- The last cycle (Section 0.7.5.11) has completed.

Note that if you set the filter to accept a narrow band of values, for example, a setting of 0.299,0.301,1 the simulation may never realize a population within the filter range. In such cases you should can set the replicate parameter (Section 0.7.5.7) to, say, 5 to 10, in order to ensure that at least one of your runs produces a population (as written to the output genepop file).

0.7.5.4 SNP het initialization

The value here determines a mean expected heterozygosity (H_e) for the N SNPs in the initial population as totaled by either the Number of SNPs parameter (Section 0.7.4.5) or as input using a loci information file (Section 0.7.4.1) using the calculation as given in Section 0.7.5.3). The SNP frequencies are drawn from a truncated normal distribution (using the python scipy.stats.truncnorm function, limiting sampling values to the interval [0.0,0.5]) whose mean is given by solving for f, in $H_e/2 = -f^2 + f$, and whose standard deviation is derived heuristically, so that the mean expected heterozygosity calculated using the frequencies is within a tolerance value t of the value set for initial heterozygosity. The program initially tries with $t = 0.01$. If this fails, it increases to $t = 0.02$. If the latter fails, the program reports an error and stops. The default value for the intital H_e is 0.5.

Note that the intital allele frequency distribution can have a large effect on the number of SNPs that run to fixation and so cannot contribute to a subsequent Nb or Ne estimation, depending on the number of cycles run. If you are concerned with minimizing the number of SNPs that run to fixation, then we recommend recording as few cycles as will meet your needs, and to start recording cycles as soon as possible after burn-in. If the number of fixed SNPs is a crucial quantity in your usage, we also recommend you make a test run and use a genepop file analysis progrem (such as the Genepop software package) to see how many SNPs have run to fixation at each cycle.

0.7.5.5 Msat het initialization

The value here determines an expected heterozygosity (H_e) for all Microsatellites in the initial population, so that it also gives the mean expected heterozygosity for the SNP set. The value will determine a set of Dirichlet-distributed allele frequencies, with the shape parameter chosen heuristically to produce through trials a set of frequencies that generate an expected heterozygosity within 0.001 of the target value. The default value for the intital H_e is 0.8. The valid range for for this parameter is $0.0 < H_e \leq 0.85$.

0.7.5.6 Nb and census adjustment

This parameter offers one or more specifications that will change the target Nb and the number of individuals in the population by a fixed rate and at a range of cycles (one or more). Entries are of the form $min - max : rate$, specifying a change in Nb and census size applied at cycle numbers min through max. The values conform to $min <= 2 <= max$, and $rate >= 0.0$. No adjustment is made with $rate = 0.0$. For example, to reduce the Nb and the total number of individuals by a tenth at cycle 3 (remaining in effect for the remaining cycles, unless another adjustment is added to the list), you would edit the entry to read, 3-3:0.1. The adjustments are different, depending whether $rate$ is less than or greater than 1.0:

- If $rate$ is less than 1.0, the target Nb value, and each age class in the current census is reduced by the proportion given by $rate$. Note that the change in Nb will result in a change to N0 as described above the in section 0.7.3.1.

- If $rate > 1.0$, the target Nb value will be multiplied by $rate$, with a resulting recalculation of N0. No change will be made to the current census.

0.7.5.7 Replicates

This value sets the number of independent simulations run with the current parameter set. These can be run in parallel if you specify more than one process in the Processes parameter (Section 0.7.1.1).

0.7.5.8 Skip breeding probability

If this value is not set to "None," it should be a list of percentages. It effects the number of available females of a given age at a given cycle number c. The $i^{th} percentp_i$ gives the probability $(p/100)$ that a female of $age = i$, is not able to breed in cycle c. Like the Litter (Section 0.7.3.10) and Force Skip parameters (Section 0.7.3.9), this parameter is not settable in the interface, but can be included in your configuration file.

0.7.5.9 Cycles of burn-in

This integer n is valid in the range $1 \leq n \leq r$, with r giving the total reproductive cycles (Section 0.7.5.11). This value tells the simulation that the Nb tolerance test for true Nb variance (Section 0.7.3.5) should not be performed for the first n cycles. The default value for this parameter equals twice the number of Ages in the model.

0.7.5.10 Start recording at cycle

This integer value c will result in the genepop file containing only the populations of cycles c through r, where r = total Reproductive cycles . This can greatly reduce the size of the output genepop file, when you are interested only in the last $r - c$ cycles, but want to simulate many cycles before recording, and when you have large populations and many loci to simulate.

0.7.5.11 Stop recording at cycle

This integer c tells the simulation to stop after recording (and end) after the c^{th} cycle. (total cycles includes the burn-in cycles).

0.7.5.12 Tolerance tries

Choices for this parameter are 100, 1000, and 10,000. For each reproductive cycle, this value is the maximum number of tries allowed for a population's PWoP calculated Nb value to be within the range given by parameter True Nb varialnce allowed (Section 0.7.3.5). Set at 100, this parameter can cause simulation interuption failures when the parameterization of the population's size, fecundity, survival values, etc., result in PWoP caluclated distributions of Nb values with a high enough variance that that 100 tries is insufficient. On the other hand, a setting of 10,000, useful when the distribution of possible PWoP values has high variance, will, when the the simulation is parameterized

such that PWoP Nb calculations cannot meet the tolerance setting, cause a very lenghtly but ultimately interrupted simulation.

0.8 Manually editing configuration files

Our program provides a collection of simulation configuration files available at https://github.com/popgengui/agestrucne/tree/data. You can directly edit a *conf file, which can sometimes be more convenient than using the GUI interface to change parameter values. If you open one of the supplied configuration files in the subdirectory, "configuration_files," youll see the parameter=value pairs, each below a section header inside square brackets. Some of the parameter names differ from their functional equivalents in the interface (see Table 0.1).

