Variational Bayes and parallel algorithms for fitting mixture models to large genotype data sets

Dan Davison

August 6, 2010

Contents

1 Introduction

Inference of population structure is often the main motivation for creating data sets of genetic variation at the level of species and populations. In addition, information about population structure is often required in other inference problems. A now familiar example is genome-wide association study of common diseases in humans. These studies seek to identify genomic regions within which genetic variation is associated with the phenotype of interest. Anticipated penetrance is typically very low for phenotypes of interest, with the result that well-powered studies require large sample sizes. In this situation, subtle correlations between genome-wide ancestry and phenotype result in spurious association signals in a naive analysis. This problem can be avoided to a large degree if information about ancestry (i.e. population structure) is available.

Starting with the work of Pritchard *et al.* (2000) (PSD), whose software package `structure` has been widely used during the last decade, several studies have developed Bayesian mixture models for multilocus genotype data with the objective of characterising population structure (Pritchard *et al.* 2000, Corander *et al.* 2003, Guillot *et al.* 2005, Huelsenbeck & Andolfatto 2007, Leslie & Donnelly ????). However, in the last five years those analysing large SNP data sets (10^5-10^6 loci, 10^3-10^4 individuals) have largely abandoned these methods due to the long computing times required by the available implementations, turning instead to Principal Components Analysis (PCA) (**???**). Here I describe two new approaches for fitting mixture models to genotype data, which can result in practical computing times when analysing such large data sets. The first approach is variational Bayes. VB offers the possibility of obtaining approximate posterior densities via an EM-like hill-climbing algorithm. The second approach is parallel computing, which results in computation times which scale inversely with the number of processors available. I describe and implement parallel algorithms for EM, VB and MCMC.

2 Overview

In section 3 I describe the models and introduce notation. The models studied are the "No Admixture Model" and the "Admixture Model" of Pritchard *et al.* (2000), and the "F Model" of Falush *et al.* (2003). In section 5 I describe new algorithms for fitting the three models. In each case I start with the EM algorithm, which provides a helpful comparison with the corresponding VB algorithm. Finally I describe parallel versions of the same algorithms. Additional derivation of the updates used are provided in the 8. Section 6 presents results of using the different methods to fit the models to simulated and real data. Finally in section 7 I discuss potential applications of these methods. The algorithms are implemented in the software package `psi` available at [psi URL]. Table 1 contains notation used throughout the paper.

3 Models

3.1 No Admixture model

In the basic mixture modelling problem, n items $\{X_i,\ldots,X_n\}$ are observed and each is assumed to belong to one of K groups. Thus for each item there is an unknown label $Z_i \in \{1,\ldots,K\}$ indicating the group to which it belongs. The main objectives are to learn about the values of these labels, and about the value of K. Group k is characterised by a probability distribution $f(x; \phi_k)$, and inference for K and Z typically also requires inference for the parameters ϕ_k.

The "No-Admixture Model" of Pritchard *et al.* (2000) is an example of this class of models: X_i is a data set of multilocus genotypes for individual i and the "groups" can be thought of as idealized biological populations from which the study individuals have ancestry. These populations are fully characterised by the allele frequencies at each locus. Hardy-Weinberg and linkage equilibrium are assumed so that the data for an individual from population k would be simulated from the prior by drawing alleles from the allele frequency distribution for population k, independently across chromosomes and loci. I use a Dirichlet prior distribution for the allele frequencies in each population at each locus with hyperparameters $\alpha^{(0)}$, independently across loci and groups. [1] The following algorithm simulates a data set from this model.

Algorithm 1 No Admixture Model

for each individual i
  $Z_i \sim Q$

for each locus l
  for each group k
    $P_{lk} \sim \text{Dirichlet}(\alpha^{(0)})$

  for each individual i
    for each chromosome a
      $X_{ila} \sim P_{lZ_i}$

[2]

See PSD for further details.

3.2 Admixture model

An important early contribution to the literature on mixture models for studying population structure (references above) is the "Admixture Model" (AM) introduced by Pritchard *et al.* (2000), in which alleles at different loci or on different chromosomes within a single individual may have been inherited from different groups. Thus the integer-valued labels Z_i become integer-valued matrices in which Z_{ila} is the label of the group from which the allele on chromosome a at locus l in individual i was inherited. In general it is not possible to infer the values of the Z_{ila}; instead we are interested in the genome-wide proportion Q_{ik} of ancestry for individual i in group k. The prior distributions of the ancestry proportions are Dirichlet with hyperparameters $\lambda^{(0)}$, independently across individuals, and the prior for the allele frequencies is the same as in the No Admixture Model. The following algorithm simulates a data set from the Admixture Model model.

[1] **FIXME** Dimension of Dirichlet varies with number of alleles at locus

[2] **FIXME** Why is Q a prior probability distribution in NAM, but something drawn from a Dirichlet prior in AM?

Algorithm 2 Admixture Model

for each individual i
$Q_i \sim \text{Dirichlet}(\lambda^{(0)})$

for each locus l
for each group k
$P_{lk} \sim \text{Dirichlet}(\alpha^{(0)})$

for each individual i
for each chromosome a
$Z_{ila} \sim Q_i$
$X_{ila} \sim P_{lZ_{ila}}$

3.3 F model

The F model of **?**Falush *et al.* (2003) models shared ancestry of populations by introducing an ancestral population into the model. The populations in the mixture are characterised by allele frequencies which depend on the frequency in the ancestral population. The following algorithm simulates a data set under the F model without admixture. The extension to admixture is obvious by comparison with algorithm **??** above. [3]

4 Introduction to Variational Bayes

The basic idea of VB is to assume a specific parametric form for the posterior density, and then to optimize the values of the hyperparameters via a hill-climbing algorithm. Thus, in principle, VB makes Bayesian posterior densities available without imposing the computational burden of exploring the support of the posterior via a Markov-chain sampler. Whether or not this results in a more attractive procedure than MCMC is discussed in section 7.

For observed data X and unobserved parameters ϕ we can write

$$\Pr(X|K) = \frac{\Pr(\phi, X|K)}{\Pr(\phi|X, K)} = \frac{p(\phi, X)}{q^*(\phi)},$$

where $q^*(\phi)$ denotes the (unknown) true posterior density of parameters ϕ and $p(\phi, X)$ is the complete data likelihood (for the purposes of this section, ϕ includes the integer-valued membership indicators Z, as well as the real-valued parameters P and Q).

[taking logs and integrating with respect to ϕ, this can be written as...]

