AwesomePackage: A Valid R Package for Ancestry Inference

Jonathon Chow
Department of Mathematical Sciences
University of Science and Technology of China
Anhui, 230026, P.R.China
jonathonchow23@gmail.com

October 4, 2022

Abstract

Ancestry inference is an important topic in genetics. Its main task is to estimate population structure from genetic data. The PSD model has been adopted as the standard way of ancestry inference. Meanwhile, many statistical learning methods can be used to fit the PSD model, such as Markov chain Monte Carlo (MCMC), Expectation-Maximization (EM), sequential quadratic programming (SQP), variational inference (VI) and stochastic variational inference (SVI). Here, we implement the algorithms to fit the PSD model based on EM, SQP, VI and SVI respectively. We evaluate and compare these algorithms from multiple perspectives. From the perspective of algorithm accuracy, we show that all these algorithms perform well, among which the SQP algorithm performs best. From the perspective of algorithm efficiency, we show that the performance of the SVI algorithm is far better than other algorithms on large-scale data, and the performance of the VI algorithm is slightly better than other algorithms on small-scale data. From the perspective of data structure, we show that the VI algorithm and the SVI algorithm tend to reveal only the main features, while the EM algorithm and the SQP algorithm tend to reveal the fine structure. From the perspective of population number, the optimal value of the VI algorithm tends to appear earlier, while the optimal values of the EM algorithm and the SQP algorithm tend to appear at the largest population number. We test these algorithms on simulated data, TGP data and HGDP data. Our R package, AwesomePackage, is freely available online at https://github.com/JONATHONCHOW/AwesomePackage.

***K*eywords** PSD model · EM algorithm · SQP algorithm · VI algorithm · SVI algorithm

1 Introduction

Ancestry inference, which reveals the structure of a population from genotypic data, has become an essential task in genetics. It is relevant to many important topics in genetics, such as inheritable diseases (Francioli et al. 2014), conservation genetics (Pearse and Crandall 2004; Randi 2008), the ancestry and migration patterns of natural populations (Rosenberg et al. 2002; Reich et al. 2009), etc. With decreasing costs in sequencing and genotyping technologies, enormous amounts of genetic data about people and other organisms have become available. There is a growing need for fast and accurate tools to uncover the structure of populations from vast amounts of genetic data.

Model-based (likelihood and Bayesian) and non-model-based (PCA and K-means clustering) methods were developed to identify populations and assign individuals to the identified populations using marker genotype data. Model-based methods are favoured because they are based on a probabilistic model of population genetics with biologically meaningful parameters and thus produce results that are easily interpretable and applicable. Furthermore, they often yield more accurate structure inferences than non-model-based methods.

The probabilistic model of Pritchard, Stephens and Donnelly (Pritchard, Stephens, and Donnelly 2000), known as the PSD model, has become a standard tool for those model-based methods.

Many statistical learning methods can be used to fit the PSD model. From the perspective of maximum likelihood estimation, we can use Expectation-Maximization (EM), sequential quadratic programming (SQP), and sparse non-negative matrix factorization (SNMF). From the perspective of estimating the Bayesian posterior distribution, we can use Markov chain Monte Carlo (MCMC), variational inference (VI), and stochastic variational inference (SVI). Almost all of these algorithms have been developed into software, such as STRUCTURE (MCMC) (Pritchard, Stephens, and Donnelly 2000), FRAPPE (EM) (Tang et al. 2005), ADMIXTURE (SQP) (Alexander, Novembre, and Lange 2009), sNMF (SNMF) (Frichot et al. 2014), fastSTRUCTURE (VI) (Raj, Stephens, and Pritchard 2014), and TeraStructure (SVI) (Gopalan et al. 2016). There are also some recent developments, such as PopCluster (Wang 2022). These algorithms have different advantages. For example, STRUCTURE, FRAPPE and ADMIXTURE can parse fine structures, fastSTRUCTURE can highlight salient features, and TeraStructure can analyze large-scale data.

Meanwhile, the collection of human genetic data is proceeding apace. The two most typical projects are the 1000 Genomes Project (Abecasis et al. 2012) and the Human Genome Diversity Project (Cann et al. 2002; Cavalli-Sforza 2005), Known as TGP and HGDP.

In *Models and Methods*, we briefly describe the PSD model and the theoretical basis of the algorithms. We also briefly illustrate the relationship between the PSD model and some other models. We then describe the implementation details of the algorithms and the schemes to accelerate computation. Finally, we introduce some criteria for algorithm evaluation to help evaluate the accuracy of the results, choose population number, and compare the performance of different algorithms. In *Applications*, we compare the accuracy and time complexity of different algorithms on simulated genotype data sets. Then we demonstrate the results, especially the selection of population number, on TGP data set and HGDP data set.

2 Models and Methods

We now briefly describe the PSD model and the various algorithms for fitting the PSD model.

2.1 PSD model

Suppose we have I diploid individuals genotyped at J biallelic loci. Let (g_{ij}^1, g_{ij}^2) represents the genotype at marker j of individual i, where g_{ij}^a represent the observed number of copies of allele 1 at seat a. Thus, (g_{ij}^1, g_{ij}^2) equals $(1,1)$, $(1,0)$, $(0,1)$, or $(0,0)$ accordingly, as i has genotype 1/1, 1/2, 2/1, or 2/2 at marker j. Let $g_{ij} = g_{ij}^1 + g_{ij}^2$. These individuals are drawn from an admixed population with contributions from K postulated ancestral populations. Population k contributes a fraction p_{ik} of individual i's genome. Note that $\sum_{k=1}^K p_{ik} = 1$, and $p_{ik} \geq 0$. Allele 1 at SNP j has frequency f_{kj} in population k. Note that $0 \leq f_{kj} \leq 1$. Note that individuals are formed by the random union of gametes. This produces the binomial distribution $g_{ij}^a \sim Binomial(1, p_{ik} f_{kj})$. We consider (z_{ij}^1, z_{ij}^2), where z_{ij}^a is an element of the set $\{1, \ldots, K\}$, denotes the population from which the genes of individual i of marker j at position a really come. Let $z_{ijk}^a = \mathbf{1}(z_{ij}^a = k)$, obviously, $z_{ijk}^a \in \{0,1\}$, and $\sum_{k=1}^K z_{ijk}^a = 1$.

In conclusion, the observed variable is the genotype matrix G, the latent variable is the matrix Z of the true origin of genes, the parameters are the population scale matrix P and the gene scale matrix F, and the hyperparameter is the population number K. The goal of the EM algorithm and the SQP algorithm is to solve the optimization problem of maximizing log-likelihood function $\mathcal{L}(G|P,F)$ under constraints

$$\begin{aligned} \underset{P,F}{max} \quad & \mathcal{L}(G|P^{(t)}, F^{(t)}) \\ s.t. \quad & \sum_{k=1}^K p_{ik} = 1, \quad i = 1, \ldots, I \\ & 0 \leq p_{ik} \leq 1, \quad i = 1, \ldots, I, \quad k = 1, \ldots, K \\ & 0 \leq f_{kj} \leq 1, \quad j = 1, \ldots, J, \quad k = 1, \ldots, K. \end{aligned}$$

The goal of the VI algorithm and the SVI algorithm is to find the variational family $Q(Z,P,F)$ that is closest to the posterior $P(Z,P,F|G)$.

