fasta2genotype.py
Version 1.10
Written for Python 2.7.10
Available on request from the author

This program takes a fasta file listing all sequence haplotypes of all individuals at all loci as well as a list of individuals/populations and list of white loci then outputs data in one of eight formats:

(1) migrate-n, (2) Arlequin, (3) DIYabc, (4) LFMM, (5) Phylip, (6) G-Phocs, or (7) Treemix
(8) Additionally, the data can be coded as unique sequence integers (haplotypes) in Structure/Genepop/SamBada/Bayescan/Arlequin/GenAlEx format or summarized as allele frequencies by population.

Execute program in the following way:
python fasta2genotype.py [fasta file] [whitelist file] [population file] [VCF file] [output name]

The fasta file:
The headers should be formatted exactly the way done by populations.pl in Stacks v. 1.12

- For example:

```
>CLocus_#_Sample_#_Locus_#_Allele_#
TCACCAGGATCATATCGTGCATCGGCTACTGCATGCGTTCCCC
...
```

- (The # denotes an integer)
- CLocus is the locus number in the catalog produced by cstacks.
- Sample is the individual number from denovo_map.pl, in the same order specified with -s flags
- Locus refers to the locus number called for that individual; this information is not used
- Allele must be 0 or 1; this script assumes diploid organisms; homozygote is assumed if only 0
  - IMPORTANT: The ‘0’ or ‘1’ only counts alleles for a given individual. A ‘0’ allele for individual 1 could be a different sequence than a ‘0’ allele for individual 2.

The whitelist file is optional:

- This file is simply a column of catalog loci that you wish to retain.
- Locus numbers can be extracted from many of the stacks output files (e.g. VCF, structure).
- Example format:

```
255
812
1256
...
```

- If no pruning is desired, simply use 'NA' as the input argument for whitelist file. For example:

```
python fasta2genotype.py ./fasta NA ./populations ./vcf MyOutput
```

The populations file should have three columns:
[1] Sample ID from stacks, [2] Individual ID, [3] Population ID

- These can have any column names
- You must use tabs not spaces
- Individual IDs must be 9 characters or shorter for [1] Migrate and [5] Phylip option.
- Population IDs must be 9 characters or shorter for [5] Phylip if outputting by population.
- Population IDs must start with a non-numeric character (A-Z or a-z) for [1] Migrate option.
- Example format:

```
SampleID       IndID          PopID
1              Y120037        Wawona
2              Y120048        Westfall
...            ...            ...
```

The VCF file is optional:

- Loci can be pruned using a coverage threshold. A VCF file from the same run of populations.pl must be provided in this case.
- If no pruning is desired, simply use 'NA' as the input argument for VCF file. For example:

```
python fasta2genotype.py ./fasta ./whitelist ./populations NA MyOutput
```

- If pruning is desired, IndividualID spelling in VCF file must match that in populations file
- This also means there should be the same list of IndividualIDs in both files. This program assumes the same VCF format found in Stacks v. 1.12
- Data are read beginning at row 9. Columns 2–4 (POS, ID, REF, ALT) describe the SNP location, locus number, primary SNP allele, and alternative SNP allele. These columns are used and are shown below. Columns 1, and 5-9 are not shown below to save space.
- Columns 10 and up contain coverage data in the following format: three elements (genotype, coverage, likelihood) separated by colons. Each column is an individual.

...	POS	ID	REF	ALT	...	1	2
...	7	1	C	T	...	0/0:139:.,.,.	0/0:72:.,.,.
...	8	1	G	A	...	0/0:17:.,.,.	0/1:10:.,.,.
...	25	2	A	G	...	0/0:26:.,.,.	0/1:12:.,.,.
...	26	2	T	C	...	0/0:33:.,.,.	0/0:12:.,.,.
...	...	...	...	...	...	...	...

  - IMPORTANT: STACKS does not provide coverage info for consensus (non-variable) sequences. Therefore, these loci are removed if coverage pruning is chosen. This will be true even if you tell the program to keep "All" loci.
- Additionally, the LFMM format relies on SNP genotypes found in the VCF file (REF and ALT columns). A VCF file must be included for this option.

