Haplotype phasing using EM & divide and conquer

July 15, 2018

The application

We developed a lightweight software for haplotype phasing, consisting of two scripts: `main.py`, the main interface and `phase.py`, containing the implementations. Usage of the scripts are described in `readme.txt`.

Methodology

EM is widely used for haplotype phasing; however it has poor scalability because its computational time increases exponentially with the length of input genotype sequence. To phase arbitrarily long genotypes, we divided the input genotypes (sequences of $\{0, 1, 2\}$) into atomic l-length segments, apply EM to each of the l-length segments, and finally merge the individually phased segments together in a reasonable way.

EM

We extend the EM algorithm such that it deals with individuals coming from an admixture of populations. Let p_{hk} be the frequency of haplotype h in population k, q_k be the probability that a randomly chosen haplotype is from popoulation k. Let $C(x_i)$ be the set of all compatible haplotypes with genotype x_i. Starting with an initial guess of p and q as p^0, q^0, repeatedly update with:

$$a^t_{i,k_1,k_2,h_1,h_2} = \frac{q^t_{k_1} q^t_{k_2} p^t_{h_1 k_1} p^t_{h_2 k_2}}{\sum_{k,k'} \sum_{h,h' \in C(x_i)} q^t_k q^t_{k'} p^t_{hk} p^t_{h'k'}}$$

q^{t+1}_k = Sum of all a with at least 1 k in subscript

p^{t+1}_{hk} = Sum of all a with at least 1 k and at least 1 h in subscript

p and q are normalized such that all q sum to 1 and all p for a particular k sum to 1. After convergence, the most probable phase (and population for each haplotype) is derived .

Merging

After the phase for each of the two neighboring segments A and B are found, an additional round of EM is applied to a w-length window C centering between A and B. This will determine whether the two segments are relatively *trans*- or *cis*-phased. For example if $w = 6$, $A = 212210$ and $B = 112200$, we attempt to find the haplotype frequencies within the window 210112. If we have already phased A to be $101110, 101100$ and B to be $111100, 001100$, and by doing EM on C we discovered that the the pair $100111, 110001$ has higher likelihood than $110111, 100001$, we can assume that the more probable phase for AB is 101110001100, 101100111100.

Additional notes

To account for the fact that this merging method is greedy and that the optimal phase for a segment may not be in the optimal phase for the full genotype, a command line option allows the program to save the top k phases for a segment, and during merging these phases are re-ranked based how how likely they appear in the window C. Experiments suggest that doing so consistently improves switch accuracy. In addition, the initial guess for q is randomized but the user can input a custom guess. As a result with multiple populations the performance is highly dependent on this initial guess. Therefore, using only 1 population is recommended.