Taking logs and integrating with respect to some distribution $q(\phi)$ (this will be the approximate posterior density, and in practice it will be chosen to have a convenient parametric form) gives

[3] **FIXME** F model here or in discussion?

$$\begin{aligned}\log \Pr(X|K) = & \int \log p(\phi, X) q(\phi) d\phi - \int \log q^*(\phi) q(\phi) d\phi \\ = & \int \log \frac{p(\phi, X)}{q(\phi)} q(\phi) d\phi - \int \log \frac{q^*(\phi)}{q(\phi)} q(\phi) d\phi \\ = & F(q, p) + d_{KL}(q \mid\mid q^*).\end{aligned}$$

The first term $F(q, p)$ is a functional of the approximate posterior q and the complete data likelihood p, and the second term is the Kullback-Leibler divergence between $q(\phi)$ and the unknown true posterior $q^*(\phi)$. While the first term can be evaluated, the second cannot. Since $\Pr(X|K)$ is a constant, maximizing $F(q, p)$ corresponds to minimizing a reasonable measure of the distance between the approximate posterior and the true posterior. The next section describes hill-climbing algorithms at each iteration of which an increase in the value of $F(q, p)$ is guaranteed. When these algorithms reach convergence, the final value of q can be used as an approximation of the true posterior density. Furthermore, since the maximum value of $F(q, p)$ approximates $\log \Pr(X|K)$, the posterior distribution of the number K of mixture components can be investigated by fitting the model at several different values of K.
[4]

5 Fitting the models via Variational Bayes

5.1 General comments

Pritchard *et al.* (2000) and Falush *et al.* (2003) described how to fit the above models using MCMC. In this section I describe how to fit these models using Variational Bayes (VB). The VB algorithms bear a strong similarity to Expectation-Maximization (EM) algorithms, and a simple heuristic descriprion is as follows.

1. Set parameters (EM) / hyperparameters (VB) to their initial values.
2. **E step** Compute the discrete probability distribution Pr(Z|X) on the unknown cluster indicators, using the current parameter estimates.
3. **M step** Use the current distribution Pr(Z|X) to update the parameter estimates.
4. Stop if converged, otherwise go to (2).

In the following sections I describe the E step, the M step and how to assess convergence. In EM, the E step is accomplished straightforwardly using Bayes rule and current point estimates of the parameters P and Q. In contrast, in VB the "parameters" are hyperparameters $\alpha^{(1)}$ and $\lambda^{(1)}$ of the posterior density, and the E step is accomplished by averaging over the current posterior densities for P and Q. **?** show that this is achieved by the following update scheme:

E Step Set $q(Z) \propto \exp\left\{\mathrm{E}_{q(\phi)} \log p(Z, X|\phi)\right\}$

M Step Set $q(\phi) \propto \Pr(\phi) \exp\left\{\mathrm{E}_{q(Z)} \log p(Z, X|\phi)\right\}$
[5]

[4] **FIXME** In the next section we switch back to separating the integer-valued Z from the real-valued ϕ
[5] **FIXME** Not going to discuss Gibbs sampler algorithm at all?

5.2 No Admixture model

In this case the parameters are P (allele frequencies) and Q (cluster intensities). We start by considering the EM algorithm for fitting this model (Algorithm 10).

5.2.1 EM algorithm

Algorithm 3 No Admixture model: EM

E step

for each individual i
  for each group k
    $\gamma_{ik} \leftsquigarrow Q_k \prod_l \prod_{a=1}^{2} P_{lkX_{ila}}$

M step

for each group k
  $Q_k \leftarrow \frac{1}{n} \sum_i \gamma_{ik}$
  for each locus l
    for each allele type j
      $P_{lkj} \leftarrow \frac{\sum_i \sum_a I(X_{ila}=j)\gamma_{ik}}{\sum_i \sum_a \gamma_{ik}}$

```
FIXME: Monitoring convergence for EM
```

5.2.2 VB algorithm

To fit the No Admixture model via VB, we specify that the approximate posterior density $q(Z, Q, P)$ can be factorised as $q(Z)q(Q)q(P)$ and that each of these three components has the same parametric form as the prior, differing only in the hyperparameters. In other words, we specify that $q(Q)$ is Dirichlet$(\lambda_1^1, \ldots, \lambda_K^1)$, and that $q(P_{lk\cdot})$ is Dirichlet$(\alpha_{lk1}^1, \ldots, \alpha_{lkJ_l}^1)$, independently for all l, k.
[6] [7]

Let $\gamma_k^i = q(z_i = k)$

- Monitoring convergence
  The E and M steps are iterated until the increase in $F(q, p)$ is sufficiently small that convergence is judged to have been reached, which means that it is necessary to evaluate $F(q, p)$ at the end of each iteration. Since $q()$ factorises by assumption/definition,

[6] **FIXME** what are we saying about q(Z)
[7] **FIXME** how is the notation going to differentiate among these different q distributions?

Algorithm 4 No Admixture model: VB (Overview)

E step

for each individual i
  for each group k
    $\gamma_{ik} \leftsquigarrow \exp\left\{\mathrm{E}_{q(\theta)} \log p(z_i = k, x_i|\theta)\right\}$

M step

$q(Q) \propto p(Q) \exp\left\{\mathrm{E}_{q(Z)} \log p(Z|Q)\right\}$
for each group k
  for each locus l
    $q(\mu) \propto p(\mu) \exp\left\{\mathrm{E}_{q(z)} \log p(x|\mu, z)\right\}$

Algorithm 5 No Admixture model: VB (Explicit)

E step

for each individual i
  for each group k
    $\gamma_{ik} \leftsquigarrow \exp\left\{\Psi\left(\lambda_k^{(1)}\right) - \Psi\left(\sum_{k'} \lambda_{k'}^{(1)}\right) + \sum_l \left[\sum_{a=1}^{2} \Psi\left(\alpha_{klX_{lia}}^{(1)}\right)\right] - 2\Psi\left(\sum_{j'=1}^{J_l} \alpha_{klj'}^{(1)}\right)\right\}$

M step

for each group k
  $\lambda_k^{(1)} \leftarrow \lambda_k^{(0)} + \sum_i \gamma_{ik}$
  for each locus l
    for each allele type j
      $\alpha_{lkj}^{(1)} \leftarrow \alpha_{lkj}^{(0)} + \sum_i \sum_a \gamma_{ik} I(X_{ila} = j)$

$$
\begin{aligned}
F(q,p) = & \int q(\theta)q(z)\log\frac{p(\theta)p(z,x|\theta)}{q(\theta)q(z)}d\theta dz \\
= & \int q(\theta)\log\frac{p(\theta)}{q(\theta)}d\theta + \int q(\theta)q(z)\log\frac{p(z,x|\theta)}{q(z)}d\theta dz \\
= & -d_{KL}(q||p) + \mathrm{E}_{q(Q,z)}\log p(z|Q) + \mathrm{E}_{q(\mu,z)}\log p(x|z,\mu) + H\left(q(z)\right),
\end{aligned}
$$

where $H\left(q(z)\right) = -\int q(z)\log q(z)dz$ is the Shannon entropy of $q(z)$. Computation of these four terms is described in Appendix 8.1.1 .

