A nonparametric estimator of population structure unifying admixture models and principal components analysis

Irineo Cabreros* and John D. Storey†

December 28, 2017

Abstract

We introduce a simple and computationally efficient method for fitting the admixture model of genetic population structure, called ALStructure. The strategy of ALStructure is to first estimate the low-dimensional linear subspace of the population admixture components and then search for a model within this subspace that is consistent with the admixture model's natural probabilistic constraints. Central to this strategy is the observation that all models belonging to this constrained space of solutions are risk-minimizing and have equal likelihood, rendering any additional optimization unnecessary. The low-dimensional linear subspace is estimated through a recently introduced principal components analysis method that is appropriate for genotype data, thereby providing a solution that has both principal components and probabilistic admixture interpretations. Our approach differs fundamentally from other existing methods for estimating admixture, which aim to fit the admixture model directly by searching for parameters that maximize the likelihood function or the posterior probability. We observe that ALStructure typically outperforms existing methods both in accuracy and computational speed under a wide array of simulated and real human genotype datasets. Throughout this work we emphasize that the admixture model is a special case of a much broader class of models for which algorithms similar to ALStructure may be successfully employed.

*Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 USA. Email: cabreros@math.princeton.edu.

†Lewis-Sigler Institute for Integrative Genomics, Princeton University, Princeton, NJ 08544 USA. Email: jstorey@princeton.edu.

Contents

1 Introduction

Understanding structured genetic variation in human populations remains a foundational problem in modern genetics. Such an understanding allows researchers to correct for population structure in GWAS studies, enabling accurate disease-gene mapping (Knowler *et al.*, 1988; Marchini *et al.*, 2004; Song *et al.*, 2015). Additionally, characterizing genetic variation is important for the study of human evolutionary history (Cavalli-Sforza *et al.*, 1988; Esteban *et al.*, 1998; Li *et al.*, 2008).

To this end, much work has been done to develop methods to estimate what Alexander *et al.* (2009) term *global ancestry*. In the global ancestry framework, the goal is to simultaneously estimate two quantities:

(i) the allele frequencies of ancestral populations

(ii) the admixture proportions of each modern individual

Many popular global ancestry estimation methods have been developed within a probabilistic framework. In these methods, which we will refer to as *likelihood-based* approaches, the strategy is to fit a probabilistic model to the observed genomewide genotype data by either maximizing the likelihood function (Alexander *et al.*, 2009; Tang *et al.*, 2005) or the posterior probability (Gopalan *et al.*, 2016; Prichard *et al.*, 2000; Raj *et al.*, 2014). The probabilistic model fit in each of these cases is the *admixture model*, described in detail in Section 2.1, in which the global ancestry quantities (i) and (ii) are explicit parameters to be estimated.

A related line of work relies on principal component analysis (PCA) and other eigendecomposition methods, rather than directly fitting probabilistic models; as such, we will refer to them collectively as *PCA-based* approaches. These methods find many of the same applications as global ancestry estimates while obviating a direct computation of global ancestry itself. For or example, the EIGENSTRAT method of Patterson *et al.* (2006) and Price *et al.* (2006) uses the principal components of observed data to correct for population stratification in GWAS, avoiding altogether the estimation of admixture proportions or ancestral allele frequencies. Similarly, Hao *et al.* (2016) observe that many important applications of global ancestry really only require *individual-specific allele frequencies*. In a sense, individual-specific allele frequencies are simpler than global ancestry; while global ancestry specifies all of the individual-specific allele frequencies, the converse is not true. Therefore, Hao *et al.* (2016) introduce a simple truncated-PCA method that accurately and efficiently estimates individual-specific allele frequencies alone.

Both likelihood-based and PCA-based methods have distinct merits and drawbacks. The PCA-based methods are computationally efficient and accurate in practice. It is shown, for instance, that the individual-specific allele frequencies obtained by truncated-PCA are empirically more accurate than those obtained by likelihood-based methods (Hao *et al.*, 2016). Another attractive feature of PCA-based methods is that they make minimal assumptions about the underlying data-generative model. However, as mentioned before, PCA-based methods do not provide the full global ancestry estimates that their corresponding likelihood-based methods do. Most notably, they do not provide direct estimates of admixture proportions, which are often of primary interest in some applications. Additionally, the PCA-based methods, as they are not supported by the statistical theory of likelihood, often have weaker theoretical justifications.

In this paper, we show that the distinction between likelihood-based and PCA-based methods is an unnecessary dichotomy, as others have investigated (Brisbin *et al.*, 2012; Zheng and Weir, 2016). The method that we develop here, which we call `ALStructure`, can be viewed as a unification of these two approaches. While computationally similar to PCA-based methods, `ALStructure` is shown to fit the probabilistic admixture model, thereby providing estimates of global ancestry. Our basic strategy will be to eliminate the primary shortcomings of PCA-based methods while retaining their important advantages over likelihood-based methods. In particular we extend the approach taken in Hao *et al.* (2016) in two ways. First, we replace classical PCA with the closely related method of *latent subspace estimation* (LSE) (Chen and Storey, 2015). In so doing, we will make mathematically rigorous the empirically effective truncated-PCA method of Hao *et al.* for estimating individual-specific allele frequencies. Second, we use the method of *alternating least squares* (ALS) (Paatero and Tapper, 1994) to transform the individual-specific allele frequencies obtained via LSE into estimates of global ancestry.

We perform a body of simulations and analyze several globally sampled human studies to demonstrate the performance of the proposed method, showing that `ALStructure` typically outperforms existing methods both in terms of accuracy and speed. We also discuss its implementation and the trade-offs between theoretical guarantees and run-time. We conclude that `ALStructure` is a computationally efficient and statistically accurate method for modeling admixture and decomposing systematic variation due to population structure.

2 Proposed method

In this section we present the `ALStructure` method and detail some of its mathematical underpinnings. In Section 2.1, we define the *admixture model*: the underlying probabilistic model assumed by `ALStructure`. Section 2.2 describes the overall strategy of `ALStructure` as an optimality search subject to constraints rather than navigating a complex likelihood surface. Section 2.3 describes how the constraints can be used to estimate individual-specific allele frequencies. In Section 2.4 we present a mathematical result from Chen and Storey (2015) upon which the `ALStructure` algorithm heavily relies. Section 2.5 describes why estimating global ancestry, given the individual-specific allele frequencies, is equivalent to a constrained matrix factorization problem. An efficient algorithm based on the method of alternating least squares (ALS) is also provided in this section for performing the constrained matrix factorization. The complete `ALStructure` algorithm is then presented in Section 2.6.

Throughout this work, we adhere to the following notational convention: for a matrix $\boldsymbol{A}$, we denote the i row vector of $\boldsymbol{A}$ by $\boldsymbol{a}_{i\cdot}$, the j column vector of $\boldsymbol{A}$ as $\boldsymbol{a}_{\cdot j}$, and the (i,j) element of $\boldsymbol{A}$ as a_{ij}.

