Manual for InStruct

Hong Gao, Scott Williamson and Carlos Bustamante

May 12, 2007

1 Preliminaries and Acknowledgements:

This is the README file for the program InStruct. It provides a description of the input file format, output file format and instructions on how to run the executable files of this program under distinct Operating System or using the Cornell Computational Biology Service Unit web server `http://cbsuapps.tc.cornell.edu/InStruct.aspx`. The initial version of InStruct was written by Hong Gao in ANSI C, under the guidance of Carlos Bustamante and Scott Williamson and Jarek Pillardy wrote the MPI code for it to run on the CBSU web server. Your running of InStruct is made possible through the support of the Cornell CBSU.

NOTE: InStruct is an alternative program to STRUCTURE especially in the cases of existence of partial self-fertilization or inbreeding. It has the similar data format and output format to facilitate the usage and spread of this software. We suggest users using both programs concurrently to compare results, if applicable.

1.1 Citing InStruct

When publishing results based on this program, please cite "Gao, H., Williamson,S., and Bustamante, C.D. 2007.An MCMC Approach for Joint Inference of Population Structure and Inbreeding Rates from Multi-Locus Genotype Data. Genetics (online)". We ask that you also please acknowledge the Cornell Computational Biology Service Unit (CBSU).

2 Disclaimer and Copyright

3 Introduction

InStruct implements the Markov Chain Monta Carlo algorithm for the generalized Bayesian clustering method to estimate the self-fertilization rates or inbreeding coefficients both at population level and individual level and cluster individuals into subpopulations simultaneously using genotype data consisting of unlinked markers. It can make inference of optimal number of subpopulations underlying a sample via the Deviance Information Criteria. And besides diploid case, it is currently generalized to jointly infer selfing rates and substructure for tetraploid individual, considering both the autotetraploid and allotetraploid. The methods used are introduced in the paper mentioned above and also some papers to be submitted. Applications of our method include inferring population structure with or without admixture, estimation of selfing rates or inbreeding coefficients for subpopulations or individuals and inferring the number of subpopulations.

In InStruct, We assume a model in which there are K populations (where K may be unknown), each of which is characterized by a set of allele frequencies at each locus and a selfing rate. Individuals in the sample are assigned (probabilistically) to populations, or jointly to two or more populations if their genotypes indicate that they are admixed. And each individual has a selfing history characterized by the number of generations of self-fertilization until the most recent event of outcrossing. Based on population composition and selfing extent, selfing rates of each population can be estimated.

This method does not assume a particular mutation process, and it can be applied to a variety of the commonly used genetic markers including microsatellites, SNPs and RFLPs, provided that they are in linkage-disequilibrium. And our model does not assume the Hardy-Weinberg equilibrium within loci.

This document includes information about how to format the data file, how to run on local computers/CBSU clusters, how to choose appropriate parameters, and how to interpret the results. And this program is mainly invoked from the command line.

4 Format for Data File

The input data file is similar to the two data input formats of STRUCTURE (PRITCHARD *et al.*, 2000; FALUSH *et al.*, 2003). It is a space/tab-delimitted plain text file of genotype data, with each haplotype or each individual starting on a separate line. The examples of format for the genotype data are shown in Table 1,2 and 3. Essentially, the entire data set is arranged as a matrix in a single file, in which the data for individuals are in rows, and the loci are in columns.

NOTE: For diploid or tetraploid individuals, the number of the sample size can be counted by InStruct automatically. If the value of sample size obtained by InStruct is not the same as that specified on the command-line, a warning message will be printed and the new value will be used for later running. The same with locus counting. However, when users prepare the input file, pay attention to the end of files, as there should be only one new end line.

4.1 Data Formats for Diploid

Two input formats are acceptable for diploid organisms. If the option "DATA_FMT" is set to be 0, then data for each individual are stored as two consecutive rows, where each locus is in one column, and the order of the alleles for a single individual does not matter. If the "LABEL" option is turned on, the first column stores the labels of individuals. And if "POPDATA" is set to be one, the next column stores the user-defined population information of individuals (for instance these might designate the geographic sampling locations of individuals and note that this predefined group information is not used in later inference, only facilitates the use of Distruct program). If the option "EXTRA_COLS" is greater than zero, there are columns containing extra information about individuals before the genotype data. The pre-genotype data columns (see below) are recorded twice for each individual. The rest columns store the genotype data of individuals. Each column stores the genotype data for a single locus and each allele type should be coded by a unique integer or string at a given locus (e.g. microsatellite repeat score). In addition, if "MARKER_NAMES" is turned on, the first row stores the marker names for all the loci. See one example below.

