Direct fast estimator for recombination frequency in the F_2 cross

Mathias Lorieux
DIADE, University of Montpellier, CIRAD, IRD, Montpellier, France
ORCID: 0000-0001-9864-3933
E-mail: mathias.lorieux@ird.fr

April 2021

Abstract

In this short note, a new unbiased maximum-likelihood estimator is proposed for the recombination frequency in the F_2 cross. The estimator is much faster to calculate than its EM algorithm equivalent, yet as efficient. Simulation studies are carried to illustrate the gain over another simple estimate proposed by Benito & Gallego (2004).

Keywords: Recombination fraction, F_2 cross, EM algorithm, efficient estimator

Conflict of interest statement: The author declared he has no conflict of interest.

1 Problem

Whole-genome sequencing allows constructing genetic maps with millions of loci. Unless the order of loci is known *a priori*, determining the order of the loci on linkage groups requires the computation of the recombination fractions for all pairs of loci. Computing time can be extremely long, especially in the F_2 cross where an iterative procedure is usually used. A faster yet efficient estimator is thus needed.

2 Development

Consider two loci for which we want to know their recombination frequency r. In the F_2 cross, it is not possible to count the recombinant individuals directly, since the double heterozygotes class is issued from either two recombinant ($AB|BA$ or $BA|AB$) or two non-recombinant ($AA|BB$) gametes, if A and B are the parental alleles. Those two configurations are generally indistinguishable. Thus,

we cannot estimate r directly since we need to solve the 3rd degree equation to get the efficient, unbiased maximum-likelihood estimator for r

$$2(c+g)+(b+d+f+h)-2\hat{r}n+2e\frac{\hat{r}^2}{(1-\hat{r})^2+\hat{r}^2}=0 \tag{1}$$

where c and g are the counts for the double recombinants $AB|AB$ or $BA|BA$, b, d, f and h are the counts for the single recombinants $AA|AB$, $AB|AA$, $AB|BB$ and $BB|AB$, and e is the count for the double heterozygotes class. Note that the non-recombinant classes $AA|AA$ and $BB|BB$, a and i, are not used since they are not informative for the estimation of r.

The common way to solve (1) is to use an iterative algorithm to estimate the fraction of the recombinants, $AB|BA$ or $BA|AB$, in the double heterozygote class. The most used in this case are the Newton-Raphson (see for instance Edwards 1972) and Expectation-maximization (EM) algorithms (Dempster et al, 1977). For instance, an EM algorithm to estimate the fraction of double recombinants in the e class was proposed by Mangin (1991)

$$\hat{t}=\frac{\hat{r}^2}{(1-\hat{r})^2+\hat{r}^2} \tag{2}$$

$$\hat{r}=\frac{2(c+g)+(b+d+f+h)+2\hat{t}e}{2n} \tag{3}$$

After convergence, we obtain the $\hat{r}_{EM}$ estimator, which asymptotic variance is

$$V\hat{r}=\frac{r(1-r)(1-2r+2r^2)}{2n(1-3r+3r^2)}$$

Iterative algorithms can require many iterations to reach convergence, depending on the starting parameter value for r, making them slow to calculate. Benito and Gallego (2014) proposed a simplified, faster estimator for r (here noted as $\hat{p}$) that ignores the double heterozygotes class. Following our notations, this estimator is

$$\hat{r}_{Benito}=\hat{p}=\frac{1}{n}\left(2(c+g)+\frac{1}{2}(b+d+f+h)\right) \tag{4}$$

With asymptotic variance

$$V\hat{p}=\frac{p(1+p)}{2n}$$

However, this estimator has greater variance than the $\hat{r}_{EM}$ estimator, especially when r is close to 0.5, which is problematic for testing linkage.

Now, note that if a value close enough to the true r is chosen to initiate the EM process, then convergence is reached in one step, allowing to reduce it to a direct estimator for r. One can use the unbiased Benito's estimator, $\hat{p}$, as the

starting value in a one-step EM process. For this, we replace $\hat{r}$ in the $\hat{t}$ EM formula (2) by $\hat{p}$

$$\hat{r}_{2021} = \frac{1}{2n}\left(2(c+g) + (b+d+f\ \ +h) + 2e\frac{\hat{p}^2}{(1-\hat{p})^2+\hat{p}^2}\right) \tag{5}$$

then we plug in (4) in (5)

$$\hat{r}_{2021} = \frac{1}{2n}\left(2(c+g) + (b+d+f+h)\right.$$
$$\left.+2e\frac{(2(c+g)/n + (b+d+f+h)/2n)^2}{(1-(2(c+g)/n+(b+d+f+h)/2n))^2 + (2(c+g)/n+(b+d+f+h)/2n)^2}\right)$$

