SPAM

Version 3.2

User's Guide

STATISTICS
PROGRAM FOR
ANALYZING
MIXTURES

Special Publication No. 15

Alaska Department of Fish and Game
Division of Commercial Fisheries
Gene Conservation Laboratory
333 Raspberry Road
Anchorage, Alaska 99518

August 2000

SPAM

(Statistics Program for Analyzing Mixtures)

Version 3.2

Special Publication No. 15

Alaska Department of Fish and Game
Division of Commercial Fisheries
Gene Conservation Laboratory
333 Raspberry Road
Anchorage, Alaska 99518

August 2000

Table of Contents

General Description

SPAM is a program developed by the Alaska Department of Fish and Game's (ADF&G) Gene Conservation Laboratory for analyzing admixtures of populations. Sometimes referred to as a mixed stock analysis (MSA) or genetic stock identification (GSI), the analysis employs maximum likelihood methods to estimate relative contributions of discrete populations in a mixture of several populations. The first step in such an analysis is to assemble a baseline of genotypic and/or phenotypic characteristics for populations that may be present in the mixture. The mixture is then sampled and the same suite of characteristics in the baseline is measured in the sample. The contribution estimates for each population in the baseline are computed as those that result in the greatest probability of obtaining the observed mixture sample.

SPAM is a FORTRAN program compiled for use under the Windows 95/98 operating system. The algorithms used in SPAM are taken from the GIRLSEM and CONJA-S programs written by Masuda et al. (1991) and Pella et al. (1996) of the National Marine Fisheries Service Auke Bay Laboratory and from the program HIGHSEAS written by Smouse et al. (1990). Multiple windows are used to monitor analysis progress and report actions taken. ASCII files are used to provide input to SPAM and report results. Each analysis uses an ASCII control file to set options and parameters that define the analysis to be performed.

Two types of analyses can be performed with SPAM: ***estimation*** of the mixture composition, and ***simulation*** of possible mixture scenarios to evaluate performance for a given baseline. The estimation procedure includes bootstrapping to evaluate bias (accuracy) and variance (precision) due to mixture sampling error and sampling error in the baseline. The simulation procedure permits the user to specify relative population sizes so that population contributions to the simulated mixture can reflect real situations. With either analysis mode, SPAM generates contribution estimates for each population in the baseline. The user has the option of generating estimates for larger aggregations of populations to coincide with specific spatial boundaries or to pool populations that are genetically indistinguishable.

The baseline file(s) contains the ***response*** (allele or phenotype) frequencies for each ***character*** (phenotype, locus, or isolocus) and can be stored in one large ASCII file or in a separate file for each population. The response frequencies can be either relative frequencies or absolute frequencies (response counts). The absolute frequency format is preferred because relative frequencies are summary statistics of the absolute frequencies and can cause round off error in subsequent computations. If relative frequencies are used, they are converted to absolute frequencies and stored as such in SPAM. During this conversion process, responses can potentially be lost due to rounding error. If this occurs, a warning message identifies the population and character. Moreover, if a baseline population is missing the response frequencies for a character (or set of characters), the program will attempt to estimate them using a generalization of the expectation-maximization (EM) algorithm (Smouse et al. 1990).

The mixture file specifies individual multi-character ***types*** (i.e., genotypes if genetic characters are used). SPAM identifies all unique types and tallies their frequencies. If information on a character for an individual is not available, zeros are entered for that character. Individuals that have information for the same set of characters comprise a ***class***.

SPAM 3.2 provides a number of control, memory, calculation, and output enhancements over SPAM 2.0. These are detailed in the rest of the manual but, briefly: the user is now provided a complete windows environment in which to edit files, perform analyses, and view output; installation and initialization of the program are greatly simplified; the user can request the bootstrap resample estimates for later investigation; the user can request the conditional probability of a multi-character type for a given population and/or the conditional probability of a population for a given multi-character type; the user can set the confidence level for interval calculations; the user can request symmetric bootstrap confidence interval estimates; the maximum number of alleles has been increased from 9 to 100; the program has undergone extensive error checking and includes expanded error message capabilities. For a full version history of the software, see the SPAM webpages on the ADF&G's Gene Conservation Lab website (http://www.cf.adfg.state.ak.us/geninfo/research/genetics/).

The examples used throughout this manual are of Columbia River chinook salmon data which have been widely evaluated with earlier versions of SPAM and other maximum likelihood programs. All files required to duplicate these analyses are included in the file COLUMBIA.ZIP, which requires a file compression utility to unzip.

Installation and Use

SPAM version 3.2 contains a single executable file, SPAM32.EXE. This file can be installed and executed from any directory. Simply copy this file into the directory of choice.

Two options listed below (selection of a text editor and turning off error message boxes) make use of a SPAM.INI file located in the C:\WINDOWS directory. SPAM will create this file and make additions and revisions as necessary so that you will not need to edit this file yourself. The use of SPAM.INI is fully compatible with previous versions of SPAM, allowing for the continued use of multiple versions.

To create a shortcut for SPAM 3.2 on your desktop, right click the file in Explorer and select **Create Shortcut** and place it on your desktop. You can change the working directory of the shortcut so that file selection always begins in a predetermined directory. Right click the shortcut, select the **Shortcut** tab, and enter a path in the **Start in:** box. From this same location, you can select **Change Icon…** to select the old-style SPAM icon if you prefer. You can also add a SPAM entry to the Windows Start menu. Consult the Windows help files for detailed instructions.

SPAM is run by executing SPAM32.EXE, either directly, from a shortcut, or from the Start menu. A standard window appears from which you will perform analyses and view and/or edit files. It remains open for the duration of your SPAM session.

To run an analysis with an existing control file, select **Run SPAM** from the **Commands** menu, or press **F10**. An open-file dialog box appears which allows you to select one or more control files. The filter is set for a default file extension of `*.CTL` so we recommend you label your control files in this way. If your file has a different extension, select **All Files (*.*)** from the **Files of type:** dialog and locate your control file. When the appropriate file or files are selected, click **Open**. A child window will appear with the name of the control file in the caption bar along with the word "**Running**." Blue text will appear in the child window as the analysis moves through different stages. You are informed of any output files that are created. When the analysis is complete, the text changes to black and informs you the analysis is complete, the system asterisk sound is heard, and the word "**Finished**" appears in the caption bar.

If a fatal error occurs during the analysis, a popup window appears that describes the error. Click **OK** and the text in the child window changes to red and informs you the analysis was aborted, the system exclamation sound is heard, and the word "**Finished**" appears in the caption bar.

The analysis window informs you of all output files that have been created. To view the results of the analysis, select the output file you wish to view from the **Results** menu that appears with a child window after the analysis is complete. All possible files associated with an analysis are included in the list, but only those that pertain to the selected child window are visible and can be selected. Select a file and it will open in an external text editor. By default, Notepad is used as the text editor, but any text editor can be chosen as described below. You can also use function keys to view the more common files: control = **F5**, log = **F6**, estimation = **F7**, and simulation = **F8**. Once activated, the text editor is a stand-alone application and can be used to edit and save any file.

You can open any file for editing or viewing from SPAM. Select **Open File** from the **Commands** menu, or press **F11**. An open-file dialog box appears from which you can select a single file. The filter is initially set for control files (`*.CTL`), but can be changed to show all SPAM files (`*.CTL`, `*.LOG`, `*.EST`, `*.SIM`, `*.BOT`, `*.ITR`, `*.RSM`, `*.BSL`, `*.CMX`, `*.GEN`, and `*.POP`) or any file (`*.*`). Select the file, click **Open**, and the file is opened in the external text editor. With this feature, you can easily make changes to your control file within your SPAM session.

Once an analysis is complete and you have closed the child window or exited SPAM altogether, you can open any SPAM file directly with your text editor. All output files can be found in the directory specified in the control file. Alternatively, you can use the original control file to load just the list of output files associated with an analysis and view them from within SPAM. Select **Load Results List** from the Commands menu, or press **F12**. An open-file dialog box appears which allows you to select one or more control files. Select the file and click **Open**. A child window will appear with text indicating that the control file is being read. Upon completion, the text changes to green and informs you that the results list has been loaded. You can then use the **Results** menu to select the file or files you would like to view. This procedure is only meaningful if the analysis was previously run. Otherwise, some or all of the files listed may not exist. If a nonexistent file is selected, the text editor will appear, but no file is loaded.

To change the external text editor, select **Select Text Editor** from the **Options** menu. An open-file dialog box appears with the filter set for executable files (`*.EXE`). Find the executable file for the text editor you wish to use and click **Open**. The path to this file will be saved in the SPAM.INI file for future sessions. Notepad has a single document interface so that only a single file can be opened at a time. Repeatedly selecting files results in multiple Notepad applications running. Selecting a text editor with a multiple document interface allows you to open several files within the one editor.

If you are running several control files successively (multiple control files selected from the **Run Spam** open-file dialog box), a fatal error in any one file will stop the process until you click **OK** in the popup error dialog box. This is undesirable if you are running several analyses overnight. To turn off the popup error messages, select **Show Error Messages** from the **Options** menu. Error message popup boxes are shown if this menu item has a check mark and not shown if the check mark is absent. Repeatedly selecting **Show Error Messages** will toggle the state of the check mark. When unchecked, error message boxes are not shown and a fatal error will only abort the current analysis allowing SPAM to continue to the next control file. Error messages are always printed to the log file so that you can determine the cause of the error. After multiple analyses are run, an aborted analysis is easy to spot because the text will be red. Select the child window where the error occurred and choose the log file from the **Results** menu to view the error message. The state of the **Show Error Messages** check mark is saved in the SPAM.INI file so that any change you make will be saved between sessions.

Baseline Files

Baseline files, used in both estimation and simulation modes, describe the genetic and/or phenotypic makeup of each population that is potentially in the mixture. The types of characters that can be used are PHENOTYPE or MTDNA (haploid), LOCUS (diploid), or ISOLOCUS (tetraploid); they can be given as either relative or absolute frequencies for each population. The number of characters that can be described or the number of possible responses per character is not limited, but the maximum line length in baseline files is 512. This is particularly relevant when using microsatellites or other highly variable loci with a large number of alleles per locus. There is no limit other than the total line length. In a practical context, general guidelines have been developed that apply to both baseline and mixture files:

- Each locus should have a maximum of 100 possible alleles. If you have more, you should pool some.
- The total number of alleles over all loci should not exceed 1,000. If loci have 100 alleles each, you can have up to 10 loci (100*10=1,000). If they have 20 alleles each, you can have up to 50 loci (20*50=1,000).

Baseline data can be put in separate files for each population or combined into a single file. In a single baseline file, a population is identified by a pound sign (#) in the first column followed by its identifier number and then a name. The # delimiter is optional in the one-file-per-population baseline file, but the first non-commented line must contain the population name. The name in the file is matched with the population names defined in the control file. A comment line can be added to the baseline file, or a line of data can be commented out, by putting either a back slash (\) or a forward slash (/) in the first column of the line.

If a #, \, or / is not found in the first column, SPAM expects the line to contain baseline data. Each character occupies one record (line). The first field is the character identifier or name, followed by the sample size (only if relative frequencies are given), followed by the character frequencies. Integers, such as the character number, sample size, and character absolute frequencies, must be separated by a space. The relative frequencies do not need to be separated by a space because a decimal point is used to delimit values. If there is at least one decimal point in the record, SPAM expects relative frequencies. Otherwise, it will expect absolute frequencies. This check is performed on each record. The SPAM parser will search for as many frequencies as the number of types defined in the control file or until an end-of-line is reached. If an end-of-line is reached first, all remaining responses are assumed to be zero. Below is an excerpt from the Columbia single baseline file containing relative frequencies (COLUMB.BSE). This example shows the delimiter record that signifies the start of the Carson data. Note that integer character identifiers and character names can be mixed within the baseline file(s).

```
[...]
 6   200 0.9870.0130.0000.0000.0000.0000.0000.0000.000
 7   199 0.9370.0630.0000.0000.0000.0000.0000.0000.000
 8   200 1.0000.0000.0000.0000.0000.0000.0000.0000.000
 9   197 1.0000.0000.0000.0000.0000.0000.0000.0000.000
#5 Carson
 PMI   200 0.8480.1520.0000.0000.0000.0000.0000.0000.000
 TO1   199 0.7410.2590.0000.0000.0000.0000.0000.0000.000
 MDH34 200 0.9800.0200.0000.0000.0000.0000.0000.0000.000
 [...]
```

The baseline excerpt below contains the same data as absolute frequencies (COLUMB1.BSE). A sample size is not required because the data are counts.

```
[...]
 6  395   5   0
 7  373  25   0
 8  400   0   0
 9  394   0   0
#5 Carson
 PMI    339  61   0
 TO1    295 103   0
 MDH34  784  16   0
 [...]
```

With absolute frequencies, the sum of the frequencies is either two, four, or one times the sample size depending on whether the character is a locus, isolocus, or a phenotype (mtDNA). Consider the PMI locus above: the sum of the absolute frequencies for this character is 400, twice the sample size of 200 given with the relative frequencies, indicating that PMI must be a locus with two responses per individual. MDH34 is obviously an isolocus because the sum of the absolute frequencies is 800, four times the sample size.

The single baseline file for the Carson population (CARSON.FRQ) is shown in its entirety using relative frequencies.

```
#Carson
PMI   200 0.8480.1520.0000.0000.0000.0000.0000.0000.000
TO1   199 0.7410.2590.0000.0000.0000.0000.0000.0000.000
MDH34 200 0.9800.0200.0000.0000.0000.0000.0000.0000.000
IDH34 200 0.8860.0000.1140.0000.0000.0000.0000.0000.000
GL1   199 1.0000.0000.0000.0000.0000.0000.0000.0000.000
ADA1  199 0.9870.0130.0000.0000.0000.0000.0000.0000.000
LGG   190 0.9610.0390.0000.0000.0000.0000.0000.0000.000
LDH4  199 0.9820.0180.0000.0000.0000.0000.0000.0000.000
LDH5  200 0.9870.0130.0000.0000.0000.0000.0000.0000.000
```

Mixture File

Mixture files, used only in estimation mode, provide data on the suite of selected characters for each individual in the mixture sample. It is important to use proper sampling design so that the mixture sample is representative of the true mixture being estimated. SPAM has the flexibility to select a subset of all characters described in the baseline and mixture files that are used in an analysis, eliminating the need to create multiple input files for a reduced set of characters. Again, the number of characters is not limited, but the maximum line length for mixture files is 1,024. When SPAM reads a mixture file, all individuals are categorized by class and type as determined by their specific set of characters. Class is determined by the set of characters with information (e.g., which loci were measured), whereas type is determined by the value of those characters (e.g., which alleles are present). All individuals in a particular class have data on the same set of characters. If there are no missing characters, then all individuals belong to a single class. If, however, some individuals are missing data for character B, then they belong to a second class. Within a class, all individuals with identical expressions of the character suite are considered one type. SPAM tallies the number of individuals of each class and type.

Mixture files are best described by providing an example. Below is the first segment of the mixture file for the Columbia example (COLUMB.MIX):

```
* characters
 1   PMI
 2   TO1
 3   MDH34
 4   IDH34
 5   GL1
 6   ADA1
 7   LGG
 8   LDH4
 9   LDH5
* end
\
     011  110  400  400  200  200  200  200  200
     011  110  400  400  200  200  200  200  200
     011  110  400  400  200  200  200  200  200
     011  110  400  400  200  200  200  200  200
     020  020  400  400  200  200  200  200  200
 [...]
```

The first 11 records define the order of the characters in the mixture file and are optional. If they are omitted, SPAM assumes that the characters are in the order given in the control file. The names in the mixture file must match those in the control file because SPAM identifies the character by its name and not by the identification number. Identifying the order of the characters allows the user to try various character combinations in the analysis without regenerating the mixture file. In the Columbia example, one could examine the influence of a character on the mixture estimates by making two quick changes to the control file (i.e., comment out the character in the *CHARACTERS section and change the number of characters in the *PARAMETERS section) and then rerunning SPAM. No change is needed in the mixture file.

