A short manual for `sNMF`
(command-line version)

Eric Frichot
efrichot@gmail.com

June 26, 2013

Please, print this reference manual only if it is necessary.

This short manual aims to help users to run `sNMF` command-line engine on Mac and Linux.

1 Description

Inference of individual admixture coefficients, which is important for population genetic and association studies, is commonly performed using compute-intensive likelihood algorithms. With the availability of large population genomic data sets, fast versions of likelihood algorithms have attracted considerable attention. Reducing the computational burden of estimation algorithms remains, however, a major challenge. Here, we present a fast and efficient method for estimating individual admixture coefficients based on sparse non-negative matrix factorization algorithms. We implemented our method in the computer program `sNMF`, and applied it to human and plant genomic data sets. The performances of `sNMF` were then compared to the likelihood algorithm implemented in the computer program `ADMIXTURE`. Without loss of accuracy, `sNMF` computed estimates of admixture coefficients within run-times approximately 10 to 30 times faster than those of `ADMIXTURE`

Eric Frichot, François Mathieu, Théo Trouillon, Guillaume Bouchard, Olivier François. *Fast Inference of Admixture Coefficients Using Sparse Non-negative Matrix Factorization Algorithms*, submitted.

2 Installation

To install `sNMF` CL version, you just have to execute the install script (install.command) in `sNMF` main directory. To execute it in a terminal shell, go to `sNMF` main directory and write "./install.command". If the script is not executable, type "chmod +x install.command" and then "./install.command". A set of binaries should be created in `sNMF` main directory.

3 Data format

Input file is composed a genotype file.

eigenstratgeno (example.eigenstratgeno)
The **genotype file** contains 1 line per SNP. Each line contains 1 character per individual: 0 means zero copies of reference allele. 1 means one copy of reference allele. 2 means two copies of reference allele. 9 means missing data.

Below, an example of genotype file for $n = 3$ individuals and $L = 5$ loci.

```
112
010
091
121
```

There are 2 **output files**.

- The first file (with extension **.Q**) contains the individual admixtures coefficients. It is a matrix a matrix of n lines (the number of individuals) and K columns (the number of ancestral populations).

- The second file (with extension **.F**) contains the ancestral allele frequencies. It is a matrix a matrix of $nc \times L$ lines (the number of alleles times the number of SNPs) and K columns (the number of ancestral populations). For each SNP, the first line contains the ancestral frequencies for allele 0, the second line for allele 1, ...

4 Run the programs

The program is executed by a command line. The format is:

```
./sNMF -g genotype_file.geno -K number_of_ancestral_populations
```

All the previous options are mandatory. There is no order for the options in the command line. Here is a more precise description of the options:

- `-g genotype_file.geno` is the path for the genotype file (in .geno format).
- `-K number_of_ancestral_populations` is the number of K of ancestral populations.

Other options are not mandatory:

- `-p p` is the number of processes that you choose to use if you run the algorithm in parrallel. Be careful, the number of process has to be lower or equal than the number of physical processes available on your computer (default: 1).
- `-i iteration_number` is the max number of iterations in algorithm (default: 1000). The algorithm should not go until the max number of iterations. The stopping criteria should depend on the tolerance error only.
- `-a alpha` is the value of the regularization parameter (by default: 100). Results can depend on the value of this parameter, especially for small data sets (see the associated paper).
- `-e tolerance` is the tolerance error (by default: 0.0001).
- `-s seed` is the initialization for the random parameter (by default: random).
- `-m ploidy` 1 if haploid, 2 if diploid (default: 2).

If you need a summary of the options, you can use the **`-h`** option by typing the command line

```
./sNMF -h
```

A full example is available at the end of this note.

5 Cross-Entropy criteria

We also provide two other programs:

- the first program creates a data set with a given percentage of missing data from your original data set. The command line format is:

  ```
  ./createDataSet -g genotype_file.geno
  ```

  The mandatory option is:

  – `-g genotype_file.geno` is the path for the genotype file (in .geno format).

  It will create by default, a file with around 5 % of missing data with the name genotype_file_**I**.geno with a _**I** extension to differentiate this file from the original file.

  Other options are not mandatory:

  – `-r percentage` is the percentage of missing data in your data set (default: 0.05).
  – `-e tolerance` is the tolerance error (by default: 0.0001).
  – `-s seed` is the initialization for the random parameter (by default: random).
  – `-m ploidy` is 1 if haploid, 2 if diploid (default: 2).

- the second program calculates the cross-entropy criterion for all data and for missing data from the output of `sNMF`. The cross-entropy criteria is useful to choose the best run for different number of ancestral populations (K) and different values of the regularization parameter (α). A smaller value of the cross-entropy with missing data means a better prediction of the data. The command line format is:

  ```
  ./crossEntropy -g genotype_file.geno -K number_of_ancestral_populations
  ```

  The mandatory option is:

  - `-g genotype_file.geno` is the path for the genotype file (in .geno format).
  - `-K number_of_ancestral_populations` is the number of K of ancestral populations.

  In this case, the output from `sNMF`, the files with missing data, the original files and the results files have to b in the same directory.

  Other options are not mandatory:

  - `-m ploidy` 1 if haploid, 2 if diploid (default: 2).

6 Tutorial

6.1 Data set

The data set that we analyze in this tutorial is an Asian human data set of SNPs data. This data is a worldwide sample of genomic DNA (10757 SNPs) from 934 individuals, taken from the Harvard Human Genome Diversity Project - Centre Etude Polymorphism Humain (Harvard HGDP-CEPH)2 . In those data, each marker has been ascertained in samples of Mongolian ancestry (referenced population HGDP01224) [1].

6.2 Create a data set with missing data

In the main directory, type:

```
./createDataSet -g examples/panel11.geno
```

A file with 5 % of missing data with path `examples/panel11_I.geno` has been created.

6.3 Run `sNMF`

Then, run `sNMF` for the data set with 5 % of missing data (with $K = 5$ for example):

```
./sNMF -g examples/panel11_I.geno -K 5
```

The results files `examples/panel11_I.Q` `examples/panel11_I.F` have been created.

6.4 Calculate the Cross-Entropy

Finally, calculate the cross-entropy criteria:

```
./crossEntropy -g examples/panel_11.geno -K 5
```

With this procedure, you can calculate a criteria for each of your analyses. It is a way to choose the best run for different number of ancestral populations (K) and different values of the regularization parameter (α).

7 Contact

If you need assistance, do not hesitate to send me an email (efrichot@gmail.com or eric.frichot@imag.fr). A FAQ (Frequently Asked Questions) section is available on our webpage (ttp://membres-timc.imag.fr/Olivier.Francois/snmf.html). `sNMF` software is still under development. All your comments and feedbacks are more than welcome.

References

[1] Nick J. Patterson, Priya Moorjani, Yontao Luo, Swapan Mallick, Nadin Rohland, Yiping Zhan, Teri Genschoreck, Teresa Webster, and David Reich. Ancient admixture in human history. *Genetics, doi:10.1534/genetics.112.145037*, 2012.