NAME

FRANz − Pedigree reconstruction.

VERSION

2.0.0

SYNOPSIS

FRANz [Options] *infile*

DESCRIPTION

FRANz reconstructs pedigrees (family trees) using polymorphic, codominant markers.

OPTIONS

Prior Information:

--femrepro *i:i*

Age range in which females can reproduce (default 0:1000).

--malerepro *i:i*

Age range in which males can reproduce (default 0:1000). For individuals with unknown sex, the maximum age range is used. For example,

--femrepro *13:43* **--malerepro** *14:60*

here, individuals with unknown sex are not excluded from age 13 to 60 as candidate parents.

--N *i* **[**--*Nf* **i** --*Nm* **i***]*

For paternity inference, N is the number of candidate fathers in the population (default auto) [NMCP01]. This is the sum of the average number of sampled (n) and unsampled (N-n) breeding males in the population. If unset, then estimated jointly with the pedigree. For parentage inference N = Nf = Nm (NOT Nf + Nm), but one can also specify Nf and Nm instead of N if these numbers differ. If N is not known, use Nmax instead.

--Nmax *i* **[**--*Nfmax* **i** --*Nmmax* **i***]*

Maximum number of candidate fathers in the population. This is the estimated upper limit of N when N is not known. FRANz will then incorporate the uncertainity of N in the pedigree reconstruction. We need this limit to avoid that the Markov Chain converges to a very high N, which would result in an empty pedigree. When mothers are unknown and the numbers of males and females differ, we can again specify Nfmax and Nmmax.

--n *i* Number of sampled candidate parents (default auto) [NMCP01]. This feature can be used in combination with --N for setting the number of unsampled candidates (N-n) directly (which might be useful in the absence of age data).

--pedigreein *FILE*

Filename of the pedigree input file.

--fullsibin *FILE*

Filename of the fullsib input file.

--halfsibin *FILE*

Filename of the halfsib input file. Individuals that are fullsibs OR halfsibs (e.g. for nest structured data). EXPERIMENTAL FEATURE.

--geofile *FILE*

Pairwise distances of sampling locations.

--coordfile *FILE*

Coordinates of sampling locations.

Parentage Options:

--selfing

Allows selfing.

--mintyped *i*
Minimum number of typed loci (default 1+numloci/2). Individuals with less typed loci will be ignored. This number also defines the minimum number of common typed loci for a pair of individuals.

--maxmismatching *i,i*
Maximum number of mismatching loci for pairs and triples. If unset, then the mismatch distributions generated by the simulation are used to calculate this values. This procedure strongly depends on the estimated typing error rate.

--numloci *i*
Use only the first i loci. For testing purposes mainly.

--typingerror *f*
Rate of typing error. If some parent-offspring relationships are known (with --pedigreein), then the observed mismatches in these relationships are used to estimate the typing error. The minimum number of relationships is calculated with the standard statistical methods (z=1.96, 20% accepted error). If the observed number is too low, we assume that the error rate is constant across loci and devide the required number by the number of loci. If this number is not reached (or no relationships are known), the default error rate of 0.01 is used. The minimum estimated error rate is 0.005.

--femmaxdist *f*
Maximal distance of sampling locations for females. This is the maximal allowed distance between mother and offspring.

--malemaxdist *f*
The same for males (candidate fathers). For individuals with unknown sex, the maximum of the both distances are used (and unlimited if only one is specified).

--parentsmaxdist *f*
And the maximal distance between mother and father.

--sibmaxdist *f*
And the same for siblings.

Fullsib Options:

--[no]fullsibtest
Detect siblings. This turns our fullsib heuristic as described in [RSK09] on or off (default --nofullsibtest).

--fullsibparental
Detect fullsibs also in parental generation?

--fullsibpvmethod *i*
The p-Value correction method for multiple testing. 1 = Benjamini-Hochberg, 2 = Holm. Default 1.

--fullsibpvth *f,f,f*
The p-Value threshold of the sibling filter. For 0, 1 or 2 common "compatible parental genotypes". Such sampled genotypes are compatible to an offspring genotype according the Mendelian laws and the typing error rate. Default is 0.001,0.001,0.05. This means that if individual A and B have no common compatible parental genotypes, a threshold of 0.001 is used. The idea behind these different thresholds is that if A and B have a common compatible parent pair, this is an additional hint that A and B are fullsibs. So we should use a less conservative p-Value threshold (0.05).

--fullsibH0 *i,i,i*
Defines the null hypotheses (PO,HS,U) and their prior probabilities. Examples:

0,0,1 : FS vs. Unrelated.

1,2,1*: FS vs. PO, 2xHS(HS,Aunt/Uncle), U.

2,4,1#: FS vs. 2xPO (PO and OP),4xHS(HS,Aunt/Uncle,Grandparent/-child), U.

*:default with age data, #: default without age data

Allele frequency Options:

--freqin *FILE*
Filename of allele frequency input file (optional). Requires --noupdatefreqs (see below).

--[no]updatefreqs
Update Allele Frequencies (default --noupdatefreqs). When turned off, individuals are treated as unrelated and all genotypes are used to estimate the population allele frequencies. This is a reasonable assumption when the dataset is large and the average family size is small. Turned on, the allele frequencies are updated by using only the founders (indegree 0 and 1). In the MH iterations, the frequencies are updated after a swap event (for technical reasons).

Simulation Options:

--simiter *i*
Number of simulation iterations (default 50000)

--proportiontyped *f*
Proportion of typed loci (default auto)

--simselfingrate *f*
Proportion of self-fertilization. Requires --selfing. If not specified, then the selfing rate estimated from observed average loss of heterozygosity is used in the simulations. You can use third party software to find better estimates (see section DATA CONVERSION).

HWE exact test options:

--hwesteps *i*
Number of steps (default 2000) [GT92].