The mixture data begin after the backslash (\). In the example, the locus PMI for the first individual had a 2 and a 3 allele (heterozygote). A single-digit frequency is permitted for each allele. The locus frequencies must sum to two, isoloci to four, and a phenotype (or mtDNA) to 1. Note that the isoloci (the third and the fourth fields) each have a total of four alleles. Each record (line) represents a single individual.

Control File

For every SPAM analysis, a control file is required to define input and output options, as well as parameters to be used in the maximum likelihood search for contribution estimates. Within the control file, the user can identify the analysis type (**estimation** or **simulation**), select **options**, set **parameters**, define the **characters** used, specify the baseline **populations**, aggregate populations into larger **regions**, provide path and filename information for all input and output **files**, and instruct SPAM to **run** the analysis. Example control files for an estimation and a simulation are listed in Appendices 1 and 2.

A control file contains eight required sections, each identified by an asterisk (*) in the first column followed by a section keyword. The keywords are:

1) *`ESTIMATION` or *`SIMULATION`
2) *`OPTIONS`
3) *`PARAMETERS`
4) *`CHARACTERS`
5) *`POPULATIONS`
6) *`REGIONS`
7) *`FILES`
8) *`RUN`

For proper execution of SPAM, each section must appear in the control file and in the order given here. The section keywords can occur anywhere in the line following the asterisk and can be upper- or lowercase. Only the first four characters of a keyword are required (except for `RUN`). As shown in the example control files, other words can be mixed with the keywords to allow for more readability; however, care should be taken not to mix keywords within a line. SPAM will parse the control file line by line until it reaches a recognizable section label keyword. Once a keyword is encountered, all subsequent lines belong to that section until the next keyword is encountered. This means that, while lines within a section can be in any order, all lines belonging to one section should remain within that section.

Each section consists of a series of program control statements that specify the SPAM analysis. As with section labels, each control statement uses one or two keywords that are recognized by SPAM by their first four characters. The following text will cover each of these sections and their respective control statements in more detail. Examples of the various control file sections are provided in shaded text boxes and come from the estimation control file found in Appendix 1.

You can insert comment lines within a section by placing a backslash (\) or forward slash (/) in the first column. This is useful for adding your own comments to the control file or altering the analysis. For example, you can remove populations or characters from an analysis by placing comment delimiters at the beginning of appropriate lines, and then easily add them back for a subsequent analysis by removing the delimiters.

* Estimation/Simulation

```
* estimation: Columbia Example
```

The first section identifies the type of analysis that is to be run. If `*Estimation` is used, SPAM computes composition estimates from a mixture sample. If `*Simulation` is used, SPAM simulates a user-defined mixture scenario. No additional control statements are allowed in this section; however, a title for the analysis can be specified after the colon. This title is echoed in all output files. The major difference between running a simulation and an estimation is that the estimation requires a mixture file as input.

* Options

```
* options selected for optimization
  use IRLS algorithm in optimal search       :  f
  print mixture file                         :  t
  print baseline relative frequencies        :  t
  print conditional genotype probabilities   :  t
  print conditional population probabilities :  t
  print bootstrap estimates                  :  t
  print iterations                           :  t
  compute likelihood confidence intervals    :  t
  compute infinitesimal jacknife std.dev.    :  t
  resample mixture frequencies               :  t
  resample baseline                          :  t
```

This section is used to select performance and output options. The keywords for the various options are listed here.

Keyword(s)	***Default***	***Description***
`IRLS`	F	Use IRLS algorithm in optimal search
`PRIN`		Print…
`BASE`	F	baseline relative frequencies
`MIXT`	F	mixture file
`GENO`	F	conditional genotype probabilities
`POPU` or `STOC`	F	conditional population (stock) probabilities
`BOOT`	F	estimates from each bootstrap resample
`ITER`	F	MLE search iterations
`COMP`		Compute…
`CONF`	F	likelihood confidence intervals
`JACK`	F	infinitesimal jackknife standard deviations
`RESA`		Resample…
`MIXT`	F	mixture frequencies
`BASE`	F	baseline

To turn on and off each option, SPAM accepts `T`, `F`, `TRUE`, `FALSE`, `YES`, `NO`, `ON` and `OFF`. The parser is not case sensitive. The switch is always the first argument to the right of the colon. SPAM will use default values for any of the control statements that are not specified in the control file. A section label is still required if all defaults are to be used.

¾ IRLS ¾

SPAM uses three numerical algorithms in the search for the maximum likelihood estimates (MLEs): conjugate gradient (CG), iteratively reweighted least squares (IRLS), and expectation-maximization (EM).

The IRLS and CG algorithms are well documented by Pella et al. (1996). The conjugate gradient algorithm is preferred for its low memory requirements and speed. However, for problems such as the Columbia Example where there are few populations in the mixture and the mixture is quite large (1,597 individuals), the IRLS algorithm performs well. This is because the IRLS algorithm is based on Fisher's scoring algorithm, and the more individuals contributing to the Fisher's information matrix, the better the IRLS algorithm performs. The down side of the IRLS algorithm is its memory requirements. The information matrix requires p^2 storage locations, where p is the number of populations in the mixture, and the simplex tableau used in the quadratic programming (a technique used to constrain the estimates to sum to one), requires an additional $2p^2$ storage locations. All computations are done in double precision, so a single storage location requires eight bytes of memory.

If loci are missing in the baseline, SPAM performs the MLE search using a generalization of the EM algorithm (GEM). In this case, SPAM initializes the missing allele frequencies to be uniform across all alleles for each missing locus. That is, each allele's relative frequency is set to $1/a$, where a is the number of alleles. In addition to the typical maximization step used in the EM algorithm, SPAM computes the missing allele frequencies utilizing the "expected" number of individuals with each genotype from each population at that stage in the iterative process. The process cycles until the missing baseline frequencies no longer change beyond 10^{-8}. If a stopping criterion is not reached at this point, SPAM stops updating the baseline frequencies and either the CG or IRLS algorithm takes over. This is a slight variation of Smouse et al. (1990), who used a similar algorithm to this to perform unconditional MLE estimation, thereby updating the entire baseline with information from the mixture sample. See Appendix 13 for mathematical details.

¾ PRINT ¾

The printing switches allow the user to create output files for the condensed mixture, baseline frequencies, conditional genotype and population probabilities, bootstrap estimates, and MLE search information. These switches require two keywords: the first is PRIN and the second can be MIXT, BASE, GENO, POPU (or STOC), BOOT, or ITER.

A complete baseline file (BASE) can be generated by SPAM with relative frequencies for each population and character. If a character is missing from the baseline for a population, SPAM will include these parameters in its maximum likelihood search and include these estimated frequencies in the baseline output file.

The condensed mixture (MIXT) shows each unique type in the mixture and its frequency of occurrence grouped by class. The file is considered to be condensed because each individual in the mixture is no longer listed separately, but grouped by common types.

The conditional genotype probabilities for each population (GENO) can be printed as well as the conditional population probabilities for each genotype (POPU). The former are computed using the baseline frequencies with the assumption that all populations are in Hardy–Weinberg (HW) equilibrium. It is therefore imperative that each population be tested for HW equilibrium before it is included in the baseline. The latter probabilities are calculated from Bayes' theorem using the conditional genotype probabilities and the final mixture estimates.

In a bootstrap or simulation, stock contribution estimates for each bootstrap resample can be output to a file (BOOT). This information is useful for examining the distribution of estimates or other analyses. The estimates can also be combined into different reporting groups without having to repeat the full analysis.

An output file can also be created that contains iteration information from the search for the maximum likelihood estimates (ITER). This information may be used for evaluating the performance of the search algorithm in finding the true maximum.

These files are described further in the chapter on Output Files.

¾ COMPUTE ¾

In addition to maximum likelihood mixture estimation, SPAM will perform two other computations at the user's request: infinitesimal jackknife standard errors for the mixture estimates (Millar 1987) and likelihood-based confidence intervals for aggregated regional estimates.

The likelihood (log likelihood) confidence intervals (CONF) based on the likelihood ratio statistic are computed only if individual populations are aggregated into regions. That is, SPAM will not compute confidence intervals for individual population contributions unless they are defined as a region composed of a single population. The procedure sequentially computes confidence intervals for each region specified under the REGION command (see the *REGIONS section for a full discussion on aggregating populations). For each confidence interval, the individual populations are assorted into two groups such that the first group contains all populations in the specified region and the second contains all other populations. SPAM then searches for the upper and lower group contributions that make the probability of the likelihood ratio statistic equal to the desired confidence level (0.90 for a 90% confidence interval). Note that the negative of two times the log of the likelihood ratio statistic has an asymptotic distribution of chi-square with one degree of freedom. For each step in the search for the two bounds of the confidence interval, a MLE search for the individual population estimates is performed such that each group is constrained to sum to its trial value.

If the infinitesimal jackknife estimates are requested (JACK) and aggregated regions are specified, SPAM computes confidence intervals for each region using the Normal (Gaussian) distribution. The infinitesimal jackknife estimates give the variance of the regional proportions due to mixture sampling error only; they assume the baseline genotype frequencies are known without error.

¾ RESAMPLE ¾

The RESAMPLE option performs parametric bootstrapping of the mixture (MIXT) and/or the baseline (BASE). In a simulation, a mixture is generated using the baseline frequencies and the HW principle. The baseline can be parametrically regenerated on each simulation iteration if the resample baseline option is turned on. If both the baseline and the mixture are to be resampled in a simulation, the mixture is parametrically resampled using the resampled baseline estimates. Unconditional confidence intervals of region proportions can be estimated by resampling both the baseline and the mixture.

In resampling isoloci, SPAM treats the data as equal frequencies at two disomic loci, which is equivalent to equal frequencies at a single tetrasomic locus. Bootstrapping the baseline is permitted even when baseline frequencies are missing.

*** Parameters**

The control parameters specify the number of populations and characters in the analysis, upper limit parameters, and tolerances to control the optimization search.

```
* control parameters
  number of populations in analysis  :  14
  number of characters in analysis   :   9
  maximum number of genotypes        : 200
  maximum number of classes          :  20
  maximum # of iterations            : 300
  maximum number of missing loci     :  4
  estimate tolerance                 : .1E-3
  likelihood tolerance               :  1.0e-10
  genotype tolerance                 :  1.0e-6
  algorithm switch tolerance         :  0.01
  GPA                                :  90
  number of resamplings              :  100
  confidence intervals               :  90
  random seed                        :  -718805
```

The keywords for the *PARAMETERS command are listed here.

Keyword(s)	***Default***	***Description***
`NUMB`		Number of…
`POPU` or `STOC`	-	populations (stocks) in the analysis
`CHAR`	-	characters in the analysis
`RESA`	100	bootstrap resamplings
`MAXI`		Maximum number of…
`GENO`	100	genotypes
`CLAS`	1	classes
`ITER`	100	iterations
`MISS`	0	missing (unscored) loci in mixture
`TOLE`		Tolerances for…
`ESTI`	10^{-4}	estimates
`LIKE` or `FUNC`	10^{-10}	likelihood (function)
`GENO`	10^{-10}	genotype probability
`ALGO` or `SWIT`	10^{-2}	algorithm switch (CG to IRLS)
`GUAR`, `PERC`, or `GPA`	90	Guaranteed percent achievement of the maximal likelihood (GPA)
`CONF`	90	Confidence interval size (percent)
`SIZE`	100	Simulation sample size
`SEED`	*From CPU clock*	Random number generator seed

¾ *NUMBER* ¾

Using the NUMBER keyword, the user specifies the number of populations in the baseline (POPU), the number of characters to be used in the analysis (CHAR) and, if bootstrapping, the number of iterations for the bootstrap (RESA). The first two parameters do not have defaults and must be specified by the user. SPAM dynamically allocates the computer memory at run-time so the number of populations or characters in the analysis are not limited beyond the computer's capability and the line limits for mixture and baseline files. Avoid adding too many loci (characters) in the analysis. Because the multilocus probability is the product of each individual locus genotype probability, as characters are added to the analysis the number of possible multilocus genotypes increases, the probability of each multilocus genotype decreases, the expected number of individuals that possess that genotype decreases, and the number of classes tends to increase. Again, all computations are done in double precision. Because setting the number of resamples to zero or one will result in an error and abort the analysis, turn off bootstrapping with the RESAMPLE keywords in the *OPTIONS section, not by setting the number of resamples to zero. There is no upper limit to the number of resamples, but consider the computation time required for larger numbers. A minimum number of 1,000 is recommended.

¾ MAXIMUM ¾

The MAXIMUM keyword is used to set upper bounds on the number of genotypes (GENO), classes (CLAS), iterations used in the optimization search (ITER), and the number of unscored characters in the mixture (MISS). MISS is used to filter individuals out of the mixture that are missing too much information. This helps to minimize the number of classes. All the MAXIMUM parameters have defaults. The default for the number of missing loci is 0, with valid input ranging from 0 to the number of loci minus 1. Any number less than 0 will be set to 0, and any number greater than or equal to the number of loci will be set to the number of loci minus 1. SPAM will not recognize individuals that are missing all characters. If the maximum number of iterations is exceeded, SPAM will stop searching and print a warning in the log file.

¾ TOLERANCE and GPA ¾

The TOLERANCE and GPA (also GUARANTEED or PERCENT) keywords allow the user to influence the performance of SPAM's search for the MLEs through the various stopping criteria of the search. The first stopping criterion (GPA, GUAR, or PERC) is the guaranteed percent achievement of the maximal likelihood, GPA (Pella et al. 1996). This value should be in the range of 0 to 100. Generally, 70, 80, or 90 percent are good choices. The second stopping criterion is the estimate tolerance (ESTI). If the maximum change in the estimates for an iteration is less than this value, search is terminated. The third stopping criterion is the function or likelihood tolerance (FUNC or LIKE). As with the estimate tolerance, if the change in the likelihood is less than the specified tolerance the search is terminated. To turn off a particular stopping criterion, set its value to the extreme. That is, set the likelihood tolerance to an extremely small number (.1E-100, or 0), or set GPA to 100. SPAM notifies the user of the termination type in the log file.

If the user requests the IRLS algorithm (in the *OPTIONS section), SPAM uses the CG algorithm initially until the maximum estimate update is less than the algorithm switch tolerance (SWIT). At that point SPAM switches to the IRLS algorithm.

The genotype probability tolerance (GENO) allows the user to filter genotypes that cannot be adequately explained by the baseline and are considered impossible by SPAM. As mentioned before, adding more (polymorphic) loci to the MSA decreases the multilocus genotype probabilities. For example, for 5 di-allelic loci, each with a low-frequency allele of probability 0.10, the probability of an “aabbccddee” genotype is 1E-10; for 10 di-allelic loci the probability reduces to 1E-20. Hence, the tolerance used will depend on the problem at hand. If genotype probabilities exist that are very small in relation to most of the others, SPAM’s performance deteriorates. This is especially true when working with microsatellites or other highly variable loci. The user should ensure that adequate estimation of baseline allele frequencies has been made prior to performing analyses with SPAM. Smouse and Chevillon (1998) reviewed the analytical aspects of population-specific discrimination for individuals. They concluded that a modest number of codominant loci, each with a small number of alleles with each allele in moderate frequency, provides the highest potential discriminatory power. These guidelines are also applicable to the admixture analyses performed by SPAM.

Genotypes declared impossible are identified in the condensed mixture output by an asterisk. If impossible genotypes exist, the mixture estimates will not sum to one, but to the proportion of individuals in the mixture that possess explainable genotypes. In this case, an “Unknown” category is printed in the mixture estimate output with the proportion of unexplainable genotypes. Impossible genotypes can also occur in simulation analyses, and cause an apparent decrease in simulated sample size. In some situations with highly variable loci and a large genotype probability tolerance, all individuals in a simulated mixture may be declared impossible and the analysis aborts. In this case, a parameter value of 0 can be used to ensure that no types are found impossible.