5.3 Admixture model

In the Admixture model, the unobserved quantities are Z, Q and P.

5.3.1 EM

Algorithm 6 Admixture model: EM

E step

for each individual i
  for each locus l
    for each chromosome a
      for each group k
        $\gamma_{ilak} \leftsquigarrow Q_{ik}\prod_l\prod_{a=1}^{2}P_{lkX_{ila}}$

M step

for each group k
  for each individual i
    $Q_{ik} \leftarrow \frac{1}{2L}\sum_l\sum_a\gamma_{ilak}$
  for each locus l
    for each allele type j
      $P_{lkj} \leftarrow \frac{\sum_i\sum_a I(X_{ila}=j)\gamma_{ilak}}{\sum_i\sum_a\gamma_{ilak}}$

5.3.2 VB

As in the No Admixture model, we specify that approximate posterior density $q(Z,Q,P)$ can be factorised as $q(Z)q(Q)q(P)$ and that each of these three components has the same parameteric form as in the prior, differing only in the hyperparameters. Specifically, we specify that the ancestry vectors $q(Q_{i\cdot})$ are each Dirichlet$(\lambda^1_{i1},\ldots,\lambda^1_{iK})$ and, as in the No Admixture model, that $q(P_{lk\cdot})$ is Dirichlet$(\alpha^1_{lk1},\ldots,\alpha^1_{lkJ_l})$, independently for all l, k.

Algorithm 7 Admixture model: VB (Overview)

E step

for each individual i
  for each locus l
    for each chromosome a
      for each group k
        $\gamma_{ilak} \leftsquigarrow \exp\left\{\mathrm{E}_{q(P,Q)} \log \Pr(Z_{ila} = k, X_{ila} | P, Q)\right\}$

M step

for each individual i
  $q(Q_{i\cdot}) \propto \Pr(Q_{i\cdot}) \exp\left\{\mathrm{E}_{q(Z)} \log \Pr(Z|Q_i)\right\}$
for each group k
  for each locus l
    $q(P_{lk\cdot}) \propto \Pr(P_{lk\cdot}) \exp\left\{\mathrm{E}_{q(Z)} \log p(X|P_{lk\cdot}, Z)\right\}$

Algorithm 8 Admixture model: VB (Explicit)

E step

for each individual i
  for each locus l
    for each chromosome a
      for each group k
        $\gamma_{ilak} \leftsquigarrow \exp\left\{\Psi\left(\lambda_{ik}^{(1)}\right) - \Psi\left(\sum_{k'=1}^{K} \lambda_{ik'}^{(1)}\right) + \Psi\left(\alpha_{klX_{lia}}^{(1)}\right) - \Psi\left(\sum_{j'=1}^{J_l} \alpha_{klj'}^{(1)}\right)\right\}$

M step

for each group k
  for each individual i
    $\lambda_{ik}^{(1)} \leftarrow \lambda_{ik}^{(0)} + \sum_{l=1}^{L} \sum_{a=1}^{2} \gamma_{ilak}$
  for each locus l
    for each allele type j
      $\alpha_{lkj}^{(1)} \leftarrow \alpha_{lkj}^{(0)} + \sum_{i=1}^{n} \sum_{a=1}^{2} \gamma_{ilak} I(X_{ila} = j)$

Let $\gamma_k^{ila} = q(z_{ila} = k)$
In Appendix **??** it is shown that

$$\log \gamma_k^{ila} = \Psi\Big(\lambda_{ik}^1\Big) - \Psi\Big(\sum_{k'} \lambda_{ik'}^1\Big) + \Psi\Big(\alpha_{klx_{lia}}^1\Big) - \Psi\Big(\sum_{j'=1}^{J_l} \alpha_{klj'}^1\Big),$$

where Ψ is the digamma function.

5.3.3 M step

Using the current distribution $p(z)$, the M step involves setting $q(\theta)$ proportional to

$$\begin{aligned} & p(\theta) \exp\left\{\mathrm{E}_{q(z)} \log p(z, x|\theta)\right\} \\ & \qquad = \quad p(Q) \exp\left\{\mathrm{E}_{q(Z)} \log p(Z|Q)\right\} \times p(\mu) \exp\left\{\mathrm{E}_{q(z)} \log p(x|\mu, z)\right\}, \end{aligned}$$

and so the updates for $q(Q)$ and $q(\mu)$ can be performed separately, by setting

$$q(Q) \propto p(Q) \exp\left\{\mathrm{E}_{q(z)} \log p(z|Q)\right\} \quad \text{and} \quad q(\mu) \propto p(\mu) \exp\left\{\mathrm{E}_{q(z)} \log p(x|\mu, z)\right\}.$$

- Updating the approximate posterior on admixture proportions
  The hyperparameters of $q(Q)$ are updated according to the following algorithm (see Appendix **??**):
- Updating the approximate posterior on allele frequencies
  The hyperparameters of $q(\mu)$ are updated according to the following algorithm (see Appendix 8.1.2):

5.3.4 Monitoring convergence

Since $q()$ factorises by definition,

$$\begin{aligned} F(q,p) = & \int q(\theta) q(z) \log \frac{p(\theta) p(z, x|\theta)}{q(\theta) q(z)} d\theta dz \\ = & \int q(\theta) \log \frac{p(\theta)}{q(\theta)} d\theta + \int q(\theta) q(z) \log \frac{p(z, x|\theta)}{q(z)} d\theta dz \\ = & - d_{KL}(q||p) + \mathrm{E}_{q(Q,z)} \log p(z|Q) + \mathrm{E}_{q(\mu,z)} \log p(x|z, \mu) + H\left(q(z)\right), \end{aligned}$$

where $H\left(q(z)\right) = -\int q(z) \log q(z) dz$ is the Shannon entropy of $q(z)$. Computation of these four terms is described in Appendix 8.1.2.

5.4 Admixture model with correlated allele frequencies

The correlated frequencies model affects how we update $q(\mu)$. The E step is unchanged, as this involves estimating $q(z)$ given the current $q(\mu, Q)$. In the M step, the update of $q(Q)$ is also unchanged, as this doesn't involve μ. I think the update of $q(\mu)$ in the correlated frequencies model differs only in that the 'prior counts' of the number of copies of allele j observed in population k at locus l are now given by α_{lkj}^0

5.5 Parallel algorithms

Available computational techniques for fitting these models typically involve iterative hill-climbing algorithms, or Markov Chain Monte Carlo, and therefore in their simplest form are not parallelisable, since each iteration depends on the preceding iterations [8]. However, at each iteration, computations can be carried out in parallel processes, provided that certain across-process averages are computed at the end of each iteration. Two strategies for parallelisation are possible in practice: division-by-loci and division-by-individuals. Here I focus on division-by-loci. One reason for doing so is that the number of loci is typically larger than the number of individuals and therefore more often exceeds p, the number of processes.