--hwechunks *i*
Number of chunks (default 200) [GT92].

--hwechunksize i
The chunk size (default 1000) [GT92].

Pedigree Constraints:

--maxdepth *i*
Max. pedigree depth (generations). Rejects pedigrees with a larger depth in the Markov Chain Monte Carlo (MCMC) sampling. Without age data and if there are many undetected fullsibs, this CAN improve the accuracy by preventing deep "fullsib cascades" (see Fig. 1a in [RSK09]). But use with care. EXPERIMENTAL FEATURE.

MCMC Parameters:

--[no]gibbsmissing
Gibbs sampling of missing data (default --nogibbsmissing). Roughly spoken, FRANz can fill missing data with random alleles during pedigree reconstruction. You can turn this on and off with this flags. EXPERIMENTAL FEATURE.

Simulated Annealing (SA) Parameters:

--sachains *i*
Number of chains (default 2) [Alm03].

--samaxiter *i*
Max. number of iterations (default 100000000) [Alm03].

--sachi *f*
Initial acceptance probability (default 0.95) [Alm03].

--sacstart *f*
Sets the initial temperature, thus deactivating the initial temp. calculation --sachi.

--sabeta *f*
Neighbourhood size factor (default 3.000) [Alm03].

--sadelta *f*
Increment (default 0.100) [Alm03].

--saepsilon *f*
The convergence tolerance (default 0.001000) [Alm03].

--sanepsilon *i*
Convergence events (default 3) [Alm03].

--saexactmax *i*
Do exhaustive enumeration instead of SA if dataset contains less than i+1 individuals (default 25) [CR09]. Does not work with --gibbsmissing or --Nmax. **The required memory grows extremely fast (2^i):** 28 is the maximum value on a computer with 4GB RAM.

Metropolis Hastings Parameters:

--mhchains *i*
Number of chains (default number of CPU cores). When i is > 1, then we do a MCMCMC sampling. See below.

--mhburniniter *i*
Number of burnin iterations. After starting from a random pedigree configuration, we start the normal MH algorithm but do not sample pedigrees in this burnin phase (default 500000).

--mhiter *i*
Number of iterations (default 3000000).

--mhsamplefreq *i*
Sample every ith pedigree (default 10).

--mhswapfreq *i*
For MCMCMC: try to swap every ith iteration (default 25). If --mhchains is greater than 1, then we do a MCMCMC sampling (the default on a multicore CPU if FRANz is compiled with the --enable-openmp flag). That is, we swap the states of a random pair of chains and accept this swap with the normal MH acceptance function. The chains 2, .. n are heated, where the temperature of the ith chain is 1 / (1 + (i - 1) * T). T is specified via --mhtemp.

As all threads have to wait during the swapping, it is a good time to update allele frequencies, so we do that if --updatefreqs is set.

--mhtemp f
For MCMCMC: the temperature of the MCMCMC (default 0.500). This temperature is used to calculate the heat of the ith chain. See --mhswapfreq.

Output options:

--out *FILE*
Filename of the summary output file (default summary.txt)

--lociout *FILE*
Filename of the loci summary output file (default locisummary.txt)

--mismatchout *FILE*
Filename of the mismatches output file (default mismatches.txt)

--freqout *FILE*
Filename of allele frequency output file.

--pout *FILEPREFIX*
Prefix of the parentage output file(s) (default parentage). A prefix is here a filename without the filename extension (.txt, .csv, ...). The filename extension (suffix) is determined by the output format, see below.

--poutformat *i,i*
Format(s) of the parentage outfile(s). The parameter is a list of output formats:

1: Most likely parentages (.csv)

2: All with positive LOD (.csv)

Default "1"

--simulationout *FILE*
Filename of the simulation result file (default simulation.txt)

--siblingsout *FILE*
Prefix of the siblings output file (default siblings)

--siblingsoutformat *i,i*
Format(s) of the siblings outfile(s). The parameter is a list of output formats:

1: FRANz format (.dat)

2: Text format (.txt)

3: CSV format (.csv)

Default "2" (Text)

--pedigreeout *FILEPREFIX*
Prefix of pedigree output files (default pedigree)

--pedigreeoutformat *i,i*
Format(s) of the pedigree outfile(s). The parameter is a list of output formats:

1: FRANz format (.dat)

2: Graphviz format (.dot));

3: Text format (Id Sire Dam) (.txt)

Default "1,2" (FRANz and Graphviz)

--mcmclog *FILE*
Filename of MCMC log file (default mcmc.log)

--hwetestout *FILE*
Filename of the detailed HWE test results. Print the output of the original implementation [GT92] in the specified file.

--missingout *FILE*
Filename of the missing data Gibbs sampler results.

Data conversion options:

--cervusgenotypeout *FILE*
Output the genotypes in CERVUS (CSV) format [KTM07].

--cervusoffspringout *FILE*
Output a CERVUS offspring file [KTM07].

--parenteout *FILE*
Output the genotypes in PARENTE format [CBM02].

--genepopout *FILE*
Output the genotypes in Genepop format [R07].

--rmesout *FILE*
Output the genotypes in RMES format [DPVCG07].

Program options:

--seed *i* seed for random numbers (default: time)

-v
--verbose
increase verbosity level (standard level: 1)

-q
--quiet quiet mode, no output except errors and warnings is generated (=verb. level 0)
-h
--help the basic options

--helpall
show all options

QUICK START

Input file

You might be confused after scrolling over so many options. However, most options have good default values and you will only need to set a few of them.