If a simulation is run and the RESAMPLE BASELINE option is on, it is possible to generate genotypes that have a probability of zero. This occurs when rare alleles are lost in the regeneration of the baseline frequencies, but are randomly generated in the mixture. The user will find a comment to this effect in the log file should this occur.

¾ CONFIDENCE INTERVALS ¾

The confidence interval can be specified using the CONFIDENCE keyword; 90% confidence intervals are the default. An error message is generated if this is less than or equal to 0 or greater than or equal to 100. This parameter is used for the normal and likelihood confidence intervals as well as the bootstrap confidence intervals.

¾ SAMPLE SIZE ¾

The SIZE keyword is used to set the sample size for the mixtures generated in a simulation analysis. Along with the simulated mixture contributions for each population, this is useful in deciding the appropriate number of individuals to sample from a mixture to achieve desired estimation results for a given baseline.

¾ SEED ¾

The SEED keyword allows the user to specify a seed for the random number generator for reproducible results. The seed takes a negative value between -2,147,483,648 and 0. If a positive seed is given, it will be changed to negative. Therefore, the seed 12,345 will give the same sequence of numbers as -12,345. The seed will be printed in the bootstrap and/or simulation output files.

If a random seed is not declared in the control file, SPAM will generate a seed based on the current CPU time. The seed will be printed in the bootstrap and/or simulation output files for reference.

* Characters

The information in the *CHARACTERS section defines each character, its identification number, its type, and the maximum number of outcomes each character can realize. The number of active characters listed here (i.e., not commented out) cannot be more than the number of characters given in the *PARAMETERS section. If so, an error is generated and the analysis will abort. If there are fewer characters defined than specified in the *PARAMETER section, the default values are applied to all undefined characters.

```
* characters used
    [id #] [# types] [kind] [character]
      1        3       locus     PMI
      2        3       locus     TO1
      3        3      isolocus   MDH34
      4        3      isolocus   IDH34
      5        3       locus     GL1
      6        3       locus     ADA1
      7        3       locus     LGG
      8        3       locus     LDH4
      9        3       locus     LDH5
```

The first line after the *CHARACTERS keyword determines the order in which the character attributes will be given. The attribute labels are listed here.

Label	***Default***	***Attribute***
`[ID #]`	*see text*	Identification number for the character
`[# TYPES]`	9	Maximum number of alleles/types the character can realize
`[KIND]`	`locus`	Character type (locus, isolocus, phenotype, mtdna)
`[CHARACTER]`	*see text*	Name of the character

The identification number is used to identify the characters and must match those in the baseline. If it is not provided, SPAM sequentially assigns identification numbers to the characters from one to the number of characters. These assigned numbers also must match the baseline file(s), so if you let SPAM assign them, be sure the characters in the control file are listed in the same order they appear for all the populations in the baseline.

The number of TYPES for each character is the maximum number of outcomes that can be realized by the character (e.g., the number of possible alleles at a locus). Hence, the baseline will have at most this many frequencies for the character, and the character genotype defined in the mixture will have no more than this many adjacent digits. If there are more types listed in either the baseline or the mixture, they are ignored. If there are less, SPAM sets the remaining undefined types to zero. The default number of types is nine with no upper limit other than line length in the baseline and mixture files.

The kind of character (KIND) is identified by keyword LOCUS, ISOLOCUS, PHENOTYPE, or MTDNA. A locus has two responses per individual (diploid) and an isolocus has four (tetraploid). Phenotype and mtDNA characters have one response per individual (haploid). The keywords PHENOTYPE and MTDNA can be used interchangeably. Both are provided merely for the convenience of the user. If the KIND attribute is not provided, the character type defaults to LOCUS.

If the character name or label is not provided, SPAM will label each character as “Char- *x*,” where *x* is a sequential integer.

* Populations

The information in the *POPULATIONS section defines the identification number, population names, baseline files, and the regional aggregation of populations.

```
* populations used in analysis
 [id #]  [population]           [file]        [lev1] [lev2]
   1      Warm Springs         : warm.frq     :  2       1
   2      Rapid                : rapid.frq    :  4       2
   3      Kooskia              : kooskia.frq  :  4       2
   4      Round Butte          : round.frq    :  5       3
   5      Carson               : carson.frq   :  3       1
   6      Eagle                : eagle.frq    :  1       1
   7      Little White Salmon  : white.frq    :  4       2
   8      South Santian        : santian.frq  :  5       3
   9      Oakridge             : oakridge.frq :  5       3
   10     Kalama               : kalama.frq   :  1       1
   11     Cowlitz              : cowlitz.frq  :  1       1
   12     McKenzie             : mckenzie.frq :  5       3
   13     Levenworth           : leven.frq    :  3       1
   14     Klickitat            : klickit.frq  :  2       1
```

The first line after the *POPULATIONS keyword defines the order in which the population attributes will be provided. The attribute labels are listed here.

Label	***Default***	***Attribute***
`[ID #]`	*see text*	Identification number for the population
`[POPULATION]`	*see text*	Population name
`[FILE]`	*see text*	Baseline file
`[LEV1]`	-	Level 1 regional identifier
`[LEV2]`	-	Level 2 regional identifier
`[LEV3]`	-	Level 3 regional identifier
`[ESTIMATE]`	*see text*	Initial estimates (estimation) or relative population contribution to mixture (simulation)

The population identification number is used only if a single baseline file is used. If the identification number is not provided, SPAM assigns sequential integers from one to the number of populations in the single file baseline. If the population name is not provided, SPAM labels each population “Population *x*,” where *x* is an integer.

If the baseline is stored in individual files, one for each population, then the FILE label is required. In the example control file, the population name and the file name are delimited by a colon. The SPAM parser uses the colon to delimit character strings, after which it will continue parsing for the next attribute if one exists. If the population name is the last attribute provided, the colon is not required.

Contribution estimates are always computed for each population in the baseline, but as previously mentioned, the user is frequently interested in estimates for larger aggregations of populations. These aggregations can be defined in this section. A maximum of three levels of aggregated regions can be defined with the first identified by LEV1, the second by LEV2, and the third by LEV3. Integer values are used to identify regions and must be positive, but do not need to be sequential. A zero excludes a population from that level. The defined regions in one level do not have to be nested within the previous level, although strict hierarchies are certainly allowed. If the levels are properly nested, their identifiers must be unique within that level. SPAM does not check for proper nesting and views each level simply as a set of classification identifiers. Further discussion on the statistics computed for regions is discussed in the *REGIONS section.

The total number of regions that can be defined between levels 1, 2, and 3 is equal to the total number of populations, regardless of whether a region is defined identically in more than one level. The sum of the number of regions in levels 1, 2, and 3 cannot exceed this limit. A contribution estimate for each population is provided in the output without the need to be defined as its own region, therefore defining regions is best limited to combining two or more populations.

When performing an estimation analysis, the ESTIMATE identifier is used to set the initial contribution estimates of the mixture. If it is not provided, the starting values in the MLE search are $1/p$, where p is the number of populations defined under the *PARAMETERS command. It is sometimes useful to try various starting values to verify that the same contribution estimates are obtained, providing evidence that the true maximum likelihood is found and not just a local maximum. For a simulation analysis, the ESTIMATE identifier defines the true mixture that is generated stochastically using the baseline frequencies. Values for ESTIMATE do not have to sum to one, and can be on any convenient scale. For example, both of the following listings would result in 10%, 20%, 30%, and 40% for populations A, B, C, and D, respectively.

```
* populations
[population]   [lev1]   [estimate]
  Pop A          1          .1
  Pop B          1          .2
  Pop C          2          .3
  Pop D          2          .4
```

```
* populations
[population]   [lev1]   [estimate]
  Pop A          1          1
  Pop B          1          2
  Pop C          2          3
  Pop D          2          4
```

ESTIMATE values for simulations can be further adjusted by entering ESTIMATE values in the *REGIONS section as described below.

* Regions

If aggregated regions are defined, the *REGIONS section permits the user to label them. SPAM computes regional estimates by summing the appropriate estimates from the individual populations and then computing variances and correlations at the regional level. It must be noted that SPAM blindly computes the covariances between each region even if one region is nested within another. This is done simply out of computational convenience, but does offer the advantage that correlations for several combinations of groups can be obtained.

```
* regions
 [level] [label] [Region]
    1       1     Lower Columbia
    1       2     Mid Columbia
    1       3     Upper Columbia
    1       4     Snake River
    1       5     Other
    2       1     Columbia
    2       2     Snake River
    2       3     Other
```

Below are the attribute labels for the *REGIONS section.

Label	***Defaults***	***Attribute***
`[LEVEL]`	-	Integer denoting the regional level (defined in the *POPULATIONS section)
`[LABEL]`	*see text*	Integer denoting the region identifier (defined in the *POPULATIONS section)
`[REGION]`	*see text*	Name of the region
`[ESTIMATE]`	-	Regional contribution to the mixture for a simulation run

LEVEL refers to the regional level defined in the *POPULATIONS section. The permissible values for LEVEL depend on the number of regional aggregation levels defined. Entries can be 1 if LEV1 is defined, 2 if LEV2 is defined, and 3 if LEV3 is defined. The LABEL is the integer value used to identify the region in the *POPULATIONS section, with defaults from one to the number of regions. REGION defines the name of region and is echoed on the output. If missing, each region is labeled by "Region *x*," where *x* is a sequential number from 1 to the number of regions defined in the *POPULATIONS section.

The ESTIMATE identifier is specifically designed for simulation studies where one is interested in investigating the identifiability of a region using a given baseline or the accuracy and precision of the contribution estimates for various mixture scenarios and sample sizes. The combination of ESTIMATE values from the *POPULATIONS and the *REGIONS sections determine the true population contributions for the simulated mixtures. If ESTIMATE values are not given in the REGIONS sections, then the contributions are determined solely from the *POPULATIONS sections as described above. If both are given, the population ESTIMATE values become relative contributions within the regional ESTIMATE values. This is perhaps best described with an example from the simulation control file in Appendix 2.

```
* populations
 [population]            [lev1]  [estimate]
  Warm Springs         :   2        10
  Rapid                :   4        20
  Kooskia              :   4        10
  Round Butte          :   5        10
  Carson               :   3        10
  Eagle                :   1        20
  Little White Salmon  :   4        10
  South Santian        :   5        10
  Oakridge             :   5        10
  Kalama               :   1        50
  Cowlitz              :   1        10
  McKenzie             :   5        10
  Levenworth           :   3        10
  Klickitat            :   2        10

 * regions
   [level] [label]  [region]         [estimate]
      1       1      Lower Columbia  :    1
      1       2      Mid Columbia    :    0
      1       3      Upper Columbia  :    0
      1       4      Snake River     :    0
      1       5      Other           :    0
```

Here we simulate a 100% Lower Columbia mixture, specified in the ESTIMATE column in the *REGIONS section. In the *POPULATIONS section, the three populations within region 1 have estimates of 20, 50, and 10. The true mixture will then be 20/80=25% Eagle, 50/80=62.5% Kalama, and 10/80=12.5% Cowlitz. Note that the 20:50:10 ratio is maintained. If we had simulated a mixture of 50% Lower Columbia and 50% some other region, then the percentages for the true mixture would be reduced by half (12.5%, 31.25%, and 6.25%) with the remaining 50% coming from the other region. ESTIMATE values are particularly convenient for adding data such as relative abundances.

When more than one level is specified in a simulation, the true contributions for the simulation are derived from level 1. Because the parser will look for ESTIMATE values for all levels even though they are only used for level 1, always include placeholder entries for levels 2 and 3.

* Files

The *FILES section permits the user to name the files associated with an analysis. You may list a single baseline file (usually identified by a .BSE extension), or the path for the individual population baseline files defined in the *POPULATIONS section. The mixture file (usually identified with a .MIX extension) is listed here when performing an estimation. The root filename for all output files is also included here.

```
* files
 path      : d:\spam32\columbia\baseline
 mixture   : d:\spam32\columbia\mixture\columb.mix
 output    : d:\spam32\columbia\output\estimation\columb
```

Below are the keywords for the *FILES section.

Keyword	***Default***	***Description***
`BASELINE`	-	Path and filename for single baseline file
`PATH`	-	Path for individual population baseline files
`MIXTURE`	-	Path and filename for mixture file (estimation only)
`OUTPUT`	-	Path and root filename for output files

If both the individual baseline files and the single baseline file are specified, the single baseline file listed in the *FILES command is used and the PATH is ignored.

The OUTPUT keyword should be followed by a root filename, including the path, that will be used for all output files created by SPAM. Although any file extension is allowed on the input files (baseline, mixture, and control), SPAM uses specific extensions for each output file. Possible output files with their extensions are listed here, and output files are described in the next section.

`.LOG` - Log file
`.EST` - Mixture estimates, standard errors, correlations and confidence intervals
`.SIM` - Simulation output
`.ITR` - Iterations information
`.BOT` - Bootstrap output
`.RSM` - Resample output
`.BSL` - Baseline relative frequencies
`.CMX` - Condensed mixture
`.GEN` - Conditional genotype probabilities
`.POP` - Conditional population probabilities

Because output files are the only means of obtaining output from SPAM, an error will be generated if no output filename is provided. Also, if an output filename is given that contains directories that do not exist, they will be created for you rather than giving an error message. This saves you from having to set up the output directory structure before running the analysis. One word of caution: ***If you specify an output path and root filename for files that already exist, they will be overwritten.***

* Run

The *RUN section notifies SPAM to begin execution. Once this command is interpreted, SPAM stops parsing the control file, closes it, proceeds to read the baseline, followed by the mixture, and begins the estimation or simulation analysis. No other control statements are permitted in the *RUN section.

Output Files

All results from a SPAM analysis are printed to a collection of ASCII text files that can be viewed through the SPAM environment or separately with any text editor. The set of files created depends on the analysis requested in the control file. All files, with the exception of the resampled estimates file, are formatted for convenient viewing and printing. Every SPAM analysis will produce a log file and either an estimation file or simulation file, depending on the type of analysis run. A brief description of each output file is included here, along with examples in the Appendices.

Log (`*.LOG`) Appendix 3

Every SPAM analysis generates a log file containing a list of the steps undertaken and any errors encountered. This file should be reviewed to make sure the estimation procedure converged properly. The log file uses the same path and root filename as the control file since the log file is initiated before the control file is parsed.

Estimation (`*.EST`) Appendix 4

The estimation file is the primary output from an estimation analysis. Here the user will find contribution estimates with jackknife standard errors (if requested) for each population in the baseline and all reporting regions specified. Infinitesimal jackknife covariance and correlation matrices are output for each reporting region. Normal and likelihood confidence intervals are also output in this file. Some terms used in the estimation output are defined here.

Number of types: the total number of distinct genotypes in the mixture. Missing loci are considered when defining genotypes.

Number of observations: the total number of individuals in the mixture after possible exclusions have been made due to missing loci.

Classes: the number of classes in the mixture. Further, the number of types and individuals per class are tallied.

Score: When a closed-form solution is possible with maximum likelihood, estimates are obtained by taking the partial derivatives of the likelihood function (also called scores), setting them to zero, and then solving simultaneously for the parameters. Here, we take our parameter estimates and plug them into the partial derivatives. If the estimation procedure is good, we should get values close to zero, or at least not very large. As an indication of how good the estimates are, you can check to make sure there are no large values of score by checking the largest one, which is marked with an asterisk. Most commonly these will be around 0.1 or 0.2. Scores with negative values are not a concern.

Simulation (`*.SIM`) Appendix 5

The simulation file is the primary output from a simulation analysis. Mean bootstrap contribution estimates with bootstrap standard deviations are given for each population in the baseline and all reporting regions. Bootstrap covariance and correlation matrices are output for each reporting level. Percentile bootstrap confidence intervals are also output from the distribution of contribution estimates generated from the replicate simulated mixtures. A warning about possible inadequacies in the confidence intervals is printed if fewer than 100 resamples are performed. Some terms used in the simulation file are defined here.

Mean estimate: the mean of the contribution estimates from all the simulations.