0.9 Simulation output

When a simulation is complete the message "simulation in progress" will disappear from the interface and editable entry boxes will no longer be grayed-out. A completed simulation delivers a genepop file for each replicate, named using the Output files base name parameter shown in the The Load/Run section of the simulation input (Section 0.7). The base name is extended with a replicate number and a "genepop" extension, so that, for example, if your simulation output base name is bulltrout, and you specified 3 replicates, the output file for the 3rd replicate would be named "bulltrout.r3.genepop". Also, there are several produced during the first replicate only, all prefixed with the output base name. The output file with extension "conf" lists the parameter settings for the simulation (and, hence, for all replicates). Other ouput files include extension "_age_counts_by_gen.tsv," and extension "_nb_values_calc_by_gen.tsv". When the simulation includes a heterozygosity filter (Section 0.7.5.2) the output will include a file with extension "_het_value_by_cycle_number.tsv". When the simulation is run with a non-zero "Number of chromosomes" (Section 0.7.4.3), the output includes a file with extension, "_loci_and_chromosome.tsv". Details on the output files follow.

1. The conf file shows the parameter settings used in the simulation (except the number of replicates, which it always sets to one). This file can be loaded into another instance of the Simulation Input (see the Load/Run parameter) and another simulation with matching parameters can be run. Conveniently, if it represents many customized settings on a former configuration file, small changes to it can be made to run a slightly adjusted simulation, without having to re-enter all of the settings used to create it.

2. The age counts file is a table with tab-delimited fields that gives a count of total individuals for each age class, for each (one-indexed) reproductive cycle. The first line in the file gives column headers, the first "generation," referring to reproductive cycle number, a count of reproductive cycles, and the rest listing age classes simply as $1, 2, 3 \cdots t$, where $t =$ total age classes. This file is created only for the first simulation replicate.

Short name in file	Interface Name	Link to description
N0	N0 (Newborns)	(Section 0.7.3.1)
Nb	Nb	Set to "None" value(Section 0.7.3.4)
NbNc	Nb/Nc	(Section 0.7.3.2)
NbNe	Nb/Ne	(Section 0.7.3.3)
NbVar	True Nb variance allowed	(Section 0.7.3.5)
ages	Ages	(Section 0.7.3.6)
cull_method	Cull method	(Section 0.7.5.1)
dataDir	Ignored	Ignored
doNegBinom	Not documented	Ignored
do_het_filter	Filter recorded pops by heterozygosity	(Section 0.7.5.2)
fecundityFemale	Female relative fecundity	(Section 0.7.3.8)
fecundityMale	Male relative fecundity	(Section 0.7.3.8)
forceSkip	Force skip	(Section 0.7.3.9)
gammaAFemale	Details not documented	If all 4 gamma vectors provided, offspring created by fitness generator
gammaAMale	Details not documented	If all 4 gamma vectors provided, offspring created by fitness generator
gammaBFemale	Details not documented	If all 4 gamma vectors provided, offspring created by fitness generator
gammaBMale	Details not documented	If all 4 gamma vectors provided, offspring created by fitness generator
gens	Stop recording at cycle	(Section 0.7.5.11)
het_filter	Het filter parameters	(Section 0.7.5.3)
het_init_snp	SNP het init	(Section 0.7.5.4)
het_init_msat	Microsat het init	(Section 0.7.5.5)
isMonog	Monogamous	(Section 0.7.3.11)
lbd	Ignored	Ignored
litter	Proportional Litter Sizes	(Section 0.7.3.10)
maleProb	Probability male birth	(Section 0.7.3.12)
model_name	Model	(Section 0.7.2.2)
mutFreq	Mutation frequency	(Section 0.7.4.2)
nbadjustment	Nb and census adjustment	(Section 0.7.5.6)
numChroms	Number of chromosomes	(Section 0.7.4.3)
numMSats	Number of microsatellites	(Section 0.7.4.4)
numSNPs	Number of SNPs	(Section 0.7.4.5)
popSize	Population size	(Section 0.7.3.13)
reps	Replicates	(Section 0.7.5.7)
skip	Skip breeding probability	(Section 0.7.5.8)
startAlleles	Starting Msat allele total	(Section 0.7.4.6)
startLambda	Cycles of burn-in	(Section 0.7.5.9)
startSave	Start recording at cycle	(Section 0.7.5.10)
survivalFemale	Female survival	(Section 0.7.3.7)
survivalMale	Male survival	(Section 0.7.3.7)
use_loci_file	Use Loci File	(Section 0.7.4.1)

Table 0.1: Short parameter names as found in configuration files, and their descriptions. Note that a few of our provided configuration files include male and female gamma A and B distribution vectors (noted in this table) which results in a fitness calulation during offpring creation, a feature currently not documented in our manual.

3. The Nb values file is a table with tab-delimited fields giving the PWoP Nb values calculated during the simulation, and used to compare to the target Nb value +/- the true Nb variance value. The first column gives the one-indexed reproductive cycle number and the second the PWoP-based Nb value that passed the tolerance test, and represents the accepted

population for that cycle. This file is created only for the first simulation replicate.

4. The heterozygosity-by-cycle output file is created only when the simulation includes a heterozygosity filter (Section 0.7.5.2). The two comma-delimited fields give the reproductive cycle number (also referred to as the "pop" number, and the mean expected heterozygosity (Section 0.7.5.3) of the population at that cycle number. The entries are restricted to those pops that are also recorded in the genepop file (i.e. that passed the filter parameters (Section 0.7.5.3).

5. The loci-and-chromosome output file is created only when the value for the "Number of chromosomes" parameter (Section 0.7.4.3) is non-zero. The two, tab-delimited values in each line give, first, a chromosome name, in the form of an integer such that N chromosomes are named $0, 1, 2 \cdots N-1$, and, second, a loci name that matches one of those in the first section of the output genepop file.