		M1	M2	M3	M4
G02	NM	2	4	2	1
G02	NM	2	4	1	2
G04	FG	2	3	2	-9
G04	FG	3	3	2	-9

The above shows an example of two diploid individuals with the first column, labels of individuals, the second column, prior population information, followed by four loci columns. The missing data is represented by -9 in this case. And the first line is the marker names. The command line for this input file is written as "-N 2 -L 4 -lb 1 -a 1 -w 1 -x 0 -m -9 -af 0 -p 2".

The second format is that each genome takes one row, i.e., one locus information takes two consecutive columns to store.

		M1		M2		M3		M4	
G02	NM	2	2	4	4	2	1	1	2
G04	FG	2	3	3	3	2	2	-9	-9

The data-set in Table 2 is the same as that in Table 1 but takes the second format. The command line for this input file is written as "-N 2 -L 4 -lb 1 -a 1 -w 1 -x 0 -m -9 -af 1 -p 2".

4.2 Data Formats for Tetraploid

And for the tetraploids, only the second format is accepted via InStruct. The rest options are similar. And there is also a pratical problem for polyploids that the true genotypes are known usually, thus users just put the distinct alleles known per locus per individual and the rest blanks can be represented by missing data. The genotypes can be inferred by InStruct itself. For example,

		M1				M2			
G02	NM	2	4	-9	-9	1	2	-9	-9
G04	FG	2	3	-9	-9	2	-9	-9	-9

Table 3 shows an example of data-set of tetraploids.The command line for this input file is written as ”-N 2 -L 2 -lb 1 -a 1 -w 1-x 0 -m -9 -p 4 -ap 0/1”.

4.3 Missing Genotype Data

Missing data should be indicated by a number or string which doesnt occur elsewhere in the data and the default setting is -9. The missing-data value can be set on the command line using the flag ”-m”.

5 Format for Initial File

The format of a file that provides the prior information is similar to a fasta file. This initial file only contains the prior information of selfing rates. Each independent starting point begins with a greater sign ”>” and its chain name, with the values of selfing rates for subpopulations on the consecutive line and delimitted by a space or spaces.
NOTE: The initial file is not necessary for running the program. If the initial file is not specified, the starting points are randomly generated. If the number of starting points is less than the chain number, the starting points not-specified will be randomly generated. If the number of starting points is greater than the chain number, an error message is reported and the program is terminiated. The following is the format of initial file and an example.
Format:
>chain_name1
num1 num2...

>chain_name2
num3 num4...
.....

Example:
>chain#1
0.4325 0.5438 0.3573

>chain#2
0.3459 0.6783 0.1334

The above example shows two independent starting points of selfing rates for three subpopulations.

6 Inference of the Number of Clusters

If users need to make inference of the number of clusters formed by classification of individuals, they have to turn on the boolean variable "INF_K" using the command-line flag "-ik". And users need to specify the range of the value of K, the number of clusters, via the command-line flag "-kv", followed by two integers separated by spaces, the first integer represents the lower bound of K and the second integer represents the upper bound of K. The value of K should be positive integers, thus if either the lower bound or the upper bound is less than 1 or the upper bound is less than the lower bound, the range of K will be reset to the default value, with the lower bound equal to 1 and the upper bound equal to $1 + [n^{0.3}]$, where n is the sample size, which is recommended by BOZDOGAN (1993).

After the output file is generated, you can find the last line indicating the optimal K value favored by the Deviance Information Criteria, which just implies that the model assuming the optimal K value fits the data most.

7 Running InStruct

Users can either download the executable files of InStruct from the web page
`http://cbsuapps.tc.cornell.edu/InStruct.aspx` and run it on the local computers or submit jobs to CBSU clusters from the above web access.
We provide the executables under three OS, Windows, Linux and Mac OS X. It can be downloaded based on the user's need and the source code can be requested from the authors if necessary. The current version of InStruct does not have GUI interface so they need to be invoked from the commandline.
If you would like to see all the parameters to be specified, you might type the following to get help information (current working directory assumed, otherwise users need to specify the directory for these executables).
For Windows: InStruct.exe -h
For Linux or OS X: InStruct -h
Users can specify many parameters on the command-line according to their needs or just use the default setting. And the parameters used in InStruct are introduced in details in the following tables.