Expected: the true simulated contributions as specified in the *POPULATIONS and *REGIONS sections of the simulation control file.

Root MSE: the square root of the mean square error. This is a measure of simulation performance incorporating both the standard deviation and bias (Estimate-Expected) of an estimate.

Angle deviation and *Distance*: tools for evaluating simulation results. The mean angle (in radians) and distance between the true contributions and the estimated contributions are given for different aggregation levels (Wood et al. 1987). The first one through *n* levels listed are for the *n* aggregation levels, and the last is for the individual populations with no aggregation.

Iterations (`*.ITR`)Appendix 6

Information regarding the maximum likelihood search is output to this file, which can be used for diagnostic purposes or evaluating overall performance. This file can be created for all SPAM estimation analyses. Due to the amount of information it contains, it is impractical to create this file for most simulation analyses. Therefore, it is not available for simulations with more than 10 resamples.

Bootstrap (`*.BOT`) Appendix 7

If a bootstrap is performed in an estimation analysis, this file gives similar output as that found in a simulation file. Mean bootstrap contribution estimates with bootstrap standard deviations are given for each population in the baseline and all reporting regions. Bootstrap covariance and correlation matrices are output for each reporting level. Percentile bootstrap confidence intervals are also output from the distribution of contribution estimates generated from the replicate resamples. See explanations for Root MSE, angle deviation, and distance in the Simulation output description. Some other terms used are defined here.

Mixture resampling goodness of fit: an overall test of whether the mixture resamples come from the same distribution of types as the true mixture. A chi-square test is used with the true number of individuals of each type in the mixture as the expected and the mean number of each type from all resamples as the observed.

Mean estimate: the mean of the bootstrap contribution estimates from all the resamples of the baseline and/or mixture.

Expected: the contribution estimates from the true baseline and mixture. These values match the estimation output.

Resamples (`*.RSM`) Appendix 8

When running a simulation or a bootstrap, a user may wish to output the contribution estimates for every population in the baseline for each resample. Standard output only gives the mean and upper and lower confidence bounds. The resample output file can be used with other analytical software to examine distributions or generate estimates for new reporting regions without rerunning the analysis.

The first two lines of the resampled estimates file specify the number of populations in the analysis and the number of resamples taken followed by a list of population names. The estimates are then given with as many columns as there are populations and as many rows as resamples. If there are more than 100 populations, the line will wrap so that there could be two or more rows per resample.

Baseline (`*.BSL`) Appendix 9

Baseline frequencies are printed in the baseline file with the locus (or character) across the top and population down the left-hand side of the output for easy population-by-population comparisons. A missing locus for a population is identified by a sample size of zero and “dashed-out” frequencies. Immediately below this line, the number of individuals in the mixture sample expected to be from the population as determined from the GEM and the corresponding allele frequencies is shown. The sample baseline file includes an example where the PMI locus was removed from the Rapid Creek and Eagle baselines. By comparison with the known frequencies, it can be seen that the GEM does quite well for Rapid Creek, a large contributor to the mixture, but poorly for Eagle, a low contributor to the mixture.

Mixture (`*.CMX`) Appendix 10

The condensed mixture file lists all unique types found in the mixture grouped by class and the number of times each type occurred. Impossible types defined in the *PARAMETERS section are flagged in the mixture file with an asterisk.

Conditional Genotype Probabilities (`*.GEN`) Appendix 11

Based on the baseline frequencies and assuming Hardy–Weinberg equilibrium, conditional probabilities are calculated for each type found in the mixture. The probabilities are presented in matrix form with unique types in rows and populations in columns. Entry P_{ij} is the probability that an individual is type i given that it belongs to population j.

With the conditional genotype probabilities and the contribution estimates for each population, SPAM calculates conditional population probabilities from Bayes' rule. Presented in the same matrix format, entry P_{ij} is the probability that an individual belongs to population j given that it is type i.

Correspondence

We welcome correspondence regarding SPAM. If you would like to be included on the mailing list and receive notifications of updates, please contact us at the address below. Please report any bugs as soon as possible so we can assess the problem and make any necessary corrections to the program.

Joel Reynolds – Joel_Reynolds@fishgame.state.ak.us
Bill Templin – Bill_Templin@fishgame.state.ak.us
Lisa Seeb – Lisa_Seeb@fishgame.state.ak.u

Alaska Department of Fish and Game
Division of Commercial Fisheries
Gene Conservation Laboratory
333 Raspberry Road
Anchorage, Alaska 99518
USA

Acknowledgments

SPAM was developed with the effort and support of many people. We thank Penny Crane, Bill Templin, Craig Busack, Robin Waples, and Dick Wilmot for their assistance in making SPAM a reality and the many users who offered suggestions and identified improvements. We thank Fred Utter for providing a vision of SPAM long before the first line of code was ever written.

Funding was provided by the *Exxon Valdez* Trustee Council, Project 9x255, and the State of Alaska.

Limited Warranty and Disclaimer

This software and accompanying written materials (including instructions for use) are provided “as is” without warranty of any kind. Further, Alaska Department of Fish and Game (ADF&G) does not warrant, guarantee, or make any representations regarding the use, or the results of use, of the software or written materials in terms of correctness, accuracy, reliability, currentness, or otherwise. The entire risk as to the results and performance of the software is assumed by you. If the software or written materials are defective, you, and not ADF&G or its employees, assume the entire cost of all necessary servicing, repair, or correction.

The above is the only warranty of any kind, either express or implied, including but not limited to the implied warranty of fitness for a particular purpose, that is made by ADF&G. No oral or written information or advice given by ADF&G or its employees shall create a warranty or in any way increase the scope of this warranty and you may not rely on any such information or advice.

Neither ADF&G nor anyone else who has been involved in the creation, production or delivery of this product shall be liable for any direct, indirect, consequential or incidental damages (including damages for loss of business profits, business interruption, loss of business information, and the like) arising out of the use or inability to use such product even if ADF&G has been advised of the possibility of such damages.

Use of this product for any period of time constitutes your acceptance of this agreement and subjects you to it contents.

Literature Cited

Masuda, M., S. Nelson, and J. Pella. 1991. The computer programs for computing conditional maximum likelihood estimates of stock composition from discrete characters. U.S. Department of Commerce, NOAA/NMFS Auke Bay Laboratories, Juneau, Alaska.

Millar R. B. 1987. Maximum likelihood estimation of mixed stock fishery composition. Canadian Journal of Fisheries and Aquatic Sciences 44:583-590.

Pella, J. J., M. Masuda, and S. Nelson. 1996. Search algorithms for computing stock composition of a mixture from traits of individuals by maximum likelihood. U.S. Department of Commerce, NOAA/NMFS Technical Memo. NMFS-AFSC-61.

Smouse, P. E., R. S. Waples, and J. A. Tworek. 1990. A genetic mixture analysis for use with incomplete source population data. Canadian Journal of Fisheries and Aquatic Sciences 47:620-634.

Smouse, P.E., and C. Chevillon. 1998. Analytical aspects of population-specific DNA fingerprinting for individuals. Journal of Heredity 2:143-150.

Wood, C. C., T. J. Mulligan, and D. A. Fournier. 1987. Stock identification with the maximum-likelihood mixture model: sensitivity analysis and application to complex problems. Canadian Journal of Fisheries and Aquatic Sciences 44:866-881.

Appendices

EXAMPLE SPAM FILES FOR THE COLUMBIA RIVER MIXED STOCK FISHERY

Control Files

Output Files

MISCELLANEOUS

Appendix 1: Estimation control file

```
* estimation: Columbia Example

* options selected for optimization
  use IRLS algorithm in optimal search       :  f
  print mixture file                         :  t
  print baseline relative frequencies        :  t
  print conditional genotype probabilities   :  t
  print conditional population probabilities :  t
  print bootstrap estimates                  :  t
  print iterations                           :  t
  compute likelihood confidence intervals    :  t
  compute infitesimal jacknife std.dev.      :  t
  resample mixture frequencies               :  t
  resample baseline                          :  t

* control parameters
  number of populations in analysis  :  14
  number of characters in analysis   :   9
  maximum number of genotypes        : 200
  maximum number of classes          :  20
  maximum # of iterations            : 300
  maximum number of missing loci     :  4
  estimate tolerance                 : .1E-3
  likelihood tolerance               :  1.0e-10
  genotype tolerance                 :  1.0e-6
  algorithm switch tolerance         :  0.01
  GPA                                :  90
  number of resamplings              :  100
  confidence intervals               :  90
  random seed                        :  -718805

* characters used
    [id #] [# types] [kind] [character]
      1        3       locus     PMI
      2        3       locus     TO1
      3        3      isolocus   MDH34
      4        3      isolocus   IDH34
      5        3       locus     GL1
      6        3       locus     ADA1
      7        3       locus     LGG
      8        3       locus     LDH4
      9        3       locus     LDH5

* populations used in analysis
    [id #]  [population]           [file]            [lev1] [lev2]
     1      Warm Springs         : warm.frq        :   2       1
     2      Rapid                : rapid.frq       :   4       2
     3      Kooskia              : kooskia.frq     :   4       2
     4      Round Butte          : round.frq       :   5       3
     5      Carson               : carson.frq      :   3       1
     6      Eagle                : eagle.frq       :   1       1
     7      Little White Salmon  : white.frq       :   4       2
     8      South Santian        : santian.frq     :   5       3
     9      Oakridge             : oakridge.frq    :   5       3
     10     Kalama               : kalama.frq      :   1       1
     11     Cowlitz              : cowlitz.frq     :   1       1
     12     McKenzie             : mckenzie.frq    :   5       3
     13     Levenworth           : leven.frq       :   3       1
     14     Klickitat            : klickit.frq     :   2       1

* regions
 [level] [label] [Region]
    1       1     Lower Columbia
    1       2     Mid Columbia
    1       3     Upper Columbia
    1       4     Snake River
    1       5     Other
    2       1     Columbia
    2       2     Snake River
    2       3     Other

* files
 path      : d:\spam32\columbia\baseline
 mixture   : d:\spam32\columbia\mixture\columb.mix
 output    : d:\spam32\columbia\output\estimation\columb

* run
```

Appendix 2: Simulation control file

```
* simulation: Columbia Example

* options selected for optimization
  print baseline relative frequencies         :  t
  print conditional population probabilities  :  t
  print bootstrap estimates                   :  t
  resample baseline                           :  f

* control parameters
  number of stocks in analysis       :  14
  number of characters in analysis   :   9
  maximum number of genotypes        : 200
  maximum number of classes          :  20
  maximum # of iterations            : 300
  estimate tolerance                 : .1E-3
  likelihood tolerance               :  1.0e-10
  genotype tolerance                 :  1.0e-6
  algorithm switch tolerance         :  0.01
  percent of maximum likelihood      :  80
  number of resamplings              : 100
  size of simulated mixture sample   : 500
  confidence intervals               : 90
  random seed                        : -123456

* characters used
[# types] [kind] [character]
    3       locus     PMI
    3       locus     TO1
    3      isolocus   MDH34
    3      isolocus   IDH34
    3       locus     GL1
    3       locus     ADA1
    3       locus     LGG
    3       locus     LDH4
    3       locus     LDH5

* populations
 [population]            [lev1]  [estimate]
  Warm Springs         :   2       10
  Rapid                :   4       20
  Kooskia              :   4       10
  Round Butte          :   5       10
  Carson               :   3       10
  Eagle                :   1       20
  Little White Salmon  :   4       10
  South Santian        :   5       10
  Oakridge             :   5       10
  Kalama               :   1       50
  Cowlitz              :   1       10
  McKenzie             :   5       10
  Levenworth           :   3       10
  Klickitat            :   2       10

 * regions
   [level] [label]  [region]         [estimate]
      1       1      Lower Columbia  :    1
      1       2      Mid Columbia    :    0
      1       3      Upper Columbia  :    0
      1       4      Snake River     :    0
      1       5      Other           :    0

* files
 baseline  : d:\spam32\columbia\baseline\columb1.bse
 output    : d:\spam32\columbia\output\simulation\columb1

* run
```

Appendix 3: Log file (excerpts)

```
*** Note: SPAM program.
     Log sent to file:
     D:\spam\Spam32\columbia\control\Columest.log.

 *** Note: Reading baseline frequencies.
     Reading Warm Springs = Warm Spring.

 *** Note: Reading baseline frequencies.
     Reading Rapid = Rapid.

 *** Note: Reading baseline frequencies.
     Reading Kooskia = Kooskia.

 *** Note: Reading baseline frequencies.
     Reading Round Butte = Round Butte.

 *** Note: Reading baseline frequencies.
     Reading Carson = Carson.

 *** Note: Reading baseline frequencies.
     Reading Eagle = Eagle.

 *** Note: Reading baseline frequencies.
     Reading Little White Salmon = Little White Salmon.

 *** Note: Reading baseline frequencies.
     Reading South Santian = South Santian.

 *** Note: Reading baseline frequencies.
     Reading Oakridge = Oakridge.

 *** Note: Reading baseline frequencies.
     Reading Kalama = Kalma.

 *** Note: Reading baseline frequencies.
     Reading Cowlitz = Cowlitz.

 *** Note: Reading baseline frequencies.
     Reading McKenzie = McKenzie.

 *** Note: Reading baseline frequencies.
     Reading Levenworth = Levenworth.

 *** Note: Reading baseline frequencies.
     Reading Klickitat = Klickitat.

 *** Warning: Reading baseline frequencies.
     Sum of allele/type frequencies is   801, but   800 is expected
     for population Klickitat, character MDH34.

 *** Note: SPAM estimation procedure.
     iteration sent to file:
     d:\spam\spam32\columbia\output\estimation\columb.itr.

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Guaranteed percent maximum value of the
     likelihood achieved =  94.5%.

 *** Note: SPAM estimation procedure.
     Estimates sent to file:
     d:\spam\spam32\columbia\output\estimation\columb.est.

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Guaranteed percent maximum value of the
     likelihood achieved =  91.3%.

 *** Note: Likelihood CI procedure.
     Lower bound search for Lower Columbia:
```

```
     Estimate:    .0073
     Chisquared:   3.09
     P-Value:     .9214

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Guaranteed percent maximum value of the
     likelihood achieved =  90.7%.

 *** Note: Likelihood CI procedure.
     Lower bound search for Lower Columbia:
     Estimate:    .0081
     Chisquared:   2.94
     P-Value:     .9135

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Guaranteed percent maximum value of the
     likelihood achieved =  96.6%.

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Maximum change in parameters =    .3346E-04.

 *** Note: Likelihood CI procedure.
     Lower bound search for Lower Columbia:
     Estimate:    .0095
     Chisquared:   2.68
     P-Value:     .8986

[...]

 *** Note: SPAM estimation procedure.
     Compressed Mixture sent to file:
     d:\spam\spam32\columbia\output\estimation\columb.cmx.

 *** Note: SPAM estimation procedure.
     Conditional Genotype Probabilities sent to file:
     d:\spam\spam32\columbia\output\estimation\columb.gen.

 *** Note: SPAM estimation procedure.
     Conditional Population Probabilities sent to file:
     d:\spam\spam32\columbia\output\estimation\columb.pop.

 *** Note: Executing Bootstrap.
     Conditional Population Probabilities sent to file:
     d:\spam\spam32\columbia\output\estimation\columb.rsm.

 *** Note: Executing Bootstrap.
     Iteration   1.

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Guaranteed percent maximum value of the
     likelihood achieved =  92.1%.

 *** Note: Executing Bootstrap.
     Impossible types exist using bound .10E-05 and are flagged in
     the condensed mixture output.   The adjusted sample size is 1596.

 *** Note: Executing Bootstrap.
     Iteration   2.

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Guaranteed percent maximum value of the
     likelihood achieved =  90.3%.

 *** Note: Executing Bootstrap.
     Iteration   3.

 *** Note: Searching for the maximum likelihood estimates.
     Convergence. Maximum change in parameters =    .9666E-04.

[...]
```