2.1 The admixture model

The observed data $\boldsymbol{X}$ is an $m \times n$ matrix in which m (the number of SNP's) is typically much larger than n (the number of individuals). An element x_{ij} of $\boldsymbol{X}$ takes values 0, 1, or 2 according to the number of reference alleles in the genotype at locus i for individual j.

`ALStructure` makes the assumption common to all likelihood-based methods that the data are generated from the *admixture model*. Under this model, there is an unobserved $m \times n$ matrix $\boldsymbol{F}$ that encodes the complete individual-specific allele frequencies. The genotypes are generated independently according to $x_{ij} | f_{ij} \sim \text{Binomial}(2, f_{ij})$. $\boldsymbol{F}$ is of rank d such that $d \ll n \ll m$, where d carries the interpretation of being the number of ancestral populations from which the observed population is derived. $\boldsymbol{F}$ then admits a factorization $\boldsymbol{F} = \boldsymbol{P}\boldsymbol{Q}$ in which $\boldsymbol{P}$ and $\boldsymbol{Q}$ have the following properties:

$$\begin{aligned} &\boldsymbol{P} \in \mathbb{R}^{m \times d} \text{ with } p_{ij} \in [0,1] \;\; \forall (i,j) \\ &\boldsymbol{Q} \in \mathbb{R}^{d \times n} \text{ with } q_{ij} \geq 0 \;\; \forall (i,j) \text{ and } \textstyle\sum_i q_{ij} = 1 \;\; \forall j \end{aligned}$$

The matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ have the following interpretations: (i) each row $\boldsymbol{p}_{i\boldsymbol{\cdot}}$ of $\boldsymbol{P}$ represents the frequencies of a single SNP for each of the d ancestral populations and (ii) each column $\boldsymbol{q}_{\boldsymbol{\cdot}j}$ of $\boldsymbol{Q}$ represents the admixture proportions of a single individual. Together, $\boldsymbol{P}$ and $\boldsymbol{Q}$ encode the global ancestry parameters of the observed population; the goal of existing likelihood-based methods is to estimate these matrices. By contrast, the truncated-PCA method of HAO *et al.* (2016) is focused on estimating $\boldsymbol{F}$ and not its factors. Eq. 1 summarizes the admixture model.

$$\underset{m \times n}{\left(\begin{array}{c} \\ \\ \\ \boldsymbol{F} \\ \\ \\ \\ \end{array}\right)} = \underset{m \times d}{\left(\begin{array}{c} \\ \\ \\ \boldsymbol{P} \\ \\ \\ \\ \end{array}\right)} \underset{d \times n}{\left(\begin{array}{c} \\ \boldsymbol{Q} \\ \\ \end{array}\right)} \tag{1}$$

The model introduced in PRICHARD, STEPHENS and DONNELLY (2000), which we refer to as the *PSD model*, is an important special case of the admixture model. It additionally assumes the following prior distributions[1] on $\boldsymbol{P}$ and $\boldsymbol{Q}$:

$$\begin{aligned} p_{ij} &\sim \text{Balding-Nichols}(F_i, p_i) \\ \boldsymbol{q}_{\boldsymbol{\cdot}j} &\sim \text{Dirichlet}(\boldsymbol{\alpha}) \end{aligned}$$

The Balding-Nichols distribution (BALDING and NICHOLS, 1995) is a reparameterization of the Beta distribution in which F_i is the F_{ST} (WEIR and COCKERHAM, 1984) at locus i and p_i is the population minor

[1] What we will refer to as the PSD model is not exactly the same as the original formulation in PRICHARD *et al.* (2000). They utilize uniform priors $p_{ij} \sim \text{Beta}(1,1)$ and $\boldsymbol{q}_{\boldsymbol{\cdot}j} \sim \text{Dirichlet}(\boldsymbol{1})$.

allele frequency at locus i. Specifically, Balding-Nichols(F, p) = Beta $\left(\frac{1-F}{F}p, \frac{1-F}{F}(1-p)\right)$. Existing Bayesian likelihood-based methods (GOPALAN *et al.*, 2016; PRICHARD *et al.*, 2000; RAJ *et al.*, 2014) fit the PSD model specifically, while the frequentist likelihood-based methods (ALEXANDER *et al.*, 2009; TANG *et al.*, 2005) and `ALStructure` require only the admixture model assumptions.

2.2 Optimal model constraints

Most existing methods for fitting the admixture model employ various optimization techniques to search for the maximum likelihood parameters in the frequentist setting (ALEXANDER *et al.*, 2009; PRICHARD *et al.*, 2000) or the maximum a posteriori estimate in the Bayesian setting (GOPALAN *et al.*, 2016; RAJ *et al.*, 2014). Our approach has a fundamentally different character: rather than searching through a rough likelihood landscape in pursuit of an optimal solution, the `ALStructure` algorithm optimizes over a set of feasible solutions subject to a set of optimal constraints. Any solution meeting these constraints is a risk-minimizing solution up to the inherent non identifiability of the admixture model; consequently, any further optimization is unnecessary.

There are several constraints that any reasonable estimate of the parameters of the admixture model must obey. The first is simply that the parameter estimates $\hat{\boldsymbol{F}}$, $\hat{\boldsymbol{P}}$, and $\hat{\boldsymbol{Q}}$ obey the relationship $\hat{\boldsymbol{F}} = \hat{\boldsymbol{P}}\hat{\boldsymbol{Q}}$. We will refer to this constraint as the “Equality" constraint. The second obvious requirement is that entries of matrices $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{Q}}$ obey the probabilistic constraints of the admixture model:

$$p_{ij} \in [0,1] \quad \forall (i,j) \tag{2}$$

$$q_{ij} \geq 0 \qquad \forall (i,j) \tag{3}$$

$$\sum_i q_{ij} = 1 \qquad \forall j \tag{4}$$

As we will encounter these constraints frequently, we refer to Eq. (2) as the “□" constraint, and Eq. (3) and (4) as the “△" constraint. This is simply because the constraints on $\boldsymbol{P}$ demarcate the boundaries of a d-dimensional unit cube (the generalization of a square) whereas the constraints on $\boldsymbol{Q}$ demarcate a d-dimensional simplex (the generalization of an equilateral triangle). Together we refer to the □ and △ constraints as the “Boundary" constraints.