6. The genepop file(s) conform(s) to the genepop file standards given at `http://genepop.curtin.edu.au/help_input.html`. The header line notes the name of the *.gen file it came from, which simply names an intermediate file from which it derived its population information. It also gives the value of Nb/Ne. If the value is non-zero, it can be loaded automatically into the Nb/Ne estimation interface (see the Parameters section 0.11.3 of the Nb/Ne Estimation interface description. The second line of the genepop file gives the name of the first loci, which is simply its ordinal, "l_0." Each consecutive loci, $l_0, l_1, l_2 \ldots l_{L-1}$ (where L gives the total number of microsatellites plus the total number of SNPs) is listed on a separate line. Note that the first M loci will represent the microsatellites, and the last S loci will represent the SNPs, with M and S the totals given in the genome parameters (Section 0.7.4) of the Simulation Input. Thereafter the file consists of separate "pop" sections, each representing a reproductive cycle. The first n cycles (as numbered $1, 2, 3 \ldots n$) will not be in the file if the Start at cycle number parameter is set to $n+1$. The population for each cycle is listed, in order of cycle number. Each is demarked by a line with "pop" as its sole entry. Individuals, one to a line, follow each "pop" entry. Each individual has an ID with multiple fields delimited by a semicolon, giving, individual id number;sex (1 = male, 2 = female);id of father;id of mother;age class. These are followed by a comma, and then a space-delimited set of alleles for each locus named in the lines 2 total number of loci. Note that these allele entries represent diploidy, and use 3-digit allele numbering so that, for each loci, allele one is named by the first 3 digits, and allele 2 by the last 3. Also, please note our remarks about SNP allele frequencies running to fixation in section 0.7.5.4.

0.10 Running an Nb or Ne Estimation

The Nb (and Ne) estimation interface performs an LD based Nb or Ne estimation from genepop file input as supplied by the user. While it was developed in

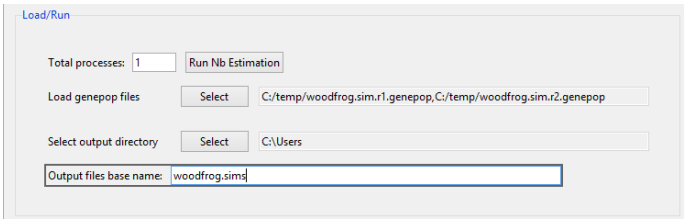


Figure 0.7: Nb/Ne estimation interface, Load/Run section

concert with the simuPOP-based simulation output from our programs interface, it will perform estimations on any genepop file input. To run estimations:

a Load a new Nb interface with the add menu's "Add new Nb estimation" option (Figure 0.1) and set the parameters with the provided controls. For details see Section 0.11

b Load a genepop file..

c Adjust the estimation parameters. The parameters are detailed in Section 0.11

d Click the button labeled "Run Nb Estimation", and the computations will start. The buttons text now changes to say "cancel simulation," and next to it a new label will note that "estimations in progress." As in the other interfaces, while the estimations are in progress, the parameter controls are disabled.

0.10.1 Running an Nb or Ne estimation from the command line

The module pgdriveneestimator.py offers a command line interface for running LDNe2 estimations. At a Linux or DOS terminal you can type a command of the form,

```
<python> <pgdriveneestimator.py> <options>
```

where

- ⟨python⟩ is the python3 or python2.7 executable,
- ⟨pgdriveneestimator.py⟩ is either the module name pgdriveneestimator.py, if your terminal already knows the path to the module, which should be the case with most installations (see the README.md listed above in Section 0.2.1 and available in our github repository's main directory), or the full path to the module (in the distributions main directory), and
- ⟨options⟩ is a list of parameters given using the option flags. To see the list of options, invoke the command without any options.

To see a detailed list of options, invoke the command with,

```
<python> <pgdriveneestimator.py> -h
```

0.11 Nb/Ne Estimations input

The interface provides for multiple subsampling schemes of both individuals and loci within the input genepop file pop sections. The sub-sections of the interface follow.

0.11.1 The Load/Run section

This section (Figure 0.7) offers an interface to load input and name the output files.

0.11.1.1 Total processes

The program will run estimations on the individual "pop" sections in parallel using the number of processes set here. It is usually advisable, unless your computer has many process already running, to use most if not all of your available (virtual) processing cores, to speed up the estimation run parameter. The program defaults to using half of the available cpu's (or virtual cores) available on your machine.

0.11.1.2 Load genepop files button

Clicking on this button produces a file loading interface to locate and load one or more genepop file(s). Note that when you load multiple genepop files, the parameter settings will be applied to all. In particular, activating an Nb bias adjustment will apply it to all the files, so that only data with which it is compatible should be loaded. This also applies to other parameters, such as the population number

0.11.1.3 Select output directory

By clicking on the button and choosing your preferred folder you select where the estimation output files will be written. Note that this will also be used as a temporary directory in which intermediate files will be written inside new directories with the "tmp" prefix, ending in random characters. These files will be removed on completion of the simulation. Sometimes, if the estimation run is cancelled or otherwise is interrupted, they will not be removed, but can be manually deleted from your directory.

0.11.1.4 Output files base name

The text entered here will become the prefix for the output files (Section 0.12).

0.11.2 The Genepop Files Loaded section

This section has a single box that shows you the names of the loaded genepop files (Figure 0.8). It is not an editable section.

0.11.3 The Parameters section

This section supplies the main parameters, including the choice of subsampling in pop sections and/or loci (Figure 0.9).

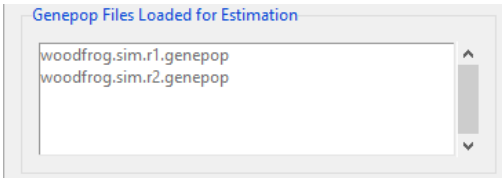


Figure 0.8: Nb/Ne estimation interface, genepop files loaded section

0.11.3.1 Use Chrom Loci File

Check this box, and click on "Chrom Loci File", to locate a file that has two tab-delimited columns, chromosome name and loci name. Each loci name in each genepop file loaded should be present in this file, so that LDNe2 will have a chromosome to associate with all loci.