2.2 EM algorithm

EM algorithm provides a way to solve MLE iteratively (Bishop and Nasrabadi 2006). At *E-step*, we compute the expectation

$$a_{ijk} = \frac{p_{ik}^{(t)} f_{kj}^{(t)}}{\sum_{k=1}^{K} p_{ik}^{(t)} f_{kj}^{(t)}}, \quad b_{ijk} = \frac{p_{ik}^{(t)} (1 - f_{kj}^{(t)})}{\sum_{k=1}^{K} p_{ik}^{(t)} (1 - f_{kj}^{(t)})}.$$

At *M-step*, we compute the maximization and update the parameters

$$p_{ik} = \frac{\sum_{j=1}^{J} g_{ij} a_{ijk}^{(t)} + \sum_{j=1}^{J} (2 - g_{ij}) b_{ijk}^{(t)}}{2J}, \quad f_{kj} = \frac{\sum_{i=1}^{I} g_{ij} a_{ijk}^{(t)}}{\sum_{i=1}^{I} g_{ij} a_{ijk}^{(t)} + \sum_{i=1}^{I} (2 - g_{ij}) b_{ijk}^{(t)}}.$$

A simple convergence criterion is that the change in the log-likelihood function

$$\mathcal{L}(G|P, F) = \sum_{i=1}^{I} \sum_{j=1}^{J} \left\{ g_{ij} log\Big[\sum_{k=1}^{K} p_{ik} f_{kj} \Big] + (2 - g_{ij}) log\Big[\sum_{k=1}^{K} p_{ik} (1 - f_{kj}) \Big] \right\}$$

is small enough.

2.3 SQP algorithm

The optimization problem describe in Section 2.1 is convex in P for F fixed and in F for P fixed. Convexity makes block iteration amenable to convex optimization techniques (Boyd and Vandenberghe 2004). We update P and F block by block alternately. When we update P, we calculate the first and second partial derivatives (Hessian matrix) of P under the condition that F is fixed

$$\frac{\partial \mathcal{L}}{\partial p_{ik}} = \sum_{j=1}^{J} \left[\frac{g_{ij} f_{kj}}{\sum_{k=1}^{K} p_{ik} f_{kj}} + \frac{(2 - g_{ij})(1 - f_{kj})}{\sum_{k=1}^{K} p_{ik} (1 - f_{kj})} \right],$$

$$\frac{\partial^2 \mathcal{L}}{\partial p_{ik} \partial p_{il}} = -\sum_{j=1}^{J} \left[\frac{g_{ij} f_{kj} f_{lj}}{(\sum_{k=1}^{K} p_{ik} f_{kj})^2} + \frac{(2 - g_{ij})(1 - f_{kj})(1 - f_{lj})}{(\sum_{k=1}^{K} p_{ik} (1 - f_{kj}))^2} \right].$$

We then solve the quadratic programming problem

$$\begin{aligned} \underset{\Delta P_i}{min} \quad & \frac{1}{2} (\Delta P_i)^T \Big[-\nabla_{P_i}^2 \mathcal{L}(G|P^{(t)}, F^{(t)}) \Big] \Delta P_i - \Big[\nabla_{P_i} \mathcal{L}(G|P^{(t)}, F^{(t)}) \Big]^T \Delta P_i - \mathcal{L}(G|P^{(t)}, F^{(t)}) \\ s.t. \quad & 1^T \Delta P_i = 0 \\ & 1 - P_i^{(t)} \geq \Delta P_i \geq -P_i^{(t)} \end{aligned}$$

When we update F, we calculate the first and second partial derivatives (Hessian matrix) of F under the condition that P is fixed

$$\frac{\partial \mathcal{L}}{\partial f_{kj}} = \sum_{i=1}^{I} \left[\frac{g_{ij} p_{ik}}{\sum_{k=1}^{K} p_{ik} f_{kj}} - \frac{(2 - g_{ij}) p_{ik}}{\sum_{k=1}^{K} p_{ik} (1 - f_{kj})} \right],$$

$$\frac{\partial^2 \mathcal{L}}{\partial f_{kj} \partial f_{lj}} = -\sum_{i=1}^{I} \left[\frac{g_{ij} p_{ik} p_{il}}{(\sum_{k=1}^{K} p_{ik} f_{kj})^2} + \frac{(2 - g_{ij}) p_{ik} p_{il}}{(\sum_{k=1}^{K} p_{ik} (1 - f_{kj}))^2} \right].$$

We then solve the quadratic programming problem

$$\begin{aligned} \underset{\Delta F_j}{min} \quad & \frac{1}{2} (\Delta F_j)^T \Big[-\nabla_{F_j}^2 \mathcal{L}(G|P^{(t)}, F^{(t)}) \Big] \Delta F_j - \Big[\nabla_{F_j} \mathcal{L}(G|P^{(t)}, F^{(t)}) \Big]^T \Delta F_j - \mathcal{L}(G|P^{(t)}, F^{(t)}) \\ s.t. \quad & 1 - F_j^{(t)} \geq \Delta F_j \geq -F_j^{(t)} \end{aligned}$$

We choose the same convergence criterion as the EM algorithm.

2.4 VI algorithm

In Bayesian statistics, we often can not directly find the true posterior, then we can use a family of densities over the latent variables, parameterized by free *variational parameters* to approximate the true posterior (Blei, Kucukelbir, and McAuliffe 2017). The choice of the variational family is restricted only by the tractability of computing expectations with respect to the variational distributions; here, we choose parametric distributions that are conjugate to the distributions in the likelihood function based on the assumptions of *mean field approximation* and independence as

$$Q(Z,P,F)=\prod_{i=1}^{I}\prod_{j=1}^{J}\prod_{a=1}^{2}Q(z_{ij}^{a})\cdot\prod_{i=1}^{I}Q(p_i)\cdot\prod_{j=1}^{J}\prod_{k=1}^{K}Q(f_{kj}),$$

where each factor can then be written as

$$Q(z_{ij}^{a})=Multinomial(\tilde{z}_{ij}^{a}),$$
$$Q(p_i)=Dirichlet(\tilde{p}_i),$$
$$Q(f_{kj})=Beta(\tilde{f}_{kj}^{1},\tilde{f}_{kj}^{2}).$$

$\tilde{z}_{ij}^{a}$, $\tilde{p}_i$, $\tilde{f}_{kj}^{1}$, $\tilde{f}_{kj}^{2}$ are the parameters of the variational distributions (variational parameters).