Appendix 4: Estimation output file

```
* Estimation: Columbia Example

  Iterations                24
  Log likelihood          -5626.04
  Percent of maximum      94.51
  Maximum estimate update .107E-03

  Number of types  187
  Number of observations 1597
  Classes     15
     - Types per class
       142    6    3    2    7    2    8    4    2    2
         4    2    1    1    1
     - Individuals per class
      1518    8    4    6   10    2   25    8    2    2
         5    4    1    1    1


  Characters used in the analysis

        Character      Type
_________________________________

   1     PMI         locus
   2     TO1         locus
   3     MDH34       isolocus
   4     IDH34       isolocus
   5     GL1         locus
   6     ADA1        locus
   7     LGG         locus
   8     LDH4        locus
   9     LDH5        locus
______________________________________________________________________

       Area                            Estimate  Std.Err.   CV
    1  Lower Columbia                   .0429     .0217    .51
    2  Mid Columbia                     .1604     .0461    .29
    3  Upper Columbia                   .1882     .0616    .33
    4  Snake River                      .4925     .0700    .14
    5  Other                            .1159     .0164    .14

    6  Columbia                         .3916     .0666    .17
    7  Snake River                      .4925     .0700    .14
    8  Other                            .1159     .0164    .14
______________________________________________________________________

  Correlation (below)/Covariances (above)

                                          1         2         3         4         5
    1  Lower Columbia                 .470E-03 -.681E-03 -.187E-03  .456E-03 -.571E-04
    2  Mid Columbia                  -.681     .213E-02 -.108E-03 -.123E-02 -.103E-03
    3  Upper Columbia                -.140     -.038     .379E-02 -.375E-02  .256E-03
    4  Snake River                    .300     -.382     -.871     .490E-02 -.364E-03
    5  Other                         -.161     -.136      .254     -.318     .268E-03
    6  Columbia                      -.276      .435      .853     -.972      .088
    7  Snake River                    .300     -.382     -.871     1.000     -.318
    8  Other                         -.003     -.002      .004     -.005      .016

                                          6         7         8
    1  Lower Columbia                -.398E-03  .456E-03 -.571E-04
    2  Mid Columbia                   .134E-02 -.123E-02 -.103E-03
    3  Upper Columbia                 .350E-02 -.375E-02  .256E-03
    4  Snake River                   -.453E-02  .490E-02 -.364E-03
    5  Other                          .958E-04 -.364E-03  .268E-03
    6  Columbia                       .444E-02 -.453E-02  .958E-04
    7  Snake River                   -.972      .490E-02 -.364E-03
    8  Other                          .001     -.005      .268E-03
```

```
      Population                          Estimate  Std.Err.   CV     Score
   1  Warm Springs                          .0180     .0162    .90       -.7
   2  Rapid                                 .2916     .0231    .79E-01    .0
   3  Kooskia                               .2009     .0723    .36        .1 *
   4  Round Butte                           .0001     .0002    2.9     -13.1
   5  Carson                                .1880     .0616    .33        .0
   6  Eagle                                 .0000     .0000    .00    -235.7
   7  Little White Salmon                   .0000     .0000    .00     -57.0
   8  South Santian                         .0349     .0235    .67       -.1
   9  Oakridge                              .0810     .0199    .25        .0
  10  Kalama                                .0000     .0000    .00     -28.2
  11  Cowlitz                               .0429     .0217    .51        .0
  12  McKenzie                              .0000     .0000    .00     -78.9
  13  Levenworth                            .0002     .0004    1.9     -14.3
  14  Klickitat                             .1424     .0436    .31        .0
```

```
  Correlation (below)/Covariances (above)

                                       1         2         3         4         5
    1  Warm Springs                 .262E-03  .831E-04  .265E-04 -.370E-03  .111E-04
    2  Rapid                        .222      .533E-03 -.431E-03 -.925E-04 -.248E-04
    3  Kooskia                      .023     -.259      .523E-02 -.366E-02 -.463E-03
    4  Carson                      -.371     -.065     -.822      .379E-02  .326E-03
    5  South Santian                .029     -.046     -.273      .225      .551E-03
    6  Oakridge                    -.014     -.079      .111     -.057     -.726
    7  Cowlitz                      .027      .049      .275     -.140     -.068
    8  Levenworth                  -.004     -.001      .005     -.008      .001
    9  Klickitat                   -.001     -.002     -.018      .004     -.001

                                       6         7         8         9
    1  Warm Springs                -.436E-05  .936E-05 -.222E-07 -.178E-04
    2  Rapid                       -.362E-04  .247E-04 -.966E-08 -.558E-04
    3  Kooskia                      .159E-03  .431E-03  .147E-06 -.129E-02
    4  Carson                      -.697E-04 -.187E-03 -.198E-06  .262E-03
    5  South Santian               -.339E-03 -.346E-04  .545E-08 -.272E-04
    6  Oakridge                     .394E-03 -.225E-04  .116E-07 -.824E-04
    7  Cowlitz                     -.052      .470E-03  .250E-07 -.691E-03
    8  Levenworth                   .002      .003      .145E-06 -.154E-06
    9  Klickitat                   -.004     -.032      .000      .190E-02
```

```
  90% Confidence Intervals

                             Normal Distribution
  Population                    lower   upper
  Lower Columbia                .007    .079
  Mid Columbia                  .085    .236
  Upper Columbia                .087    .290
  Snake River                   .377    .608
  Other                         .089    .143
  Columbia                      .282    .501
  Snake River                   .377    .608
  Other                         .089    .143

                                 Likelihood
  Population                    lower   upper
  Lower Columbia                .010    .080
  Mid Columbia                  .082    .237
  Upper Columbia                .095    .295
  Snake River                   .373    .603
  Other                         .089    .166
  Columbia                      .285    .506
  Snake River                   .373    .603
  Other                         .089    .166
```

Appendix 5: Simulation output file

```
______________________________________________________________________

  Simulation          : Columbia Example

  Average number of genotypes :       75.85
  Simulated sample size       :         500
  Number of resamplings       :         100
  Random seed                 :     -123456

                                            Mean
      Area                                Estimate  Std.Dev.  Root MSE  Expected
      1  Lower Columbia                    .9741     .0260     .0366    1.0000
      2  Mid Columbia                      .0065     .0154     .0167     .0000
      3  Upper Columbia                    .0008     .0031     .0032     .0000
      4  Snake River                       .0057     .0106     .0120     .0000
      5  Other                             .0127     .0164     .0207     .0000

      90% Bootstrap Confidence Intervals
      Area                                 Lower     Upper
      1  Lower Columbia                    .9160     .9996
      2  Mid Columbia                      .0000     .0356
      3  Upper Columbia                    .0000     .0041
      4  Snake River                       .0000     .0263
      5  Other                             .0000     .0403

______________________________________________________________________

  Correlation (below)/Covariances (above)

                                          1         2         3         4         5
    1  Lower Columbia                  .677E-03 -.239E-03 -.897E-05 -.125E-03 -.301E-03
    2  Mid Columbia                    -.594     .239E-03 -.113E-05 -.111E-04  .103E-04
    3  Upper Columbia                  -.112     -.024     .955E-05  .520E-06  .122E-06
    4  Snake River                     -.452     -.067      .016     .113E-03  .221E-04
    5  Other                           -.012      .001      .000      .002     .268E-03

______________________________________________________________________


                     angle deviation          distance
          level     mean      std.dev.     mean      std.dev.
            1       .022      .023         .033      .033
            2       .103      .053         .077      .038

______________________________________________________________________

                                        Mean
      Population                      Estimate  Std.Dev.  Root MSE  Expected
   1  Warm Springs                     .0005     .0014     .0015     .0000
   2  Rapid                            .0044     .0092     .0102     .0000
   3  Kooskia                          .0001     .0006     .0006     .0000
   4  Round Butte                      .0010     .0033     .0035     .0000
   5  Carson                           .0000     .0000     .0000     .0000
   6  Eagle                            .2536     .0407     .0406     .2500
   7  Little White Salmon              .0012     .0059     .0060     .0000
   8  South Santian                    .0019     .0055     .0058     .0000
   9  Oakridge                         .0056     .0118     .0130     .0000
  10  Kalama                           .6157     .0538     .0544     .6250
  11  Cowlitz                          .1048     .0405     .0450     .1250
  12  McKenzie                         .0042     .0094     .0103     .0000
  13  Levenworth                       .0008     .0031     .0032     .0000
  14  Klickitat                        .0060     .0153     .0164     .0000
      Unknown                          .0001

______________________________________________________________________
```

```
Correlation (below)/Covariances (above)

                                    1         2         3         4         5
  1  Warm Springs               .198E-05 -.603E-06 -.294E-06  .357E-05 -.541E-07
  2  Rapid                     -.046      .848E-04 -.188E-05 -.171E-04 -.324E-05
  3  Round Butte               -.063     -.061      .111E-04  .289E-04  .108E-05
  4  Eagle                      .062     -.046      .214      .165E-02 -.215E-04
  5  Little White Salmon       -.007     -.060      .055     -.090      .345E-04
  6  South Santian              .133      .117      .067     -.016     -.071
  7  Oakridge                   .015      .032      .088      .052      .119
  8  Kalama                     .049     -.053     -.127     -.653     -.076
  9  Cowlitz                   -.160     -.138     -.136     -.247      .061
 10  McKenzie                  -.085      .142     -.092      .148     -.070
 11  Levenworth                -.087      .054     -.060      .012     -.052
 12  Klickitat                  .001     -.001      .000      .003      .000

                                    6         7         8         9        10
  1  Warm Springs               .103E-05  .249E-06  .371E-05 -.909E-05 -.113E-05
  2  Rapid                      .597E-05  .349E-05 -.262E-04 -.515E-04  .123E-04
  3  Round Butte                .123E-05  .347E-05 -.227E-04 -.184E-04 -.290E-05
  4  Eagle                     -.368E-05  .248E-04 -.143E-02 -.406E-03  .567E-04
  5  Little White Salmon       -.229E-05  .827E-05 -.240E-04  .145E-04 -.386E-05
  6  South Santian              .304E-04 -.422E-05 -.271E-04  .931E-05 -.436E-05
  7  Oakridge                  -.065      .139E-03 -.141E-03 -.733E-04  .620E-05
  8  Kalama                    -.091     -.223      .290E-02 -.918E-03 -.209E-03
  9  Cowlitz                    .042     -.154     -.422      .164E-02  .750E-04
 10  McKenzie                  -.084      .056     -.413      .197      .889E-04
 11  Levenworth                 .054      .029     -.021     -.056     -.041
 12  Klickitat                 -.001      .003     -.002     -.006     -.002

                                   11        12
  1  Warm Springs              -.377E-06  .107E-05
  2  Rapid                      .152E-05 -.774E-05
  3  Round Butte               -.616E-06  .872E-06
  4  Eagle                      .149E-05  .105E-03
  5  Little White Salmon       -.953E-06 -.260E-05
  6  South Santian              .915E-06 -.683E-05
  7  Oakridge                   .105E-05  .319E-04
  8  Kalama                    -.353E-05 -.937E-04
  9  Cowlitz                   -.695E-05 -.248E-03
 10  McKenzie                  -.121E-05 -.155E-04
 11  Levenworth                 .955E-05 -.743E-06
 12  Klickitat                  .000      .234E-03
```

Appendix 6: Iterations output file (excerpts)

```
Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5805.540     .0
       5       .270E-01  (.000    )   -5629.042     .0     CG
      10       .494E-02  (.000    )   -5626.236     .0     CG
      15       .220E-02  (.000    )   -5626.089   17.2     CG
      20       .102E-02  (.000    )   -5626.045   36.0     CG
      24       .107E-03  (.000    )   -5626.038   94.5     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5629.685     .0
       5       .712E-02  (.000    )   -5627.602    1.8     CG
      10       .511E-03  (.000    )   -5627.586   42.2     CG
      14       .147E-03  (.000    )   -5627.585   91.3     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5627.509   44.9
       5       .334E-03  (.000    )   -5627.508   90.7     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5627.384   27.2
       5       .396E-03  (.000    )   -5627.380   89.2     CG
       6       .335E-04  (.000    )   -5627.380   96.6     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5628.592     .0
       5       .127E-01  (.000    )   -5627.285    2.3     CG
      10       .742E-03  (.000    )   -5627.276   72.2     CG
      12       .401E-03  (.000    )   -5627.276   94.1     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5627.319   67.2
       3       .162E-03  (.000    )   -5627.318   96.1     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5627.388   56.1
       3       .268E-03  (.000    )   -5627.387   94.9     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5629.420     .0
       5       .623E-02  (.000    )   -5627.465     .0     CG
      10       .514E-02  (.000    )   -5627.410     .0     CG
      15       .113E-02  (.000    )   -5627.366    2.0     CG
      20       .413E-02  (.000    )   -5627.312   59.1     CG
      25       .109E-03  (.000    )   -5627.311   90.7     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5627.340   32.3
       2       .789E-04  (.000    )   -5627.340   85.2     CG

                   Max                  Log
   Iteration     Update  (Baseline)  Likelihood   % Max  Algorithm
       0                               -5627.390   14.9
       5       .136E-03  (.000    )   -5627.389   80.0     CG
       6       .572E-04  (.000    )   -5627.389   87.4     CG

[...]
```