Although FRANz is a command line tool, it is quite user friendly once you have your data in the input file format. This format is very similar to the one of the Migrate and Phylip programs:

```
1 3 / SIMPSONS
7 Springfield
Grampa     1 1920 ? M 110/100 200/208 ?/?
Homer      1 1950 ? M 110/170 200/210 300/302
Bart       1 1982 ? M 110/120 200/212 302/304
Lisa       1 1980 ? F 140/170 200/218 302/306
Maggie     1 1988 ? F 110/140 210/212 300/304
Marge      1 1952 ? F 120/140 212/218 ?/306
Flanders   1 ? ? ? 150/160 214/220 300/?
```

(Note: We know that this format is not as common as Excel (CSV) files, but it has several advantages and we provide an user friendly conversion tool on our website. See section IMPORT FROM CSV at the end of this manual.)

The first line in this file,

```
1 3 / SIMPSONS
```

says the dataset includes one sampling location and three loci. The alleles of diploid genotypes are separated by a slash (/), and the dataset title is "SIMPSONS". The second line is for the first (and in this case the only) sampling location:

```
7 Springfield
```

This means 7 genotypes in sampling location "Springfield". Now we come to the genotypes:

```
Grampa     1 1920 ? M 110/100 200/208 ?/?
```

The first ten characters (just like in Migrate or Phylip) are a description of the genotype or individual. If the genotype ID is shorter than 10 characters, you have to fill the remaining characters with spaces:

```
Grampa     1 1920 ? M 110/100 200/208 ?/?   #VALID
Grampa 1 1920 ? M 110/100 200/208 ?/?       #INVALID
```

Then, the next number is how often this genotype was observed. This is meant for clonal organisms as

desribed in [RSK10]. The 1920 is year of birth of Grampa, ? his year of death (unknown), M his sex (F for females and ? if unknown). The rest of the line is reserved for the 3 diploid loci.

First FRANz run

Now, run FRANz with this Simpsons example file (the $ visualizes the Command Prompt, don't type it):

```
$ FRANz --Nmax 2 simpsons.dat
```

The leading "--" before the parameters is important! With --Nmax 2 we say that every offspring has not more than two candidate fathers in our population - and for parentage inference also not more than two candidate mothers.

IMPORTANT: if you have a good estimate of the number of unsampled candidate parents, use the --N instead of the --Nmax options. See also the section FRANz RUNS FOREVER. If both --N or --Nmax are omitted, then a complete sampling is assumed and the pedigree that maximizes the mendelian segregation probabilities is returned.

Now you will get a warning because you have to specify the age range in which an individual can reproduce sexually:

```
$ FRANz --Nmax 2 --femrepro 14:45 --malerepro 14:45 simpsons.dat
```

You can also specify that FRANz should update the allele frequencies during Simulated Annealing (SA) optimization and Markov Chain Monte Carlo (MCMC) sampling with the --updatefreqs option. This is a good idea here because the dataset is quite small and we have one big family:

```
$ FRANz --Nmax 2 --femrepro 14:45 --malerepro 14:45 --updatefreqs simpsons.dat
```

The output:

```
[====================] 100% Initializing Mersenne Twister
[====================] 100% Allele Frequency Analysis
[====================] 100% Simulation
[====================] 100% LOD Calculation
[====================] 100% SA Optimization
[====================] 100% MCMC (Sampling)
```

In the first step, we initialize the random number generator (Mersenne Twister [MN00]). After the "Allele Frequency Analysis" we simulate individuals with known relationship. In the "LOD Calculation" step we determine all possible (with the allowed number of mismatching loci) parent-offspring pairs and triples. "SA Optimization" is the Simulated Annealing step that searches efficiently for the Maximum Likelihood pedigree. If the dataset contains less than --saexactmax individuals, we do an exhaustive pedigree enumeration as described in [CR09] if N is known (this does not work with the --Nmax feature). The Markov Chain Monte Carlo sampler finally estimates the statistical significance of the parentages.

Now open the file summary.txt. You will get some summary statistics (more detailed in locisummary.txt). The most important file is parentage.csv, which lists the likeliest parents of each individual:

```
Grampa,2,,,,,0.000000E+00,1.0000,2,0,0,0,,,<
Homer,3,Grampa,2,,,-2.613851E-01,0.6662,2,0,0,1,1.366295E+00,,<
 ...
Flanders,2,,,,,0.000000E+00,0.9980,3,0,3,3,,,<
```

The most important values are the LOD scores in column 7 [MT86] and the posterior probabilities in column 8 [NMCP01]. MCMC and SA are necessary when individuals cannot be ordered in generations a priori. This is the case when not all individuals have a known year of birth. In addition, femrepro.min and malerepro.min must be both greater than 0. If you have specified --updatefreqs, --Nmax and/or --gibbsmissing, we have to do a MCMC sampling. In the case of MCMC sampling, the posterior probability is simply the fraction of sampled pedigrees with this parentage [RSK09]. For example, a posterior probability of 1.0 (Grampa) means that in all MCMC sampled pedigrees, this individual had the same parentage. In only 66% of all pedigrees, Grampa was identified as father of Homer. You can use the --poutformat option if you want

to list all considered parentages, not only the likeliest:

$ FRANz --Nmax 2 --femrepro 14:45 --malerepro 14:45 --updatefreqs --poutformat 2 simpsons.dat

See also section OUTPUT FILES.