We then calculate the variational parameterized ELBO

$$\begin{aligned}ELBO=&\sum_{i=1}^{I}\sum_{j=1}^{J}\Bigg\{\sum_{k=1}^{K}\Big(\mathbb{E}[z_{ijk}^{1}]+\mathbb{E}[z_{ijk}^{2}]\Big)\Big(\mathbf{1}(g_{ij}=0)\mathbb{E}[log(1-f_{kj})]+\mathbf{1}(g_{ij}=2)\mathbb{E}[logf_{kj}]+\mathbb{E}[logp_{ik}]\Big)\\&+\mathbf{1}(g_{ij}=1)\sum_{k=1}^{K}\Big(\mathbb{E}[z_{ijk}^{1}]\mathbb{E}[logf_{kj}]+\mathbb{E}[z_{ijk}^{2}]\mathbb{E}[log(1-f_{kj})]\Big)-\mathbb{E}[logz_{ij}^{1}]-\mathbb{E}[logz_{ij}^{2}]\Bigg\}\\&+\sum_{j=1}^{J}\sum_{k=1}^{K}log\frac{B(\tilde{f}_{kj}^{1},\tilde{f}_{kj}^{2})}{B(\beta^{1},\beta^{2})}+(\beta^{1}-\tilde{f}_{kj}^{1})\mathbb{E}[logf_{kj}]+(\beta^{2}-\tilde{f}_{kj}^{2})\mathbb{E}[log(1-f_{kj})]\\&+\sum_{i=1}^{I}\Bigg\{\sum_{k=1}^{K}(\alpha_k-\tilde{p}_{ik})\mathbb{E}[logp_{ik}]+log\Gamma(\alpha_k)-log\Gamma(\tilde{p}_{ik})\Bigg\}+log\Gamma(\sum_{k=1}^{K}\tilde{p}_{ik})-log\Gamma(\sum_{k=1}^{K}\alpha_k),\end{aligned}$$

where α_k, β^1 and β^2 are the parameters of the prior distribution. We choose the simple priors as $P(p_i)=Dirichlet(\frac{1}{K}\mathbf{1}_K)$, $P(f_{kj})=Beta(1,1)$. We take the partial derivative of ELBO and obtain the parameter update formula

$$\tilde{z}_{ijk}^{1}\propto exp\Big\{\mathbf{1}(g_{ij}=0)\psi(\tilde{f}_{kj}^{2})+\mathbf{1}(g_{ij}=1)\psi(\tilde{f}_{kj}^{1})+\mathbf{1}(g_{ij}=2)\psi(\tilde{f}_{kj}^{1})-\psi(\tilde{f}_{kj}^{1}+\tilde{f}_{kj}^{2})+\psi(\tilde{p}_{ik})-\psi(\sum_{k=1}^{K}\tilde{p}_{ik})\Big\},$$

$$\tilde{z}_{ijk}^{2}\propto exp\Big\{\mathbf{1}(g_{ij}=0)\psi(\tilde{f}_{kj}^{2})+\mathbf{1}(g_{ij}=1)\psi(\tilde{f}_{kj}^{2})+\mathbf{1}(g_{ij}=2)\psi(\tilde{f}_{kj}^{1})-\psi(\tilde{f}_{kj}^{1}+\tilde{f}_{kj}^{2})+\psi(\tilde{p}_{ik})-\psi(\sum_{k=1}^{K}\tilde{p}_{ik})\Big\},$$

$$\tilde{p}_{ik}=\alpha_k+\sum_{j=1}^{J}(\tilde{z}_{ijk}^{1}+\tilde{z}_{ijk}^{2}),$$

$$\tilde{f}_{kj}^{1}=\beta^{1}+\sum_{i=1}^{I}\Big[\mathbf{1}(g_{ij}=1)\tilde{z}_{ijk}^{1}+\mathbf{1}(g_{ij}=2)(\tilde{z}_{ijk}^{1}+\tilde{z}_{ijk}^{2})\Big],$$

$$\tilde{f}_{kj}^{2}=\beta^{2}+\sum_{i=1}^{I}\Big[\mathbf{1}(g_{ij}=1)\tilde{z}_{ijk}^{2}+\mathbf{1}(g_{ij}=0)(\tilde{z}_{ijk}^{1}+\tilde{z}_{ijk}^{2})\Big].$$

The convergence criterion is that the change in ELBO is small enough. Using the Dirichlet distribution and the beta distribution expectations, we have $\mathbb{E}[p_{ik}]=\frac{\tilde{p}_{ik}}{\sum_{k=1}^{K}\tilde{p}_{ik}}$, $\mathbb{E}[f_{kj}]=\frac{\tilde{f}_{kj}^{1}}{\sum_{k=1}^{K}\tilde{f}_{kj}^{1}+\tilde{f}_{kj}^{2}}$.

2.5 SVI algorithm

The traditional variational inference using the coordinate ascent method needs to traverse all the data in the process of each iteration, resulting in high computational cost for large data sets. SVI (Hoffman et al. 2013) is an ideal alternative. Our goal is to update the global variable P iteratively. In each iteration, we first sample a SNP location j and all observations g_{ij} at that location. Then, in the sampled data, we update F with fixed P in the same way as VI until convergence. Next, we update the global variable

$$\tilde{p}_{ik}^{(t+1)} = (1-\rho_t)\tilde{p}_{ik}^{(t)} + \rho_t\big[\alpha_k + J(\tilde{z}_{ijk}^1 + \tilde{z}_{ijk}^2)\big],$$

where step size $\rho_t = (\tau_0 + t)^{-\kappa}$. We set τ_0 to 1 and κ to 0.5. We take a validation set that does not participate in the training, and then compute the log-likelihood function on the validation set until the change is small enough.

2.6 Relationships with other models

The PSD model is closely related to the multinomial topic model. More precisely, The PSD model (log-likelihood) is similar to the probabilistic latent semantic analysis (PLSA) model (Hofmann 2001), and the PSD model (Bayesian posterior) is similar to the latent dirichlet allocation (LDA) model (Blei, Ng, and Jordan 2003). They are very similar in the representation of the model, the derivation of the algorithm and so on. Even using the algorithm of the multinomial topic model to fit the diploid genotype data can get good results. But this is practical, not strictly mathematically equivalent. In fact, the multinomial topic model have been shown to be equivalent to the Poisson NMF model (Carbonetto et al. 2021) and are widely used to analyze the population structure of single-cell genes such as RNA.

3 Applications

We fit the PSD model on simulated data sets, TGP data set and HGDP data set.

3.1 Simulated data sets

To evaluate the performance of the different learning algorithms, we generated two groups of simulated genotype data sets.

3.1.1 Simulated Data Set A

We generated the simulated data set A (Raj, Stephens, and Pritchard 2014) in three steps. First, generate the population scale matrix P using the Dirichlet distribution; In the second step, the gene scale matrix F is generated using beta distribution. The third step is to generate the genotype matrix G using the binomial distribution. We set the number of individuals I to 600, the number of SNPs J to 2500, and the number of populations K to 3.

Step 1. The population scales for each sample are drawn from a symmetric Dirichlet distribution to simulate small amounts of gene flow between the three populations. Here we use $Dirichlet(\frac{1}{10}\mathbf{1}_3)$. Step 2. The ancestral allele frequencies $\bar{f}_j$ for each SNP are drawn from a natural data set to simulate allele frequencies in natural populations. Here we use the HGDP data set. First, $\bar{f}_j$ is equal to the total number of suballeles observed at the jth SNP divided by twice the number of individuals. Then, we assume that the samples are drawn from a three-population demographic model. The edge weights correspond to the parameter F_k (Wright 1949) in the model that quantifies the genetic drift of each of the three current populations from an ancestral population. Here we choose $(F_1, F_2, F_3) = (0.1, 0.05, 0.01)$ to simulate strong structure and $(F_1, F_2, F_3) = 0.5 \times (0.1, 0.05, 0.01)$ to simulate weak structure. Thus, the allele frequency at a given locus for each population is drawn from a beta distribution (Balding and Nichols 1995) $f_{kj} \sim Beta\Big(\frac{1-F_k}{F_k}\bar{f}_j, \frac{1-F_k}{F_k}(1-\bar{f}_j)\Big)$. Step 3. According to the PSD model, each element g_{ij} of the matrix G follows a binomial distribution with probability $(PF)_{ij} = \sum_{i=1}^{K} p_{ik} f_{kj}$ and number of trials 2.