The LDNe2 program will use this file to compute LDNe using only the alleles of loci pairs $l_i,\ l_j$ on chromosomes $c_k,\ c_l$, such that $c_k \neq c_l$.

Note that the command line module pgdrivesimulation.py offers the user use a second parameter related to this file name, and that will instruct LDNe2 to compute using only the loci pairs such that $c_k = c_l$. This option is not currently available from the graphical interface.

In the absense of a file name (or when the Use Chrom Loci File box is unchecked), LDNe2 will use all loci pairs to compute LDNe.

0.11.3.2 Minimum allele frequency

This value sets the threshold below which the LDNe program will ignore an allele in its LDNe calculation. The interface defaults to 0.05, as a value that is commonly used, and which reduces upward bias [3]

0.11.3.3 Monagamy

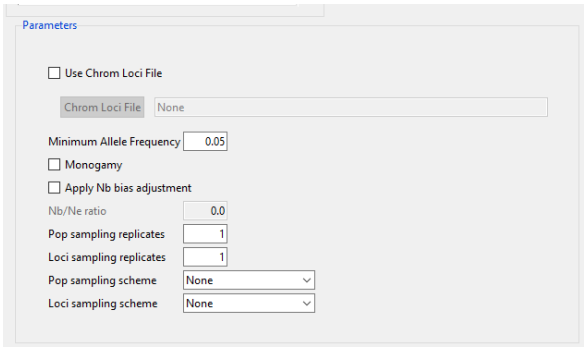
When this box is checked, the LDNe2 parameter "mating" is set to monogamy. If the box is unchecked the parameter is set to "random mating."

0.11.3.4 Nb bias adjustment check box

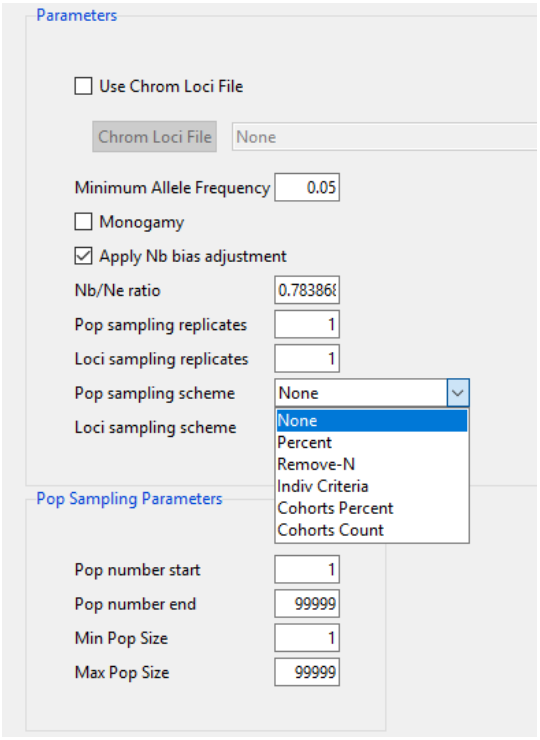
Checking this allows you to apply a bias adjustment to the estimations [3]. See Section 0.11.3.5.

0.11.3.5 Nb/Ne ratio

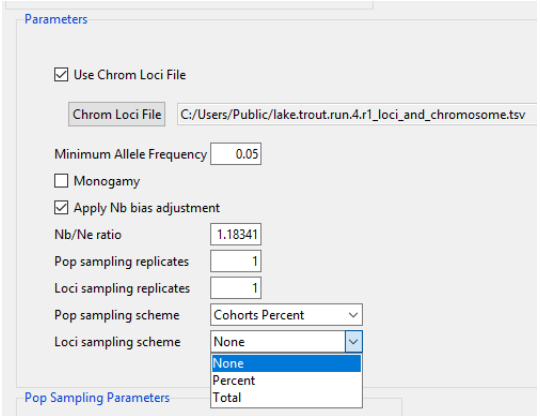
This parameter is the value used for the bias adjustment, when it is checked. A zero value or an un-checked box means no bias adjustment will be done. Note that when you load a genepop file generated by the simulation interface, and you check the box labeled "applyNb bias adjustment", the program will load the Nb/Ne value as set in the simulation interface. You can accept it or enter another value. If no value is available in the genepop file, then you will need to enter a non-zero value to make any bias adjustment.



(a) No subsampling



(b) Pop subsampling options



(c) Loci subsampling options

Figure 0.9: Nb/Ne estimations interface, parameters section

0.11.3.6 Pop sampling replicates

When set to an integer n, for each estimate, for each pop sampling parameter, the estimate is repeated n times. While you can set this to any value for any subsampling scheme, note that it is sensible only when your subsampling parameter involves a random sample of individuals (Section 0.11.3.8) more than 1 loci subsampling parameter subsampling schemes (Section 0.11.3.9), or both. If there is no random subsampling, the replicates will be performed, with identical results.

0.11.3.7 Loci sampling replicates.

When set to n, for each pop sampling replicate, for each loci subsampling parameter value, the estimate is repeated n times. As with the Pop sampling replicates parameter, if the loci subsampling scheme has no randomized subsample (Section 0.11.3.9), then the estimates will be identical.

0.11.3.8 Pop sampling scheme

This drop-down box offers the following subsampling schemes (Figure 0.9b).

1. None. This scheme uses all individuals with the pop section for the Nb or Ne estimation, unless the value for Indiv max per pop reduces the pop size to m randomly selected individuals. If the pop section has fewer individuals than the value given by Indiv min per pop , the pop section is skipped, with a message written to the Messages file (Figure 0.10a).