The maximum likelihood pedigree is stored in our own format as pedigree.dat and also for visualization as Graphviz dot file. You can convert this dot file for example in a SVG file with

$ dot -Tsvg pedigree.dot > pedigree.svg

You can use the FRANz pedigree.dat file again as input file. For example if you know some mother-offspring relationships:

$ FRANz --Nmax 2 --femrepro 14:45 --malerepro 14:45 --pedigreein simpsons.mothers simpsons.dat

The age fields (year of birth and death) might be confusing. This does not necessarily mean that you have to know the exact years. You can use this feature to order the individuals in generations if this is known a priori. For example, you have a set of offspring and a list of candidate parents. In this case, just build the sets by giving them a common age, for example 2001 for offspring and 2000 for candidate parents:

```
Grampa    1 2000 ? M 110/100 200/208 ?/?
Homer     1 2000 ? M 110/170 200/210 300/302
Bart      1 2001 ? M 110/120 200/212 302/304
Lisa      1 2001 ? F 140/170 200/218 302/306
Maggie    1 2001 ? F 110/140 210/212 300/304
Marge     1 2000 ? F 120/140 212/218 ?/306
Flanders  1 2000 ? ? 150/160 214/220 300/?
```

Then run FRANz with

$ FRANz --femrepro 1:1 --malerepro 1:1

Now please run FRANz with your data. If the input file parser complains about your files, then please read the section INPUT FILES thoroughly. If FRANz accepts your input files and something else is not working, then please read at least the next two sections!

FRANz RUNS FOREVER

The first thing you should make sure is that you really use all the prior information you have. The most valuable information you maybe have is the age of the individuals. You should specify this now (see above or in the reference under "INPUT FILES, Genotypes" below). Run FRANz and you will see a huge drop in the runtime.

For known parent-offspring relationships, you have to input a pedigree file. You can either create such a file by hand (see section INPUT FILES), with our webservice (see IMPORT FROM CSV) or you can use the output file, pedigree.dat, of a FRANz test run and remove all the wrong/unknown relationships and add the missing ones. Again, you will find some help about the data format below in the reference. Do not forget to rename the altered pedigree.dat (for example in mothers.dat), otherwise FRANz will overwrite it the next time. Then start FRANz as before, but with this pedigree file:

--pedigreein *mothers.dat*

If your marker suite is not very powerful (parent-pair exclusion probabilities < 0.95, this means the probability that a random pair of individuals in the population has a 5% chance of having a genotype pair compatible to an offspring genotype. See also next section), the simulated annealing and MCMC sampling might take a very long time without the known relationships. For testing purposes, you can control the runtime with the --sa* and mh* parameters. For example:

--sachains *0* **--mhburnin** *10000* **--mhiter** *20000*

Will turn off the SA optimization and will only run a very short MCMC. The progress of the SA optimization is reported in the file mcmc.log. On Linux and Mac, you can observe the progress with:

$ tail -f mcmc.log

If you have a good estimate of the number of breeding males and females, you should specify this number with --N instead of using --Nmax. See section INCOMPLETE SAMPLING.

Finally, if you expect many fullsibs in your data, then please read the section FULLSIBS.

FRANz RUNS OUT OF MEMORY

If FRANz uses huge amounts of memory or if you even get an error message such as "Error: malloc failed" then there are a couple of things you should try. In principle, FRANz is quite memory efficient. But if your dataset is large and your marker suite is not very powerful (see next section), then the number of possible parent-offspring pairs and triples might explode. Again, make sure that you use all prior knowledge you have. Then, apart from running FRANz on a modern computer with enough RAM, you could try a smaller typing error rate or allow fewer mismatches (check the mismatch distributions in the output file simulations.txt to get reasonable numbers here):

--typingerror *0.01* **--maxmismatching** *i,i*

If you use the multi-core version of FRANz and if you have specified --Nmax or --gibbsmissing, then every thread will have its own copy of all possible parentages. So you could try to run FRANz on fewer cores:

$ OMP_NUM_THREADS=4; FRANz ...

Alternatively, try:

--nogibbsmissing --N *i*

If your dataset is small, then it might be that the exhaustive pedigree enumeration needs more memory than available on your machine. In this case, you might want to set the --saexactmax option to a lower value, say 20.

POWER OF THE MARKER SUITE

When using parentage or paternity inference methods, there are typically two central questions: First, is the sampling rate of candidate parents high enough? A low sampling will not catch enough parentages to estimate the parameters of interest. Second, is the amount of genomic information high enough to identify parent-offspring pairs and triples in the data? The number of required marker loci mainly depends on the expected heterozygosity of each locus. But also ecological data is very helpful, most importantly the age of the individuals. Especially with low sampling rates, it is often not possible without age data to identify parent and offspring in a parent-offspring pair. Known relationships (e.g. mother-offspring) are also very informative. A good knowledge about the number of unsampled candidate mothers and fathers and knowledge of the sex of the individuals can also reduce the required number of marker loci.

Furthermore, the family structure in the data also influences the required genomic signal. If we cannot exclude relatives as candidate parents, we need more loci. On the other hand, fullsibs we can exclude as parents (e.g. because of age prior knowledge) will reduce the amount of required loci [Wan07].

INCOMPLETE SAMPLING

As already stated in the previous section, the sampling rate of candidate parents is very important for a successful application of parentage inference methods. As all other tools out there, FRANz requires some prior knowledge about this sampling rate for the estimation of the statistical significance of parentages. But in contrast to most other tools, FRANz can also incorporate the uncertainty of this sampling rate estimation in the pedigree reconstruction. You only have to provide an upper limit of the number of breeding individuals in the population with the --Nmax option. And again, if you have a good estimate of the number of breeding males and females, you should specify this number with --N instead of using --Nmax. Otherwise, FRANz has to search for the true N in one (or two if the sex of individuals is known) additional MCMC dimensions.