The final constraint we require is that the row vectors of $\hat{\boldsymbol{F}}$ lie in the linear subspace spanned by the rows vectors of $\boldsymbol{Q}$. If we denote $\langle \boldsymbol{A} \rangle$ to be the rowspace of a matrix $\boldsymbol{A}$, we can summarize this condition as:

$$\langle \hat{\boldsymbol{F}} \rangle = \langle \boldsymbol{Q} \rangle \tag{5}$$

We will refer to Eq. 5 as the “LS" (linear subspace) constraint. The LS constraint is the only nontrivial constraint that `ALStructure` enforces. The fact that $\langle \boldsymbol{F} \rangle = \langle \boldsymbol{Q} \rangle$ is a simple consequence of the linearity of the admixture model; indeed, all rows of $\boldsymbol{F}$ are linear combinations of rows of $\boldsymbol{Q}$ since $\boldsymbol{F} = \boldsymbol{PQ}$. The LS constraint thus requires the same property for our estimate $\hat{\boldsymbol{F}}$. It is important to note that the LS constraint is not the same as requiring that $\langle \hat{\boldsymbol{F}} \rangle = \langle \hat{\boldsymbol{Q}} \rangle$: this is ensured by the Equality constraint. Rather, the LS constraint requires that the row vectors of $\hat{\boldsymbol{F}}$ belong to the rowspace of the *true* $\boldsymbol{Q}$ matrix.

The apparent challenge of enforcing the LS constraint is that *a priori*, one does not have access to $\langle \boldsymbol{Q} \rangle$. However, `ALStructure` takes advantage of a recent result from CHEN and STOREY (2015) that $\langle \boldsymbol{Q} \rangle$ can be consistently estimated directly from the data matrix $\boldsymbol{X}$ in the asymptotic regime of interest, when the number of SNP's m grows large. The result of CHEN and STOREY (2015) is in fact much more general than is needed in our setting and therefore will likely be useful in many other problems. Because of its importance to this work, we further discuss this result in the context of the admixture model in Section 2.4, and show that a modified PCA of $\boldsymbol{X}$ consistently recovers $\langle \boldsymbol{Q} \rangle$.

2.3 Leveraging constraints to estimate $\hat{\boldsymbol{F}}$

The key step in `ALStructure` is to note that enforcing the LS constraint provides us with an immediate estimate for $\boldsymbol{F}$. To motivate our estimator, first observe that the simple estimate $\tilde{\boldsymbol{F}} = \frac{1}{2}\boldsymbol{X}$ is in some sense a reasonable approximation of $\boldsymbol{F}$: it is unbiased since $f_{ij} = \frac{1}{2}\,\mathrm{E}[x_{ij}]$ under the admixture model. However, this estimate leaves much to be desired — most importantly, the estimate $\tilde{\boldsymbol{F}}$ will in general be of full rank (n) rather than of low rank (d) and it will have a large variance. Assuming, for now, that we are provided with the true rowspace $\langle \boldsymbol{Q} \rangle$ of $\boldsymbol{F}$, a natural thing to try will be to project the rows of $\frac{1}{2}\boldsymbol{X}$ onto this linear subspace. Below we show that this estimator has some appealing properties.

Let us denote the the operator $\text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{X})$ such that the rows of the matrix $\boldsymbol{X}$ are projected onto the linear subspace $\langle \mathcal{S} \rangle$.[2] If we are given an orthonormal basis $\{\boldsymbol{s}_i\}$ of the k-dimensional linear subspace $\langle \mathcal{S} \rangle$, then:

$$\text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{X}) \equiv \boldsymbol{X} \left(\sum_{i=1}^{k} \boldsymbol{s}_i \boldsymbol{s}_i^T \right)$$

We will now show that the estimate

$$\hat{\boldsymbol{F}} = \frac{1}{2}\text{Proj}_{\langle Q \rangle}(\boldsymbol{X}) \tag{6}$$

is both unbiased and has minimal risk (as defined below) among all projections.

To see that $\hat{\boldsymbol{F}}$ is unbiased, we simply note that:

$$\begin{aligned} \mathrm{E}[\hat{\boldsymbol{F}}] &= \frac{1}{2}\,\mathrm{E}[\text{Proj}_{\langle \boldsymbol{Q} \rangle}(\boldsymbol{X})] \\ &= \frac{1}{2}\text{Proj}_{\langle \boldsymbol{Q} \rangle}(\mathrm{E}[\boldsymbol{X}]) \\ &= \text{Proj}_{\langle \boldsymbol{Q} \rangle}(\boldsymbol{F}) \\ &= \boldsymbol{F} \end{aligned}$$

Between the first and second line, we note that the projection operator is linear and take advantage of linearity of expectation. Between second and third line, we used the observation that $\frac{1}{2}\,\mathrm{E}[\boldsymbol{X}] = \boldsymbol{F}$. Finally, $\text{Proj}_{\langle \boldsymbol{Q} \rangle}\boldsymbol{F} = \boldsymbol{F}$ since all rows of $\boldsymbol{F}$ belong to $\langle \boldsymbol{Q} \rangle$. From an identical argument one can see that

[2] The notation $\text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{X})$ typically refers to projection of the columns of $\boldsymbol{X}$ onto the linear subspace $\langle \mathcal{S} \rangle$, but here we use this notation to denote projection of the rows of $\boldsymbol{X}$ onto $\langle \mathcal{S} \rangle$.

for projection onto any other subspace $\langle \mathcal{S} \rangle$, the corresponding estimator $\hat{\boldsymbol{F}}_{\mathcal{S}} \equiv \frac{1}{2}\text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{X})$ will have the property that

$$\mathrm{E}[\hat{\boldsymbol{F}}_{\mathcal{S}}] = \text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{F})$$

It is clear that if $\langle \boldsymbol{Q} \rangle \subseteq \langle \mathcal{S} \rangle$, then $\text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{F}) = \boldsymbol{F}$ since the projection operators act as the identity operator for vectors belonging to the subspace $\langle \mathcal{S} \rangle$. However, if $\langle \boldsymbol{Q} \rangle \not\subseteq \langle \mathcal{S} \rangle$ then $\mathrm{E}[\hat{\boldsymbol{F}}_{\mathcal{S}}] \neq \boldsymbol{F}$ in general. In conclusion, we make the following simple observation.

Lemma 1. *For a rank d matrix $\boldsymbol{F}$ that admits a factorization $\boldsymbol{F} = \boldsymbol{P}\boldsymbol{Q}$ and a random matrix $\boldsymbol{X}$ such that $\frac{1}{2}\mathrm{E}[\boldsymbol{X}] = \boldsymbol{F}$, any estimator of $\boldsymbol{F}$ of the form $\hat{\boldsymbol{F}}_{\mathcal{S}} \equiv \frac{1}{2}\text{Proj}_{\langle \mathcal{S} \rangle}(\boldsymbol{X})$ is unbiased if and only if $\langle \boldsymbol{Q} \rangle \subseteq \langle \mathcal{S} \rangle$.*

We now show that among all unbiased estimators constructed by projecting $\boldsymbol{X}$ onto a linear subspace, the projection onto $\langle \boldsymbol{Q} \rangle$ has the minimal risk, where our loss function is defined as the distance between $\hat{\boldsymbol{F}}$ and $\boldsymbol{F}$ via the squared Frobenius matrix norm (the squared sum of matrix elements):

$$\begin{aligned} L(\boldsymbol{F}, \hat{\boldsymbol{F}}) &\equiv ||\hat{\boldsymbol{F}} - \boldsymbol{F}||^2 \\ &= \text{Tr}[(\hat{\boldsymbol{F}} - \boldsymbol{F})^T(\hat{\boldsymbol{F}} - \boldsymbol{F})] \\ &= \text{Tr}[\hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}}] - 2\text{Tr}[\hat{\boldsymbol{F}}^T\boldsymbol{F}] + \text{Tr}[\boldsymbol{F}^T\boldsymbol{F}] \end{aligned} \tag{7}$$