According to previous running experience, many users tend to request large amount of memory either on their local machines or on CBSU cluster. Thus, one option "MAX_MEM" is added to InStruct to limit the maximum memory required, which can be changed via the command-line flag "-mm". Normally if the sample size or the number of loci of a data-set is very large, users can enlarge burn-in time and increase thinning interval to avoid the memory allocation problem.

For diploid individuals, there are six modes of analysis. Mode 0 is to infer population structure only without admixture and the rest modes infer population structure with admixture. And you need to select the "MODE" via the command-line flag "-v", no matter whether you make inference of the number of clusters or not. For Mode 3 and Mode 5, you can select the choice of the Prior for individual selfing rates/inbreeding coefficients. However, the Dirichlet Process prior is not avail-

able for Mode 5 currently.
InStruct currently can only jointly infer selfing rates and population structure for tetraploid individuals, for the two distinct types of polyploids, autotetraploid and allotetraploid. As diploid is the default setting, users need to use the flag "-p 4 -ap 0/1" to tell InStruct that the species is tetraploid and it is autotetraploid/allotetraploid.

Here presents an example of running InStruct:

InStruct -d data.txt -o output.txt -K 10 -v 2 -x 0 -w 1 -j 2000 -e 0 -f 0 -L 17 -N 293 -p 2 -u 200000 -b 100000 -t 10 -c 5 -s 862638 370749 135155 -m -1 -sl 0.95 -lb 1 -a 0 -g 1 -r 2000 -pi 1 -pf 1 -ik 1 -kv 1 5 -df 1 -af 0 -mm 2.0e9

The above illustrates the followings:
"-d data.txt" means the input data file is data.txt;
"-o output.txt" means the output file's name is output.txt;
"-K 10" means ten subpopulations are assumed for this sample; "-v 2" means Mode 2 is selected to perform joint inference of population selfing rates and population substructure;
"-x 0" means there are no extra columns of information of individuals besides labels or predefined subpopulations;
"-w 1" means the first row of the input file is the marker names;
"-j 2000" means the number of retained iterations to check whether there is any empty cluster;
"-e 0" means the proposal method for selfing rates is adaptive independence sampler;
"-f 0" means the prior for selfing rates is uniform distribution;
"-L 17" means the number of loci is 17;
"-N 293" means the sample size is 293;
"-p 2" means individuals are diploid;
"-u 200000" means the total number of iterations is 200000;
"-b 100000" means the number of burn-in iterations is 100000;
"-t 10" means the length of thinning interval is 10 iterations;
"-c" means the number of chains to run is 5;
"-s 862638 370749 135155 " means to reset the seeds for the random number generator;
"-m -1" means the missing data are represented by -1;
"-sl 0.95" means the significance level used in summarizing posterior credible intervals of parameters is 0.95;
"-lb 1" means the first column of the input file is the labels of individuals;
"-a 0" means there is no column of information about geographical locations of individuals;
"-g 1" means to use the Gelman-Rubin statistics to check the convergence among chains;
"-r 2000" indicates the number of retained iterations to be used to check the convergence of chains;
"-pi 1" means to print the likelihood and some parameters along the MCMC run;
"-pf 1" means to print the summarized allele frequencies of subpopulations into the result file;
"-ik 1 -kv 1 5" means to infer the number of clusters underlying the sample by trying K=1 to 5, then choosing the K value with the lowest DIC values;
"-df 1" means to output results in the Distruct format to draw the genome assignment plot;
"-af 0" means the input file format is that each haploid takes one row;
"-mm 2.0e9" means the maximum memory allowed for this run is 2.0e9 byte.