Quality filtering options:

- In addition to coverage filtering, several other quality control measures can be selected.
- Monomorphic loci can be removed.
- Loci suspected of being paralogs (assembled as one 'Frankenstein' locus, but including alleles from different loci) can be removed. Loci surpassing the desired threshold value of heterozygosity are tested for Hardy-Weinberg genotype proportions. If they fail (based on a chi-squared test), and have higher heterozygosity than expected under Hardy-Weinberg, they are removed.
- Alleles under a locus-wide frequency can be removed
- Alleles represented under a given frequency of populations can be removed (e.g. 4 pops of 16, frequency of 0.25)
- Loci can be removed from populations where those populations fall under a missing data threshold for those loci
- Loci under a given locus-wide missing data threshold can be removed.
- Individuals missing a threshold frequency of loci can be removed.
- Restriction enzyme sites can be removed if requested, for single- or double-digest setups. Simply select this option and provide the 5' and/or 3' sequence(s). There can be multiple sequences for either 5' or 3' end. For example, you might have double-digest RAD data with READ 1 and READ 2 in the same fasta file. You can tell the program to remove either 'TGCAGG' or 'CGG' depending on which is found on the 5' end of a given sequence. Regardless of restriction enzymes or adapters used, it might help to examine the fasta file before choosing this option. Your sequences might be reversed or reverse-complemented.
- If you are exporting an Phylip alignment file, there are several options.
  - SNPs or sequences: concatenate just the polymorphic sites (SNPs), or full sequences, including invariable sites.
  - The alignment can be summarized at the level of haplotypes, individuals, or populations, using IUPAC ambiguity codes.
  - You can select a subset of loci that are "phylogenetically informative" (PI), meaning fixed for alternate alleles at 2+ taxa, or simply "fixed" (at 1+ taxa).
  - If you choose the option for "PI"/"fixed" loci, you will probably want to output SNPs rather than full sequences, since you probably want only PI or fixed sites. However, if you choose "PI"/"fixed" loci and "full sequences," then the program will output complete sequences containing at least 1 PI or fixed SNP.
  - The alignment can be separated by locus with "!" symbols.
  - A header of tab-delimited locus names can be added.

TO DO:

- Transfer database system to sqlite or similar, looping nested dictionaries is very slow
  - **Particularly** slow: Treemix, coverage filtering
- Allow LFMM function to operate free of VCF file (like Phylip function)
- Allow order of individuals, pops, loci to be specified from input files

What follows is a brief demo of a single sample dataset in all available output formats.

Migrate-n file format:

```
<Number of populations> <number of loci>
<number of sites for locus1> <number of sites for locus 2> ...
<Number of gene copies> <title for population>
Ind1a <locus 1 gene copy 1 sequence>
Ind1b <locus 1 gene copy 2 sequence>
Ind2a <locus 1 gene copy 1 sequence>
Ind2b <locus 1 gene copy 2 sequence>
Ind1a <locus 2 gene copy 1 sequence>
Ind1b <locus 2 gene copy 2 sequence>
Ind2a <locus 2 gene copy 1 sequence>
Ind2b <locus 2 gene copy 2 sequence>
<Number of gene copies> <title for population>
Ind1a <locus 1 gene copy 1 sequence>
Ind1b <locus 1 gene copy 2 sequence>
Ind2a <locus 1 gene copy 1 sequence>
Ind2b <locus 1 gene copy 2 sequence>
Ind1a <locus 2 gene copy 1 sequence>
Ind1b <locus 2 gene copy 2 sequence>
Ind2a <locus 2 gene copy 1 ????????>
Ind2b <locus 2 gene copy 2 ????????>
...   ...
```

Arlequin file format:

```
[Profile]
      "ProjectName"
             Project
             NbSamples=#
             GenotypicData=1
             GameticPhase=0
             DataType=DNA
             LocusSeparator=TAB
             MissingData="?"
[Data]
      [[Samples]]
              SampleName="PopID"
              SampleSize=#
              SampleData={
Ind1  1       <Locus 1 copy 1>        <Locus 2 copy 1> ...
              <Locus 1 copy 2>        <Locus 2 copy 2> ...
Ind1  2       <Locus 1 copy 1>        <Locus 2 copy 1> ...
              <Locus 1 copy 2>        <Locus 2 copy 2> ...
}
              SampleName="PopID"
              SampleSize=#
              SampleData={
Ind1  1       <Locus 1 copy 1>        <Locus 2 copy 1> ...
              <Locus 1 copy 2>        <Locus 2 copy 2> ...
Ind1  2       <Locus 1 copy 1>        <?????????????> ...
              <Locus 1 copy 2>        <?????????????> ...
...   ...     ...                     ...
}
```