First let us compute the risk of our projection estimator $\hat{\boldsymbol{F}}$. Suppose we have an orthonormal basis $\{\boldsymbol{v}_i\}$ of $\langle \boldsymbol{Q} \rangle$. Using the definition of $\hat{\boldsymbol{F}}$ from Eq. 6 and the fact that the rows of both $\boldsymbol{F}$ belong to $\langle \boldsymbol{Q} \rangle$, we note that we can write any row of either matrix in terms of the basis vectors $\{\boldsymbol{v}_i\}$:

$$\boldsymbol{f}_{i\bullet} = \sum_j \langle \boldsymbol{f}_{i\bullet}^T, \boldsymbol{v}_j \rangle \boldsymbol{v}_j^T \tag{8}$$

$$\hat{\boldsymbol{f}}_{i\bullet} = \frac{1}{2}\sum_j \langle \boldsymbol{x}_{i\bullet}^T, \boldsymbol{v}_j \rangle \boldsymbol{v}_j^T \tag{9}$$

By rewriting the matrices $\hat{\boldsymbol{F}}$ and $\boldsymbol{F}$ with respect to the basis $\{\boldsymbol{v}_i\}$ and using Eq. 8 and 9, it is a straightforward calculation to show that

$$\text{Tr}[\boldsymbol{F}^T\boldsymbol{F}] = \sum_{i=1}^{m}\sum_{j=1}^{k} \langle \boldsymbol{f}_{i\bullet}^T, \boldsymbol{v}_j \rangle^2$$

$$\text{Tr}[\hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}}] = \sum_{i=1}^{m}\sum_{j=1}^{k} \langle \frac{1}{2}\boldsymbol{x}_{i\bullet}^T, \boldsymbol{v}_j \rangle^2$$

Substituting this result into Eq. 7 and taking expectations, we have the following expression for our loss function:

$$\begin{aligned} R(\boldsymbol{F}, \hat{\boldsymbol{F}}) &= \mathrm{E}[L(\boldsymbol{F}, \hat{\boldsymbol{F}})] \\ &= \mathrm{E}\left[\mathrm{Tr}[\hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}}] - 2\mathrm{Tr}[\hat{\boldsymbol{F}}^T\boldsymbol{F}] + \mathrm{Tr}[\boldsymbol{F}^T\boldsymbol{F}]\right] \\ &= \mathrm{E}\left[\mathrm{Tr}[\hat{\boldsymbol{F}}^T\hat{\boldsymbol{F}}] - \mathrm{Tr}[\boldsymbol{F}^T\boldsymbol{F}]\right] \\ &= \sum_{i=1}^{m}\sum_{j=1}^{k}\mathrm{E}\left[\langle\frac{1}{2}\boldsymbol{x}_{i\bullet}^T, \boldsymbol{v}_j\rangle^2\right] - \langle\boldsymbol{f}_{i\bullet}^T, \boldsymbol{v}_j\rangle^2 \\ &= \frac{1}{4}\sum_{i=1}^{m}\sum_{j=1}^{k}\mathrm{Var}[\langle\boldsymbol{x}_{i\bullet}^T, \boldsymbol{v}_j\rangle] \end{aligned} \tag{10}$$

By studying Eq. 10, we can see the estimator $\hat{\boldsymbol{F}}$ has several favorable properties. First note that the risk is a sum of $m \times k$ nonnegative numbers since $\mathrm{Var}[\boldsymbol{Z}] \geq 0$ for any random variable $\boldsymbol{Z}$. If we were to project onto a larger subspace $\langle\mathcal{S}\rangle \supset \langle\boldsymbol{Q}\rangle$, we would add terms to Eq. 10 and consequently increase our risk. If we were to project onto a smaller subspace $\langle\mathcal{S}\rangle \subset \langle\boldsymbol{Q}\rangle$, then the risk may decrease, however our new estimator will now be biased by Lemma 1. From these observations, we conclude that $\hat{\boldsymbol{F}}$ is optimal in the sense described in the following Lemma 2.

Lemma 2. *For a rank d matrix $\boldsymbol{F}$ that admits a factorization $\boldsymbol{F} = \boldsymbol{PQ}$ and a random matrix $\boldsymbol{X}$ such that $\frac{1}{2}\mathrm{E}[\boldsymbol{X}] = \boldsymbol{F}$, the estimator $\hat{\boldsymbol{F}} \equiv \frac{1}{2}\mathrm{Proj}_{\langle\boldsymbol{Q}\rangle}(\boldsymbol{X})$ is an unbiased estimator of $\boldsymbol{F}$ and has the smallest risk of any unbiased estimator of the form $\tilde{\boldsymbol{F}} \equiv \frac{1}{2}\mathrm{Proj}_{\langle\mathcal{S}\rangle}(\boldsymbol{X})$.*

We note that this strategy is related to the strategy taken in HAO *et al.* (2016) in which $\boldsymbol{F}$ was estimated by projecting $\frac{1}{2}\boldsymbol{X}$ onto the space spanned by the first d principal components. In that work, it was observed that this simple strategy of estimating $\hat{\boldsymbol{F}}$ typically outperformed existing methods in terms of estimating $\boldsymbol{F}$. We will see in Section 2.4 that the space spanned by the first d principal components is a good estimator for $\langle\boldsymbol{Q}\rangle$ itself, but it can be improved practically and with theoretical guarantees by performing a modified PCA. Therefore, Lemma 2 provides a theoretical justification for the empirically accurate method put forward in HAO *et al.* (2016).

2.4 Latent subspace estimation

We have shown that the linear subspace $\langle\boldsymbol{Q}\rangle$ can be leveraged to provide a desirable estimate of $\boldsymbol{F}$. Here we show how we can compute a consistent estimate of $\langle\boldsymbol{Q}\rangle$ from the observed data $\boldsymbol{X}$ using a general technique developed in CHEN and STOREY (2015), which we will refer to as *Latent Subspace Estimation* (LSE).