3.1.2 Simulated Data Set B

We also use three steps to generate simulated data set B. In the second step, we set all F_k to 0.1. The third step is the same as for data set A. We just consider the first step. We set a Gaussian density for each

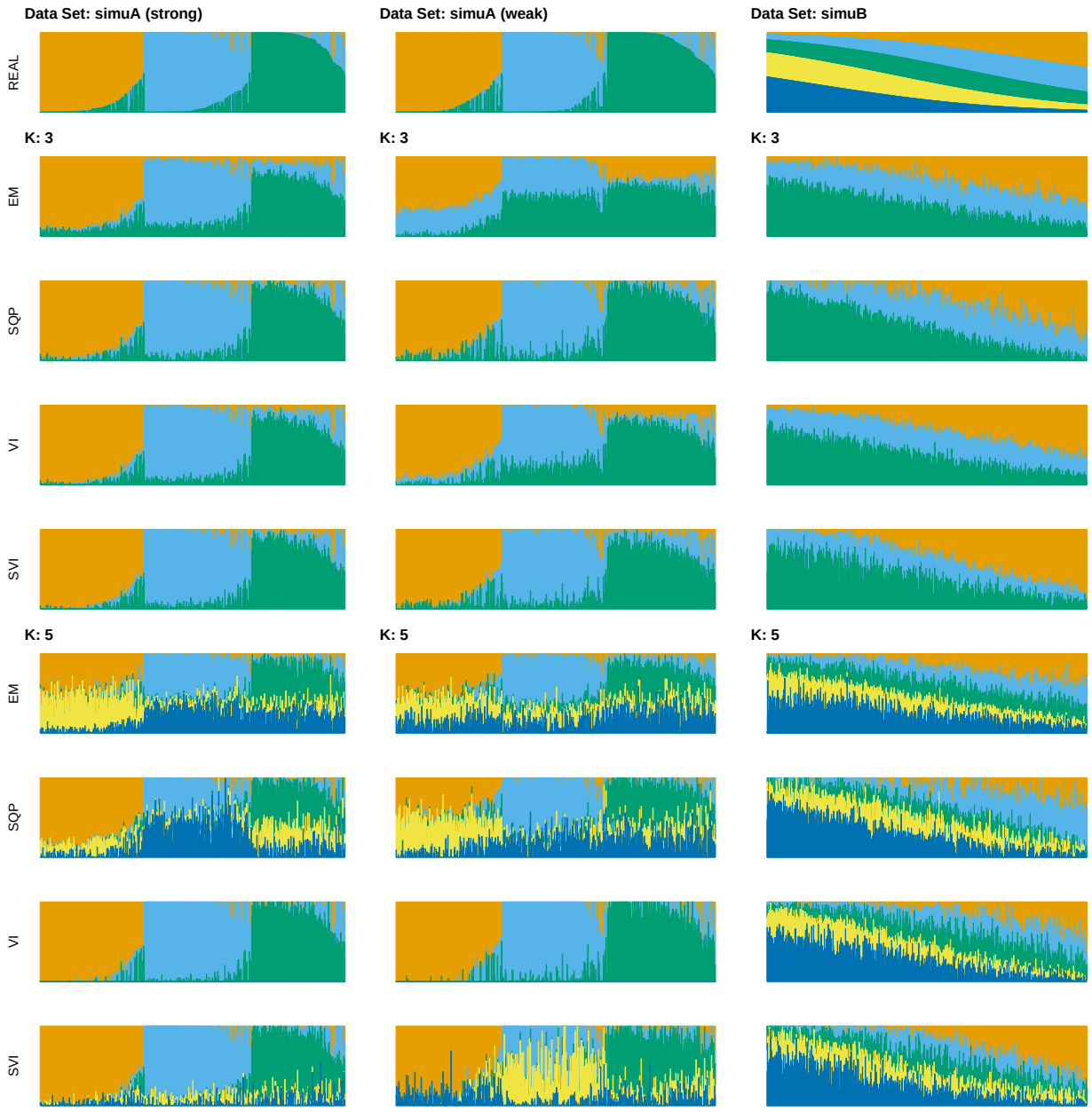


Figure 1: The structure plot of simulated data sets. The first column is simulated data set A with strong structure. The second column is simulated data set A with weak structure. The third column is simulated data set B. Each row represents the structure diagram obtained by different K and different methods.

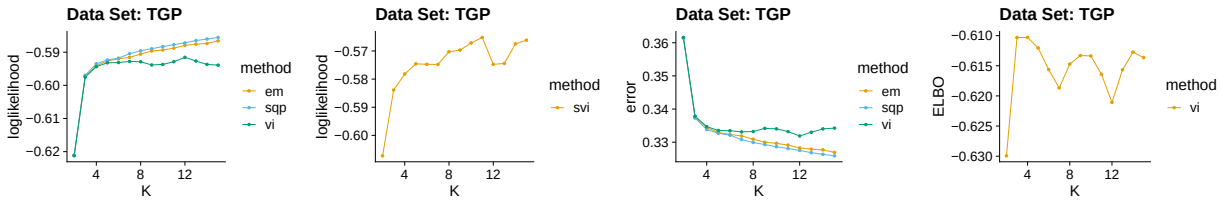


Figure 2: The evaluation indicators of TGP data set. The figures are the log-likelihood curves of EM, SQP, VI, the log-likelihood curve of SVI on the validation set, the error curves of EM, SQP, VI, and the ELBO curve of VI.

ancestral population centered at its location and normalizing each individual such that all proportions sum to 1 (Gopalan et al. 2016). In this case, each ancestral population is placed at a location evenly spaced along a line. Individuals are also positioned evenly on the line, and their proportions p_{ik} are a function of their proximity to each population's location. We set the number of individuals I to 1000, the number of SNPs J to 5000, and the number of populations K to 5.

3.1.3 Results

The main purpose of simulated data set A is to study the influence of the strength of population structure and the choice of parameter K on the performance of different algorithms. See Column 1 and 2 of Figure 1. For EM and SQP algorithms, they tend to reveal details, that is, they are sensitive to parameter K and structure strength. With the appropriate parameter K, this may be an advantage, as it reveals a finer structure. However, when the parameter K is too large, the phenomenon of overfitting is easy to occur. In addition, SQP algorithm is more accurate than EM algorithm. For the VI algorithm, we notice that the results of VI are almost consistent for both parameter K and structure strength changes. This means that the VI algorithm only tends to reveal the main factors, thereby ignoring some smaller contributions. This is both a strength and a weakness. The SVI algorithm can both highlight the main parts like VI, and react acutely when the structure is not obvious like EM and SQP.

The main purpose of simulated data set B is to study the performance of different algorithms when the mixing ratio gap between different individuals is small. See Column 3 of Figure 1. In this case, EM algorithm and SQP algorithm can more faithfully reflect the structure of the data set, while VI algorithm and SVI algorithm will overemphasize some features.