5.5.1 Admixture model

- EM

Algorithm 9 Admixture model: EM: Parallel

for each process c **in parallel**

E step

for each individual i
for each locus $l \in {}_c$
for each chromosome a
for each group k
$\gamma_{ilak} \leftarrow\!\!\sim Q_{ik} \prod_l \prod_{a=1}^{2} P_{lkX_{ila}}$

M step

for each group k
for each individual i
$Q_{ik}^{(c)} \leftarrow \frac{1}{2L_p} \sum_{l=1}^{L_p} \sum_{a=1}^{2} \gamma_{ilak}$
for each locus $l \in {}_c$
for each allele type j
$P_{lkj} \leftarrow \frac{\sum_i \sum_a I(X_{ila}=j)\gamma_{ilak}}{\sum_i \sum_a \gamma_{ilak}}$

for each individual i
for each group k
$Q_{ik} \leftarrow \sum_{c=1}^{C} \frac{L_c Q_{ik}^{(c)}}{L}$

- VB

[8] **FIXME** Strategies involving simulating multiple Markov chains, or repeated hill-climbinging from different starting points, are of course amenable to parallelisation

Algorithm 10 Admixture model: VB (Explicit)

E step

for each individual i
  for each locus l
    for each chromosome a
      for each group k

$$\gamma_{ilak} \leftarrow \exp\left\{\Psi\left(\lambda_{ik}^{(1)}\right) - \Psi\left(\sum_{k'=1}^{K} \lambda_{ik'}^{(1)}\right) + \Psi\left(\alpha_{klX_{lia}}^{(1)}\right) - \Psi\left(\sum_{j'=1}^{J_l} \alpha_{klj'}^{(1)}\right)\right\}$$

M step

for each group k
  for each individual i

$$\lambda_{ik}^{(1)} \leftarrow \lambda_{ik}^{(0)} + \sum_{l=1}^{L} \sum_{a=1}^{2} \gamma_{ilak}$$

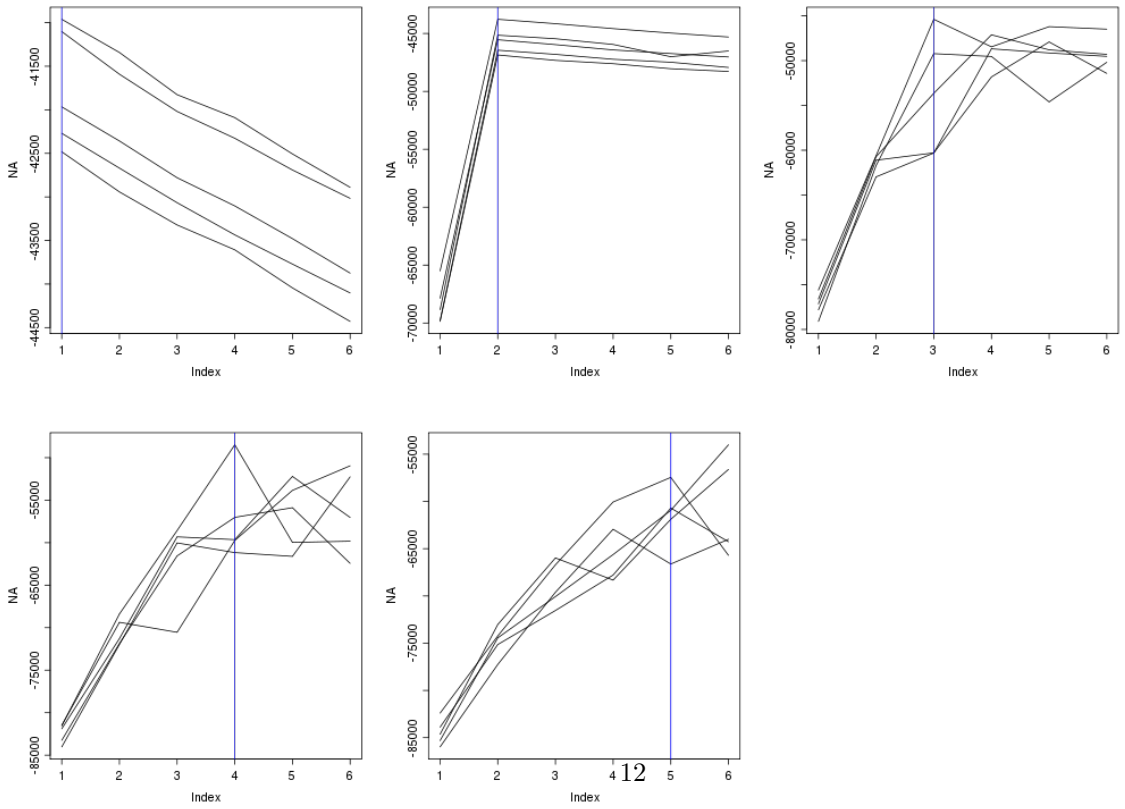
  for each locus l
    for each allele type j

$$\alpha_{lkj}^{(1)} \leftarrow \alpha_{lkj}^{(0)} + \sum_{i=1}^{n} \sum_{a=1}^{2} \gamma_{ilak} I(X_{ila} = j)$$

6 Results

6.1 Known K

6.2 Inferring K



6.3 Parallel processing

7 Discussion

Pritchard *et al.* (2000) introduced an AM for loosely linked markers in which the ancestry labels $\mathrm{Z_{i.a}}$ are autocorrelated along a chromosome due to linkage. In this situation it can be possible to estimate $\mathrm{Z_{ila}}$ at each locus. A disadvantage of methods based on PCA is that they are not easily extended in this manner: the principal components are eigenvectors of a covariance matrix which is estimated by averaging across all loci.