PCA, a popular technique to apply to population genetic data, is a method for identifying linear combinations of variables that sequentially maximize variance explained in the data (JOLLIFFE, 2002). We would like to employ SNP-wise PCA to estimate $\langle\boldsymbol{Q}\rangle$. A standard SNP-wise PCA applied to $\boldsymbol{X}$ computes the singular value decomposition of the $n \times n$ sample covariance matrix, $\frac{1}{m}\boldsymbol{X}^T\boldsymbol{X}$ (perhaps by first mean centering SNP-wise). However, we can calculate that

$$\begin{aligned}\mathrm{Var}[x_{ij}] &= \mathrm{Var}[\mathrm{E}[x_{ij}|f_{ij}]] + \mathrm{E}[\mathrm{Var}[x_{ij}|f_{ij}]] \\ &= \mathrm{Var}[2f_{ij}] + \mathrm{E}[2f_{ij}(1-f_{ij})] \\ \mathrm{Cov}[x_{ij}, x_{ik}] &= \mathrm{Cov}[\mathrm{E}[x_{ij}|f_{ij}], \mathrm{E}[x_{ik}|f_{ik}]] + \mathrm{E}[\mathrm{Cov}[x_{ij}, x_{ik}|f_{ij}, f_{ik}]] \\ &= \mathrm{Cov}[2f_{ij}, 2f_{ik}]\end{aligned}$$

The information about $\langle \boldsymbol{Q} \rangle$ is contained in terms composed of $\mathrm{Var}[2f_{ij}]$ and $\mathrm{Cov}[2f_{ij}, 2f_{ik}]$, but not $\mathrm{E}[2f_{ij}(1-f_{ij})]$. As fully detailed in Chen and Storey (2015), the term $\mathrm{E}[2f_{ij}(1-f_{ij})]$ captures variation specific to x_{ij}, so we must correct for $\mathrm{E}[2f_{ij}(1-f_{ij})]$ in the PCA.

Let $\delta_{ij} = \mathrm{E}[2f_{ij}(1-f_{ij})]$, and let $\boldsymbol{\Delta}$ be a diagonal matrix with jth element equal to $\frac{1}{m}\sum_{i=1}^{n}\delta_{ij}$. Also, let $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d\}$ corresponding to the top d eigenvalues of the matrix $\boldsymbol{\Gamma} = \frac{1}{m}\boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{\Delta}$. Chen and Storey (2015) shows that

$$\lim_{m\to\infty} \langle \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d\} \rangle \triangle \langle \boldsymbol{Q} \rangle = \emptyset$$

with probability 1, where $\triangle$ denotes the symmetric set difference.

However, since we don't know the term $\mathrm{E}[2f_{ij}(1-f_{ij})]$, we must estimate it. Since $\mathrm{E}[2x_{ij} - x_{ij}^2] = \mathrm{E}[2f_{ij}(1-f_{ij})]$, Chen and Storey (2015) further show the following theorem (which we have re-written to reflect our special case).

Theorem 1 (Chen and Storey (2015))**.** *Let us define $\hat{\delta}_j = \frac{1}{m}\sum_i 2x_{ij} - x_{ij}^2$ and let $\boldsymbol{D}$ be the diagonal matrix with jth entry equal to $\hat{\delta}_j$. The d eigenvectors $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d\}$ corresponding to the top d eigenvalues of the matrix $\boldsymbol{G} = \frac{1}{m}\boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{D}$ span the latent subspace $\langle \boldsymbol{Q} \rangle$ in the sense that*

$$\lim_{m\to\infty} \langle \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_d\} \rangle \triangle \langle \boldsymbol{Q} \rangle = \emptyset$$

with probability 1, where $\triangle$ denotes the symmetric set difference. Further, the smallest $n-d$ eigenvalues of $\boldsymbol{G}$ conerge to 0 with probability 1.

The row space $\langle \boldsymbol{Q} \rangle$ is therefore estimated by

$$\widehat{\langle \boldsymbol{Q} \rangle} = \boldsymbol{V}_{(1:d)}^T \tag{11}$$

where $\boldsymbol{V}_{1:d}$ are the first d columns from the singular value decomposition of $\boldsymbol{G}$. We note that the singular value decomposition is $\boldsymbol{G} = \boldsymbol{V}\boldsymbol{W}\boldsymbol{V}^T$, where the diagonal matrix $\boldsymbol{W}$ contains the eigenvalues and the columns of $\boldsymbol{V}$ are the eigenvectors.

Importantly, the first d rows of

$$\boldsymbol{W}^{1/2}\boldsymbol{V}^T \tag{12}$$

are the top d principal components of $\boldsymbol{G}$. These capture the systematic variation in the rowspaces of $\boldsymbol{X}$ and $\boldsymbol{F}$ due to $\boldsymbol{Q}$. These are the principal components of interest for understanding structure, not those calculated directly from $\boldsymbol{X}$.

CHEN and STOREY (2015) also provides results on the distribution of the eigenvalues of $\boldsymbol{G}$ as $m \to \infty$, along with results on how to determine d. We stress that the general form of Theorem 1 from CHEN and STOREY (2015) makes LSE applicable to a vast array of models beyond factor models and the admixture model discussed here. As a further benefit to the LSE methodology, it is both easy to implement and computationally appealing. The entire computation of $\widehat{\langle \boldsymbol{Q} \rangle}$ requires a single eigendecomposition of an $n \times n$ matrix where the accuracy depends only on large m.

2.5 Leveraging constraints to estimate $\boldsymbol{P}$ and $\boldsymbol{Q}$

Now that we have a method for obtaining the estimate $\hat{\boldsymbol{F}}$ by leveraging the LS constraint, what remains is to find estimates for $\boldsymbol{P}$ and $\boldsymbol{Q}$. Since the estimate $\hat{\boldsymbol{F}}$ has several appealing properties, as outlined in Section 2.3, the approach of `ALStructure` is simply to keep $\hat{\boldsymbol{F}}$ fixed and seek matrices $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{Q}}$ that obey the Equality and Boundary constraints of the admixture model. Below we discuss some of the general properties of this approach: namely the question of existence and uniqueness of solutions. We will briefly discuss the general problem of nonidentifiability in the admixture model and provide simple and interpretable conditions under which the admixture model is identifiable. Finally, we will provide simple algorithms for computing $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{Q}}$ from $\hat{\boldsymbol{F}}$ based on the method of Alternating Least Squares (ALS).

Existence, uniqueness, and anchor conditions. First we develop some terminology. We will say that an $m \times n$ matrix $\boldsymbol{A}$ *admits an admixture-factorization* if the following feasibility problem has a solution:

$$\begin{aligned} \text{find:} \quad & (\boldsymbol{B}, \boldsymbol{C}) \\ \text{subject to:} \quad & \boldsymbol{A} = \boldsymbol{B}\boldsymbol{C} \text{ and } (\square, \triangle) \end{aligned} \tag{13}$$

In words, the feasibility problem in Eq. 13 simply seeks a factorization of $\boldsymbol{A}$ that obeys the Equality and Boundary constraints from Section 2.2 imposed by the admixture model. The smallest integer d for which $(\boldsymbol{B}, \boldsymbol{C})$ is a solution to Eq. 13 with $\boldsymbol{B}$ an $m \times d$ matrix and $\boldsymbol{C}$ a $d \times n$ matrix is the *admixture-rank* of $\boldsymbol{A}$, which we denote $\text{rank}_{\text{ADM}}(\boldsymbol{A})$. By seeking a rank d admixture-factorization of $\hat{\boldsymbol{F}}$, `ALStructure` converts a problem of high-dimensional statistical inference to a matrix factorization problem.