3.2 TGP data set

We use the TGP data of the first phase (Abecasis et al. 2012). We first make a mapping of individuals and populations. Then we do data preprocessing, that is, we convert the data into a matrix with elements 0, 1, 2, and deal with missing values. The final data set contained 1092 samples with genotypes at 470,349 loci. We fit the data in different methods with different parameters K.

We notice that when K is large, the index does not change significantly. This suggests the following criteria for choosing K: select the lowest model complexity when there is no obvious gap in indicators. See Figure 2. The log-likelihood curves of EM and SQP slow down from K equals 4. The log-likelihood curve of VI flattens out from about K equals 4, and shows that the optimal K is 12. The log-likelihood curve of SVI shows that the optimal K is 11, and 8, 9, 10, and 11 are all good choices for K. The error curves of EM, SQP and VI are almost identical with the log-likelihood curves of EM, SQP and VI. The ELBO curve of VI shows the curve oscillating from K equals 3.

In conclusion, we note that when K is around 4, the fit is already doing very well. The optimal K should be reached around 11, but from the structure diagram, the populations appear redundant at this time. For the **best** K (equals 4 and 11), we draw the structure plot. See Figure 3.

3.3 HGDP data set

We use the Harvard HGDP-CEPH data (Lu et al. 2011). We do the same with the HGDP data set as we do with the TGP data set, and finally we get the data set contained 942 samples with genotypes at 451,689 loci. We fit the data in different methods with different parameters K.

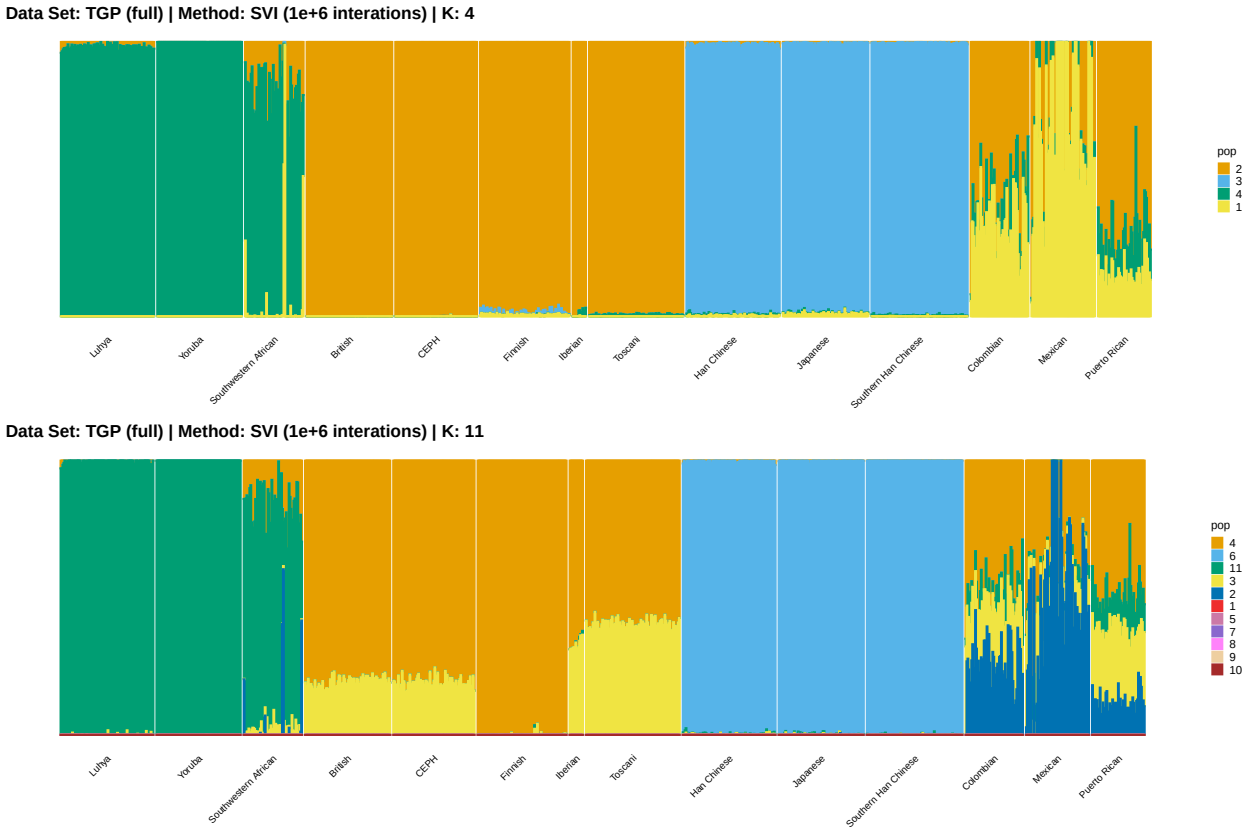


Figure 3: The structure plot of TGP data set with the beat K.

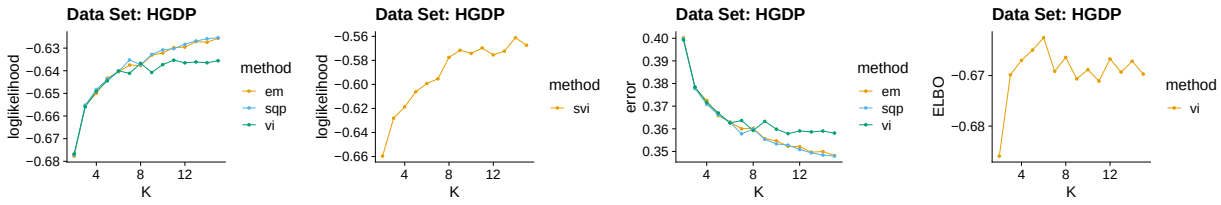


Figure 4: The evaluation indicators of HGDP data set. The figures are the log-likelihood curves of EM, SQP, VI, the log-likelihood curve of SVI on the validation set, the error curves of EM, SQP, VI, and the ELBO curve of VI.