2. Percent. When you select this scheme the pop sampling interface (Section 0.11.4) shows a percent box with two buttons below it (Figure 0.10b), "Add Value," and "Trim." You can edit any box currently in the list. Clicking "Add Value" will append a box to the list, its default value taken from its nearest neighbor. For each percentage p in the list, each pop section will be reduced to p percent of its total individuals, unless its census is not in the pop number range (Sections 0.11.4.1 and 0.11.4.2), in which case the population will be skipped, and a message written to the messages file (Section 0.12.1). For each subsample the individuals are randomly selected. Note that any multiple loci subsampling values and/or Pop sampling replicates and Loci sampling replicates replicates will result in an estimate for each percentage value, repeated for each of the loci subsampling and replicate values.

3. Remove-N. This scheme also offers an editable list, which behaves as described for the Percent scheme (Figure 0.10c). For this scheme, when you enter an integer N, the pop subsample will be that given by its total minus N, the removed N individuals randomly selected except in the case of $N = 1$, in which case each of the $remove - 1$ cases will be estimated (i.e. there will be t estimations for a pop section with t individuals. Pop sections are skipped if the total individuals in the population are not in the pop number range (see Sections 0.11.4.1, and 0.11.4.2). Note, as with the Percent scheme, estimates for each N value will be repeated for loci sample values, pop sampling replicates, and Loci sampling replicates.

4. Cohorts Percent. Because this scheme selects individuals by age, it requires input genepop files produced by the Simulation output (Section 0.9). If your input is empirical data and you wish to calculate Nb, it is assumed that each pop in the genepop file is a single cohort and all individuals should be used in the Nb calculation. When Cohorts is selected the interface shows, besides its usual pop number and min/max pop size parameters, an entry labelled "Indiv max age" (Figure 0.10d). For each population, individuals outside the ages given by [0.0, max age] are excluded. However, because the simulations produce genepop files with no individuals aged less than 1.0, the interval is effectively [1.0, max age]. For each pop, and for each percentage value in the list, subsampling steps are:

   a. For each age value in $[0.0, maxage]$, count the total individuals t_a in the age group a.

   b. Find the smallest of the age group totals, $t_{smallest}$.

   c. Randomly subsample $t_{smallest}$ individuals from each age group to get a total sample size s.

   d. s is in the range [Indiv min per pop, Indiv max per pop] then, for each percentage p in the percentage list, randomly select p percent of the collected individuals. If s is outside the range an error occurs and the analysis is terminated.

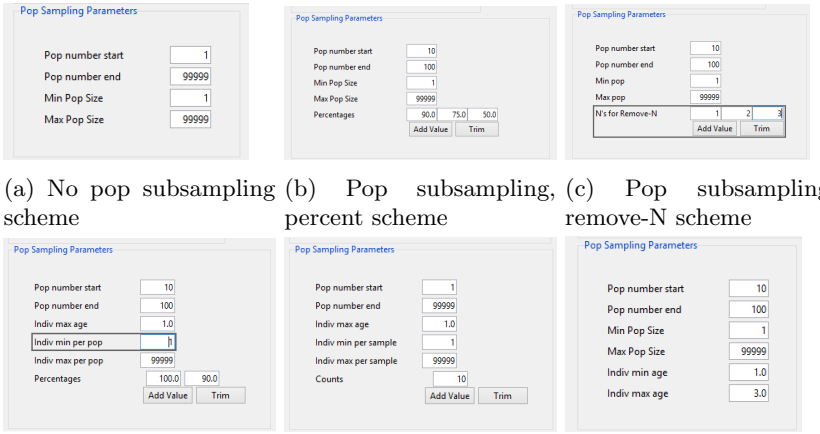
5. Cohorts Count. This scheme behaves very similarly to the Cohorts Percent (above, 4), with nearly identical parameters (Figure 0.10e). However, instead of sampling percentages, for each sample value c_i in the list of sampling values, c_i individuals will be randomly selected from each age class. If for any pop section, any of the age classes has a count of individuals less than c_i, an error is raised and the estimations terminate.

6. Individual Criteria. This scheme allows you to select a range of cohorts by contiguous age group. Like the cohorts schemes, it requires your input to be genepop files produced by the Simulation output. When Individual Criteria is selected the interface shows, besides its the pop number and min/max pop size parameters, two entries for a minimum and maximum age (Figure 0.10f). The program pools all individuals within the minimum and maximum age range (inclusive), and calculates the estimation using the pool. If the total of pooled individuals is outside the range given by [Indiv min per pop Indiv max per pop], an error is thrown and the analysis is terminated.

0.11.3.9 Loci sampling scheme

1. None. All loci will be used in the estimations, from the i^{th} to the j^{th}, i and j given by the Loci number start and Loci number end (Figure 0.11a). If the Max Loci count m is less than the total loci, then m loci will be randomly selected.

2. Percent. For each percentage p listed in the "percentages" boxes (Figure 0.11b), an estimation is calculated using a random selection of p percent



(a) No pop subsampling scheme

(b) Pop subsampling, percent scheme

(c) Pop subsampling, remove-N scheme

(d) Pop subsampling, cohorts percent scheme

(e) Pop subsampling, cohorts count scheme

(f) Pop subsampling, individual criteria scheme

Figure 0.10: Nb/Ne estimations interface, pop subsampling parameters

of the loci from those within the range given by Loci number start to Loci number end.

3. Total. For each total t listed in the "totals" boxes (Figure 0.11c), an estimation is calculated using a random of t loci from those within the range given by Loci number start to Loci number end.

0.11.4 Pop sampling parameters section

In this section (Figure 0.10), you set the pop section sampling parameters, which are presented according to the scheme selected in the Pop sampling scheme parameter.

0.11.4.1 Pop number start

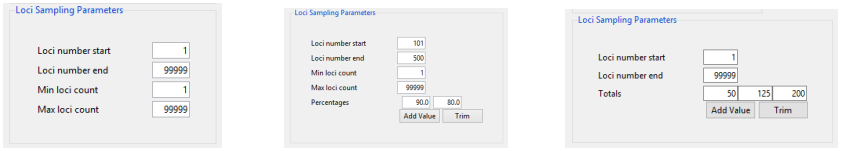
When this is set to integer n, the estimates will skip pop section numbers (as ordered in the genepop file) in the range [1, n 1].