Let $R = c+g$ and $H = b+d+f+h$ in (4). Then

$$\hat{r}_{2021} = \frac{1}{2n}\left(2R + H + 2e\frac{(2R/n+H/2n)^2}{(1-(2R/n+H/2n))^2+(2R/n+H/2n)^2}\right) \tag{6}$$

The fastest way to compute $\hat{r}$ may depend on the programming language. However, multiplication is usually faster than division in computers, so one might want to pre-calculate R, H, $D = 1/2n$, $E = 4DR+DH$ and $F = E^2$ then apply the formula

$$\hat{r}_{2021} = D\left(2R + H + 2e\frac{F}{1-2E+2F}\right) \tag{7}$$

3 Discussion

This new estimator $\hat{r}_{2021}$ is equivalent to $\hat{r}_{EM}$ after convergence and is therefore also unbiased and has the same asymptotic variance. Moreover, it can be more precise than $\hat{r}_{EM}$ especially when few iterations are allowed to speed-up the computation. An estimator's efficiency is obtained by calculating its mean-squared error (MSE), which is the sum of its variance and the squared difference between its estimated and expected values (*i.e.*, its bias). The smaller the MSE, the greater the efficiency. Note that if an estimator is unbiased, the asymptotic MSE is simply the variance, however in a finite number of observations it is preferable to keep the observed bias in the MSE formula. Figure 1 shows plots of the MSEs and standard deviations of three estimators for r: $\hat{r}_{EM}$, $\hat{r}_{Benito}$ or $\hat{p}$ and formula (7) $\hat{r}_{2021}$ in simulated F_2 populations. We see that the relative efficiency of the new estimator over Benito and Gallego's $\hat{p}$ increases continuously with r. In the test carried in this study, formula (7) was 3 to 17.5 times faster than the EM algorithm estimator to compute in 95% of the cases. The relative computation time depended mostly on both the values of r and the initialization value r_{Start} (Figure 2). This of course also depends on the stop value set for the convergence. In this simulation study, it was set to 0.00001 to ensure comparable precision with the two other estimators. In Supplemental data 1, an Excel workbook is provided to replicate the simulation studies with any population size, number of repetitions and value for r.

4 Conclusion

The new proposed estimator has the desirable properties of efficiency and is considerably faster to compute than the EM algorithm. It is more efficient than Benito and Gallego's estimator. Also, in case of segregation distortion that would favor the double heterozygotes class e, Benito and Gallego's estimator would rely on a smaller number of observations than expected and would have a larger standard deviation. The new estimator should thus be used instead of r_{EM} in F_2 crosses when the number of loci is very large, and instead of or r_{Benito} in any case.

5 References

Benito, C., and Gallego, A. (2004). A simple method for the estimation of recombination frequencies and genetic distances. Cellular and Molecular Biology Letters, 9(4A), 617–634.

Dempster A.P., N.M. Laird et D.B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society 39: 1-38.

Edwards. 1972. Likelihood. The John Hopkins University Press, Baltimore, 275 pp.

Mangin B. 1991. Construction de cartes génétiques : quelques méthodes. In: Méribel 91, Méribel, France, p. 1-4.