DIYabc file format:

```
<Project Name> <NF=NF>
Locus 1 <A>
Locus 2 <A>
Pop
Ind1  ,       <[Locus 1 copy 1][Locus 1 copy 2]>      <[Locus 2 copy 1][Locus 2 copy 2]> ...
Ind2  ,       <[Locus 1 copy 1][Locus 1 copy 2]>      <[Locus 2 copy 1][Locus 2 copy 2]> ...
Pop
Ind1  ,       <[Locus 1 copy 1][Locus 1 copy 2]>      <[Locus 2 copy 1][Locus 2 copy 2]> ...
Ind2  ,       <[Locus 1 copy 1][Locus 1 copy 2]>      <[?????????????][?????????????]> ...
...           ...             ...                     ...             ...
```

LFMM file format (assuming 1 snp at first locus, 2 snps at second locus):

```
Ind1  Pop1    0       0       0       0       A       A       C       T       G       G ...
Ind2  Pop1    0       0       0       0       T       T       C       C       G       G ...
Ind1  Pop2    0       0       0       0       A       T       C       T       T       T ...
Ind1  Pop2    0       0       0       0       T       T       0       0       0       0 ...
...   ...     ...     ...     ...     ...     ...     ...     ...     ...     ...     ...
```

Phylip file format (assuming 1 SNP at first locus, 2 SNPs at second locus):
(Only SNPs will be output, and loci will be concatenated.)
(Alternatively full sequences can be output, and loci will be concatenated.)
(Other options are available, see above.)

```
<number of individuals> <number of base pairs>
Ind1_Pop1 AYG ...
Ind2_Pop1 TCG ...
Ind1_Pop2 WYT ...
Ind1_Pop2 TNN ...
...       ...    ...
```

G-Phocs format:

```
<Number of loci>

<Locus name> <number of individuals> <number of sites for locus 1>
Ind1 <locus 1 sequence>
Ind2 <locus 1 sequence>
Ind1 <locus 1 sequence>
Ind2 <locus 1 sequence>

<Locus name> <number of individuals> <number of sites for locus 2>
Ind1 <locus 1 sequence>
Ind2 <locus 1 sequence>
Ind1 <locus 1 sequence>
Ind2 <locus 1 ????????>
...   ...
```

Treemix format:

```
Pop1     Pop2
2,2      1,3 ...
3,1      1,1 ...
4,0      2,0 ...
...      ...
```

Structure format:

```
              Loc1    Loc2 ...
Ind1  Pop1    1       1 ...
Ind1  Pop1    1       2 ...
Ind2  Pop1    2       1 ...
Ind2  Pop1    2       1 ...
Ind1  Pop2    1       3 ...
Ind1  Pop2    2       4 ...
Ind2  Pop2    2       0 ...
Ind2  Pop2    2       0 ...
...   ...     ...     ...
```

Genepop format (six digit):

```
<Project Name>
Locus 1
Locus 2
Pop
Ind1 ,  001001        001002 ...
Ind2 ,  002002        001001 ...
Pop
Ind1 ,  001002        003004 ...
Ind2 ,  002002        000000 ...
```

Allele Frequency:

```
      Loc1_1          Loc1_2          Loc2_1          Loc2_2          Loc2_3          Loc2_4 ...
Pop1  0.50000         0.50000         0.75000         0.25000         0.00000         0.00000 ...
Pop2  0.25000         0.75000         0.00000         0.00000         0.50000         0.50000 ...
...   ...             ...             ...             ...             ...             ...
```

SamBada format:

```
      Loc1_1  Loc1_2  Loc2_1  Loc2_2  Loc2_3  Loc2_4 ...
Ind1a 1       0       1       0       0       0 ...
Ind1b 1       0       0       1       0       0 ...
Ind2a 0       1       1       0       0       0 ...
Ind2b 0       1       1       0       0       0 ...
Ind1a 1       0       0       0       1       0 ...
Ind1b 0       1       0       0       0       1 ...
Ind2a 0       1       0       0       NaN     NaN ...
Ind2b 0       1       0       0       NaN     NaN ...
...   ...     ...     ...     ...     ...     ...
```

Bayescan format:

```
[loci]=2

[populations]=2

[pop]=1
1     4       2       2       2
2     4       4       3       1       0       0

[pop]=2
1     4       2       1       3
2     2       4       0       0       1       1
```

GenAlEx format:

```
2     4       2       2       2
                      Pop1    Pop2
IndID PopID   Loc1            Loc2
Ind1  Pop1    1       1       1       2 ...
Ind2  Pop1    2       2       1       1 ...
Ind1  Pop2    1       2       3       4 ...
Ind2  Pop2    2       2       0       0 ...
...   ...     ...     ...     ...     ...
```