0.11.4.2 Pop number end

If the genepop file has t total pop sections, then, when this parameter is set to integer n, the estimates will skip pop section numbers in the range $[n + 1, t]$.

0.11.4.3 Indiv min per pop

For the None, Percent, and Remove-N sampling schemes, a pop section must contain at least this many individuals, or it will be skipped. Skipped populations will be noted in the Messages file. For the Cohorts and Individual Criteria sampling schemes, the total cohort sample must meet or exceed this minimum, or an error occurs and the estimation run is terminated.



(a) No loci subsampling scheme (b) Loci subsampling, percent scheme (c) Loci subsampling, total scheme

Figure 0.11: Nb/Ne estimations interface, loci subsampling parameters

0.11.4.4 Indiv max per pop

For the None sampling scheme, if total individuals in a pop section exceed this value i_max, than i_max individuals will be randomly selected from the pop. For the Percent and Remove-N schemes, pop sections with individuals totaling more than this value will be skipped, with a message written to the Messages file. For the Cohorts scheme, if sample size exceeds this value, then an error occurs and the estimation run is terminated; for the Individual Criteria scheme, the sample is reduced to the value by random selection of individuals.

0.11.4.5 Scheme-specific parameters

See the descriptions in Section 0.11.3.8.

0.11.5 Loci sampling parameters section

In this part of the interface you can limit the loci used to determine the Nb or Ne estimation.

0.11.5.1 Loci number start

Loci used in the estimation will be drawn from the i^{th} to the j^{th} Loci as ordered in the genepop file. This value gives the i^{th}. Note that this range allows you, for example, to run a simulation in which the first m loci are microsatellites and the last s loci are SNPs, and then use the same genepop file in two different estimations, one based on microsatellites using loci range 1 to m, and the second based on SNPs using range loci $m + 1$ to $m + s$.

0.11.5.2 Loci number end

This value gives the j^{th} loci in the range as described for Loci number start.

0.11.5.3 Min Loci count

This value sets a minimum for the total loci to be used in the estimation. If the loci sample in the genepop file is less than this value, the program generates an error message and the run is terminated.

0.11.5.4 Max Loci count

This sets a maximum value m on the number of loci to be used in the estimation. If the range gives a larger total, then m loci will be randomly selected from those in the range, Loci number start to Loci number end.

0.11.5.5 Scheme specific parameters

See the descriptions in Section 0.11.3.9.

0.12 Nb estimation output

0.12.1 Messages file

With extension *.msgs, this file shows the parameter settings used in the estimations, and also logs error messages.

0.12.2 Estimates table

With extension *.tsv, this file gives tab-delimited quantities associated with the Nb or Ne estimations (Table 0.2)

0.13 Visualization Interfaces

The program offers 2 interfaces that plot the Ne/Nb estimations as output by the program's LDNe output (Section 0.12).

0.13.1 Boxplot Interface

This interface is available by clicking on the "Add" menu in the main menu (Figure 0.1), then clicking "Add new boxplot interface." After using the Button labeled "Select to load a *tsv file (Section 0.12), you can group and filter input values to get a boxplot of the LDNe Nb/Ne estimations according to the groups and filters (Figure 0.12a).

0.13.1.1 Grouping.

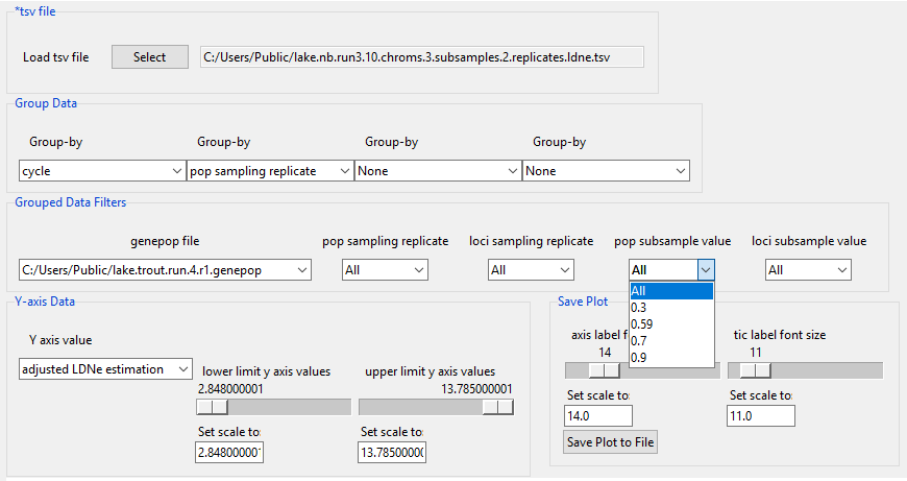
The interface offers four Group-by fields allowing you to group the data using up to four value sets according to the input fields as given in the Ne/Nb estimation output (See Table 0.2):

- genepop file from original_file column in the Ne/Nb estimation output tsv file.
- pop number, from the pop column.
- total indiv. sampled, from the indiv_count column.
- pop subsample value, from the sample_value column.
- pop sampling replicate, from the replicate_number column.
- loci subsample value, from the loci_sample_value column.