FULLSIBS

Fullsib relationships are very informative and reduce the candidate parents tremendously. FRANz can identify highly probable fullsibs in the data with the --fullsibtest option. If it is very unlikely that your data contains many fullsibs, you should not turn this on. False positives can decrease the accuracy of the

reconstruction. As we have already said, true positives can greatly enhance the accuracy, but if there are no fullsibs, you can only loose. As an alternative to avoid false positives, use very conservative p-Value thresholds:

--fullsibpvth *0.0001,0.0001,0.001*

Or select the Holm instead of the Benjamini-Hochberg correction:

--fullsibpvmethod *2*

For a good choice of the p-Value cutoff, it is recommended to check the file siblings.txt. This file also lists all rejected fullsib candidates. These are pairs where the likelihood that they are fullsibs is higher than the likelihoods that they are halfsib, parent-offspring or unrelated, but the likelihood differences were not significant (i.e. did not pass the p-Value filter). You will see that most pairs did not pass the halfsib p-Value cutoff. If you don't expect many halfsibs (or aunts/uncles) in the data, try less conservative cutoffs. You can even turn the halfsib test off with:

--fullsibH0 *1,0,1*

FRANz will then test every pair against the null hypotheses parent-offspring and unrelated. The integer numbers can be used as weighting factor (prior probabilities) in the p-Value calculation (see OPTIONS).

Per default, FRANz only searches in the offspring generation for fullsibs. This means all individuals without candidate parents in the data are omitted. You can include these individuals with the flag --fullsib-parental.

If you know some fullsib or fullsib/halfsib relationships a priori (by field observation or determined with other tools), you can also specify them with:

--fullsibin *filename*

or

--halfsibin *filename*

See the INPUT FILES section for the format of this file.

FRANz tries to detect inconsistencies in the fullsib assignments: if A,B and B,C are fullsibs, then A and C must be fullsibs, too. Another explanation would be that either A,B or B,C are false positives. FRANz uses a simple heuristic here: if it is unlikely that A and C are fullsibs and either A,B or B,C are close to the p-Value cutoff, then it marks A,B (or B,C, respectively) as false positive. Otherwise FRANz marks A,C as fullsib (these are the "indirect" fullsibs in siblings.txt).

INPUT FILES

Genotypes

See the Tutorial above for a description of the main genotype file. Here the complete example:

```
1 3 / SIMPSONS
7 Springfield
Grampa    1 1920 ? M 110/100 200/208 ?/?
Homer     1 1950 ? M 110/170 200/210 300/302
Bart      1 1982 ? M 110/120 200/212 302/304
Lisa      1 1980 ? F 140/170 200/218 302/306
Maggie    1 1988 ? F 110/140 210/212 300/304
Marge     1 1952 ? F 120/140 212/218 ?/306
Flanders  1 ? ? ? 150/160 214/220 300/?
```

If you want to provide loci ids, you can add them after the first line, one id per line (max. length 10 characters):

```
1 3 / SIMPSONS
L1
L2
```

```
L3
7 Springfield
Grampa     1 1920 ? M 110/100 200/208 ?/?
Homer      1 1950 ? M 110/170 200/210 300/302
Bart       1 1982 ? M 110/120 200/212 302/304
Lisa       1 1980 ? F 140/170 200/218 302/306
Maggie     1 1988 ? F 110/140 210/212 300/304
Marge      1 1952 ? F 120/140 212/218 ?/306
Flanders   1 ? ? ? 150/160 214/220 300/?
```

Here it is important that at least the first locus ID is NOT a number. Otherwise the input file parser assumes that it is the number of individuals.

Here an example for multiple sampling locations:

```
2 3 / SIMPSONS
7 Springfield
Grampa     1 1920 ? M 110/100 200/208 ?/?
Homer      1 1950 ? M 110/170 200/210 300/302
Bart       1 1982 ? M 110/120 200/212 302/304
Lisa       1 1980 ? F 140/170 200/218 302/306
Maggie3210 1 1988 ? F 110/140 210/212 300/304
Marge      1 1952 ? F 120/140 212/218 ?/306
Flanders   1 ? ? ? 160/160 214/214 300/?
3 NYC
Fry        1 1974 ? M 110/120 220/220 300/306
Farnsworth 1 2801 ? M 140/142 240/242 340/342
Leela      1 2900 ? F 144/144 244/240 340/300
```

Known relationships

Known parent-offspring relationships are defined in FRANz with a pedigree infile. The probably simplest thing to generate one is to run FRANz once (but see FRANz RUNS FOREVER). It outputs a pedigree file as pedigree.dat. You can alter this file accordingly and input in a second run with the --pedigreein argument. Example:

```
7
   Grampa
    Homer
     Bart
     Lisa
   Maggie
    Marge
 Flanders
    Marge      Bart
    Marge      Lisa
    Marge    Maggie
```

The first line is the number n of individuals, the next n lines are the exactly (!) 10 characters long names or descriptions of the individuals. They must be identical (leading or trailing whitespaces are ignored and I recommend right aligned ids) to the ones in the genotype file. Then, the remaining lines are the pedigree arcs in the format

```
   parent     child
```

(each again exactly 10 characters long).

Known fullsib relationships are defined with --fullsibin. If you know that some individuals are either full-sibs or halfsibs, you can specify a --halfsibin file. This is useful for example in nest structered data when

one or both sexes are monogamous - if not, see [J07]. If the genotype of the monogamous parent of the halfsib/fullsib group is known, then just specify a pedigreein file instead of a halfsibin file.

Example:

```
1
3
    Bart
    Lisa
  Maggie
```

The first line is the number of fullsib or fullsib/halfsib groups, the 3 is the number of fullsibs in the first group and the following 3 lines contain the ids of the individuals as in the pedigree infile.

Allele frequencies

The allele frequency file is a little bit complicated, but it is also automatically generated by FRANz. If you want to use different genotypes for the allele frequency estimation than for the pedigree reconstruction, then run FRANz once with the allele frequency genotypes and the command line parameters

--maxmismatching *0,0* **--noreconstruction --freqout** *alleles.dat*

Then run FRANz with the genotypes for the pedigree reconstruction and the command line parameter

--freqin *alleles.dat*

Example file:

```
3
7 100 170
100 0.071429
110 0.285714
120 0.142857
140 0.214286
150 0.071429
160 0.071429
170 0.142857
7 200 220
200 0.285714
208 0.071429
210 0.142857
212 0.214286
214 0.071429
218 0.142857
220 0.071429
4 300 306
300 0.300000
302 0.300000
304 0.200000
306 0.200000
```

The first line is the number of loci (3 in this example). The second line is for the first locus and says that there are 7 different alleles in range 100 to 170. The next 7 lines are the alleles with their frequency (space separated).