Appendix 7: Bootstrap output file

```
______________________________________________________________________

  Bootstrap           : Columbia Example

  Average number of genotypes :      150.37
  Number of resamplings       :         100
  Random seed                 :     -718805
  Mixture resampling goodness of fit
     Chi-Squared   DF   P-value
            1.4    186.  1.0000

                                            Mean
      Area                                Estimate  Std.Dev.  Root MSE  Expected
      1  Lower Columbia                    .0574     .0288     .0321     .0429
      2  Mid Columbia                      .1252     .0642     .0729     .1604
      3  Upper Columbia                    .1831     .0960     .0957     .1882
      4  Snake River                       .5074     .1073     .1078     .4925
      5  Other                             .1270     .0285     .0304     .1159

      6  Columbia                          .3657     .1065     .1091     .3916
      7  Snake River                       .5074     .1073     .1078     .4925
      8  Other                             .1270     .0285     .0304     .1159

      90% Bootstrap Confidence Intervals
      Area                                 Lower     Upper
      1  Lower Columbia                    .0053     .1008
      2  Mid Columbia                      .0145     .2205
      3  Upper Columbia                    .0409     .3461
      4  Snake River                       .3143     .6480
      5  Other                             .0871     .1753

      6  Columbia                          .2117     .5626
      7  Snake River                       .3143     .6480
      8  Other                             .0871     .1753

______________________________________________________________________

  Correlation (below)/Covariances (above)

                                          1         2         3         4         5
    1  Lower Columbia                  .831E-03 -.138E-02 -.433E-03  .112E-02 -.140E-03
    2  Mid Columbia                    -.744     .412E-02  .399E-03 -.285E-02 -.289E-03
    3  Upper Columbia                  -.157      .065     .922E-02 -.929E-02  .105E-03
    4  Snake River                      .362     -.414     -.902      .115E-01 -.487E-03
    5  Other                           -.171     -.158      .038     -.159      .812E-03
    6  Columbia                        -.319      .460      .898     -.965     -.107
    7  Snake River                      .362     -.414     -.902     1.000     -.159
    8  Other                           -.005     -.004      .001     -.005      .028

                                          6         7         8
    1  Lower Columbia                 -.980E-03  .112E-02 -.140E-03
    2  Mid Columbia                    .314E-02 -.285E-02 -.289E-03
    3  Upper Columbia                  .918E-02 -.929E-02  .105E-03
    4  Snake River                    -.110E-01  .115E-01 -.487E-03
    5  Other                          -.325E-03 -.487E-03  .812E-03
    6  Columbia                        .113E-01 -.110E-01 -.325E-03
    7  Snake River                     -.965     .115E-01 -.487E-03
    8  Other                           -.003     -.005     .812E-03

______________________________________________________________________

                     angle deviation          distance
          level     mean      std.dev.     mean      std.dev.
            1       .246      .118         .152      .071
            2       .203      .118         .135      .080
            3       .379      .139         .173      .070
```

```
                                    Mean
   Population                     Estimate  Std.Dev.  Root MSE  Expected
 1  Warm Springs                    .0167     .0177     .0176     .0180
 2  Rapid                           .2830     .0332     .0341     .2916
 3  Kooskia                         .2233     .1140     .1156     .2009
 4  Round Butte                     .0107     .0194     .0220     .0001
 5  Carson                          .1672     .1010     .1026     .1880
 6  Eagle                           .0000     .0000     .0000     .0000
 7  Little White Salmon             .0011     .0100     .0100     .0000
 8  South Santian                   .0458     .0297     .0315     .0349
 9  Oakridge                        .0704     .0269     .0288     .0810
10  Kalama                          .0112     .0140     .0179     .0000
11  Cowlitz                         .0463     .0299     .0299     .0429
12  McKenzie                        .0001     .0014     .0014     .0000
13  Levenworth                      .0159     .0237     .0283     .0002
14  Klickitat                       .1085     .0613     .0698     .1424
```

```
 Correlation (below)/Covariances (above)

                                      1         2         3         4         5
  1  Warm Springs                 .312E-03  .138E-03 -.122E-04 -.577E-04 -.428E-03
  2  Rapid                         .236     .110E-02 -.136E-02  .974E-04 -.241E-03
  3  Kooskia                      -.006     -.360     .130E-01 -.878E-04 -.943E-02
  4  Round Butte                  -.169      .152     -.040     .374E-03 -.187E-03
  5  Carson                       -.240     -.072     -.819     -.096     .102E-01
  6  Little White Salmon          -.081      .107     -.011      .094     -.091
  7  South Santian                 .074      .012     -.216     -.081      .118
  8  Oakridge                      .037     -.186      .119      .044      .039
  9  Kalama                       -.052      .189     -.030      .053     -.074
 10  Cowlitz                      -.069      .016      .311      .000     -.157
 11  McKenzie                      .023      .159      .042     -.056     -.147
 12  Levenworth                    .063      .120      .145     -.169     -.322
 13  Klickitat                     .002      .005     -.027     -.004      .011

                                      6         7         8         9        10
  1  Warm Springs                 -.143E-04  .387E-04  .175E-04 -.128E-04 -.365E-04
  2  Rapid                         .354E-04  .120E-04 -.166E-03  .879E-04  .163E-04
  3  Kooskia                      -.123E-04 -.732E-03  .364E-03 -.480E-04  .106E-02
  4  Round Butte                   .182E-04 -.465E-04  .227E-04  .144E-04 -.877E-07
  5  Carson                       -.923E-04  .353E-03  .107E-03 -.105E-03 -.472E-03
  6  Little White Salmon           .100E-03  .390E-04 -.453E-04  .145E-04 -.923E-05
  7  South Santian                 .131      .881E-03 -.552E-03  .232E-04 -.124E-03
  8  Oakridge                     -.168     -.692      .723E-03 -.961E-04  .464E-04
  9  Kalama                        .103      .056     -.254      .197E-03 -.129E-03
 10  Cowlitz                      -.031     -.140      .058     -.307      .892E-03
 11  McKenzie                     -.010     -.038     -.129     -.081     -.068
 12  Levenworth                   -.074      .032     -.153      .174      .121
 13  Klickitat                    -.002      .003     -.012      .000     -.044

                                     11        12        13
  1  Warm Springs                  .559E-06  .264E-04  .270E-04
  2  Rapid                         .736E-05  .944E-04  .175E-03
  3  Kooskia                       .670E-05  .391E-03 -.312E-02
  4  Round Butte                  -.152E-05 -.772E-04 -.695E-04
  5  Carson                       -.208E-04 -.768E-03  .108E-02
  6  Little White Salmon          -.146E-06 -.175E-04 -.163E-04
  7  South Santian                -.156E-05  .222E-04  .902E-04
  8  Oakridge                     -.485E-05 -.971E-04 -.321E-03
  9  Kalama                       -.159E-05  .578E-04 -.384E-05
 10  Cowlitz                      -.283E-05  .853E-04 -.132E-02
 11  McKenzie                      .195E-05  .471E-05  .120E-04
 12  Levenworth                    .142      .560E-03 -.284E-03
 13  Klickitat                     .009     -.012      .376E-02
```

Appendix 8: Resamples output file

```
Number of populations:  14
Number of resamples:   100
Warm Springs
Rapid
Kooskia
Round Butte
Carson
Eagle
Little White Salmon
South Santian
Oakridge
Kalama
Cowlitz
McKenzie
Levenworth
Klickitat
 .0000  .2225  .2295  .0082  .1820  .0000  .0000  .1013  .0251  .0000  .0408  .0000  .0000  .1899
 .0258  .2606  .1032  .0020  .2624  .0000  .0000  .0679  .0753  .0000  .0000  .0000  .0004  .2024
 .0000  .2728  .3346  .0000  .0702  .0000  .0000  .0338  .0682  .0155  .0485  .0000  .0411  .1152
 .0000  .2394  .2908  .0000  .2467  .0000  .0000  .0659  .0692  .0000  .0879  .0000  .0000  .0002
 .0232  .2663  .1063  .0000  .2751  .0000  .0000  .0890  .0464  .0017  .0036  .0000  .0000  .1885
 .0000  .2767  .1891  .0021  .2479  .0000  .0000  .0051  .0974  .0354  .0001  .0000  .0000  .1463
 .0059  .3255  .2023  .0000  .1742  .0000  .0000  .0722  .0528  .0000  .1100  .0000  .0000  .0570
 .0448  .3024  .0875  .0000  .2228  .0000  .0000  .0665  .0780  .0027  .0275  .0000  .0000  .1679
 .0208  .2503  .2752  .0000  .1690  .0000  .0000  .0883  .0651  .0284  .0223  .0000  .0048  .0758
 .0015  .2727  .2279  .0635  .1263  .0000  .0000  .0154  .1253  .0317  .0000  .0000  .0124  .1227
 .0532  .3205  .2696  .0065  .0409  .0000  .0000  .0143  .0703  .0000  .0879  .0000  .0000  .1369
 .0322  .2863  .2188  .0264  .1586  .0000  .0000  .0173  .1026  .0000  .0644  .0000  .0000  .0934
 .0289  .2628  .3538  .0207  .1412  .0000  .0000  .0242  .0863  .0000  .0823  .0000  .0000  .0000
 .0243  .3305  .1069  .0028  .2641  .0000  .0000  .0381  .0721  .0379  .0415  .0000  .0000  .0817
 .0000  .3199  .2237  .0297  .0666  .0000  .0998  .0843  .0253  .0259  .0405  .0000  .0000  .0843
 .0000  .2374  .2377  .0674  .1841  .0000  .0000  .0556  .0638  .0000  .0750  .0000  .0001  .0789
 .0406  .2278  .4340  .0000  .0227  .0000  .0000  .0646  .0759  .0000  .0684  .0000  .0654  .0000
 .0000  .2867  .1876  .0466  .2079  .0000  .0000  .0204  .0702  .0236  .0310  .0000  .0273  .0985
 .0031  .2650  .3669  .0019  .0681  .0000  .0000  .0313  .0423  .0000  .0600  .0000  .0258  .1355
 .0000  .3264  .0081  .0224  .3362  .0000  .0000  .0415  .0748  .0000  .1109  .0000  .0325  .0472
 .0072  .2819  .3034  .0000  .1700  .0000  .0000  .0142  .1010  .0257  .0751  .0000  .0003  .0212
 .0000  .2218  .1675  .0000  .3436  .0000  .0000  .0289  .1098  .0068  .0154  .0000  .0005  .1056
 .0000  .2984  .2778  .0000  .0456  .0000  .0000  .0715  .0415  .0336  .0904  .0000  .0914  .0497
 .0002  .2513  .3801  .0029  .0841  .0000  .0000  .0560  .0801  .0000  .0777  .0000  .0677  .0000
 .0000  .2902  .0000  .0528  .3199  .0000  .0000  .0555  .0618  .0000  .0013  .0000  .0000  .2185
 .0029  .3019  .2300  .0000  .2181  .0000  .0000  .0043  .1244  .0007  .0104  .0000  .0000  .1071
 .0000  .2441  .3553  .0000  .0975  .0000  .0000  .0722  .0627  .0239  .0457  .0000  .0000  .1011
 .0013  .2372  .4211  .0000  .0772  .0000  .0000  .0187  .1062  .0000  .0820  .0000  .0557  .0007
 .0418  .2744  .2987  .0040  .0703  .0000  .0000  .0256  .1070  .0000  .0393  .0000  .0000  .1395
 .0137  .2584  .2647  .0000  .1458  .0000  .0000  .0000  .1119  .0001  .0303  .0000  .0009  .1736
 .0002  .2722  .2586  .0040  .2126  .0000  .0000  .0979  .0203  .0053  .0785  .0000  .0004  .0501
 .0015  .2800  .0006  .0359  .3676  .0000  .0000  .0540  .0526  .0194  .0251  .0000  .0326  .1308
 .0002  .2567  .3436  .0000  .0978  .0000  .0000  .0481  .0655  .0102  .0347  .0000  .0118  .1314
 .0483  .3099  .2664  .0000  .0903  .0000  .0000  .0453  .0568  .0184  .0147  .0000  .0000  .1523
 .0043  .2833  .3337  .0000  .0987  .0000  .0000  .0688  .0292  .0478  .0118  .0000  .0001  .1223
 .0572  .2591  .3843  .0000  .0565  .0000  .0000  .0281  .0918  .0000  .0622  .0000  .0000  .0607
 .0140  .2968  .2703  .0751  .1293  .0000  .0000  .0253  .1253  .0000  .0634  .0000  .0000  .0004
 .0486  .2823  .2637  .0000  .1624  .0000  .0000  .0488  .0486  .0020  .0935  .0000  .0000  .0502
 .0000  .2880  .1364  .0101  .2549  .0000  .0000  .0530  .0722  .0000  .0463  .0000  .0000  .1389
 .0106  .2925  .4132  .0785  .0000  .0000  .0000  .0193  .0594  .0000  .0992  .0000  .0000  .0267
 .0000  .3526  .0400  .0000  .2468  .0000  .0000  .0320  .0966  .0000  .0680  .0000  .0000  .1639
 .0000  .2976  .0750  .0000  .3219  .0000  .0000  .0532  .0624  .0000  .0604  .0000  .0054  .1240
 .0233  .3331  .1625  .0086  .1545  .0000  .0000  .0961  .0495  .0186  .0819  .0000  .0552  .0160
 .0000  .3181  .2959  .0232  .1001  .0000  .0000  .0705  .0290  .0423  .0287  .0000  .0000  .0917
 .0231  .3054  .0218  .0000  .3461  .0000  .0000  .0291  .0663  .0062  .0402  .0000  .0000  .1619
 .0150  .2919  .1184  .0000  .2839  .0000  .0000  .0000  .0908  .0196  .0000  .0000  .0000  .1804
 .0273  .2867  .1891  .0000  .2061  .0000  .0000  .0477  .0816  .0000  .0919  .0000  .0000  .0696
 .0198  .2395  .1239  .0000  .3101  .0000  .0000  .0185  .0998  .0000  .0811  .0000  .0481  .0593
 .0029  .2565  .1402  .0000  .2973  .0000  .0000  .0000  .1196  .0001  .0841  .0000  .0000  .0993
 .0083  .2088  .0172  .0000  .4862  .0000  .0000  .0671  .0791  .0000  .0402  .0000  .0000  .0931
 .0186  .2762  .2359  .0011  .1745  .0000  .0001  .0581  .0537  .0000  .0219  .0000  .0000  .1598
 .0098  .2834  .2590  .0005  .0968  .0000  .0000  .0019  .0499  .0550  .0174  .0000  .0465  .1800
```

```
.0000  .2160  .5447  .0147  .0188  .0000  .0000  .0009  .0841  .0001  .0901  .0000  .0006  .0338
.0195  .2856  .2548  .0000  .1179  .0000  .0000  .0594  .0298  .0335  .0242  .0000  .0737  .1017
.0240  .3210  .1532  .0304  .1655  .0000  .0000  .0382  .1066  .0074  .0291  .0000  .0000  .1245
.0199  .2790  .1818  .0355  .1296  .0000  .0000  .0094  .0929  .0297  .0259  .0000  .0298  .1665
.0027  .3139  .0003  .0006  .3068  .0000  .0000  .0518  .0457  .0162  .0442  .0000  .0217  .1960
.0000  .2293  .2385  .0000  .3155  .0000  .0000  .0591  .0456  .0143  .0794  .0000  .0052  .0132
.0096  .2297  .3164  .0000  .1316  .0000  .0000  .0657  .0581  .0033  .0611  .0000  .0000  .1244
.0000  .3038  .3408  .0183  .1016  .0000  .0000  .0064  .0842  .0281  .0760  .0000  .0176  .0226
.0165  .2993  .1802  .0000  .2010  .0000  .0000  .0472  .0670  .0420  .0568  .0000  .0211  .0688
.0022  .3025  .2120  .0000  .1620  .0000  .0000  .0404  .0577  .0179  .0329  .0000  .0000  .1723
.0154  .3132  .0296  .0442  .2311  .0000  .0000  .1349  .0192  .0000  .0109  .0000  .0000  .2051
.0000  .3223  .2663  .0346  .1352  .0000  .0001  .0096  .0736  .0173  .0821  .0000  .0260  .0329
.0483  .2611  .0663  .0000  .2827  .0000  .0085  .0475  .0756  .0082  .0098  .0001  .0000  .1918
.0000  .3006  .2552  .0000  .1296  .0000  .0000  .0567  .0463  .0003  .0802  .0000  .0722  .0589
.0183  .2358  .3504  .0001  .0757  .0000  .0000  .0355  .1109  .0000  .0329  .0000  .0079  .1325
.0001  .2794  .2548  .0000  .2202  .0000  .0000  .0075  .0793  .0000  .0988  .0000  .0109  .0490
.0281  .2535  .3372  .0001  .0732  .0000  .0000  .0173  .1052  .0056  .0604  .0000  .0085  .1110
.0146  .2640  .2282  .0000  .1879  .0000  .0000  .0000  .1034  .0257  .0369  .0000  .0000  .1394
.0260  .2573  .2043  .0000  .2271  .0000  .0000  .0899  .0701  .0256  .0000  .0000  .0000  .0996
.0245  .3122  .1380  .0004  .2476  .0000  .0000  .0097  .0855  .0000  .0639  .0000  .0000  .1182
.0000  .2777  .3051  .0000  .0623  .0000  .0006  .0645  .0275  .0000  .0000  .0000  .0000  .2622
.0494  .3397  .2585  .0255  .0435  .0000  .0000  .0604  .0575  .0000  .0600  .0000  .0314  .0735
.0233  .3217  .2324  .0196  .1306  .0000  .0000  .0663  .0715  .0000  .0382  .0000  .0032  .0930
.0000  .2454  .3659  .0443  .0502  .0000  .0000  .0043  .0951  .0235  .0216  .0000  .0059  .1439
.0072  .3001  .1689  .0087  .2543  .0000  .0000  .0431  .0586  .0071  .0060  .0000  .0000  .1460
.0323  .2850  .1154  .0004  .2095  .0000  .0000  .0440  .0798  .0000  .0100  .0000  .0000  .2229
.0311  .2834  .2647  .0000  .1506  .0000  .0000  .0894  .0633  .0000  .0794  .0000  .0000  .0380
.0502  .3001  .0782  .0000  .1956  .0000  .0000  .0887  .0262  .0322  .0000  .0000  .0742  .1546
.0207  .2978  .0001  .0243  .2861  .0000  .0000  .0455  .0484  .0168  .0089  .0000  .0016  .2491
.0602  .2743  .3737  .0000  .0333  .0000  .0000  .0887  .0516  .0194  .0453  .0000  .0024  .0511
.0386  .3277  .2571  .0113  .0056  .0000  .0000  .0302  .0585  .0228  .0263  .0000  .0418  .1789
.0212  .3160  .0020  .0000  .3646  .0000  .0000  .0507  .0688  .0196  .0583  .0000  .0316  .0672
.0318  .2681  .2633  .0029  .1670  .0000  .0000  .0070  .0791  .0000  .0419  .0000  .0281  .1108
.0634  .3628  .2199  .0000  .0699  .0000  .0000  .0744  .0534  .0399  .0684  .0000  .0400  .0078
.0378  .2959  .2615  .0000  .0780  .0000  .0000  .0239  .0823  .0000  .0641  .0000  .0460  .1104
.0169  .2300  .1860  .0000  .1988  .0000  .0000  .1183  .0055  .0307  .0242  .0000  .0314  .1581
.0000  .2344  .4332  .0000  .0026  .0000  .0000  .0285  .0586  .0006  .0706  .0000  .0529  .1194
.0076  .2887  .1517  .0636  .1740  .0000  .0000  .0618  .0629  .0285  .0360  .0000  .0092  .1160
.0209  .3208  .1609  .0270  .2316  .0000  .0000  .0515  .0492  .0294  .0372  .0000  .0000  .0717
.0205  .3356  .2720  .0000  .0185  .0000  .0000  .0346  .0361  .0000  .0265  .0140  .0494  .1928
.0000  .2738  .0748  .0000  .3211  .0000  .0000  .0909  .0751  .0000  .0000  .0000  .0000  .1643
.0254  .3066  .2381  .0000  .1338  .0000  .0000  .0559  .0866  .0049  .0459  .0000  .0437  .0592
.0047  .2561  .3320  .0000  .1611  .0000  .0000  .0000  .1412  .0000  .0546  .0000  .0000  .0504
.0280  .3217  .2607  .0631  .0274  .0000  .0000  .0435  .0610  .0096  .0539  .0000  .0000  .1310
.0188  .2854  .3405  .0000  .0087  .0000  .0000  .0464  .1054  .0156  .0036  .0000  .0838  .0919
.0683  .2804  .2466  .0000  .0666  .0000  .0000  .0389  .0984  .0000  .0325  .0000  .0326  .1357
.0000  .2538  .2538  .0000  .1749  .0000  .0000  .0935  .0478  .0025  .0579  .0000  .0000  .1157
.0197  .3511  .1581  .0000  .1021  .0000  .0000  .0408  .0533  .0000  .0443  .0000  .0586  .1720
```