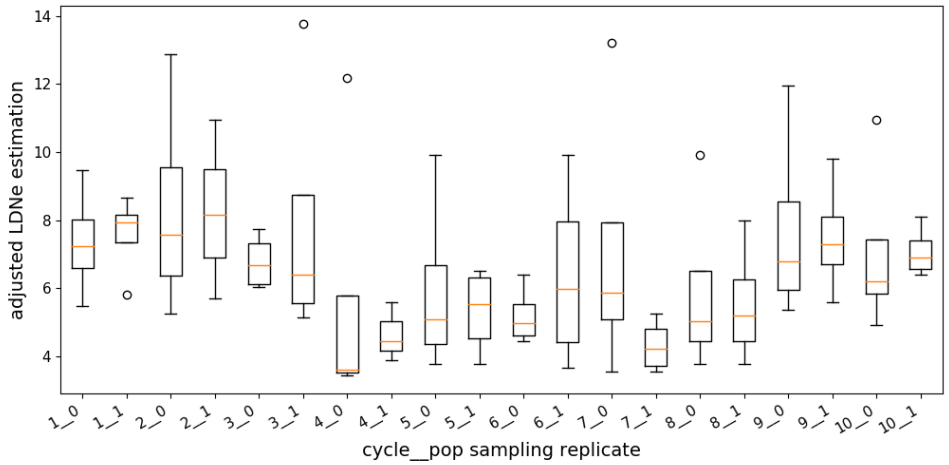
Output file column	Description
original_file	Source genepop (gp) file from which the pop sections are derived.
pop	The i^{th} reproductive cycle as ordered in the source gp file.
census	The total individuals in the pop.
indiv_count	The total individuals used in the estimation.
sample_value	The parameter, if any, used in the pop subsampling (e.g. percent).
replicate_number	The pop replicate number.
loci_sample_value	The parameter value, if any, used in the loci subsamlint (e.g. percent).
loci_replicate_number	The loci replicate number.
min_allele_freq	An LDNe2 parameter [no citation yet for LDNe2], the min allele frequency required to include an allele in the estimation.
est_type	Refers to the estimation type, currently only implemented fo "ld", that is, the estimation is based on the LDNe method.
est_ne	The Ne or Nb estimation [no citation.
95ci_low	The lower of a 95% confidence interval for the estimate, based on the jackknife method described in [no citation yet for LDNe2].
95ci_high	The upper 95% confidence interval for the estimate, based on the jackknife method described in [no citation yet for LDNe2]
overall_rsquared	A weighted mean of the allelic pairwise estimators $\hat{r}_{\Delta}$ (see [no citation yet for LDNe2]).
expected_rsquared	The expected value of the estimator $\hat{r}_{\Delta}$ (see [no citation yet for LDNe2]).
indep_comparisons	The number of independent comparisons used to determine the estimate (see [no citation yet for LDNe2]).
harmon_mean_samp_size	Sample size to calculate the LDNe estimate ([no citation yet for LDNe2]).
alt_ci_low	Lower, parametric 95% confidence interval ([no citation yet for LDNe2]).
alt_ci_high	Upper, parametric 95% confidence interval ([no citation yet for LDNe2]).
nbne	If not "None," then this is the Nb/Ne ratio used in a bias adjustment for the estimate (see [2]).
ne_est_adj	The estimate as bias-adjusted (see [2]). If no bias adjustment was calculated, then this value will be identical to that in the est_ne column.
95ci_low_adj	The 95ci_low value as bias-adjusted (see [2]). If no bias adjustment was calculated, then this value will be identical to that in the 95ci_low column.
95ci_high_adj	The 95ci_high value as bias-adjused (see [2]). If no bias adjustment was calculated, then this value will be identical to that in the 95ci_high column.
mean_het	The mean expected heterozygosity of the population, calculated as defined in Section 0.7.5.3. The calculation employs all individuals in the pop (no subsampling is applied), and only the loci with the range given by the start loci number parameter (Section 0.11.5.1) and the end loci number parameter (Section 0.11.5.2).

Table 0.2: Nb or Ne estimation output values

- loci sampling replicate, from the loci_repicate_number column.



(a) The boxplot interface, grouping on two input fields, pop (reproductive cycle number) and pop sampling replicate number.



(b) The resulting boxplot.

Figure 0.12: The LDNe Ne/Nb boxplot interface.

0.13.1.2 Filtering.

Filtering allows you to specify one of the set of values for several input fields, so that output entries without that value will be excluded from the plot data. For example, if you have output from a run in which you set a percentage subsampling scheme for pops (Section 0.11.3.8), and you sampled by 50, 80, and 100% , you can select one of the percentages, and all entries used to make the boxplot will be those with the selected pop subsampling value.

0.13.1.3 Y-axis field.

You can select the values for the y-axis field using the "Y-axis values" drop-down box. The following are the availbale output fields in the LDNe Nb/Ne estimation output (See Table 0.2):

- LDNe estimation, from the est_ne column.
- low 95% CI, from the 95ci_low column.
- high 95% CI, from the 95ci_high column.
- adjusted LDNe estimation, from the ne_est_adj column.

0.13.1.4 Y-axis value limits.

You can restrict the range of Y values plotted using the sliders or their associated entry boxes. Note that in all cases values of Infinity, and "NA," which can be present in the ouput fields, have been excluded.

0.13.1.5 Axis and tick label font size adjustment

You can adjust the font size for the axis and the tick labels using either the slidling scale controls or the text boxes. These can be especially helpful when the size of tic labels causes axis labels to be pushed out of the frame.