You can also add the sampling locations, either as pairwise distances (--geofile) or coordinates (--coordfile). In both cases, the order of the locations must be the same as the one in the genotype file. Examples:

Distances

```
3
AcquaAzz1 0.000 0.000 1030.116
AcquaAzz2 0.000 0.000 1030.116
```

Addaia 1030.116 1030.116 0.000

Coordinates

3
AcquaAzz 36.43 15.09
AcquaAzz2 36.43 15.09
Addaia 40.016 4.207

You can specify the maximum distance between mother and child, between father and child, between mother and father and between fullsibs. See OPTIONS.

OUTPUT FILES

Summary

A file with a summary of the data analysis is generated as *summary.txt.* Here you will find a compact statistic about the marker suite. For every locus, following values are printed:

- **Number of alleles and the allele range**
- **Observed and expected Heterozygosity**
- **Polymorphic Information Content (PIC) [BWSD80]**
- **Exclusion Probabilities [JT97, Wan07]**
- **Probability of genotype identity for random individuals and siblings [WLT01]**
- **Estimation of Null allele frequency [KT06].** Note that with --pedigreein, the genotypes of observed homozygote/homozygote mismatches are incorporated in the original formula as a_i.a_n/a_j.a_n.
- **p-Value of deviation from Hardy-Weinberg-Equilibrium and its standard error [GT92].**

See the references for explanations. More detailed allele frequency statistics can be found in locisummary.txt. This file also lists the allele frequency SEs [Boe91].

The following paragraph lists basically the same values, but now for the complete marker suite (all loci combined). The exclusion probabilities are listed for more sampling scenarios, such as n sampled siblings.

If --selfing was specified, then the selfing rate estimated from the allele frequencies is also reported. This is only a rough estimate as it assumes that self-fertilization is the only reason for deviations from HWE. If --simselfingrate was not specified, then this estimation is used in the simulations.

The "Files" section lists the paths to the input and output files.

The next sections list the settings. For a description, see above in OPTIONS.

If some genotypes are not unique, you will find a list of these in an "identical genotypes" section. If you have specified a pedigree infile (with arcs), then observed mismatches are also reported.

The "Simulation" section lists the critical values for the test statistics Delta LOD, PO and HS. See [RSK09].

The "Maximum Likelihood Pedigree" section lists the log-likelihoods of the best, the ML pedigree. Some statistics about this ML pedigree are also given.

Finally, the "MCMC" section lists some statistics of the MCMC sampling.

Parentages

This file lists the likeliest parentage for each genotype or individual.

The LOD score in column 7 is the ratio of L(H1)/L(H2), with L(H1) being the parentage in the current line [MT86, KTM07].

Posterior is the posterior probability of the parentage in a pedigree, defined as the probability of observing the parentage when drawing a pedigree from the posterior distribution. This posterior distribution is

generated with the standard Metropolis-Hastings algorithm or the MCMCMC algorithm when compiled with --enable-openmp on a multi-core CPU. The parentage in the ML pedigree is marked with a '<' in the last column. This is not a sign of statistical significance!

Offspring,Loci Typed,Parent 1,Loci Typed,Parent 2,Loci Typed,LOD,Posterior,Common Loci Typed,Mismatches,n_f,n_m,Pair LOD Parent 1,Pair LOD Parent 2
Grampa,2,,,,,0.000000E+00,1.0000,2,0,0,0,,,<
Homer,3,Grampa,2,,,-2.613851E-01,0.6662,2,0,0,1,1.366295E+00,,<
Bart,3,Marge,2,Homer,3,7.354571E-01,0.8473,3,0,1,1,9.704751E-03,1.585450E+00,<
Lisa,3,Marge,2,Homer,3,3.570038E+00,0.9298,3,0,1,1,1.234198E+00,1.585450E+00,<
Maggie,3,Marge,2,Homer,3,1.411737E+00,0.6927,3,0,1,1,-6.734426E-01,9.081603E-01,<
Marge,2,,,,,0.000000E+00,1.0000,2,0,0,1,,,<
Flanders,2,,,,,0.000000E+00,0.9980,3,0,3,3,,,<

Pedigree

The maximum likelihood pedigree is per default stored in two formats. The first is our own format, *pedigree.dat* , which is the same as for input pedigrees (see above). The second, *pedigree.dot* , is a "dot" file. Dot is a free graph drawing program and is part of the Graphviz package. See man dot for details. A third available format is a simple text file with three columns:

ID	SIRE	DAM
Grampa	*	*
Bart	Homer	Marge
Lisa	Homer	Marge
...		

If you'd like to see support for another format, just let us know. IMPORTANT NOTE: if you use the Maximum Likelihood pedigree to estimate parameters, always check the parentage file. This file lists the probabilities of each arc in the pedigree. Instead of just using the ML pedigree, one could also incorporate the uncertainty of the pedigree reconstruction by using all MCMC sampled pedigrees. Please contact us if you are interested here!

SA and MCMC

The logfile of the SA optimization is stored as *mcmc.log.* This file also lists the settings of SA and MCMC. Statistics of the sampled pedigrees are stored in mhparam.dat.

Siblings

If --fullsibtest was specified on the command line, then high probable siblings are listed in the file *siblings.txt.* The log-likelihood ratios (H2: unrelated) for the relationships PO (parent-offspring), FS (full-sib), HS (half-sib) are also listed. pV are the Benjamini-Hochberg corrected p-Values. It is possible to generate a FRANz fullsib file with the --siblingsoutformat option.