Appendix 9: Baseline output file

```
Columbia Example

        locus                                   PMI                         TO1
                                        (N)     1     2     3      (N)     1     2     3
 population                            ______________________     ______________________
______________________________________________________________________________________
   1 Warm Springs                       200  .915  .085  .000      200  .492  .507  .000
   2 Rapid                              196  .862  .138  .000      195  .959  .041  .000
   3 Kooskia                            199  .822  .178  .000      200  .748  .253  .000
   4 Round Butte                        189  .857  .143  .000      191  .592  .408  .000
   5 Carson                             200  .848  .153  .000      199  .741  .259  .000
   6 Eagle                              195  .477  .523  .000      196  .798  .202  .000
   7 Little White Salmon                200  .818  .183  .000      197  .784  .216  .000
   8 South Santian                      198  .553  .447  .000      198  .846  .154  .000
   9 Oakridge                           200  .427  .573  .000      200  .920  .080  .000
  10 Kalama                             198  .508  .487  .005      200  .728  .273  .000
  11 Cowlitz                            199  .490  .503  .008      200  .615  .385  .000
  12 McKenzie                           200  .448  .552  .000      200  .858  .142  .000
  13 Levenworth                         200  .895  .105  .000      200  .688  .313  .000
  14 Klickitat                          200  .693  .300  .007      200  .693  .308  .000
______________________________________________________________________________________

        locus                                  MDH34                       IDH34
                                        (N)     1     2     3      (N)     1     2     3
 population                            ______________________     ______________________
______________________________________________________________________________________
   1 Warm Springs                       200 1.000  .000  .000      191  .832  .000  .168
   2 Rapid                              193 1.000  .000  .000      196  .974  .000  .026
   3 Kooskia                            197  .973  .027  .000      200  .902  .005  .092
   4 Round Butte                        191  .999  .001  .000      193  .900  .000  .100
   5 Carson                             200  .980  .020  .000      200  .886  .000  .114
   6 Eagle                              198 1.000  .000  .000      197 1.000  .000  .000
   7 Little White Salmon                200  .979  .021  .000      199  .926  .000  .074
   8 South Santian                      188  .939  .061  .000      197  .921  .079  .000
   9 Oakridge                           194  .976  .024  .000      194  .854  .129  .017
  10 Kalama                             196  .994  .006  .000      197  .909  .089  .003
  11 Cowlitz                            197  .987  .013  .000      199  .951  .030  .019
  12 McKenzie                           200  .950  .050  .000      200  .914  .086  .000
  13 Levenworth                         198  .994  .006  .000      197  .924  .001  .075
  14 Klickitat                          200  .983  .018  .001      200  .938  .020  .043
______________________________________________________________________________________

        locus                                   GL1                        ADA1
                                        (N)     1     2     3      (N)     1     2     3
 population                            ______________________     ______________________
______________________________________________________________________________________
   1 Warm Springs                       200  .998  .003  .000      200  .988  .013  .000
   2 Rapid                              196 1.000  .000  .000      196 1.000  .000  .000
   3 Kooskia                            198 1.000  .000  .000      200 1.000  .000  .000
   4 Round Butte                        191  .979  .021  .000      186 1.000  .000  .000
   5 Carson                             199 1.000  .000  .000      199  .987  .013  .000
   6 Eagle                              198 1.000  .000  .000      198 1.000  .000  .000
   7 Little White Salmon                200 1.000  .000  .000      200 1.000  .000  .000
   8 South Santian                      195 1.000  .000  .000      197 1.000  .000  .000
   9 Oakridge                           193 1.000  .000  .000      199 1.000  .000  .000
  10 Kalama                             199 1.000  .000  .000      199  .970  .030  .000
  11 Cowlitz                            198  .980  .020  .000      198  .977  .023  .000
  12 McKenzie                           200  .998  .003  .000      200  .998  .003  .000
  13 Levenworth                         200  .993  .007  .000      199  .972  .018  .010
  14 Klickitat                          199  .995  .005  .000      200  .998  .003  .000
______________________________________________________________________________________

        locus                                   LGG                        LDH4
                                        (N)     1     2     3      (N)     1     2     3
 population                            ______________________     ______________________
______________________________________________________________________________________
   1 Warm Springs                       199  .937  .063  .000      200 1.000  .000  .000
   2 Rapid                              189  .899  .101  .000      196  .992  .008  .000
```

3 Kooskia	198	.932	.068	.000	198	.985	.015	.000
4 Round Butte	190	.955	.045	.000	192	1.000	.000	.000
5 Carson	190	.961	.039	.000	199	.982	.018	.000
6 Eagle	193	1.000	.000	.000	198	1.000	.000	.000
7 Little White Salmon	192	.951	.049	.000	151	1.000	.000	.000
8 South Santian	197	.838	.162	.000	195	1.000	.000	.000
9 Oakridge	183	.852	.148	.000	193	1.000	.000	.000
10 Kalama	200	.975	.025	.000	200	1.000	.000	.000
11 Cowlitz	198	.912	.088	.000	198	1.000	.000	.000
12 McKenzie	192	.904	.096	.000	200	1.000	.000	.000
13 Levenworth	198	.967	.033	.000	200	1.000	.000	.000
14 Klickitat	198	.934	.066	.000	200	1.000	.000	.000

locus population	LDH5 (N)	 1	 2	 3
1 Warm Springs	197	1.000	.000	.000
2 Rapid	195	1.000	.000	.000
3 Kooskia	195	.992	.008	.000
4 Round Butte	193	.992	.008	.000
5 Carson	200	.988	.013	.000
6 Eagle	197	1.000	.000	.000
7 Little White Salmon	200	1.000	.000	.000
8 South Santian	199	1.000	.000	.000
9 Oakridge	200	.985	.015	.000
10 Kalama	195	.977	.023	.000
11 Cowlitz	200	.960	.040	.000
12 McKenzie	200	1.000	.000	.000
13 Levenworth	194	.959	.041	.000
14 Klickitat	200	.965	.035	.000

```
Columbia Example: Rapid and Eagle missing PMI locus from baseline (100 resamples)

          locus                                 PMI                      TO1
                                        (N)    1     2     3      (N)    1     2     3
 population                            _______________________    ______________________
_______________________________________________________________________________________
   1 Warm Springs                       200  .915  .085  .000     200  .492  .507  .000
   2 Rapid                                0 ----- ----- -----     195  .959  .041  .000
               Estimate               471.0  .850  .149  .001
               Mean                   458.5  .843  .155  .002
               Std.Dev.                57.2  .021  .021  .002

   3 Kooskia                            199  .822  .178  .000     200  .748  .253  .000
   4 Round Butte                        189  .857  .143  .000     191  .592  .408  .000
   5 Carson                             200  .848  .153  .000     199  .741  .259  .000
   6 Eagle                                0 ----- ----- -----     196  .798  .202  .000
               Estimate                 3.2  .000  .645  .355
               Mean                     4.3  .184  .528  .287
               Std.Dev.                 4.2  .360  .248  .131

   7 Little White Salmon                200  .818  .183  .000     197  .784  .216  .000
   8 South Santian                      198  .553  .447  .000     198  .846  .154  .000
   9 Oakridge                           200  .427  .573  .000     200  .920  .080  .000
  10 Kalama                             198  .508  .487  .005     200  .728  .273  .000
  11 Cowlitz                            199  .490  .503  .008     200  .615  .385  .000
  12 McKenzie                           200  .448  .552  .000     200  .858  .142  .000
  13 Levenworth                         200  .895  .105  .000     200  .688  .313  .000
  14 Klickitat                          200  .693  .300  .007     200  .693  .308  .000
_______________________________________________________________________________________

          locus                                MDH34                    IDH34
                                        (N)    1     2     3      (N)    1     2     3
 population                            _______________________    ______________________
_______________________________________________________________________________________
   1 Warm Springs                       200 1.000  .000  .000     191  .832  .000  .168
   2 Rapid                              193 1.000  .000  .000     196  .974  .000  .026
   3 Kooskia                            197  .973  .027  .000     200  .902  .005  .092
   4 Round Butte                        191  .999  .001  .000     193  .900  .000  .100
   5 Carson                             200  .980  .020  .000     200  .886  .000  .114
   6 Eagle                              198 1.000  .000  .000     197 1.000  .000  .000
   7 Little White Salmon                200  .979  .021  .000     199  .926  .000  .074
   8 South Santian                      188  .939  .061  .000     197  .921  .079  .000
   9 Oakridge                           194  .976  .024  .000     194  .854  .129  .017
  10 Kalama                             196  .994  .006  .000     197  .909  .089  .003
  11 Cowlitz                            197  .987  .013  .000     199  .951  .030  .019
  12 McKenzie                           200  .950  .050  .000     200  .914  .086  .000
  13 Levenworth                         198  .994  .006  .000     197  .924  .001  .075
  14 Klickitat                          200  .983  .018  .001     200  .938  .020  .043
_______________________________________________________________________________________

          locus                                 GL1                     ADA1
                                        (N)    1     2     3      (N)    1     2     3
 population                            _______________________    ______________________
_______________________________________________________________________________________
   1 Warm Springs                       200  .998  .003  .000     200  .988  .013  .000
   2 Rapid                              196 1.000  .000  .000     196 1.000  .000  .000
   3 Kooskia                            198 1.000  .000  .000     200 1.000  .000  .000
   4 Round Butte                        191  .979  .021  .000     186 1.000  .000  .000
   5 Carson                             199 1.000  .000  .000     199  .987  .013  .000
   6 Eagle                              198 1.000  .000  .000     198 1.000  .000  .000
   7 Little White Salmon                200 1.000  .000  .000     200 1.000  .000  .000
   8 South Santian                      195 1.000  .000  .000     197 1.000  .000  .000
   9 Oakridge                           193 1.000  .000  .000     199 1.000  .000  .000
  10 Kalama                             199 1.000  .000  .000     199  .970  .030  .000
  11 Cowlitz                            198  .980  .020  .000     198  .977  .023  .000
  12 McKenzie                           200  .998  .003  .000     200  .998  .003  .000
  13 Levenworth                         200  .993  .007  .000     199  .972  .018  .010
  14 Klickitat                          199  .995  .005  .000     200  .998  .003  .000
_______________________________________________________________________________________

          locus                                 LGG                     LDH4
```

population	(N)	1	2	3	(N)	1	2	3
1 Warm Springs	199	.937	.063	.000	200	1.000	.000	.000
2 Rapid	189	.899	.101	.000	196	.992	.008	.000
3 Kooskia	198	.932	.068	.000	198	.985	.015	.000
4 Round Butte	190	.955	.045	.000	192	1.000	.000	.000
5 Carson	190	.961	.039	.000	199	.982	.018	.000
6 Eagle	193	1.000	.000	.000	198	1.000	.000	.000
7 Little White Salmon	192	.951	.049	.000	151	1.000	.000	.000
8 South Santian	197	.838	.162	.000	195	1.000	.000	.000
9 Oakridge	183	.852	.148	.000	193	1.000	.000	.000
10 Kalama	200	.975	.025	.000	200	1.000	.000	.000
11 Cowlitz	198	.912	.088	.000	198	1.000	.000	.000
12 McKenzie	192	.904	.096	.000	200	1.000	.000	.000
13 Levenworth	198	.967	.033	.000	200	1.000	.000	.000
14 Klickitat	198	.934	.066	.000	200	1.000	.000	.000

locus		LDH5		
population	(N)	1	2	3
1 Warm Springs	197	1.000	.000	.000
2 Rapid	195	1.000	.000	.000
3 Kooskia	195	.992	.008	.000
4 Round Butte	193	.992	.008	.000
5 Carson	200	.988	.013	.000
6 Eagle	197	1.000	.000	.000
7 Little White Salmon	200	1.000	.000	.000
8 South Santian	199	1.000	.000	.000
9 Oakridge	200	.985	.015	.000
10 Kalama	195	.977	.023	.000
11 Cowlitz	200	.960	.040	.000
12 McKenzie	200	1.000	.000	.000
13 Levenworth	194	.959	.041	.000
14 Klickitat	200	.965	.035	.000

Appendix 10: Condensed mixture output file (excerpts)

```
Columbia Example

 Class   1

  Type  N PMI TO1 MDH34 IDH34 GL1 ADA1 LGG LDH4 LDH5
 _____________________________________________________
    1   4 011 110 400   400   200 200  200 200  200
    2   7 020 020 400   400   200 200  200 200  200
    3   1 020 020 400   310   200 200  200 200  200
    4   1 020 020 400   400   200 200  200 200  110
    5   9 020 110 400   400   200 200  200 200  200
    6   3 020 110 400   400   200 200  110 200  200
    7   2 020 110 310   310   200 200  200 200  200
    8   6 020 110 400   310   200 200  200 200  200
    9   1 020 110 400   301   200 200  110 200  200
   10   1 020 110 400   400   200 200  020 200  200
   11   4 020 110 400   301   200 200  200 200  200
   12   1 020 110 400   310   200 200  200 200  110
   13   1 020 110 400   310   200 200  110 200  110

[...]

  139   1 200 200 310   400   200 200  200 110  200
  140   1 200 200 310   310   200 200  110 200  200
  141   2 200 200 400   301   200 110  200 200  200
  142   1 200 200 400   211   200 200  110 200  200



 Class   2

  Type  N PMI TO1 MDH34 IDH34 GL1 ADA1 LGG LDH4 LDH5
 _____________________________________________________
  143   1 020 110 400   301   200 000  110 200  200
  144   1 020 200 400   400   200 000  110 200  200
  145   2 200 110 400   400   200 000  200 200  200
  146   2 200 200 400   400   200 000  110 200  200
  147   1 200 200 400   400   200 000  200 200  200
  148   1 200 200 400   301   200 000  200 200  200


 Class   3

  Type  N PMI TO1 MDH34 IDH34 GL1 ADA1 LGG LDH4 LDH5
 _____________________________________________________
  149   1 020 200 400   000   000 200  200 200  200
  150   1 110 200 400   000   000 200  200 110  200
  151   2 200 200 400   000   000 200  200 200  200


[...]

 Class  14

  Type  N PMI TO1 MDH34 IDH34 GL1 ADA1 LGG LDH4 LDH5
 _____________________________________________________
  186   1 200 200 000   310   000 200  200 200  200


 Class  15

  Type  N PMI TO1 MDH34 IDH34 GL1 ADA1 LGG LDH4 LDH5
 _____________________________________________________
  187   1 200 200 400   400   000 200  110 000  000
```

