Documentation for EMALAM

Carola Sophia Heinzel[1,*], Franz Baumdicker[2,3,4], Peter Pfaffelhuber[1]

October 11, 2024

(1) Department of Mathematical Stochastics, Albert-Ludwigs-University Freiburg, 79104 Freiburg im Breisgau, Germany
(2) Cluster of Excellence "Controlling Microbes to Fight Infections", Mathematical and Computational Population Genetics, University of Tübingen, Sand 14, 72076 Tübingen, Germany
(3) Institute for Bioinformatics and Medical Informatics (IBMI), University of Tübingen, Germany
(4) Cluster of Excellence "Machine Learning in Science", University of Tübingen, Germany
(∗) Corresponding author: E-Mail: carola.heinzel@stochastik.uni-freiburg.de
E-Mail-Adresses: carola.heinzel@stochastik.uni-freiburg.de (CSH),
franz.baumdicker@uni-tuebingen.de (FB)
peter.pfaffelhuber@stochastik.uni-freiburg.de (PP)

1 Introduction and Overview

To estimated the ancestry of individuals, we can use the Admixture Model and a maximum likelihood estimator. We proved that this estimator is usually not unique. The software EMALAM calculates some other maximum likelihood estimators (MLEs). The user determines which other estimators EMALAM calculates given one MLE. This MLE can e.g. be calculated with STRUCTURE [5] or ADMIXTURE [1].

Here, we provide an explanation how to use the software EMALAM and how EMALAM works. Briefly, the input of EMALAM are the estimated IAs and the estimated allele frequencies. To estimate them, we could e.g. use STRUCTURE [5]. Additionally, EMALAM requires some additional information from the user that is specified in section 3. The output are MLEs, i.e. estimated individuals ancestries (IAs) and the estimated allele frequencies. The user defines which MLEs EMALAM calculates, e.g. the MLEs with the maximal admixture.

First, we explain how EMALAM works. Then, we describe the decisions the user has to make to use EMALAM and the input of EMALAM. Finally, we describe the interpretation of the output and we name a possibility to depict the results.

2 Explanation EMALAM

We use `scipy.minimze` to minimize our objective function under some constraints. This is an often considered problem, see e.g. [4, 2, 3]. Here, we solved this problem with a new method, EMALAM.

For the optimization, recall that we are starting with some STRUCTURE outputs $q = (q_{ik})_{i=1,...,I,k=1,...,K}$ and $p = (p_{kjm})_{k=1,...,K,j=1,...,J_m,m=1,...,M}$. Our task is to minimize/maximize, for some $\mathcal{I} \subseteq \{1, ..., I\}$, using $e_{\mathcal{I}} = (1_{i\in\mathcal{I}})_{i=1,...,I}$,

$$f_k : S \mapsto \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} q_{ik'}S_{k'k} = \frac{1}{|\mathcal{I}|}e_{\mathcal{I}}qSe_k^\top \text{ for some } k, \text{ or}$$

$$g : S \mapsto \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\sum_k h\big(q_{i\cdot}Se_k^\top\big) \text{ with } h(a) = -a\log a$$

using the constraints $S1^\top = 1$, $q_{i\cdot}S \geq 0$, $S^{-1}p_{\cdot jm}^\top \geq 0$ for all k, j, m.

We use (for $K = 2$, other cases are analogous)

$$S = S(x) = \begin{pmatrix} x_0 & x_1 \\ x_2 & x_3 \end{pmatrix}, \qquad S(x)^{-1} = \begin{pmatrix} y_0 & y_1 \\ y_2 & y_3 \end{pmatrix}. \tag{2.1}$$ {?}

Note that, e.g.

$$Se_0^\top = A_0x^\top \text{ with } A_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

In general, the matrix A_k with

$$Se_k^\top = A_kx^\top$$

is given by the python command

`np.array([[1 if i == k + j*K else 0 for i in range(K*K)] for j in range(K)])`.

For the constraints, we have with the above notation

$$\begin{aligned} x_0 + x_1 &= 1, \\ x_2 + x_3 &= 1, \\ q_{i0}x_0 + q_{i1}x_2 &\geq 0, \\ q_{i0}x_1 + q_{i1x_2} &\geq 0, \\ p_{0jm}y_0 + p_{1jm}y_1 &\geq 0, \\ p_{0jm}y_2 + p_{1jm}y_3 &\geq 0. \end{aligned}$$

In order to numerically facilitate the optimization, we compute the Jacobi matrices

$$\nabla f_k(x) = \nabla\frac{1}{|\mathcal{I}|}e_{\mathcal{I}}qSe_k^\top = \frac{1}{|\mathcal{I}|}e_{\mathcal{I}}qA_k,$$

and

$$\nabla g(x) = \nabla\frac{1}{|\mathcal{I}|}\sum_i\sum_k h'(q_{i\cdot}A_kx)q_{i\cdot}A_k$$

with $h'(a) = -1 - \log(a)$.

3 Application Decisions for the User

information⟩ There are two different decisions that the user can make: The choice for the function that should be maximized and the choice which individuals should be considered. We explain the choice concerning the function that should be maximized first. Please note that there might occur slightly negative values or values that are slightly bigger than 1.

3.1 Choice of the Target Function

Researchers can choose four different pre-defined objective functions $f_{obj}(a)$. We discuss them in more detail now and explain how the user defines an other objective function by himself or herself.