This simple approach has two apparent shortcomings:

(i) A rank d admixture-factorization of $\hat{\boldsymbol{F}}$ may not exist.

(ii) If a valid factorization exists, it will not be unique.

Item (i) is a technical problem; though $\boldsymbol{F}$ admits a rank d admixture factorization by assumption, the same is not true for $\hat{\boldsymbol{F}}$ in general. Even though the rank of $\hat{\boldsymbol{F}}$ is d by construction, $\text{rank}(\hat{\boldsymbol{F}}) \neq \text{rank}_{\text{ADM}}(\hat{\boldsymbol{F}})$ in general. `ALStructure` avoids this problem changing the feasibility problem expressed in Eq. 13 to the following optimization problem:

$$\begin{aligned} \underset{(\boldsymbol{B},\boldsymbol{C})}{\text{minimize}} \quad & ||\boldsymbol{A} - \boldsymbol{B}\boldsymbol{C}|| \\ \text{subject to:} \quad & (\square, \triangle) \end{aligned} \tag{14}$$

It is important to note that (ii) is not a problem unique to `ALStructure`, but is a fundamental limitation for any maximum likelihood (ML) method as well. This is because the likelihood function depends on $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{Q}}$ only through their product $\hat{\boldsymbol{F}}$; more formally, the admixture model is non-identifiable. One unavoidable source of nonidentifiability is that any solution $(\hat{\boldsymbol{P}}, \hat{\boldsymbol{Q}})$ to the matrix factorization problem in Eq. 13 will remain a valid solution after applying a permutation to the columns of $\hat{\boldsymbol{P}}$ and the rows of $\hat{\boldsymbol{Q}}$. A natural question to ask is: “When is there a unique factorization $\hat{\boldsymbol{F}} = \hat{\boldsymbol{P}}\hat{\boldsymbol{Q}}$ up to permutations?"

Two important types of sufficient conditions under which unique factorizations exist up to permutations are (i) *anchor SNPs* and (ii) *anchor individuals*. We note that the concept of anchors has been previously employed in the field of topic modeling, where *anchor words* are of interest (Arora *et al.*, 2013). In the case of anchor SNP's, every ancestral population has at least one SNP for which the allele frequency is nonzero in that particular population but fixed in all other ancestral populations. Analogously, in the case of anchor individuals, there exists at least one individual per ancestral population whose entire genome is inherited from that population. The fact that either a set of d anchor SNPs or d anchor individuals makes the admixture model identifiable up to permutations follows from a simple argument found in Appendix B. For the special case of $d = 3$, Fig. 1 graphically displays the anchor conditions. It is important to remember that `ALStructure` does not *require* anchors to function. Rather, anchors provide interpretable conditions under which solutions provided by `ALStructure`, or any likelihood based method, can be meaningfully compared to the underlying truth.

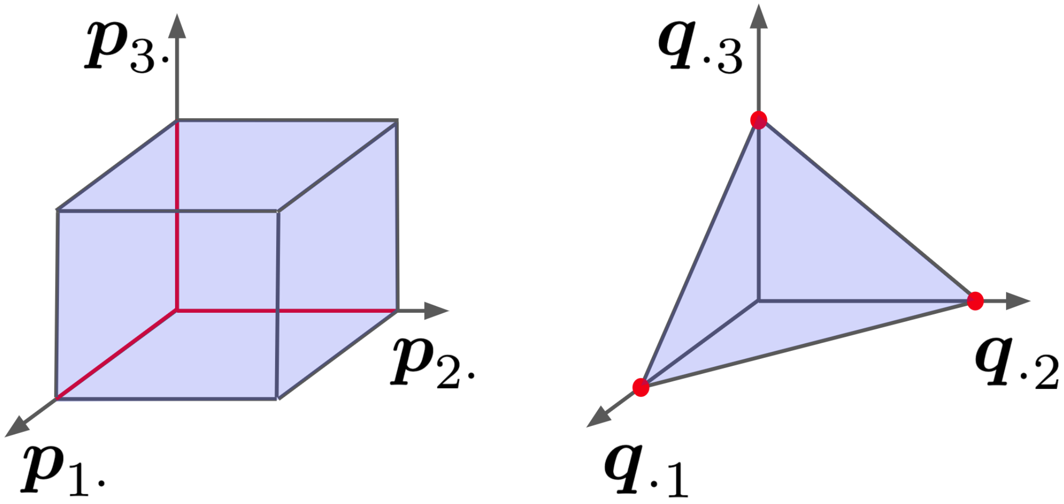

$\boldsymbol{p}_{3\cdot}$
$\boldsymbol{p}_{2\cdot}$
$\boldsymbol{p}_{1\cdot}$
$\boldsymbol{q}_{\cdot 3}$
$\boldsymbol{q}_{\cdot 2}$
$\boldsymbol{q}_{\cdot 1}$

Figure 1: Summary of sufficient conditions for a factorization $\boldsymbol{F} = \boldsymbol{P}\boldsymbol{Q}$ to be unique for $d = 3$. Axes represent the components of the row vectors of $\boldsymbol{P}$ and the column vectors of $\boldsymbol{Q}$ respectively. (left) Anchor SNP's: there is at least one row of $\boldsymbol{P}$ on each on each of the red lines. (right) Anchor genotypes: there is at least one column of $\boldsymbol{Q}$ on each of the red dots.

The anchor SNP and anchor individual conditions are not necessarily the only sufficient conditions for ensuring identifiability of the admixture model and indeed to the best or our knowledge, there is not currently a complete characterization of conditions for which the admixture model is identifiable. We regard this as an important open problem. In practice, `ALStructure` is capable of retrieving solutions remarkably close to the underlying truth even in simulated scenarios far from satisfying the anchor conditions, including conditions which are challenging for existing methods.

Computation. Here we present two simple algorithms for solving the optimization problem:

$$\begin{aligned} \underset{(\boldsymbol{P},\boldsymbol{Q})}{\text{minimize}} \quad & ||\hat{\boldsymbol{F}} - \boldsymbol{PQ}|| \\ \text{subject to:} \quad & (\square, \triangle) \end{aligned} \tag{15}$$

The first algorithm, which we call `cALS` (constrained Alternating Least Squares) has the advantage that it is guaranteed to converge to a stationary point of the objective function in (15). While theoretically appealing, this algorithm relies on solving many constrained quadratic programming problems and is consequently potentially slow. To overcome this problem, we introduce a second algorithm called `uALS` (unconstrained Alternating Least Squares), which simply ignores the problematic quadratic constraints in `cALS`. Though lacking a theoretical guarantee of convergence, the increase in speed is significant and the outputs of the two algorithms are often practically indistinguishable. We note that a similar approach was taken in Berry *et al.* (2007) for the problem of nonnegative matrix factorization (NNMF).