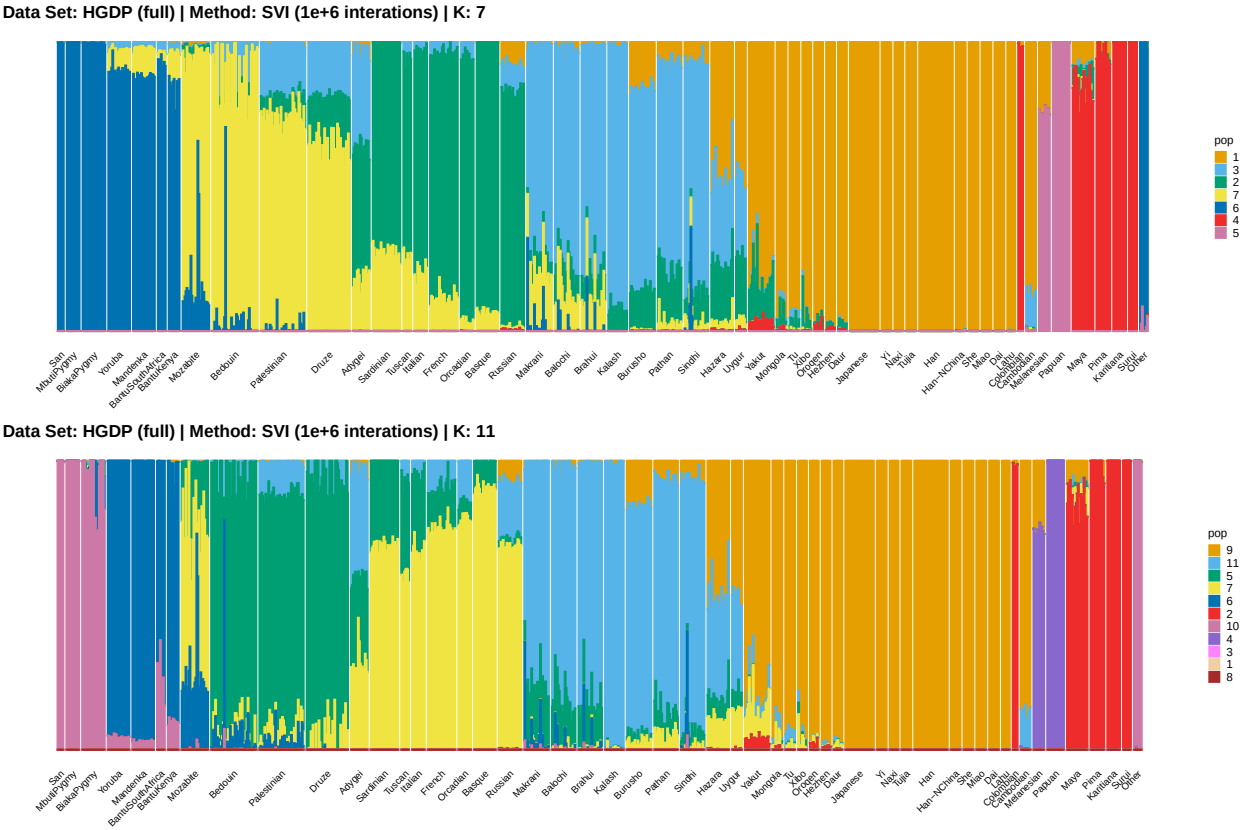


Figure 5: The structure plot of HGDP data set with the beat K.

As with the TGP data set, we choose the best K. See Figure 4. The log-likelihood curves of EM and SQP slow down from `K` equals 7. The log-likelihood curve of VI flattens out from about `K` equals 6, and shows that the optimal `K` is 8 and 11. The log-likelihood curve of SVI shows that the optimal `K` is 11 and 14, and 8, 9, 10, and 11 are all good choices for `K`. The error curves of EM, SQP and VI are almost identical with the log-likelihood curves of EM, SQP and VI. The ELBO curve of VI shows the curve oscillating from `K` equals 7.

In conclusion, we note that when `K` is around 7, the fit is already doing very well. The optimal `K` should be reached around 11, but from the structure diagram, the populations appear redundant at this time. For the **best** K (equals 7 and 11), we draw the structure plot. See Figure 5.

4 Discussion

We evaluate the algorithm from two perspectives: accuracy and efficiency.

For suitable K, the SVI algorithm and SQP algorithm perform best in terms of convergence accuracy, followed by VI algorithm and finally EM algorithm. For the unknown K, due to the lack of prior constraints, the EM algorithm and SQP algorithm are prone to overfitting when the population number is redundant. Therefore, we had better use VI algorithm and SVI algorithm to select the appropriate K.

In addition to measuring the accuracy of convergence, we still need to consider the efficiency of convergence. We have two indicators to measure the convergence efficiency, which are convergence speed (the number of iterations required to achieve convergence) and convergence time (the time required for a single iteration). We can see the convergence time plots in Figure 6, and we can see the convergence speed plots in Appendix. EM algorithm is poor in terms of convergence time and convergence speed, and the convergence time increases rapidly with the increase of K. Although SQP algorithm has a good performance in terms of convergence speed, the convergence time of the unaccelerated SQP algorithm is extremely slow, which increases rapidly with the increase of K. VI algorithm has similar convergence speed with EM algorithm (both of them have poor performance), but in terms of convergence time, VI algorithm has excellent performance, especially

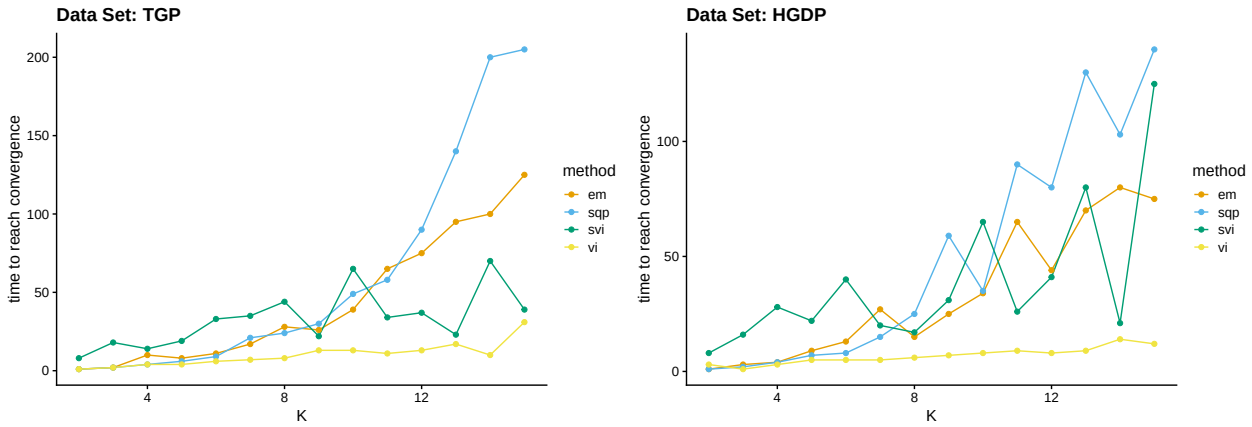


Figure 6: Convergence time. Since we only recorded integer values, we made a modest estimate of the value in integer hours.

with the increase of K, the required time increases slowly. Due to different principles, we only consider the convergence time for the SVI algorithm. Although the performance of convergence time of SVI algorithm is poor on small data sets, the time of SVI algorithm is almost only related to the length of single sampling (the number of individuals), that is to say, for complete data sets, the convergence time of SVI is almost unchanged. This means that SVI has irreplaceable advantages for large data sets. Meanwhile, similar to VI algorithm, the change of convergence time of SVI algorithm is relatively insensitive to K. By the way, compared with other algorithms, the convergence time of SVI algorithm is irregular due to the randomness of sampling.

In conclusion, we should consider both algorithm accuracy and algorithm efficiency. For small data sets, we can get good results by using VI directly. Or we can first use VI algorithm to reach the vicinity of the optimal value, and then use SQP algorithm to improve the convergence accuracy. The reason why the SQP algorithm is not directly used here is that the unaccelerated SQP algorithm is inefficient and the SQP algorithm is extremely easy to converge to local minima. For large data sets, we use the SVI algorithm without question. Of course, if K is unknown, we should pick K first, in the same way as above.

Acknowledgments

We thank 1000 Genome Project team and Human Genome Diversity Project team.

Literature Cited

Abecasis, Goncalo R, Adam Auton, Lisa D Brooks, Mark A DePristo, Richard M Durbin, Robert E Handsaker, Hyun Min Kang, et al. 2012. "An Integrated Map of Genetic Variation from 1,092 Human Genomes." *Nature* 491 (7422): 56–65.