Parameter	Data_type	Command line flag	Default value	Interpretation
DATAFILE	string	-d	NULL	name of input genotype data file
INITFILE	string	-i	NULL	name of the file containing the prior information
OUTFILE	string	-o	NULL	name of output file
NUM_INDV	integer	-N	100	number of diploid individuals in data file
NUM_LOCI	integer	-L	100	number of loci in data file
LABEL	boolean	-lb	1	indicate whether datafile contains labels (names) for each individual. 1 = Yes,0 = No
POPDATA	boolean	-a	1	indicate whether datafile contains original population classification information of each individual. 1 = Yes,0 = No
EXTRA_COLS	integer	-x	0	indicate the number of columns containing extra information about individuals besides Label and PopData
MARKER_NAMES	boolean	-w	0	indicate whether there exists a line of marker names or not at the beginning of data file 1 = Yes, 0 = No
PLOID	integer	-p	2	number of haplotypes in an individual's genome
MISS_DATA	integer	-m	-9	value used to represent the missing data
POP_NUM	integer	-K	2	number of subpopulations assumed
DATA_FMT	boolean	-af	0	two data formats accepted, 0=one haploid per line, 1=one individual per line
MAX_MEM	double	-mm	1.0e9	maximum memory allowed

Table 1: Description of parameters related to input and output.

Note: The accurate values of number of individuals or loci are not necessary as the program will automatically count the number of individuals and loci.

Parameter	Data_type	Command line flag	Default value	Interpretation
MODE	integer	-v	1	indicate which function of this program to use 0 = infer population structure only without admixture 1 = infer population structure only with admixture 2 = infer population structure and population selfing rates 3 = infer population structure and individual selfing rates 4 = infer population structure and population inbreeding coefficients 5 = infer population structure and individual inbreeding coefficients
INF_K	boolean	-ik	0	infer the number of subpopulations existing (1) or not (0)
K_LOWER K_UPPER	integer integer	-kv -kv	1 0	the lower bound of K the upper bound of K
CHN_NUM	integer	-c	5	number of independent chains for the MCMC algorithm ($\geq$1)
ITER_NUM	integer	-u	1000000	total number of iterations for each chain including burn-in
BURNIN	integer	-b	500000	number of initial steps of the MCMC to discard before retaining draws ($\geq 10^3$)
THINNING	integer	-t	10	length of an interval between two draws in MCMC
GR_FLAG	boolean	-g	1	indicate whether the Gelman-Rudin statistic is used to check convergence of MCMC's.1 = Yes,0 = No
CONV_ITER	integer	-r	2000	number of retained iterations after burn-in that are used for convergence checking
CONVFILE	string	-cf	NULL	the file name to which users would like to output the updated results for convergence assessment along the MCMC after burn-in

Table 2: Description of MCMC parameters.

Parameter	Data_type	Command line flag	Default value	Interpretation
SAMPLER	integer	-e	0	indicate which sampler to propose selfing rates or inbreeding coefficient under Mode 2 or Mode 4 0 = Adaptive Independence Sampler 1 = Back-Reflection Sampler
PRIOR_CHOICE	integer	-f	0	indicate which sampler to propose individual selfing rates or inbreeding coeffic under Mode 3 or Mode 5 0 = Back-Reflection Sampler 1 = Dirichlet Process Prior
PRINT_ITER	boolean	-pi	0	indicate whether to print the updated information of sampled iteration to the standard output (1) or not (0)
PRINT_FREQ	boolean	-pf	0	indicate whether to print the summarized allele frequencies to output file (1) or not (0)
EMPTY_CLUSTER	integer	-j	2000	indicate the number of iterations after burn-in used to determine the existence of empty clusters. If zero, then not check for the existence of empty cluster
DISTR_FMT	boolean	-df	1	indicate whether to use the Distruct format for output file (1) or not (0)
SIG_LEVEL	float	-sl	0.9	significance level for the confidence interval of the posterior distribution of parameters
SEEDS	three integers	-s	13,4,1972	three seeds for the Wichmann random number generator

Table 3: Description of Miscellaneous parameters related to MCMC.

8 Output files

One output file is available from InStruct, containing the summary of the marginal posterior distribution of the parameters. The beginning of an output file looks similar to that of STRUCTURE, which just reports the parameter setting for running of this program, followed by summary of parameter estimation from the program.

NOTE: If one would like to use the Distruct software (ROSENBERG *et al.*, 2002) to draw the bar plot of individual genome assignments, one needs to set the option "DISTR_FMT" to be 1, otherwise the format is shown below.

An example is provided below:

Chain#1:

The log Likelihood:
Posterior Mean = -2491.0177
Posterior Median = -2486.4640
95% Posterior Credible Interval is 0.0000 ∼ -2465.7651

The Deviance information criterion of this model is 6849.403864.