8 Appendix

8.1 Updates in variational Bayes algorithm

8.1.1 No-admixture model

- E step
  We need to evaluate
  $\gamma_k^i \propto \exp\left\{\mathrm{E}_{q(\theta)} \log p(z_i = k, x_i|\theta)\right\}$.
  The complete-data log likelihood is

$$\begin{aligned}\log p(z_i = k, x_i|\theta) &= \log Q_k + \sum_l \sum_{a=1}^{2} \log p(x_{ila}|\mu_{kl\cdot}) \\ &= \log Q_k + \sum_l \sum_{a=1}^{2} \log \mu_{klx_{ila}},\end{aligned}$$

  so we need to evaluate integrals of the form
  $\int q(Q) \log Q_k dQ$ and $\int q(\mu_{kl\cdot}) \log \mu_{klj} d\mu_{kl\cdot}$.
  Since the distributions $q(Q)$ and $q(\mu_{kl\cdot})$ are both Dirichlet, these have the same form. The first is

$$\begin{aligned}\int q(Q) \log Q_k dQ &= \int \left[\frac{\Gamma\left(\sum_{k'} \lambda_{k'}^1\right)}{\prod_{k'} \Gamma\left(\lambda_{k'}^1\right)} \prod_k Q_k^{\lambda_k^1 - 1}\right] \log Q_k dQ \\ &= \Psi\left(\lambda_k^1\right) - \Psi\Big(\sum_{k'} \lambda_{k'}^1\Big),\end{aligned}$$

  where Ψ is the digamma function, and the second one is

$$\int q(\mu_{kl\cdot}) \log \mu_{klj} d\mu_{kl\cdot} = \Psi\left(\alpha_{klj}^1\right) - \Psi\Big(\sum_{j'} \alpha_{klj'}^1\Big).$$

The expectation that we are trying to evaluate is then

$$\begin{aligned}
\log \gamma_k^i &= \mathrm{E}_{q(\theta)} \log p(z_i = k, x_i|\theta) \\
&= \int q(Q) \log Q_k dQ + \sum_l \sum_{a=1}^{2} \int q(\mu_{lk\cdot}) \log \mu_{lkx_{ila}} d\mu_{lk\cdot} \\
&= \Psi\Big(\lambda_k^1\Big) - \Psi\Big(\sum_{k'} \lambda_{k'}^1\Big) + \sum_l \left[\sum_{a=1}^{2} \Psi\Big(\alpha_{klx_{lia}}^1\Big)\right] - 2\Psi\Big(\sum_{j'=1}^{J_l} \alpha_{klj'}^1\Big).
\end{aligned}$$

- M step
  - Updating the hyperparameters of $q(Q)$
    We want to set $q(Q)$ proportional to
    $p(Q) \exp\big\{\mathrm{E}_{q(z)} \log p(z|Q)\big\}$.
    The expectation is

$$\begin{aligned}
\mathrm{E}_{q(z)} \log p(z|Q) = \mathrm{E}_{q(z)} \sum_i \log Q_{z_i} &= \sum_{z_1,\ldots,z_n} \sum_i \left[\log Q_{z_i}\right] \gamma_{1z_1}, \ldots, \gamma_{nz_n} \\
&= \sum_i \sum_k \gamma_k^i \log Q_k \\
&= \sum_k \log Q_k^{n_k}
\end{aligned}$$

    where $n_k = \sum_i \gamma_k^i$ is the current approximate posterior expected number of individuals assigned to population k. Therefore

$$p(Q) \exp\big\{\mathrm{E}_{q(z)} \log p(z|Q)\big\} \propto \prod_k Q_k^{\lambda_k^0 - 1 + n_k},$$

    and the update is achieved by setting the hyperparameters equal to the sum of the prior counts and the current approximate posterior expected counts:

$$\lambda_k^1 \leftarrow \lambda_k^0 + n_k.$$

  - Updating the hyperparameters of $q(\mu)$
    We want to set $q(\mu)$ proportional to
    $p(\mu) \exp\big\{\mathrm{E}_{q(z)} \log p(x|\mu, z)\big\}$.
    This factorises across loci and populations as

$$
\begin{aligned}
p(\mu)\exp\left\{\mathrm{E}_{q(z)}\log p(x|\mu,z)\right\} &= \left[\prod_l\prod_k p(\mu_{lk})\right]\exp\left\{\sum_l\sum_i \mathrm{E}_{q(z_i)}\log p(x_{li\cdot}|\mu_{lz_i})\right\} \\
&= \prod_l\left[\prod_k p(\mu_{lk})\right]\exp\left\{\sum_i\sum_k \gamma_k^i \log p(x_{li\cdot}|\mu_{lk})\right\} \\
&= \prod_l\prod_k p(\mu_{lk})\exp\left\{\sum_i \gamma_k^i \log p(x_{li\cdot}|\mu_{lk})\right\},
\end{aligned}
$$

so the approximate posterior distributions on allele frequencies can be updated separately in each population and at each locus.

$$
\begin{aligned}
p(\mu_{lk})\exp\left\{\sum_i \gamma_k^i \log p(x_{li}|\mu_{lk})\right\} &= p(\mu_{lk})\exp\left\{\sum_i \gamma_k^i \sum_a\sum_j \log \mu_{lkj}^{I(x_{lia}=j)}\right\} \\
&\propto \prod_j \mu_{lkj}^{\alpha^0_{lkj}}\exp\left\{\sum_j \log\mu_{lkj}\sum_i\sum_a \gamma_k^i I(x_{lia}=j)\right\} \\
&= \prod_j \mu_{lkj}^{\alpha^0_{lkj}}\exp\left\{n_{lkj}\log\mu_{lkj}\right\},
\end{aligned}
$$

where $n_{lkj} = \sum_i\sum_a \gamma_k^i I(x_{lia}=j)$ is the expected number of j alleles observed at locus l in population k, with the expectation taken w.r.t. $q(z)$. This results in

$$
q(\mu_{lk}) \propto \prod_j \mu_{lkj}^{\alpha^0_{lkj}-1+n_{lkj}},
$$

which is fulfilled by setting the hyperparameters equal to the sum of the prior counts and the current approximate posterior expected counts:

$$
\alpha^1_{lkj} \leftarrow \alpha^0_{lkj} + n_{lkj}.
$$

- Monitoring convergence
  - The K-L divergence between prior and approximate posterior

$$
\begin{aligned}
d_{KL}(q||p) &= \int q(\theta)\log\frac{q(\theta)}{p(\theta)}d\theta \\
&= \int q(\mu)\log\frac{q(\mu)}{p(\mu)}d\mu + \int q(Q)\log\frac{q(Q)}{p(Q)}dQ \\
&= \sum_l\sum_k d_{KL}\Big(q(\mu_{lk\cdot})||p(\mu_{lk\cdot})\Big) + d_{KL}\Big(q(Q_\cdot)||p(Q_\cdot)\Big),
\end{aligned}
$$

in which the component densities are all Dirichlet. The K-L divergence of two Dirichlet densities with parameters $\alpha_1, \ldots, \alpha_S$ and $\beta_1, \ldots, \beta_S$ is given in equation 52 of **?** as

$$d_{KL}(\mathrm{Dir}(\alpha)||\mathrm{Dir}(\beta)) = \log \frac{\Gamma(\sum_s \alpha_s)}{\Gamma(\sum_s \beta_s)} + \sum_s \log \frac{\Gamma(\beta_s)}{\Gamma(\alpha_s)} + \sum_s (\alpha_s - \beta_s) \left(\Psi(\alpha_s) - \Psi(\sum_s \alpha_s) \right)$$