6 Figures

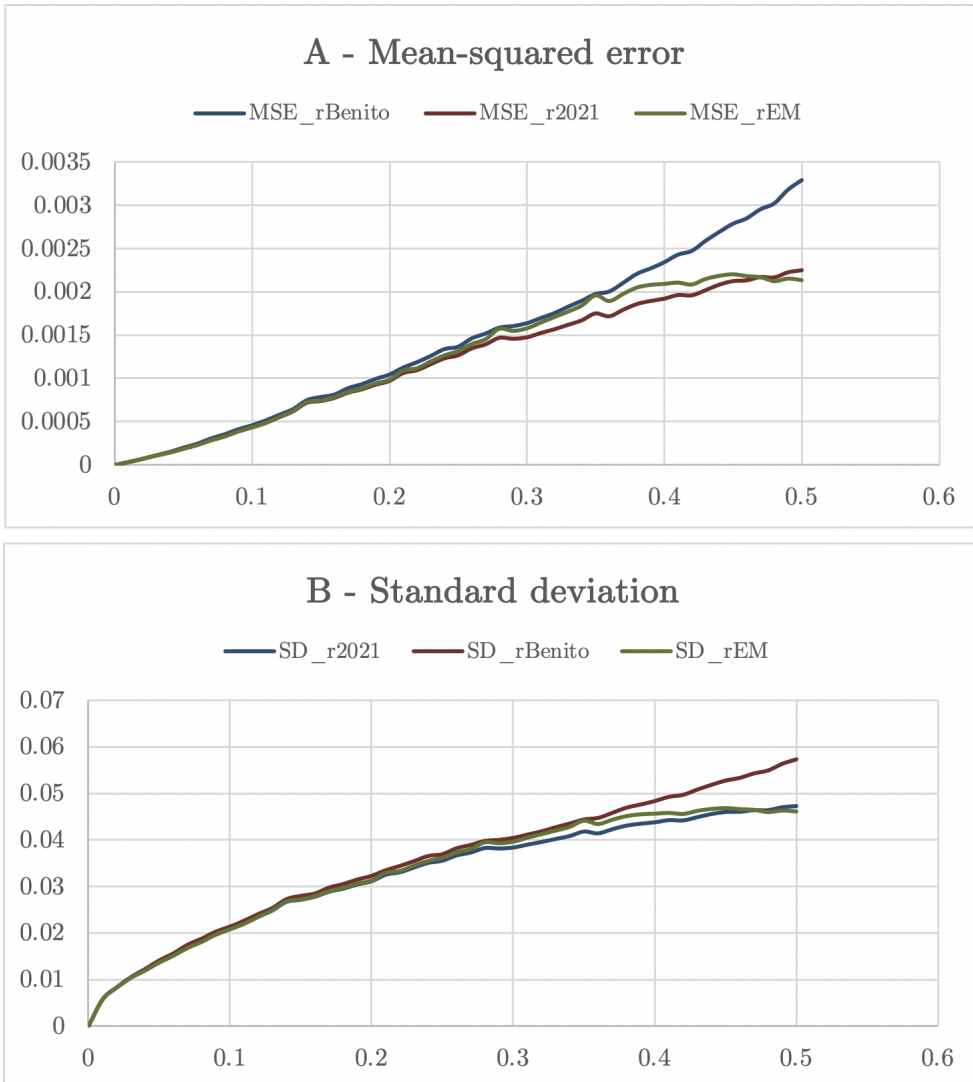

Figure 1: Mean-squared errors and standard deviations of three estimators for r, obtained by simulation of 1,000,000 F_2 populations of size $n = 100$. Horizontal axis: expected recombination frequency r. A: Mean-squared errors for maximum-likelihood/EM (MSE_r_EM) r, Benito and Gallego's $\hat{p}$ (MSE_r_Benito) and formula (7) (MSE_r_2021). B: standard deviations. Convergence of $\hat{r}_{EM}$ was stopped at 0.00001 and the initialization parameter was set to 0.25.

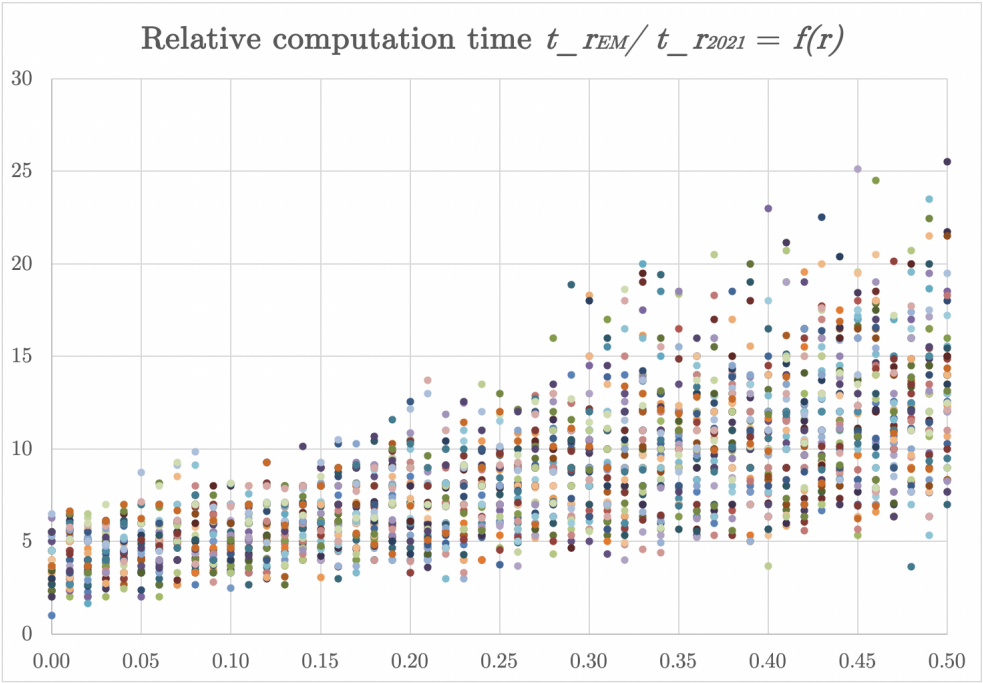

Figure 2: Relative computation time t_{rEM}/t_{r2021}, obtained by simulation of 500,000 F_2 populations of size $n = 100$. Horizontal axis: expected recombination frequency r. Convergence of $\hat{r}_{EM}$ was stopped at 0.00001. Each curve represents a different value for the initialization parameter, which varied between 0 and 0.5.