The Posterior distribution of Selfing Rates:

	Mean	Median	95% Credible Interval	Min	Max
Cluster 1	0.3795	0.3705	0.0000 ∼ 0.4005	0.3568	0.4273
Cluster 2	0.7975	0.7954	0.0000 ∼ 0.8271	0.7781	0.8271
Cluster 3	0.6900	0.7137	0.3750 ∼ 0.9223	0.0091	0.9939
Cluster 4	0.7372	0.7372	0.7372 ∼ 0.7372	0.7372	0.7372

To test whether the selfing rates of subpopulations are significant different.
Here is a p-value table of tests that self_rate_i is not equal self_rate_j,
where i is the row index and j is the column index.

	Cluster 2	Cluster 3	Cluster 4
Cluster 1	0.000000	0.003000	0.000000
Cluster 2		0.398120	0.000000
Cluster 3			0.892480

The Posterior distribution of Generations:

	Mean	Median	90% Credible Interval	Min	Max
Individual 1	2.0944	2.0000	1.0000 ∼ 4.0000	1.0000	50.0000
Individual 2	1.5446	1.0000	1.0000 ∼ 3.0000	1.0000	35.0000
Individual 3	1.6167	1.0000	1.0000 ∼ 3.0000	1.0000	22.0000
Individual 4	1.1462	1.0000	1.0000 ∼ 2.0000	1.0000	41.0000

.....

Inferred ancestry of individuals:

Indv Label (Miss) Pop Cluster 1:Mean Median 95% CI Cluster 2:Mean Median 95% CI
1 1 0.0000 1 0.0179 0.0165 0.0000 ∼ 0.0290 0.9821 0.9808 0.0000 ∼ 0.9975
2 2 0.0000 1 0.0106 0.0074 0.0000 ∼ 0.0186 0.9894 0.9879 0.0000 ∼ 0.9974
3 3 0.0000 1 0.4681 0.6994 0.0000 ∼ 0.7974 0.5319 0.2456 0.0000 ∼ 0.9981
.....

The result for each chain is reported consecutively. First the distribution of log likelihood is presented, followed by the estimated selfing rates for each subpopulations. And the difference between any pair of subpopulations is reported, too.
Then the results for Q are presented in the last part shown above (here K was set to 2). This is read as follows. Reading from row 1: Individual label (taken from data file) = 1; percentage of missing data for this individual = 0%; user-assigned population = 1; estimated percentage in clusters 1 = 0.0179, i.e., the mean value of q_{11} ; the median value of q_{11} =0.0165 ; 95% probability interval of q_{11} is (0.0000, 0.0290); next are the mean, median and 95% probability interval of q_{12}.
If the option "DISTR_FMT" is set to be 1, then only the mean values of Q is output so that they can be copied and pasted directly in the Distruct's format.

Along the MCMC running, the output from each iteration has the following format. For each update step, the current log likelihood and the selfing rates for each subpopulations are reported. The below is an example with two subpopulations assumed:

Step=0 log_likelihood=-3627.873236 selfing rate=0.290911 selfing rate=0.786113
Step=1 log_likelihood=-2954.802754 selfing rate=0.307254 selfing rate=0.814279
Step=2 log_likelihood=-2817.757203 selfing rate=0.316779 selfing rate=0.844672
Step=3 log_likelihood=-2622.290645 selfing rate=0.318240 selfing rate=0.834975
.....

References

BOZDOGAN, H., 1993 *Choosing the number of component clusters in the mixture-model using a new informational complexity criterion of the inverse-Fisher information matrix, Information and Classification, O. Opitz, B. Lausen, and R. Klar (eds.)*. Heidelberg: Springer-Verlag.

FALUSH, D., M. STEPHENS and J. K. PRITCHARD, 2003 Inference of population structure using multilocus genotype data: linked loci and correlated allele frequencies. Genetics **164**: 1567–87.

PRITCHARD, J. K., M. STEPHENS and P. DONNELLY, 2000 Inference of population structure using multilocus genotype data. Genetics **155**: 945–59.

ROSENBERG, N., J. K. PRITCHARD, J. L. WEBER, H. CANN, K. KIDD *et al.*, 2002 Genetic structure of human populations. Science **298**: 2381–5.