- The average missing data probability term

$$\begin{aligned}
\mathrm{E}_{q(Q,z)} \log p(z|Q) = & \sum_i \mathrm{E}_{q(z_i)} \mathrm{E}_{q(Q_\cdot)} \log Q_{z_i} \\
= & \sum_i \sum_k \gamma_k^i \int q(Q_\cdot) \log Q_k dQ_\cdot \\
= & \sum_i \sum_k \gamma_k^i \left[\Psi(\lambda_k^1) - \Psi(\sum_{k'} \lambda_{k'}^1) \right] \\
= & \left[\sum_i \sum_k \gamma_k^i \Psi(\lambda_k^1) \right] - n\Psi(\sum_{k'} \lambda_{k'}^1) \\
= & \left[\sum_k m_k \Psi(\lambda_{ik}^1) \right] - n\Psi(\sum_{k'} \lambda_{k'}^1),
\end{aligned}$$

  where $m_k = \sum_i \gamma_k^i$ is the expected number of individuals that derive from population k.
- The average log likelihood term

$$\begin{aligned}
\mathrm{E}_{q(\mu,z)} \log p(x|z,\mu) = & \sum_l \sum_i \sum_{a=1}^{2} \mathrm{E}_{q(z_i)} \mathrm{E}_{q(\mu_{lz_i \cdot})} \log p(x_{ila}|z_i, \mu_{lz_i x_{ila}}), \\
= & \sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^i \int q(\mu_{lk\cdot}) \log \mu_{lkx_{ila}} d\mu_{lk\cdot}. \\
= & \sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^i \left[\Psi(\alpha_{lkx_{ila}}^1) - \Psi(\sum_j \alpha_{lkj}^1) \right] \\
= & \sum_l \sum_k \sum_j \left[\Psi(\alpha_{lkj}^1) - \Psi(\sum_{j'} \alpha_{lkj'}^1) \right] \sum_i \sum_{a=1}^{2} \gamma_k^i I(x_{ila} = j) \\
= & \sum_l \sum_k \sum_j \left[\Psi(\alpha_{lkj}^1) - \Psi(\sum_{j'} \alpha_{lkj'}^1) \right] m_{lkj},
\end{aligned}$$

  where $m_{lkj} = \sum_i \sum_{a=1}^{2} \gamma_k^i I(x_{ila} = j)$ is the expected number of alleles of type j at locus l that derive from population k.

$$= \sum_l \sum_k \left[\sum_i \gamma_k^i \sum_{a=1}^{2} \Psi(\alpha_{lkx_{ila}}^1) \right] - n\Psi(\sum_{j'} \alpha_{lkj'}^1)$$

- The entropy of the probability distribution over the missing indicators

$$\begin{aligned} H\left(q(z)\right) = & -\mathrm{E}_{q(z)} \log q(z) \\ = & -\sum_i \sum_k \gamma_k^i \log \gamma_k^i \end{aligned}$$

8.1.2 Admixture model

- E step
  We need to evaluate
  $\gamma_k^{ila} \propto \exp\left\{\mathrm{E}_{q(\theta)} \log p(z_{ila} = k, x_{ila}|\theta)\right\}$.
  The complete-data log likelihood is

$$\log p(z_{ila} = k, x_{ila}|\theta) = \log Q_{ik} + \log \mu_{klx_{ila}},$$

  so we need to evaluate integrals of the form
  $\int q(Q_{i\cdot}) \log Q_{ik} dQ_{i\cdot}$ and $\int q(\mu_{kl\cdot}) \log \mu_{klj} d\mu_{kl\cdot}$.
  Since the distributions $q(Q_{i\cdot})$ and $q(\mu_{kl\cdot})$ are both Dirichlet, these have the same form. The first is

$$\begin{aligned} \int q(Q_{i\cdot}) \log Q_{ik} dQ_{i\cdot} &= \int \left[\frac{\Gamma\Big(\sum_{k'} \lambda_{ik'}^1\Big)}{\prod_{k'} \Gamma\Big(\lambda_{ik'}^1\Big)} \prod_{k'} Q_{ik'}^{\lambda_{ik}^1 - 1} \right] \log Q_{ik} dQ_{i\cdot} \\ &= \Psi\Big(\lambda_{ik}^1\Big) - \Psi\Big(\sum_{k'} \lambda_{ik'}^1\Big), \end{aligned}$$

  where Ψ is the digamma function, and the second one is

$$\int q(\mu_{kl\cdot}) \log \mu_{klj} d\mu_{kl\cdot} = \Psi\Big(\alpha_{klj}^1\Big) - \Psi\Big(\sum_{j'} \alpha_{klj'}^1\Big).$$

  The expectation that we are trying to evaluate is then

$$\begin{aligned} \log \gamma_{ilk} &= \mathrm{E}_{q(\theta)} \log p(z_{il} = k, x_{il}|\theta) \\ &= \int q(Q_{i\cdot}) \log Q_{ik} dQ_{i\cdot} + \int q(\mu_{lk\cdot}) \log \mu_{lkx_{ila}} d\mu_{lk\cdot} \\ &= \Psi\Big(\lambda_{ik}^1\Big) - \Psi\Big(\sum_{k'} \lambda_{ik'}^1\Big) + \Psi\Big(\alpha_{klx_{lia}}^1\Big) - \Psi\Big(\sum_{j'=1}^{J_l} \alpha_{klj'}^1\Big). \end{aligned}$$

- M step

- Updating the hyperparameters of $q(Q)$
  We want to set $q(Q)$ proportional to
  $p(Q)\exp\left\{\mathrm{E}_{q(z)}\log p(z|Q)\right\}$.
  This factorises across individuals as

$$p(Q)\exp\left\{\mathrm{E}_{q(z)}\log p(z|Q)\right\} = \prod_i p(Q_{i\cdot})\exp\left\{\mathrm{E}_{q(z_{i\cdot\cdot})}\log p(z_{i\cdot\cdot}|Q)\right\},$$

  so we can update the hyperparameters of $p(Q_{i\cdot})$ independently for each individual i. The expectation is

$$\begin{aligned}
\mathrm{E}_{q(z_{i\cdot\cdot})}\log p(z_{i\cdot\cdot}|Q) &= \mathrm{E}_{q(z\cdot\cdot)}\sum_l\sum_{a=1}^{2}\log Q_{iz_{ila}} \\
&= \sum_l\sum_{a=1}^{2}\sum_k \gamma_k^{ila}\log Q_{ik} \\
&= \sum_k [\log Q_{ik}]\sum_l\sum_{a=1}^{2}\gamma_k^{ila} \\
&= \sum_k \log Q_{ik}^{m_{ik}}
\end{aligned}$$

  where $m_{ik} = \sum_l\sum_{a=1}^{2}\gamma_k^{ila}$ is the current approximate posterior expected number of allele copies at all loci in individual i that derive from population k. Therefore

$$p(Q_{i\cdot})\exp\left\{\mathrm{E}_{q(z_{i\cdot\cdot})}\log p(z_{i\cdot\cdot}|Q_{i\cdot})\right\} \propto \prod_k Q_{ik}^{\lambda^0_{ik}-1+m_{ik}},$$

  and the update is achieved by setting the hyperparameters equal to the sum of the prior counts and the current approximate posterior expected counts:

$$\lambda^1_{ik} \leftarrow \lambda^0_{ik} + m_{ik}.$$

  [9]