Our algorithm for exploring the set of MLEs starts with a single MLE $\hat{q}, \hat{p}$, as e.g. provided by the output of STRUCTURE or ADMIXTURE. We then search for $S \in \mathbb{S}_{\hat{q},\hat{p}}$ as follows: Pick a subset J of individuals and (recalling that $H(x) = -\sum_k x_k \log x_k$ is the entropy of x which satisfies $x1^\top = 1$, which is maximal for the uniform distribution, and minimal for point measures) either

$$\text{(I): (a) minimize or (b) maximize } \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}(\hat{q}_{i\cdot}S_K)_k \text{ for some } k$$

$$\text{or (II): (a) minimize or (b) maximize } \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} H(\hat{S}_K q_{i,\cdot}^\top).$$

For (I), we have to fix the population that we consider and for (II) we have to name the individuals that are considered. To summarize, we can apply EMALAM to single individuals or to whole populations as described above to find the most extreme optima.

3.2 Label Switching

Additionally, the user can choose whether EMALAM should name the output so that label switching does not occur. The default is that EMALAM takes label switching into account. This means that we choose the permutation $\ell_1, \ldots, \ell_K$ of $\{1, \ldots, K\}$ of the output $\tilde{q}_{i,\ell_j} = (\hat{q}S)_{i,\ell_j}$ of EMALAM is so that

$$\sum_{i=1}^{N}\sum_{r=1}^{K}\sum_{k=1}^{K}|\hat{q}_{i,k} - (\hat{q}S)_{i,\ell_r}|$$

is minimized. If you wish to prevent this, you can use `--no_switch_labels`. You can also use `-minp` so that EMALAM considers the estimated allele frequencies instead of the estimated allele frequencies. Specifically, then EMALAM chooses $\ell_1, \ldots, \ell_K$ in order to minimize

$$\sum_{m=1}^{M}\sum_{j=1}^{J_m}\sum_{r=1}^{K}\sum_{k=1}^{K}|\hat{p}_{k,j,m} - (S^{-1}\hat{p})_{\ell_r,j,m}|.$$

4 Input of EMALAM

There is some information that the user has to provide to use EMALAM. We differentiate between the information that is required, i.e. has to be written in the command line and the information that is optional which you have to change directly in the code.

The command is e.g.

```
python emalam.py --structure_filename Example_Input/CEU_IBS_TSI_enhanced_corr_K3_f
--out out_q.txt out_p.txt --fun entropy --min
```

and every part of it is described below.

- –structure_filename : File path to a STRUCTURE output file. You can find an example for this file at our github Website.
- – hatq_filename (either this and – hatp_filename or –structure_filename is required): File path to the estimated IAs.
- – hatp_filename (either this and – hatp_filename or –structure_filename is required): File path to the estimated allele frequencies.
- –out: Names of the output files for the IAs and the allele frequencies. In the example above, out_q.txt is the name of the output file for the IAs and out_p.txt is the name of the output file for the allele frequencies.
- fun: The function that should be considered. You can choose entropy (i.e. EMALAM considers (II)) or size (i.e. EMALAM considers (I)). The default is entropy.
- –pop (required for fun == entropy): If fun == size, the index of the population which is to be optimized.
- –min (required): You can either choose min or max. For min, the function fun is minimized and for max this function is maximized.
- –n (optional): Number of iterations for the optimization. The default is 1.
- –inds (required for fun == size): The individuals which are used for the target function. If no names are given, a number starting with 0 is used. If missing, optimization is over all individuals.
- –no_switch_labels (optional): EMALAM does not take label switching into account.
- – minp (optional): Takes label switching of the allele frequencies into account. Do not use this together with –no_switch_labels.

5 Interpretation of the Output

We also provide an example output. The files are called
q_minimal.csv (for the IAs),
p_minimal.csv (for the allele frequencies),
and have the same format as the input. The rows in the output for the IAs stand for the individuals and the columns for the populations. For example, this file could contain
0.404 0.596
0.315 0.685
0.333 0.667
0.36 0.64
for four individuals and two populations. The rows of the output file for the allele frequencies represent the markers and the columns represent the populations, e.g.
0.385 0.246
0.615 0.754
0.001 0.002
0.800 0.817
0.199 0.181
represent two populations and two markers, the first one is bi-allelic and the second one has three alleles.

You can use the code `Create_Figures.py` to depict the different estimated IAs.

References

ader2009fast [1] David H Alexander, John Novembre, and Kenneth Lange. Fast model-based estimation of ancestry in unrelated individuals. *Genome research*, 19(9):1655–1664, 2009.

kraft1988 [2] Dieter Kraft. A software package for sequential quadratic programming. *Forschungsbericht-Deutsche Forschungs- und Versuchsanstalt fur Luft- und Raumfahrt*, 1988.

lalee1998 [3] Marucha Lalee, Jorge Nocedal, and Todd Plantenga. On the implementation of an algorithm for large-scale equality constrained optimization. *SIAM Journal on Optimization*, 8(3):682–706, 1998.

powell1994 [4] Michael JD Powell. *A direct search optimization method that models the objective and constraint functions by linear interpolation*. Springer, 1994.

ritchard2000 [5] Jonathan K Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.