Simulation

Detailed results of the simulation are stored in *simulation.txt.* After listing the settings used in the simulation, this file lists the observed numbers of mismatches. First for true parent-offspring pairs, then for two unrelated randomly chosen individuals. Now for true offspring-mother-father triples and finally for offspring-mother-unrelated triples.

For the fullsib p-Value calculation (--fullsibtest), the observed delta values are reported. Delta Parent-Offspring for example is defined as:

deltaPO = P(A,B|FS) - P(A,B|PO)

and is generated for A and B being fullsibs and A and B being parent-offspring. So deltaPO should be always positive for fullsibs and always negative for parent-offspring pairs. A p-Value of 0.05 can be interpreted as 5% off all pairs with a value larger than this delta value, say 1.4, were parent-offspring pairs in the simulation, NOT fullsibs despite the fact that 1.4 is positive. This delta value is reported in summary.txt, section "Simulation Results". As another example, assume a delta value of 0 that has a corresponding p-Value of 0.11. Then we would make in 11% of all comparisons an error if we would just look at the sign of

deltaPO. The sensitivity is the fraction of the fullsibs we would detect with the corresponding delta value.

HWE

A more detail output of the HWE tests is generated when the filename is specified with --hwetestout. This is basically the concatenated output of the original implementation [GT92] (which we use) of all loci.

Missing Alleles

If --missingout is specified, then the Gibbs sampled missing alleles are logged during the MCMC sampling and statistics how often the alleles were observed are printed in the specified file.

Mismatches

The file mismatches.txt lists for every locus the mismatches observed during MCMC. It also lists the percentage of sampled pedigrees showing the corresponding mismatches.

IMPORT FROM CSV

In extras/input, you will find a small perl script that transforms a CSV file in a valid input file. For example, assume this test.csv file:

Grampa,1920,?,M,110,100,200,208,?,?
Homer,1950,,M,110,170,200,210,300,302
Bart,1982,?,M,110,120,200,212,302,304

Now run the script with following parameters:

$ perl csv.pl --in test.csv --birth_col 1 --death_col 2 --sex_col 3 --data_col 4

The column ids start with 0.

You will find an user friendly GUI for this script on our website.

SIMULATED ANNEALING AND MCMC PARAMETERS

Setting good parameters in a MCMC experiment is an essential but, unfortunately, not a trivial step. A very simple test is whether the outcomes of two or more runs are equal. If the ML pedigree looks completely different or if the posterior probabilities in parentage.csv differ significantly, you have to fine-tune the parameters. The Simulated Annealing optimization should determine good parameters automatically. You should check in mcmc.log that the initial acceptance probability is close to 1.0. If not, then you have to set the initial temperature manually with

--sacstart *c*

For example if the initial acceptance ratio is 0.6 and the starting temperature 110, then try a starting temperature of 300. You have to play here until you get good results. Note that with missing age data, the maximum acceptance probability can be significantly smaller than 1.0 because steps that introduce cycles in the pedigree are always rejected, no matter how high the temperature is.

You might also want to increase

--sabeta *n*

or decrease

--sadelta *n*

to do a more exhaustive search.

For the MCMC runs, you maybe have to increase the iterations:

--mhburnin *n* **--mhiter** *n*

Future versions might ship with better diagnostic tools.

MISSING DATA

If an offspring lacks both alleles at a particular locus, this locus is ignored in all LOD calculations for this offspring. The same applies for the case that all candidate parents lack both alleles. All other missing alleles can be filled by Gibbs sampling with the --gibbsmissing option. I recommend to place loci with lots of missing alleles at the end of the input file. This way, you can easily remove them (and study the influence) with the

--numloci *n*

option, which effects that only the first n loci are used. The Gibbs missing is especially useful for datasets with high exclusion probabilities, many known relationships and datasets with only a limited number of loci with missing alleles. In this case, it might be also interesting to add the command line argument

--missingout *missing.txt*

which turns on logging of the sampled missing alleles during MCMC. This gives you then for every missing value the allele probabilities. Example:

$ FRANz simpsons.dat --Nmax 2 --femrepro 15:45 --malerepro 14:45 --updatefreqs --gibbsmissing --missingout miss.txt

miss.txt:

```
Genotype       300     302     304     306
Grampa      *0.3982  0.3032  0.1995  0.0991
Grampa       0.0989  0.2004 *0.2998  0.4009
Marge        0.0010  0.0025 *0.9953  0.0013
Flanders     0.2504  0.2565  0.2432 *0.2498
```

Note that if the number of typed loci differs between offspring and parental generations, only the intersection is of course informative for parentage inference. However, more typed loci in the offspring generation are informative for the fullsib calculation. The fullsib calculation only uses loci where both alleles are genotyped.

You probably also want to adjust the

--mintyped *n*

parameter, especially when your data contains many loci with low heterozygosity and many loci with missing data.

Genotypes with only one missing allele are ignored in the allele frequency estimations.

TYPING ERRORS

FRANz uses the error model described in [KTM07]. If some relationships are known, it estimates the typing error as described in [MSKP98]. How FRANz now uses this estimates depends on the number of observed relationships. With only a low number, FRANz uses the default typing error rate of 0.01. With a medium number, it uses the estimated average typing error over all loci and, finally, with a high number of observed relationships it uses the estimated error rates for every loci. The thresholds are determined with standard statistical methods (see --typingerror). You should always check the error rate and if necessary provide a better one with the --typingerror option.

DATA CONVERSION

If you want to compare the FRANz results with CERVUS, you can easily do that with the --cervus... options. It is also possible to convert the FRANz input into the PARENTE [CBM02] file format with the --parenteout option and into the Genepop [R07] format with the --genepopout option. The --rmesout option can be used to generate an input file for the RMES [DPVCG07] program for selfing-rate estimations.