Appendix 11: Conditional genotype probability output file (excerpts)

```
Columbia Example
  Conditional Genotype Probabilities

  Mixture Sample Summary

  Number of types       :    187
  Number of classes     :     15
  Number of individuals :   1597

     Class  Types  Indviduals
         1    142      1518
         2      6         8
         3      3         4
         4      2         6
         5      7        10
         6      2         2
         7      8        25
         8      4         8
         9      2         2
        10      2         2
        11      4         5
        12      2         4
        13      1         1
        14      1         1
        15      1         1

  Class   1

                                        Population
   Type  Size      1        2        3        4        5        6        7        8        9
  _____________________________________________________________________________________________
      1      4 .000     .000     .000     .000     .000     .000     .000     .000     .000
      2      7 .762E-03 .229E-04 .100E-02 .192E-02 .750E-03 .111E-01 .944E-03 .186E-02 .714E-03
      3      1 .000     .000     .222E-04 .000     .000     .000     .000     .636E-03 .431E-03
      4      1 .000     .000     .155E-04 .300E-04 .190E-04 .000     .000     .000     .217E-04
      5      9 .148E-02 .107E-02 .593E-02 .555E-02 .430E-02 .881E-01 .687E-02 .204E-01 .164E-01
      6      3 .198E-03 .240E-03 .868E-03 .520E-03 .353E-03 .000     .715E-03 .793E-02 .568E-02
      7      2 .000     .000     .144E-04 .000     .000     .000     .000     .182E-02 .994E-03
      8      6 .000     .000     .131E-03 .000     .000     .000     .000     .698E-02 .990E-02
      9      1 .160E-03 .251E-04 .356E-03 .230E-03 .181E-03 .000     .229E-03 .000     .446E-03
     10      1 .664E-05 .134E-04 .318E-04 .122E-04 .726E-05 .000     .186E-04 .769E-03 .492E-03
     11      4 .119E-02 .112E-03 .243E-02 .246E-02 .221E-02 .000     .220E-02 .000     .129E-02
     12      1 .000     .000     .204E-05 .000     .000     .000     .000     .000     .302E-03
     13      1 .000     .000     .298E-06 .000     .000     .000     .000     .000     .104E-03

[...]

    139      1 .000     .000     .628E-03 .000     .556E-03 .000     .000     .000     .000
    140      1 .000     .000     .662E-04 .000     .000     .000     .000     .297E-02 .110E-02
    141      2 .169E-02 .000     .000     .000     .248E-02 .000     .000     .000     .000
    142      1 .000     .000     .186E-03 .000     .000     .000     .000     .000     .647E-03
  _____________________________________________________________________________________________
  total  1518  .92420   .99521   .95511   .95303   .94985  1.00000   .98140   .89922   .88749
  max freqs      156      445       49      151      107      256       26      115       79

                                        Population
   Type  Size     10       11       12       13       14
  _________________________________________________________
      1      4 .111E-02 .196E-02 .000     .000     .110E-02
      2      7 .100E-01 .204E-01 .285E-02 .612E-03 .491E-02
      3      1 .391E-02 .259E-02 .107E-02 .336E-05 .419E-03
      4      1 .472E-03 .170E-02 .000     .527E-04 .356E-03
      5      9 .534E-01 .653E-01 .342E-01 .269E-02 .221E-01
      6      3 .274E-02 .127E-01 .730E-02 .183E-03 .311E-02
      7      2 .536E-03 .426E-03 .272E-02 .376E-06 .134E-03
      8      6 .209E-01 .828E-02 .129E-01 .148E-04 .189E-02
      9      1 .306E-04 .100E-02 .000     .593E-04 .563E-03
     10      1 .351E-04 .613E-03 .389E-03 .310E-05 .109E-03
```

```
     11      4 .597E-03 .517E-02 .000     .874E-03 .401E-02
     12      1 .987E-03 .690E-03 .000     .127E-05 .137E-03
     13      1 .506E-04 .134E-03 .000     .865E-07 .192E-04

[...]

    139      1 .000     .000     .000     .000     .000
    140      1 .398E-04 .626E-04 .115E-02 .204E-05 .113E-03
    141      2 .537E-04 .183E-03 .000     .253E-02 .120E-03
    142      1 .130E-04 .724E-04 .000     .195E-04 .216E-03
  _________________________________________________________
  total  1518  .90130   .86909   .91325   .93396   .93455
  max freqs        32       64        7       21       10

  Class   2

                                       Population
   Type  Size      1        2        3        4        5        6        7        8        9
  _____________________________________________________________________________________________
    143      1 .164E-03 .251E-04 .356E-03 .230E-03 .186E-03 .000     .229E-03 .000     .446E-03
    144      1 .986E-04 .280E-02 .128E-02 .376E-03 .519E-03 .000     .130E-02 .218E-01 .327E-01
    145      2 .176     .420E-01 .126     .200     .136     .732E-01 .138     .313E-01 .915E-02
    146      2 .114E-01 .110     .273E-01 .136E-01 .160E-01 .000     .261E-01 .333E-01 .182E-01
    147      1 .852E-01 .491     .186     .145     .195     .145     .250     .859E-01 .526E-01
    148      1 .686E-01 .514E-01 .764E-01 .641E-01 .100     .000     .802E-01 .000     .413E-02
  _____________________________________________________________________________________________
  total     8 .34118   .69733   .41735   .42276   .44794   .21822   .49605   .17236   .11724
  max freqs         0        3        0        0        1        0        0        0        1

                                       Population
   Type  Size     10       11       12       13       14
  _________________________________________________________
    143      1 .325E-04 .105E-02 .000     .627E-04 .566E-03
    144      1 .389E-02 .106E-01 .221E-01 .213E-03 .351E-02
    145      2 .616E-01 .650E-01 .226E-01 .207     .118
    146      2 .421E-02 .101E-01 .145E-01 .155E-01 .187E-01
    147      1 .822E-01 .519E-01 .679E-01 .228     .133
    148      1 .918E-03 .411E-02 .000     .738E-01 .242E-01
  _________________________________________________________
  total     8 .15282   .14265   .12708   .52441   .29850
  max freqs         0        1        0        2        0

  Class   3

                                       Population
   Type  Size      1        2        3        4        5        6        7        8        9
  _____________________________________________________________________________________________
    149      1 .150E-02 .139E-01 .132E-01 .638E-02 .998E-02 .174     .170E-01 .779E-01 .177
    150      1 .000     .268E-02 .375E-02 .000     .397E-02 .000     .000     .000     .000
    151      2 .174     .545     .281     .230     .308     .145     .341     .119     .988E-01
  _____________________________________________________________________________________________
  total     4 .17543   .56131   .29774   .23621   .32210   .31946   .35779   .19720   .27589
  max freqs         0        2        0        0        1        0        0        0        1

                                       Population
   Type  Size     10       11       12       13       14
  _________________________________________________________
    149      1 .105     .664E-01 .149     .413E-02 .325E-01
    150      1 .000     .000     .000     .000     .000
    151      2 .113     .631E-01 .974E-01 .300     .173
  _________________________________________________________
  total     4 .21799   .12949   .24599   .30424   .20587
  max freqs         0        0        0        0        0

[...]
```

Appendix 12: Conditional population probability output file (excerpts)

```
Columbia Example
  Conditional Population Probabilities

  Mixture Sample Summary

  Number of types       :    187
  Number of classes     :     15
  Number of individuals :   1597

     Class  Types  Indviduals
         1    142      1518
         2      6         8
         3      3         4
         4      2         6
         5      7        10
         6      2         2
         7      8        25
         8      4         8
         9      2         2
        10      2         2
        11      4         5
        12      2         4
        13      1         1
        14      1         1
        15      1         1

  Class   1

                                        Population
   Type  Size      1        2        3        4        5        6        7        8        9
  ____________________________________________________________________________________________
      1      4  .0000    .0000    .0000    .0000    .0000    .0000    .0000    .0000    .0000
      2      7  .0067    .0032    .0977    .0001    .0685    .0000    .0000    .0315    .0280
      3      1  .0000    .0000    .0192    .0000    .0000    .0000    .0000    .0955    .1501
      4      1  .0000    .0000    .0236    .0000    .0270    .0000    .0000    .0000    .0133
      5      9  .0026    .0303    .1154    .0000    .0782    .0000    .0000    .0690    .1287
      6      3  .0018    .0343    .0856    .0000    .0326    .0000    .0000    .1358    .2259
      7      2  .0000    .0000    .0157    .0000    .0000    .0000    .0000    .3445    .4369
      8      6  .0000    .0000    .0156    .0000    .0000    .0000    .0000    .1437    .4729
      9      1  .0105    .0266    .2599    .0001    .1239    .0000    .0000    .0000    .1311
     10      1  .0010    .0325    .0531    .0000    .0113    .0000    .0000    .2230    .3310
     11      4  .0116    .0177    .2634    .0001    .2237    .0000    .0000    .0000    .0562
     12      1  .0000    .0000    .0055    .0000    .0000    .0000    .0000    .0000    .3304
     13      1  .0000    .0000    .0035    .0000    .0000    .0000    .0000    .0000    .4976

[...]

    139      1  .0000    .0000    .5469    .0000    .4531    .0000    .0000    .0000    .0000
    140      1  .0000    .0000    .0591    .0000    .0000    .0000    .0000    .4602    .3971
    141      2  .0584    .0000    .0000    .0000    .8928    .0000    .0000    .0000    .0000
    142      1  .0000    .0000    .3020    .0000    .0000    .0000    .0000    .0000    .4237
  ____________________________________________________________________________________________

                                        Population
   Type  Size     10       11       12       13       14
  ____________________________________________________
      1      4  .0000    .3482    .0000    .0000    .6518
      2      7  .0000    .4254    .0000    .0001    .3389
      3      1  .0000    .4787    .0000    .0000    .2566
      4      1  .0000    .5527    .0000    .0001    .3833
      5      9  .0000    .2712    .0000    .0001    .3045
      6      3  .0000    .2667    .0000    .0000    .2171
      7      2  .0000    .0991    .0000    .0000    .1038
      8      6  .0000    .2095    .0000    .0000    .1583
      9      1  .0000    .1565    .0000    .0000    .2914
     10      1  .0000    .2189    .0000    .0000    .1292
     11      4  .0000    .1197    .0000    .0001    .3076
     12      1  .0000    .4005    .0000    .0000    .2635
```

```
    13       1  .0000     .3378     .0000     .0000     .1611

[...]

   139       1  .0000     .0000     .0000     .0000     .0000
   140       1  .0000     .0120     .0000     .0000     .0717
   141       2  .0000     .0150     .0000     .0010     .0328
   142       1  .0000     .0252     .0000     .0000     .2491
 ______________________________________________________________

 Class    2

                                    Population
  Type   Size       1         2         3         4         5         6         7         8         9
 _____________________________________________________________________________________________________
   143       1  .0106     .0263     .2567     .0001     .1256     .0000     .0000     .0000     .1296
   144       1  .0003     .1476     .0466     .0000     .0176     .0000     .0000     .1373     .4780
   145       2  .0361     .1395     .2879     .0002     .2915     .0000     .0000     .0124     .0084
   146       2  .0044     .6892     .1179     .0000     .0649     .0000     .0000     .0250     .0318
   147       1  .0062     .5790     .1513     .0000     .1482     .0000     .0000     .0121     .0172
   148       1  .0228     .2758     .2822     .0001     .3462     .0000     .0000     .0000     .0061
 _____________________________________________________________________________________________________

                                    Population
  Type   Size      10        11        12        13        14
 ______________________________________________________________
   143       1  .0000     .1619     .0000     .0000     .2893
   144       1  .0000     .0821     .0000     .0000     .0904
   145       2  .0000     .0317     .0000     .0005     .1918
   146       2  .0000     .0093     .0000     .0001     .0574
   147       1  .0000     .0090     .0000     .0002     .0767
   148       1  .0000     .0032     .0000     .0003     .0633
 ______________________________________________________________

 Class    3

                                    Population
  Type   Size       1         2         3         4         5         6         7         8         9
 _____________________________________________________________________________________________________
   149       1  .0008     .1222     .0802     .0000     .0566     .0000     .0000     .0820     .4325
   150       1  .0000     .3428     .3301     .0000     .3270     .0000     .0000     .0000     .0000
   151       2  .0099     .5027     .1785     .0001     .1834     .0000     .0000     .0132     .0253
 _____________________________________________________________________________________________________

                                    Population
  Type   Size      10        11        12        13        14
 ______________________________________________________________
   149       1  .0000     .0859     .0000     .0000     .1397
   150       1  .0000     .0000     .0000     .0000     .0000
   151       2  .0000     .0086     .0000     .0002     .0781
 ______________________________________________________________

[...]
```

Appendix 13: The generalized EM (GEM) algorithm

Variables:

n - Number of genotypes observed in the mixture.

M_{ij} - Number of individuals in the mixture with multilocus genotype i from population j: unobservable, random.

m_{ij} - Realization of M_{ij} : unobservable, not random.

$m_{i.}$ - $\Sigma_j\, m_{ij}$: observed.

g_{ij} - Probability of genotype i given the individual is from population j : computed using baseline relative frequencies and assuming each population is in HW equilibrium.

q_j - True contribution of population j to the mixture in the sample: unobservable, unknown.

l_i - Unconditional probability of observing genotype i in the mixture.

$$l_i = \Sigma_j\, g_{ij}\, q_j\ .$$

F_{kl} - Frequency (absolute) of allele k of locus l in the mixture for a random genotype: random, observable.

f_{kli} - Observed allele frequencies of genotype i, $F_{kl} = \{f_{kli}\}$, i=1,n.

p_{jkl} - Relative frequency of allele k, locus l, population j : unobservable for missing baseline populations, known and (assumed) fixed otherwise.

In the typical application of the EM algorithm to GSI, we have for the expectation step

$$E\left[M_{ij}\right] = m_i \frac{q_j g_{ij}}{l_i}, \tag{1}$$

and for the maximization step

$$\hat{q}_j = \frac{\sum_{i=1}^{n} m_{ij}}{m_{..}}. \tag{2}$$

Substituting E[M_{ij}] in (1) for the unobservable m_{ij} in (2) gives the EM algorithm for the GSI problem (Pella et al. 1996).

A problem arises in computing g_{ij} if loci are missing for populations in the baseline. If the m_{ij}s were known, the MLE for the missing baseline allele relative frequencies would be

$$\hat{p}_{jkl} = \frac{\sum_{i=1}^{n} m_{ij} f_{kli}}{\sum_{i=1}^{n} m_{ij} f_{.li}}. \tag{3}$$

As before, substitute E[M_{ij}] in (1) for m_{ij} in (3) to produce estimates for the missing baseline frequencies (Smouse et al. 1990). Hence, a second stage to the M step is created. SPAM performs the GEM in the following algorithm.

1. Initialize missing frequencies to be uniform.
2. Compute g_{ij}.
3. Perform the E step defined by (1).
4. Perform the M_1 step defined by (2).
5. Perform the M_2 step defined by (3).
6. Repeat steps 2-5 until Max($\hat{p}_{jkl}$) < 10^{-8} or GPA achieved.