Alexander, David H, John Novembre, and Kenneth Lange. 2009. "Fast Model-Based Estimation of Ancestry in Unrelated Individuals." *Genome Research* 19 (9): 1655–64.

Balding, David J, and Richard A Nichols. 1995. "A Method for Quantifying Differentiation Between Populations at Multi-Allelic Loci and Its Implications for Investigating Identity and Paternity." *Genetica* 96 (1): 3–12.

Bishop, Christopher M, and Nasser M Nasrabadi. 2006. *Pattern Recognition and Machine Learning*. Vol. 4. 4. Springer.

Blei, David M, Alp Kucukelbir, and Jon D McAuliffe. 2017. "Variational Inference: A Review for Statisticians." *Journal of the American Statistical Association* 112 (518): 859–77.

Blei, David M, Andrew Y Ng, and Michael I Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning Research* 3 (Jan): 993–1022.

Boyd, Stephen, and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

Cann, Howard M, Claudia De Toma, Lucien Cazes, Marie-Fernande Legrand, Valerie Morel, Laurence Piouffre, Julia Bodmer, et al. 2002. "A Human Genome Diversity Cell Line Panel." *Science* 296 (5566): 261–62.

Carbonetto, Peter, Abhishek Sarkar, Zihao Wang, and Matthew Stephens. 2021. "Non-Negative Matrix Factorization Algorithms Greatly Improve Topic Model Fits." *arXiv Preprint arXiv:2105.13440*.

Cavalli-Sforza, L Luca. 2005. "The Human Genome Diversity Project: Past, Present and Future." *Nature Reviews Genetics* 6 (4): 333–40.

Francioli, Laurent C, Andronild Menelaou, Sara L Pulit, Freerk Van Dijk, Pier Francesco Palamara, Clara C Elbers, Pieter BT Neerincx, et al. 2014. "Whole-Genome Sequence Variation, Population Structure and Demographic History of the Dutch Population." *Nature Genetics* 46 (8): 818–25.

Frichot, Eric, François Mathieu, Théo Trouillon, Guillaume Bouchard, and Olivier François. 2014. "Fast and Efficient Estimation of Individual Ancestry Coefficients." *Genetics* 196 (4): 973–83.

Gopalan, Prem, Wei Hao, David M Blei, and John D Storey. 2016. "Scaling Probabilistic Models of Genetic Variation to Millions of Humans." *Nature Genetics* 48 (12): 1587–90.

Hoffman, Matthew D, David M Blei, Chong Wang, and John Paisley. 2013. "Stochastic Variational Inference." *Journal of Machine Learning Research* 14: 1303–47.

Hofmann, Thomas. 2001. "Unsupervised Learning by Probabilistic Latent Semantic Analysis." *Machine Learning* 42 (1): 177–96.

Lu, Yontao, Nick Patterson, Yiping Zhan, Swapan Mallick, and David Reich. 2011. "Technical Design Document for a SNP Array That Is Optimized for Population Genetics."

Pearse, Devon E, and Keith A Crandall. 2004. "Beyond FST: Analysis of Population Genetic Data for Conservation." *Conservation Genetics* 5 (5): 585–602.

Pritchard, Jonathan K, Matthew Stephens, and Peter Donnelly. 2000. "Inference of Population Structure Using Multilocus Genotype Data." *Genetics* 155 (2): 945–59.

Raj, Anil, Matthew Stephens, and Jonathan K Pritchard. 2014. "fastSTRUCTURE: Variational Inference of Population Structure in Large SNP Data Sets." *Genetics* 197 (2): 573–89.

Randi, Ettore. 2008. "Detecting Hybridization Between Wild Species and Their Domesticated Relatives." *Molecular Ecology* 17 (1): 285–93.

Reich, David, Kumarasamy Thangaraj, Nick Patterson, Alkes L Price, and Lalji Singh. 2009. "Reconstructing Indian Population History." *Nature* 461 (7263): 489–94.

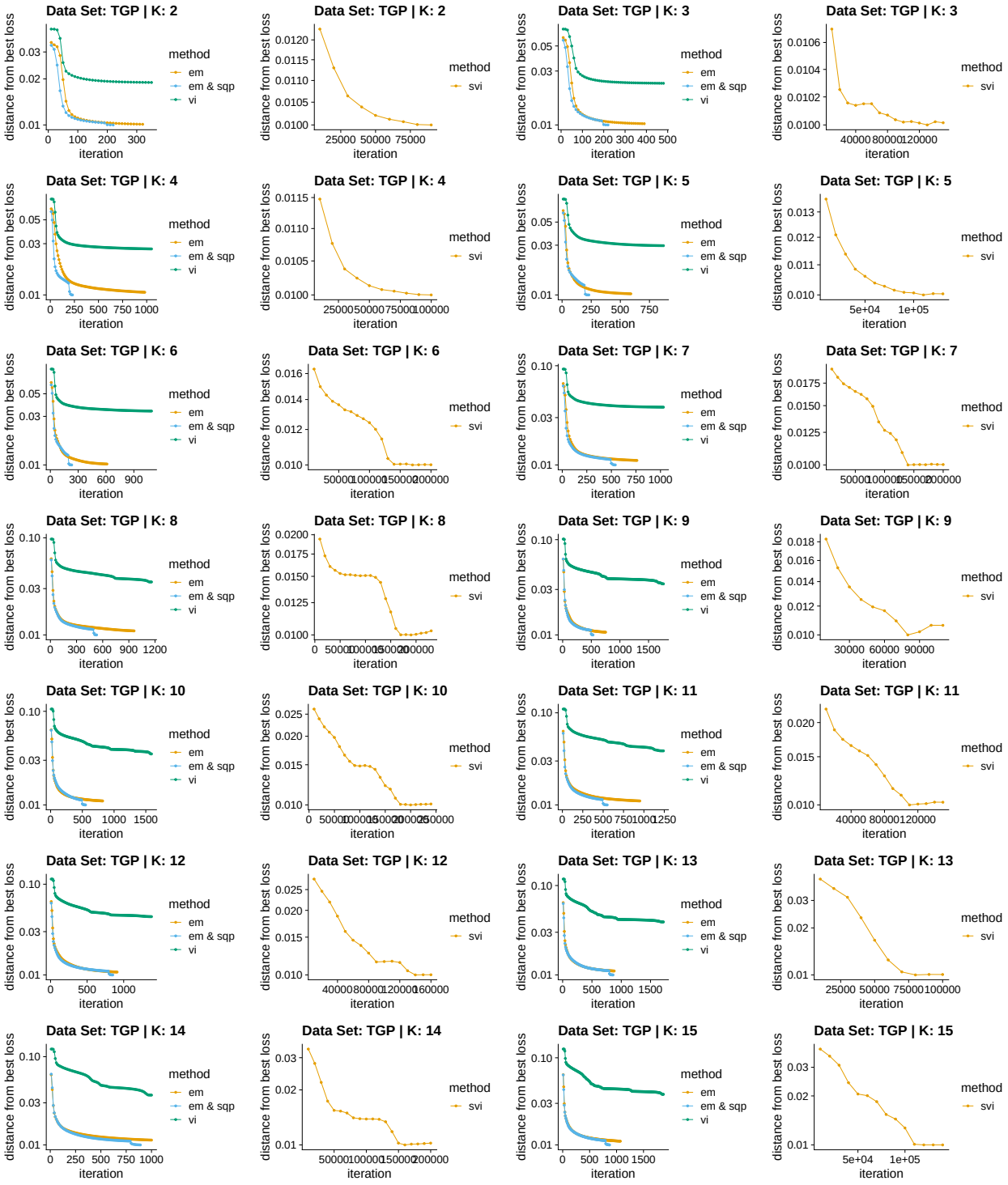
Rosenberg, Noah A, Jonathan K Pritchard, James L Weber, Howard M Cann, Kenneth K Kidd, Lev A Zhivotovsky, and Marcus W Feldman. 2002. "Genetic Structure of Human Populations." *Science* 298 (5602): 2381–85.