- Updating the hyperparameters of $q(\mu)$
  We want to set $q(\mu)$ proportional to
  $p(\mu)\exp\left\{\mathrm{E}_{q(z)}\log p(x|\mu,z)\right\}$.
  This factorises across loci and populations as

[9] **FIXME** Clarify use of $\propto$ notation

$$
\begin{aligned}
p(\mu)\exp\left\{\mathrm{E}_{q(z)}\log p(x|\mu,z)\right\} &= \left[\prod_l\prod_k p(\mu_{lk})\right]\exp\left\{\sum_l\sum_i\sum_{a=1}^{2}\mathrm{E}_{q(z_i)}\log p(x_{ila}|\mu_{lz_i})\right\} \\
&= \prod_l\left[\prod_k p(\mu_{lk})\right]\exp\left\{\sum_i\sum_{a=1}^{2}\sum_k\gamma_k^{ila}\log p(x_{ila}|\mu_{lk})\right\} \\
&= \prod_l\prod_k p(\mu_{lk})\exp\left\{\sum_i\sum_{a=1}^{2}\gamma_k^{ila}\log p(x_{ila}|\mu_{lk})\right\},
\end{aligned}
$$

so the approximate posterior distributions on allele frequencies can be updated separately in each population and at each locus.

$$
\begin{aligned}
p(\mu_{lk})\exp\left\{\sum_i\sum_{a=1}^{2}\gamma_k^{ila}\log p(x_{ila}|\mu_{lk})\right\} &= p(\mu_{lk})\exp\left\{\sum_i\sum_{a=1}^{2}\gamma_k^{ila}\sum_j\log\mu_{lkj}^{I(x_{lia}=j)}\right\} \\
&\propto \prod_j\mu_{lkj}^{\alpha^0_{lkj}-1}\exp\left\{\sum_j\left[\log\mu_{lkj}\right]\sum_i\sum_a\gamma_k^{ila}I(x_{lia}=j)\right\} \\
&= \prod_j\mu_{lkj}^{\alpha^0_{lkj}-1+m_{lkj}},
\end{aligned}
$$

where $m_{lkj}=\sum_i\sum_a\gamma_k^{ila}I(x_{ila}=j)$ is the expected number of j alleles observed at locus l in population k, with the expectation taken w.r.t. $q(z)$. The update is therefore achieved by setting

$$\alpha^1_{lkj}\leftarrow\alpha^0_{lkj}+m_{lkj}.$$

- Monitoring convergence
  - The K-L divergence between prior and approximate posterior
    This is similar to the no-admixture case (section 8.1.1); whereas Q previously comprised a single distribution over $\{1,\ldots,K\}$, it now comprises n such distributions:

$$d_{KL}(q||p)=\sum_l\sum_k d_{KL}\Big(q(\mu_{lk\cdot})||p(\mu_{lk\cdot})\Big)+\sum_i d_{KL}\Big(q(Q_{i\cdot})||p(Q_{i\cdot})\Big),$$

    in which the component densities are all Dirichlet.
  - The average missing data probability term

$$
\begin{aligned}
\mathrm{E}_{q(Q,z)} \log p(z|Q) = & \sum_l \sum_i \sum_{a=1}^{2} \mathrm{E}_{q(z_{ila})} \mathrm{E}_{q(Q_{i\cdot})} \log Q_{iz_{ila}} \\
= & \sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^{ila} \int q(Q_{i\cdot}) \log Q_{ik} dQ_{i\cdot} \\
= & \sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^{ila} \left[\Psi(\lambda_{ik}^1) - \Psi(\sum_{k'} \lambda_{ik'}^1) \right] \\
= & \sum_i \left[\sum_l \sum_{a=1}^{2} \sum_k \gamma_k^{ila} \Psi(\lambda_{ik}^1) \right] - 2L\Psi(\sum_{k'} \lambda_{ik'}^1) \\
= & \sum_i \left[\sum_k m_{ik} \Psi(\lambda_{ik}^1) \right] - 2L\Psi(\sum_{k'} \lambda_{ik'}^1),
\end{aligned}
$$

where $m_{ik} = \sum_l \sum_{a=1}^{2} \gamma_k^{ila}$ is the expected number of allele copies in individual i that derive from population k.

- The average log likelihood term

$$
\begin{aligned}
\mathrm{E}_{q(\mu,z)} \log p(x|z,\mu) = & \sum_l \sum_i \sum_{a=1}^{2} \mathrm{E}_{q(z_{ila})} \mathrm{E}_{q(\mu_{lz_{ila}\cdot})} \log p(x_{ila}|z_{ila}, \mu_{lz_{ila}x_{ila}}), \\
= & \sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^{ila} \int q(\mu_{lk\cdot}) \log \mu_{lkx_{ila}} d\mu_{lk\cdot}. \\
= & \sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^{ila} \left[\Psi(\alpha_{lkx_{ila}}^1) - \Psi(\sum_j \alpha_{lkj}^1) \right] \\
= & \sum_l \sum_k \sum_j \left[\Psi(\alpha_{lkj}^1) - \Psi(\sum_{j'} \alpha_{lkj'}^1) \right] \sum_i \sum_{a=1}^{2} \gamma_k^{ila} I(x_{ila} = j) \\
= & \sum_l \sum_k \sum_j \left[\Psi(\alpha_{lkj}^1) - \Psi(\sum_{j'} \alpha_{lkj'}^1) \right] m_{lkj},
\end{aligned}
$$

where $m_{lkj} = \sum_i \sum_{a=1}^{2} \gamma_k^{ila} I(x_{ila} = j)$ is the expected number of alleles of type j at locus l that derive from population k.