An algorithm with provable convergence. While problem (15) is nonconvex as stated, the following two subproblems are convex:

$$\begin{aligned} \underset{\boldsymbol{P}}{\text{minimize}} \quad & ||\hat{\boldsymbol{F}} - \boldsymbol{PQ}|| \\ \text{subject to:} \quad & \square \end{aligned} \tag{16}$$

$$\begin{aligned} \underset{\boldsymbol{Q}}{\text{minimize}} \quad & ||\hat{\boldsymbol{F}} - \boldsymbol{PQ}|| \\ \text{subject to:} \quad & \triangle \end{aligned} \tag{17}$$

That (16) and (17) are convex problems is clear; norms are always convex functions and $\square$ and $\triangle$ are convex constraints. In particular (16) and (17) are both members of the well-studied class of Quadratic Programs (QP) for which many efficient algorithms exist (Boyd and Vandenberghe, 2009). We propose as our procedure for factoring $\hat{\boldsymbol{F}}$ the following simple alternating procedure, called the *Constrained ALS Algorithm*.

Algorithm 1 Constrained ALS Algorithm

```
procedure cALS($\hat{F}$, d)
    Initialize $\hat{P}$ arbitrarily.
    repeat
        Solve (17) with $P = \hat{P}$ and return $\hat{Q}$.
        Solve (16) with $Q = \hat{Q}$ and return $\hat{P}$.
    until Convergence of $\hat{P}$ and $\hat{Q}$
return $(\hat{P}, \hat{Q})$
```

Despite the original problem being nonconvex, Algorithm 1 is guaranteed to converge to a stationary point of the objective function in (15) as a result of the following theorem from Grippo and Sciandrone (2000).

Theorem 2. *For the two block problem,*

$$\underset{\boldsymbol{P},\boldsymbol{Q}}{\textit{minimize}} \quad f(\boldsymbol{P},\boldsymbol{Q})$$

if $\{\boldsymbol{P}_i\}$ and $\{\boldsymbol{Q}_i\}$ are a sequence of optimal solutions to the two subproblems:

$$\underset{\boldsymbol{P}}{\textit{minimize}} \quad f(\boldsymbol{P},\boldsymbol{Q}_i)$$
$$\underset{\boldsymbol{Q}}{\textit{minimize}} \quad f(\boldsymbol{P}_i,\boldsymbol{Q})$$

then any limit point $(\boldsymbol{P},\boldsymbol{Q})$ will be a stationary point of the original problem. [3]

An efficient heuristic algorithm. If we remove all constraints on $\boldsymbol{P}$ and $\boldsymbol{Q}$ from Eq. 16 and 17, the resulting optimization problems are simple linear least squares (LS).

$$\underset{\boldsymbol{P}}{\text{minimize}} \quad ||\boldsymbol{F} - \boldsymbol{P}\boldsymbol{Q}|| \tag{18}$$
$$\underset{\boldsymbol{Q}}{\text{minimize}} \quad ||\boldsymbol{F} - \boldsymbol{P}\boldsymbol{Q}|| \tag{19}$$

Our algorithm proceeds by alternating between solving the unconstrained LS problems (18) and (19). After each step, the optimal solution will not necessarily obey the constraints of problem (15). To keep our algorithm from converging on an infeasible point, we truncate the solution to force it into the feasible set. More precisely, each element of the solution $\boldsymbol{P}^*$ to (18) is truncated to satisfy $\square$ and each entry of the solution $\boldsymbol{Q}^*$ to (19) is projected to the closest point on the simplex defined by the $\triangle$ constraints. Simplex-projection is nontrivial, however it is a well-studied optimization problem. Here we use a particularly simple and fast algorithm from CHEN and YE (2011). This algorithm, which we call the *Unconstrained ALS Algorithm*, works as follows.

Algorithm 2 Unconstrained ALS Algorithm

```
procedure uALS($\hat{\boldsymbol{F}}$, d)
    Initialize $\hat{\boldsymbol{P}}$ arbitrarily.
    repeat
        Solve (19) with $\boldsymbol{P} = \hat{\boldsymbol{P}}$, and return the simplex-projected solution $\hat{\boldsymbol{Q}}$.
        Solve (18) with $\boldsymbol{Q} = \hat{\boldsymbol{Q}}$ and return the truncated solution $\hat{\boldsymbol{P}}$.
    until Convergence of $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{Q}}$
return $(\hat{\boldsymbol{P}}, \hat{\boldsymbol{Q}})$
```

An example dataset. Figure 2 displays the output of cALS and uALS on a dataset from the PSD model with the parameters: $m = 100\,000$, $n = 500$, $k = 3$, $\boldsymbol{\alpha} = (0.1, 0.1, 0.1)$. As can be seen, the output fits for $\boldsymbol{Q}$ provided by cALS and uALS are practically indistinguishable to the eye and are both excellent approximations of the ground truth. The cALS algorithm performed slightly better than the

[3] The result from GRIPPO and SCIANDRONE (2000) is actually more general than this. We reproduce the special case above in order to make clear the connection to our problem.

uALS algorithm (6.9×10^{-3} and 7.1×10^{-3} mean absolute error, respectively). However, cALS took 3.9 hours to complete while uALS terminated in under 2 minutes. Because of the significant gains in efficiency, we use uALS exclusively throughout the remainder of this paper. The analyst who requires theoretical guarantees can, of course, use the cALS algorithm instead.

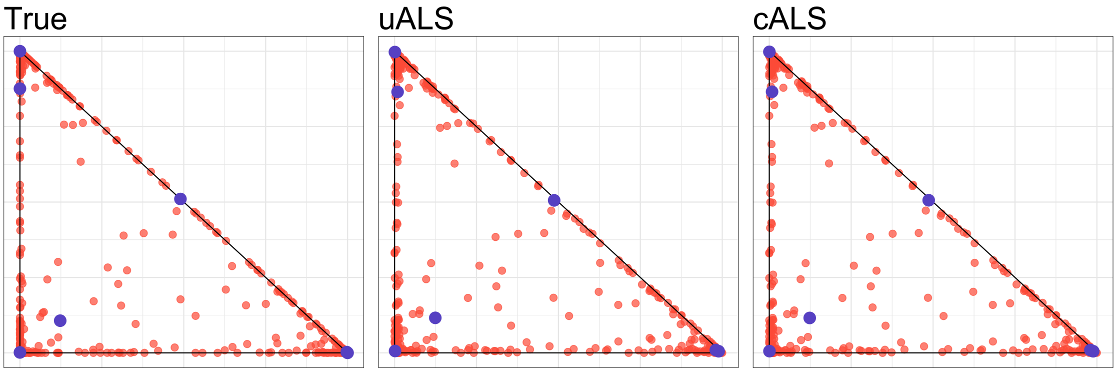


Figure 2: Biplots of the first two rows of $\boldsymbol{Q}$ (left), $\hat{\boldsymbol{Q}}_{\text{uALS}}$ (middle) and $\hat{\boldsymbol{Q}}_{\text{cALS}}$ (right). Blue points are provided as a visual aid and delineate a common subset of individuals.