0.13.1.6 The plot.

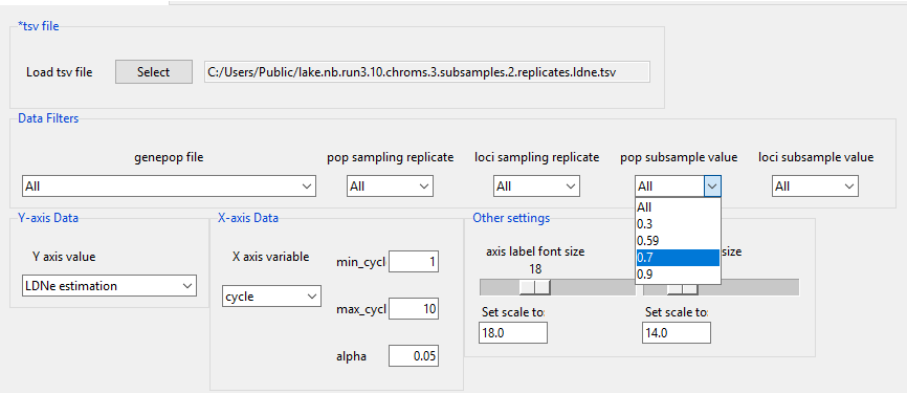
The plots produced (Figure 0.12b) result from a call to the maplotlib "boxplot" command , with its "x" parameter set to the grouped/filtered data, and the "labels argument" set to show the grouped field values (separated by double underscores). Otherwise, arguments to boxplot are not specified, and so use default arguments. You can use the "Save" button in the "Save Plot" subframe to save the plot to file. A *png image format will result if you use any file name, except when you add ".pdf," which will then be the format of the saved file image. png and pdf are the only formats used.

0.13.2 Regression Interface

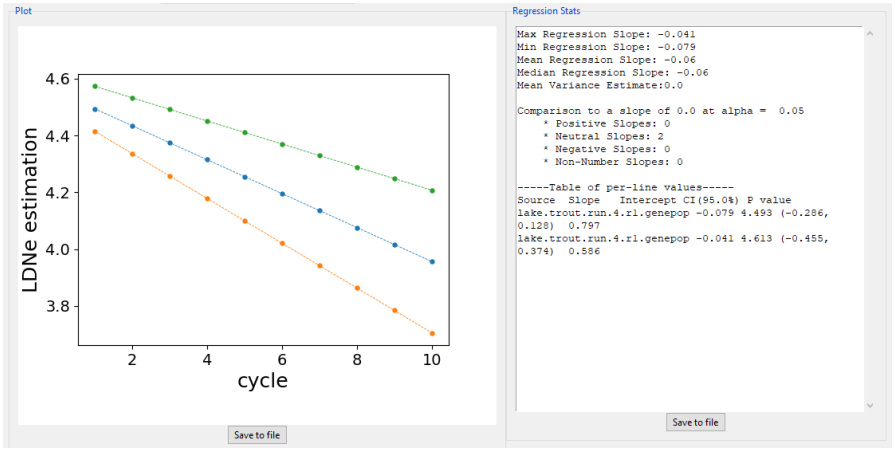
This interface is available by clicking on the "Add" menu in main menu (Figure 0.1), then clicking "Add new regress plot interface." After loading a LDNe Nb/Ne estimation output file (Section 0.12.2), you can plot regression lines (Figure 0.13).

0.13.2.1 Filtering.

These selections offer the same filtering as do those for the boxplot (Section 0.13.1.2). In this case, restricting an input field to a single value reduces the number of lines regressed, such as regressing over pop values, but only for one of multiple genepop files (Figure 0.13a), or regressing over a series of genepop files, but only for the estimations associated with pop sampling replicate 2.



(a) Settings, filtering set to select one of 3 pop subsampling values, 0.7.



(b) The resulting plot.

Figure 0.13: The Nb/Ne estimation regression interface with cycle number as the x-axis variable.

0.13.2.2 Y-axis field

The "Y axis value" drop-down box allows you to select which set of LDNe output values are used as the response variable in the regression. See the description for the boxplots in section 0.13.1.3.

0.13.2.3 X axis variable

You can regress using either the pop number (i.e. the values in the pop column in the output tsv file, see Table 0.2), or the genepop file (i.e. the file names in the "original_file" column in the output tsv file, stripped of their directory paths. When you use file names as x-axis values the program will assign numeric values to the file names in one of two ways.

- Genepop file names are numbers. If your genepop files are named using all-numbers, they will become the numerical values used along the x-axis.

- Genepop file names are not all-numbers. In this case the files will be assigned integers $1, 2, 3 \ldots N$ according to a natural sorting algorithm, that will sort non-number parts of the name alphabetically, but will sort number-parts numerically. This means, for example, that files with form myfile_1.txt, myfile_11.txt, and myfile_2.txt will be sorted in order, and assigned x-values 1,2,3:

  1. myfile_1.txt
  2. myfile_2.txt
  3. myfile_11.txt

  This genepop-file based numbering was created to accommodate regression over a series of genepop files with a single pop entry, with each file representing, for example, an annual sample of a real population, and each file named text giving the population name and a number representing the year, or an otherwise ordinal number.

0.13.2.4 Min cycle and Max cycle text boxes

You can limit the cycle range over which the regression is calculated by entering a minimum and maximum cycle number in these boxes.

0.13.2.5 Axis and tick label font size adjustment

You can adjust the font size for the axis and the tick labels using either the slidling scale controls or the text boxes. This will be helpfull as changing tic axis values can push labels out of frame.

0.13.2.6 The plot

The plot shows a regression line for each combination of input fields, as well as a line based on an expected slope.

You can use the "Save" button to save the current plot to file, either as a png (the default image format when you use a name with no extension, or an extension other than ".pdf" which is the sole alternative format.

0.13.2.7 The regression stats text box

This text box shows regression statistics and quantities. You can save this text to file using the "Save" button.

Bibliography

[1] Bo Peng and Marek Kimmel. simuPOP: a forward-time population genetics simulation environment. *Bioinformatics*, 21(18):3686–3687, September 2005.

[2] Robin S. Waples, Tiago Antao, and Gordon Luikart. Effects of Overlapping Generations on Linkage Disequilibrium Estimates of Effective Population Size. *Genetics*, 197(2):769–780, June 2014.

[3] Robin S. Waples and Chi Do. ldne: a program for estimating effective population size from data on linkage disequilibrium. *Molecular Ecology Resources*, 8(4):753–756, July 2008.

[4] Robin S. Waples and Ryan K. Waples. Inbreeding effective population size and parentage analysis without parents. *Molecular Ecology Resources*, 11:162–171, March 2011.