- The entropy of the probability distribution over the missing indicators

$$
\begin{aligned}
H\left(q(z)\right) = & -\mathrm{E}_{q(z)} \log q(z) \\
= & -\sum_l \sum_i \sum_{a=1}^{2} \sum_k \gamma_k^{ila} \log \gamma_k^{ila}
\end{aligned}
$$

8.2 EM algorithm update for μ in correlated frequencies model

The complete-data posterior density (assuming a flat prior on q) is

$$\begin{aligned} p(\theta|x,z) = p(\mu,q|x,z) &\propto p(\mu)p(q)p(z|q)p(x|z,\mu) \\ &= \prod_l \left(\prod_k p(\mu_{lk})\right)\left(\prod_i p(z_{li}|q_{iz_{li}})p(x_{li}|\mu_{lz_{li}})\right), \\ &= \prod_l \left(\prod_k p(\mu_{lk})\right)\left(\prod_i q_{iz_{li}}p(x_{li}|\mu_{lz_{li}})\right), \end{aligned}$$

so the complete-data log posterior (up to an additive constant) is

$$\log p(\theta|x,z) = \sum_l \left(\sum_k \log p(\mu_{lk})\right) + \left(\sum_i \log\left(q_{iz_{li}}p(x_{li}|\mu_{lz_{li}})\right)\right),$$

the expectation of which (with respect to the current distribution on the missing data z) is

$$\begin{aligned} \mathrm{E}_{z|x,\theta^*} \log p(\theta|x,z) &= \sum_l \sum_k \log p(\mu_{lk}) + \sum_l \sum_k \sum_i \log\Big(\gamma_{ik}p(x_{li}|\mu_{lk})\Big)p_{\theta^*}(k|x_{li}) \\ &= \sum_l \sum_k \log p(\mu_{lk}) + \sum_l \sum_k \sum_i \left(\log\gamma_{ik}\right)p_{\theta^*}(k|x_{li}) \\ &\quad + \sum_l \sum_k \sum_i \Big(\log p(x_{li}|\mu_{lk})\Big)p_{\theta^*}(k|x_{li}). \end{aligned}$$

With ancestral allele frequency α_l at locus l, and a Beta$(\alpha_l F'_k, (1-\alpha_l)F'_k)$ prior on the frequency in population k ($F'_k = \frac{1-F_k}{F_k}$), and a Bernoulli likelihood, this is

$$\begin{aligned} &\sum_l \sum_k \log\left(\mu_{lk}^{\alpha F'_k - 1}(1-\mu_{lk})^{(1-\alpha_k)F'_k - 1}\right) + \sum_l \sum_k \sum_i \left(\log\gamma_{ik}\right)p_{\theta^*}(k|x_{li}) \\ &\qquad + \sum_l \sum_k \sum_i \log\Big(\mu_{lk}^{x_{li}}(1-\mu_{lk})^{(1-x_{li})}\Big)p_{\theta^*}(k|x_{li}). \end{aligned}$$

The update for μ_{lk} maximises the locus l, population k terms in the above expression. Temporarily drop l and k subscripts, and let $p_i(k) = p_{\theta^*}(k|x_{li})$. Differentiating the locus l, population k terms in the above expression with respect to μ and setting equal to zero gives

$$\begin{aligned} \frac{\alpha F' - 1}{\mu} - \frac{(1-\alpha)F' - 1}{1-\mu} + \sum_i \left(\frac{x_i}{\mu} - \frac{1-x_i}{1-\mu}\right)p_i(k) &= 0 \\ \frac{1}{\mu(1-\mu)}\left[(1-\mu)(\alpha F' - 1) - \mu\left((1-\alpha)F' - 1\right) + \sum_i \left((1-\mu)x_i - \mu(1-x_i)\right)p_i(k)\right] &= 0 \\ \alpha F' - 1 - \mu\left((1-\alpha)F' - 1 + \alpha F' - 1 + \sum_i p_i(k)\right) + \sum_i x_i p_i(k) &= 0, \end{aligned}$$

giving

$$\mu = \frac{\sum_i x_i p_i(k) + \alpha F' - 1}{\sum_i p_i(k) + F' - 2}$$

9 References

References

Corander, J., Waldmann, P. & Sillanpaa, M. J. (2003). Bayesian analysis of genetic differentiation between populations. *Genetics* **163**, 367–374.

Falush, D., Stephens, M. & Pritchard, J. (2003). Inference of population structure using multilocus genotype data: linked loci and correlated allele frequencies. *Genetics* **164**, 1567–87.

Guillot, G., Estoup, A., Mortier, F. & Cosson, J. F. (2005). A spatial statistical model for landscape genetics. *Genetics* **170**, 1261–1280.

Huelsenbeck, J. & Andolfatto, P. (2007). Inference of population structure under a Dirichlet process prior. *Genetics* .

Leslie, S. & Donnelly, P. (????). Tbc. *TBC* page TBC.

Pritchard, J., Stephens, M. & Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics* **155**, 945–59.

Table 1: Notation

Notation	Meaning
n	Number of individuals
L	Number of marker loci
J_l	Number of alleles at locus l
K	Number of groups in mixture model
i	Indexes individuals: i $\in \{1,\ldots,n\}$
l	Indexes loci: l $\in \{1,\ldots,L\}$
j	Indexes alleles: j $\in \{1,\ldots,J_l\}$
a	Indexes chromosomes: a $\in \{1,2\}$ = {maternal,paternal}
X_{ila}	Identity of allele on chromosome a at locus l in individual i: $X_{ila} \in \{1,\ldots,J_l\}$
ϕ_k	Model parameters in mixture model with k groups: ϕ_k=(μ_k, ...?)
P_{lk}	Allele frequencies at locus l in group k
$\alpha_j^{(0)}$	Hyperparameter of Dirichlet prior on allele frequencies (same for all l, k)
$\alpha_{lkj}^{(1)}$	Hyperparameter of Dirichlet posterior on allele frequencies at locus l in group k
No Admixture Model	
Z_i	Unknown group label for individual i: $Z_i \in \{1,\ldots,K\}$
Q_k	Intensity of group k in the mixture
γ_{ik}	Posterior probability that individual i was generated from group k
Admixture Model	
Z_{ila}	Unknown group label for allele on chromosome a at locus l in individual i: $Z_{ila} \in \{1,\ldots,K\}$
Q_{ik}	Unknown genome-wide proportion of ancestry of individual i from group k
$\lambda_k^{(0)}$	Hyperparameter of Dirichlet prior on ancestry proportions (same for all i)
$\lambda_{ik}^{(1)}$	Hyperparameter of Dirichlet posterior on ancestry proportions for individual i
γ_{ilak}	Posterior probability that the allele on chromosome a at locus l in individual i was generated from gro
$\sim$	Draw value from probability distribution
$\leftarrow$	Assign value
$\rightsquigarrow$	Assign value requiring subsequent normalisation of discrete probability distribution
Ψ	Digamma function