**Running hapferret**

(This documentation is for running the executable on a Mac)

1. At present (12/2015) hapferret takes no command line input. The names of the input and output files are fixed; to run it you need to create a folder and put the input files—and a copy of ferret—there. The output files will be created there also. So the strategy is to either create a new folder for each run—easy enough on a Mac—or to keep one folder but replace the input and move or rename the output after each run.
2. **Running**

Open a terminal window, navigate to the folder containing ferret and the input files (just type “cd”, space, and drag the folder to the terminal command line), and type ./hapferret. The required input files are: files.data.txt, hap\_search\_settings.txt, and the input data files specified in files.data.txt.

1. **Specifying input files**

Input files and their formats are specified in a file with default (currently required) name “files\_data.txt”, which must be in the directory from which hapferret is run.

Sample files\_data.txt:

**wide format** (all genotypes for an individual on one line)

fileformat c

genotype\_file hla\_ceu\_haps\_14long.txt

Each line starts with an identifier, as shown. The two lines give the file format (here “c” for comma, specifying allele,allele format for each genotype), then the genotype file name.

**long format** (one genotype—one locus for one individual—on each line)

fileformat s

genotype\_file c22mge\_ceu.txt

var\_info\_file c22mge\_ceu.i.txt

Again each line starts with an identifier. The three lines give the file format (“s” stands for scan input) then the genotype file name, then the associated locus info file.

1. **File format**

HapFerret accepts two input formats (“wide” and “long” in database terminology).

**Wide format** has all genotype information for one individual on one line. Each line starts with a subject identifier, and has one genotype entry for each locus. This file must have a title line; the first entry, e.g. “PID”, identifies the column of subject IDs and is ignored. This is followed by the names of each locus. This format is specified by fileformat “c” in the files\_data.txt file. Example:

PID rs136160 rs136161 rs713753 rs4419330 rs4350853 rs136168 rs2239785

UX19193 G,C G,G C,C T,C T,T A,A A,G

UX18501 G,C C,G T,C T,T T,T A,G A,G

UX19093 G,C C,G C,C T,C T,T A,A A,G

UX19209 G,G G,G C,C T,T T,T A,G A,G

UX19144 G,C C,G C,C T,T T,T A,G G,G

UX19222 G,G C,G T,C T,T T,T A,G A,G

UX19193 G,C G,G C,C T,T T,T A,G A,G

UX19101 C,C G,G C,C T,T T,T G,G G,G

UX19101 C,C G,G C,C T,C T,T G,A G,G

UX19127 C,C G,G C,C T,T T,T G,G G,G

UX19140 C,C G,G C,C T,T T,T G,G G,G

UX19209 G,C G,G C,C T,C T,T A,G A,G

UX18522 C,G G,G C,C T,C T,G G,A G,A

UX19138 C,C G,G C,C T,T T,T G,G G,G

These are standard .txt files, with entries divided by white space, one or more spaces or tabs. Note that the loci are given by the standard (rs #) name, and the genotypes are allele,allele, with the allele given as the actual base. The locus and the allele can be identified by any alphanumeric (without spaces, and without spaces between the allele names and the comma); e.g. an example with loci from the HLA region:

Macintosh HD:Users:nelsong:Documents:analysis:development:ferret output and writing:hla_ceu_haps_7long for documentation.txt.pdf

**Long format**

Each line is one genotype: PID, race, polymorphism identifier, and the two alleles seen. Entries are separated by white space;here the two alleles are separated by white space and not by a comma. This is specified by fileformat “c” in the files\_data.txt file. Example files\_data file:

fileformat s

genotype\_file c22mge\_ceu.txt

var\_info\_file c22mge\_ceu.i.txt

The third line, var\_info\_file, specifies the file with locus information

44O00768 1 rs3752462 C C

44O00769 1 rs3752462 C T

44O00774 1 rs3752462 T T

44O00779 1 rs3752462 C C

44O00781 1 rs3752462 C T

44O00782 1 rs3752462 C C

44O00785 1 rs3752462 C T

44O00787 1 rs3752462 C C

44O00793 1 rs3752462 C T

44O00794 1 rs3752462 C T

44O00797 1 rs3752462 C T

44O00803 1 rs3752462 T T

44O00813 1 rs3752462 C T

44O00825 1 rs3752462 C T

44O00837 1 rs3752462 C T

44O00840 1 rs3752462 C T

44O00841 1 rs3752462 C T

44O00844 1 rs3752462 C C

44O00846 1 rs3752462 C T

44O00849 1 rs3752462 C T

44O00852 1 rs3752462 T T

44O00853 1 rs3752462 C C

44O00855 1 rs3752462 C C

44O00859 1 rs3752462 C C

44O00860 1 rs3752462 C C

44O00862 1 rs3752462 C C

44O00868 1 rs3752462 C C

44O00869 1 rs3752462 C C

44O00872 1 rs3752462 C T

Each line is one genotype: PID, race, polymorphism identifier, and the two alleles seen; entries separated by white space—here the two alleles are separated by white space and not by a comma.

Long format requires a locus info file, with each line giving a polymorphism id, followed by its genome coordinates (no chromosome info is used in the current version). The current algorithm needs this information to know the order of the loci, but doesn’t use the actual position for the calculation.

rs1557529 35035474

rs2157256 35037606

rs2413396 35038030

rs5750250 35038428

rs3830104 35038569

rs4820229 35038699

rs3752462 35040128

rs8141971 35041308

rs5756152 35042417

rs9610489 35043476

rs2239784 35044580

rs1005570 35045219

rs12159211 35049108

rs8136336 35052479

rs16996672 35055916

rs16996677 35057228

rs11704382 35058098

rs4820234 35059020

1. **Specifying parameters for haplotype inference**

A file is also required specifying settings for the hap inference run, this has the (currently fixed) name “hap\_search\_settings.txt”. Example content:

accept\_params 1

accept\_params 1

mode 1

blocksequence 0

race 1

disease\_data 0

target\_delta 0.00003

max\_iterations 1000

full\_hap\_call 1

subseq\_hap\_call 0

max\_subblock = 30

calc\_hap\_call\_entropy 1

n\_bootstrap\_reps 0

These need not be in order. The parameters input:

**-accept\_params** (1 or 0) Should these parameters be used as input (1), or should the user queried to enter possible changes

-**mode** 1 (keep this setting, ignore this)

**-blocksequence** 0 (keep this setting, ignore this)

-**race** Used if genotype input file is long format, then this specifies which race—identifier in 2nd column of file—to use. Ignored in wide format.

-**disease\_data** (0 or 1) 1 if there is a disease data file; not documentation yet

-**target\_delta** The EM algorithm iterates and sums the frequency difference between iterations for each haplotype, stopping when this difference is less than this parameter

**-max\_iterations** Stop the inference at this count of iterations even if target\_delta hasn’t been reached

**-full\_hap\_call** Infer haplotypes for the complete (ordered) set of loci in the input file

**-subseq\_hap\_call** Infer haplotypes for subsequences of the input loci. hapferret runs along the set of loci, inferring haplotypes first for all sequences of two (contiguous) loci, then for sequences of 3, 4, etc. loci, to the limit given by max\_subblock

**-max\_subblock** Maximum length of sequences of loci to infer in subseq\_hap\_call.

**-calc\_hap\_call\_entropy** Calculated entropy—uncertainty—of the inference?

**-n\_bootstrap\_reps** Number of bootstrap replications, set to 0 for no bootstrapping—keep this setting for now.