OTHER TOOLS

See [JA03] for a comprehensive (but now slightly outdated) comparison of other tools, for example: **CERVUS [MSKP98, KTM07], COLONY [Wan04], FAMOZ [GMSBK], MasterBayes [HRB06], NEST [J07], Parente [CBM02], PATRI [NMCP01], PedApp [Alm07].**

REFERENCES

[Alm03] A. Almudevar. A simulated annealing algorithm for maximum likelihood pedigree reconstruction. Theor Popul Biol, 63:63-75, Mar. 2003

[Alm07] A. Almudevar. A graphical approach to relatedness inference. Theoretical Population Biology, 71, 213-229. 2007.

[Boe91] M. Boehnke. Allele frequency estimation from data on relatives. Am J Hum Genet. 1991 January; 48(1): 22â25.

[BWSD80] D. Botstein, R.L. White, M. Skolnick, R.W. Davis. Construction of a genetic linkage map in man using restriction fragment length polymorphisms. Am J Hum Genet. 32(3): 314â331, May. 1980.

[CBM02] A. Cercueil, E. Bellemain, and S. Manel. PARENTE: Computer Program for Parentage Analysis. The Journal of Heredity 93(6). 2002

[CR09] R. Cowell. Efficient maximum likelihood pedigree reconstruction. Theoretical Population Biology. 2009, 76, 285-291.

[DPVCG07] P. David, B. Pujol, F. Viard,V. Castella, and J. Goudet. Reliable selfing rate estimates from imperfect population genetic data. Mol Ecol. 2007 Jun;16(12):2474-87.

[GMSBK] S. Gerber, S. Mariette, R. Streiff, C. Bodenes C, A. Kremer A Comparison of microsatellites and amplified fragment length polymorphism markers for parentage analysis. Molecular Ecology, 9, 1037â1048. 2000.

[GT92] S.W. Guo, E.A. Thompson. Performing the exact test of Hardy-Weinberg proportion for multiple alleles. Biometrics, 48, 2:361-72. 1992.

[HRB06] J.D. Hadfield, D.S. Richardson and T. Burke.Towards unbiased parentage assignment: combining genetic, behavioural and spatial data in a Bayesian framework. Mol. Ecol, 15. 3715-3730. 2006.

[J07] B. Jones, D. Grossman, D.C.I. Walsh, B.A. Porter, J.C. Avise and A.C. Fiumera. Estimating Differential Reproductive Success From Nests of Related Individuals, With Application to a Study of the Mottled Sculpin, Cottus bairdi. Genetics 176: 2427â2439. 2007

[JA03] A.G. Jones and W.R. Ardren. Methods of parentage analysis in natural populations. Molecular Ecology, 2511-2523. 2003.

[JT97] A. Jamieson, Taylor. Comparisons of three probability formulae for parentage exclusion. Animal Genetics, 28, 6:397-400(4), Dec 1997.

[KT06] S.T. Kalinowski, M.L. Taper. Maximum likelihood estimation of the frequency of null alleles at microsatellite loci. 2006.

[KTM07] S.T. Kalinowski, M.L. Taper, and T.C. Marshall. Revising how the computer program CERVUS accommodates genotyping error increases success in paternity assignment. Mol. Ecol., 16:1099--1106, Mar 2007.

[MN00] M. Matsumoto, T. Nishimura. Dynamic Creation of Pseudorandom Number Generators. Monte Carlo and Quasi-Monte Carlo Methods 1998, Springer, 2000, pp 56--69. http://www.math.sci.hiroshima-u.ac.jp/˜m-mat/MT/DC/dgene.pdf

[MSKP98] Marshall, TC, Slate, J, Kruuk, LEB & Pemberton, JM (1998) Statistical confidence for likelihood-based paternity inference in natural populations. Molecular Ecology 7: 639-655.

[MT86] T.R. Meagher, E.A. Thompson. The relationship between single parent and parent pair genetic likelihoods in genealogy reconstruction. Theoretical Population Biology,

29(1):87--106, Feb. 1986.

[NMCP01] R. Nielsen, D.K. Mattila, P.J. Clapham, and P.J. PalsbÃ¸ll. Statistical approaches to paternity analysis in natural populations and applications to the North Atlantic humpback whale. Genetics, 157:1673--1682, Apr 2001.

[R07] F. Roussett. GENEPOP'007: a complete re-implementation of the genepop software for Windows and Linux. Molecular Ecology Resources, 8,1:103-106. 2007

[RSK09] M. Riester, P.F.Stadler, K.Klemm. FRANz: Reconstruction of wild multi-generation pedigrees. Bioinformatics, 2009. 25(16):2134-2139.

[RSK10] M. Riester, P.F.Stadler, K.Klemm. Reconstruction of pedigrees in clonal plant populations. Theoretical Population Biology, 2010 (in press).

[Wan04] J. Wang. Sibship reconstruction from genetic data with typing errors. Genetics. 166(4):1963-79. 2004.

[Wan07] J. Wang. Parentage and sibship exclusions: higher statistical power with more family members. Heredity, 99, 2:205-17. 2007.

[WLT01] L.P. Waits, G. Luikart, P. Taberlet. Estimating the probability of identity among genotypes in natural populations: cautions and guidelines. Mol Ecol. 10(1):249-56, Jan. 2001.

AUTHOR

Markus Riester (University of Leipzig)

Peter F. Stadler (University of Leipzig, University of Vienna, Santa Fe Institute)

Konstantin Klemm (University of Leipzig)

Robert Cowell

FEEDBACK

Any comments, questions, critics or suggestions are gratefully received. So please don't hesitate to contact us! We would be happy to help. Your feedback will help us improving this software.

REPORTING BUGS

If you find a bug in this software, please send a mail to **markus@bioinf.uni-leipzig.de.** If possible, please include the input files and the command line parameters.

COPYRIGHT