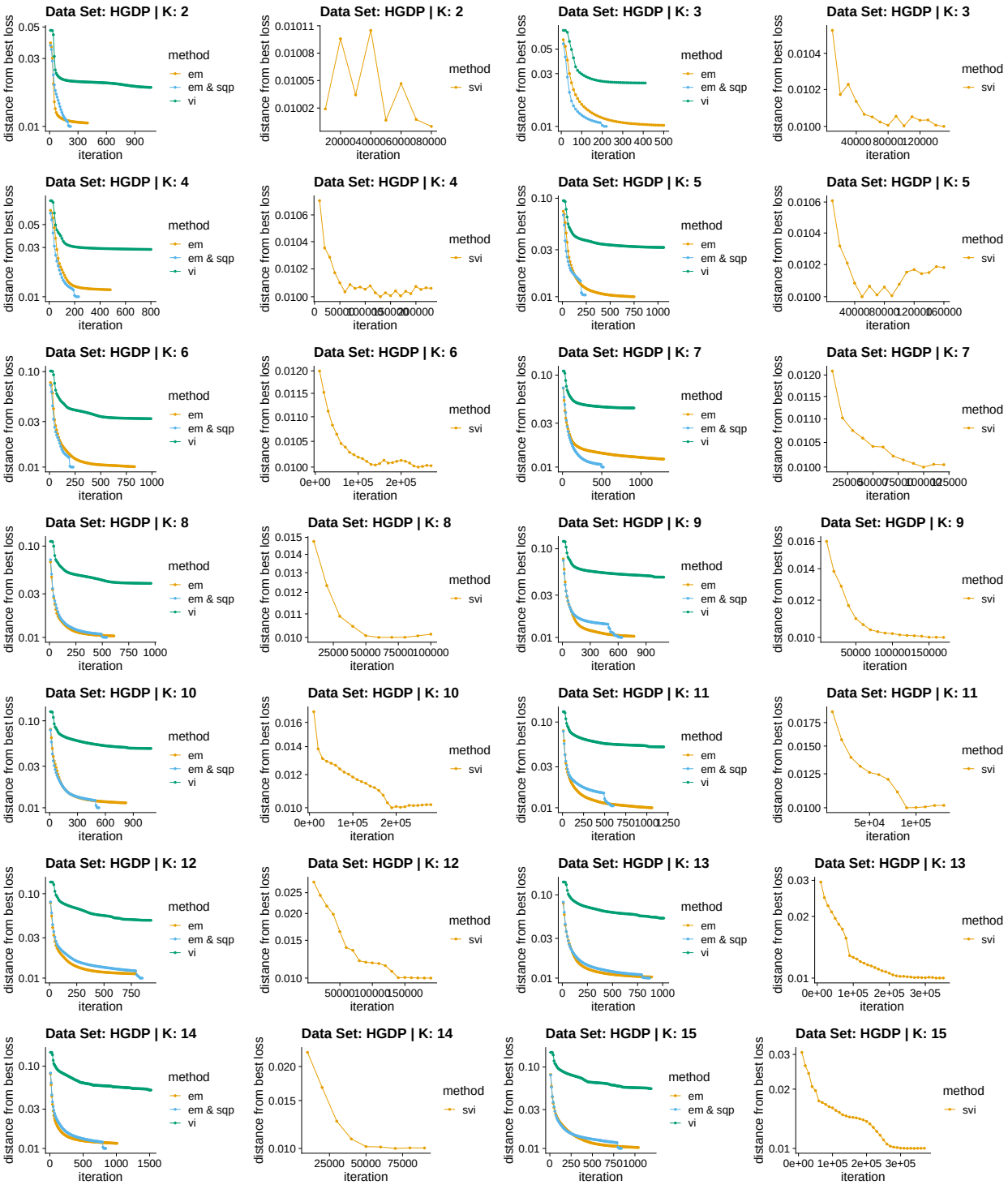
Tang, Hua, Jie Peng, Pei Wang, and Neil J Risch. 2005. "Estimation of Individual Admixture: Analytical and Study Design Considerations." *Genetic Epidemiology: The Official Publication of the International Genetic Epidemiology Society* 28 (4): 289–301.

Wang, Jinliang. 2022. "Fast and Accurate Population Admixture Inference from Genotype Data from a Few Microsatellites to Millions of SNPs." *Heredity*, 1–14.

Wright, Sewall. 1949. "The Genetical Structure of Populations." *Annals of Eugenics* 15 (1): 323–54.

Appendix


Data Set: TGP | K: 2
Data Set: TGP | K: 2
Data Set: TGP | K: 3
Data Set: TGP | K: 3
Data Set: TGP | K: 4
Data Set: TGP | K: 4
Data Set: TGP | K: 5
Data Set: TGP | K: 5
Data Set: TGP | K: 6
Data Set: TGP | K: 6
Data Set: TGP | K: 7
Data Set: TGP | K: 7
Data Set: TGP | K: 8
Data Set: TGP | K: 8
Data Set: TGP | K: 9
Data Set: TGP | K: 9
Data Set: TGP | K: 10
Data Set: TGP | K: 10
Data Set: TGP | K: 11
Data Set: TGP | K: 11
Data Set: TGP | K: 12
Data Set: TGP | K: 12
Data Set: TGP | K: 13
Data Set: TGP | K: 13
Data Set: TGP | K: 14
Data Set: TGP | K: 14
Data Set: TGP | K: 15
Data Set: TGP | K: 15
distance from best loss
iteration
method
em
em & sqp
vi
svi


Data Set: HGDP | K: 2
Data Set: HGDP | K: 2
Data Set: HGDP | K: 3
Data Set: HGDP | K: 3
Data Set: HGDP | K: 4
Data Set: HGDP | K: 4
Data Set: HGDP | K: 5
Data Set: HGDP | K: 5
Data Set: HGDP | K: 6
Data Set: HGDP | K: 6
Data Set: HGDP | K: 7
Data Set: HGDP | K: 7
Data Set: HGDP | K: 8
Data Set: HGDP | K: 8
Data Set: HGDP | K: 9
Data Set: HGDP | K: 9
Data Set: HGDP | K: 10
Data Set: HGDP | K: 10
Data Set: HGDP | K: 11
Data Set: HGDP | K: 11
Data Set: HGDP | K: 12
Data Set: HGDP | K: 12
Data Set: HGDP | K: 13
Data Set: HGDP | K: 13
Data Set: HGDP | K: 14
Data Set: HGDP | K: 14
Data Set: HGDP | K: 15
Data Set: HGDP | K: 15
distance from best loss
iteration
method
em
em & sqp
vi
svi