2.6 The ALStructure algorithm

In this section we briefly outline the entire ALStructure method whose components were motivated in depth in Sections 2 above. In order to fit the admixture model, we obtain estimates $\hat{\boldsymbol{F}}$, $\hat{\boldsymbol{P}}$, and $\hat{\boldsymbol{Q}}$ from the SNP matrix $\boldsymbol{X}$ through the following three part procedure:

(i) Estimate the linear subspace $\langle \boldsymbol{Q} \rangle$ from the data $\boldsymbol{X}$.

(ii) Project $\frac{1}{2}\boldsymbol{X}$ onto the estimate $\widehat{\langle \boldsymbol{Q} \rangle}$ to obtain an estimate of $\boldsymbol{F}$.

(iii) Factor the estimate $\hat{\boldsymbol{F}}$ subject to the Equality and Boundary constriants to obtain $\hat{\boldsymbol{P}}$ and $\hat{\boldsymbol{Q}}$.

For convenience, we detail the entire ALStructure algorithm in Algorithm 3 and annotate each of the three steps described above [4].

We emphasize here that ALStructure's estimate of global ancestry $\hat{\boldsymbol{Q}}$ is ultimately derived from the LSE-based estimate of the latent subspace $\widehat{\langle \boldsymbol{Q} \rangle}$. As the method of LSE is closely linked to PCA, we consider ALStructure to be a unification of PCA-based and likelihood-based approaches.

Perhaps the most striking feature of Algorithm 3 is its brevity. One advantage of this simplicity is its ease of implementation. Although Algorithm 3 has been implemented in the R package ALStructure, it can clearly be reimplemented in any language quite easily. Equally important is that all of the operations in Algorithm 3 are very standard. The only two computationally expensive components are (i) a single eigendecomposition (line 6) if n is large and (ii) QR decompositions to find linear least squares

[4] We note that we have decided to use the uALS function rather than the cALS function in our definition of the ALStructure algorithm, valuing the speed advantage of uALS over the theoretical guarantees of cALS. If desired, one could of course choose to use the cALS function instead without making any other alterations to the ALStructure.

Algorithm 3 ALStructure

```
procedure ALSTRUCTURE($X$, $d$)
    for $j = 0$ to n do                                                          ▷ (i)
        $\hat{\delta}_j \leftarrow \frac{1}{m}\sum_{i=1}^{m} 2x_{ij} - x_{ij}^2$
    $\boldsymbol{D} \leftarrow \text{diag}(\{\hat{\delta}_1, \ldots, \hat{\delta}_n\})$
    $\boldsymbol{G} \leftarrow \frac{1}{m}\boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{D}$
    Compute eigendecomposition $\boldsymbol{G} = \boldsymbol{V}\boldsymbol{W}\boldsymbol{V}^T$
    $\hat{\boldsymbol{F}} \leftarrow \frac{1}{2}\text{Proj}_{\widehat{\langle \boldsymbol{Q} \rangle}}(\boldsymbol{X}) = \frac{1}{2}\boldsymbol{X}\boldsymbol{V}_{(1:d)}\boldsymbol{V}_{(1:d)}^T$      ▷ (ii)
    $(\hat{\boldsymbol{P}}, \hat{\boldsymbol{Q}}) \leftarrow \text{uALS}(\hat{\boldsymbol{F}}, d)$            ▷ (iii)
return $(\hat{\boldsymbol{F}}, \hat{\boldsymbol{P}}, \hat{\boldsymbol{Q}})$
```

(LLS) solutions in the uALS algorithm. Both of these problems have a rich history and consequently have many efficient algorithms. It is likely that the ALStructure implementation of Algorithm 3 can be significantly sped up by utilizing approximate or randomized algorithms for the eigendecomposition and/or LLS computations. In its current form, ALStructure simply uses the base R functions eigen() and solve() for the eigendecomposition and LLS computations respectively. Despite this, the current implementation of ALStructure is typically faster than existing algorithms as can be seen in Sections 3 and 4 below.

For choosing the dimensionality of the model d, we recommend utilizing the recently proposed structural Hardy-Weinberg equilibrium (sHWE) test (Hao and Storey, 2017). This test can perform a genomewide goodness of fit test to the assumptions made in the admixture model over a range of d. It then identifies the minimal value of d that obtains the optimal goodness of fit. There are other ways to choose d, such as by using the theory and methods in Chen and Storey (2015) or by using other recent proposals (Hao *et al.*, 2016; Patterson *et al.*, 2006).

3 Results from simulated data

3.1 Simulated data sets

In this section we compare the performance of ALStructure to three existing methods for global ancestry estimation, Admixture, fastSTRUCTURE and terastructure. Admixture, developed by Alexander *et al.* (2009), is a popular algorithm which takes a maximum-likelihood approach to fit the admixture model. Both fastSTRUCTURE (Raj *et al.*, 2014) and terastructure (Gopalan *et al.*, 2016) are Bayesian methods that fit the PSD model using variational Bayes approaches. We abbreviate these methods as ADX, FS, and TS in the plots. A comparison among these three methods appears in Gopalan *et al.* (2016), so we will focus on how they compare to ALStructure.

To this end, we first tested all algorithms on a diverse array of simulated datasets. The bulk of our simulated data sets come from the classical PSD model (defined in Section 2.1) in which columns of $\boldsymbol{Q}$ are distributed according to the Dirichlet($\boldsymbol{\alpha}$) distribution and the rows of $\boldsymbol{P}$ are drawn from the

m	10^5, 5×10^5
n	5×10^2, 10^3, 5×10^3, 10^4
d	3, 6, 9
$\boldsymbol{\alpha}$-prototypes	$(10, 10, 10)$ $(1, 1, 1)$ $(0.1, 0.1, 0.1)$ $(10, 1, 0.1)$

Table 1: Parameters of all simulated datasets

Balding-Nichols distribution. We varied the following parameters in our simulated datasets: m, n, d, and $\boldsymbol{\alpha}$. Of particular note is the variation of $\boldsymbol{\alpha}$. For this we used four $\boldsymbol{\alpha}$-prototypes: $\boldsymbol{\alpha}_1 = (10, 10, 10)$, $\boldsymbol{\alpha}_2 = (1, 1, 1)$, $\boldsymbol{\alpha}_3 = (0.1, 0.1, 0.1)$, and $\boldsymbol{\alpha}_4 = (10, 1, 0.1)$. These four prototypes were chosen because they represent four qualitatively different distributions on the Dirichlet simplex as shown in Fig. 3: i) $\boldsymbol{\alpha}_1$ corresponds to points distributed near the center of the simplex, ii) $\boldsymbol{\alpha}_2$ corresponds to points distributed evenly across the simplex, iii) $\boldsymbol{\alpha}_3$ corresponds to points distributed along the edges of the simplex, and iv) $\boldsymbol{\alpha}_4$ corresponds to an asymmetric distribution in which points are concentrated around one of the corners of the simplex.

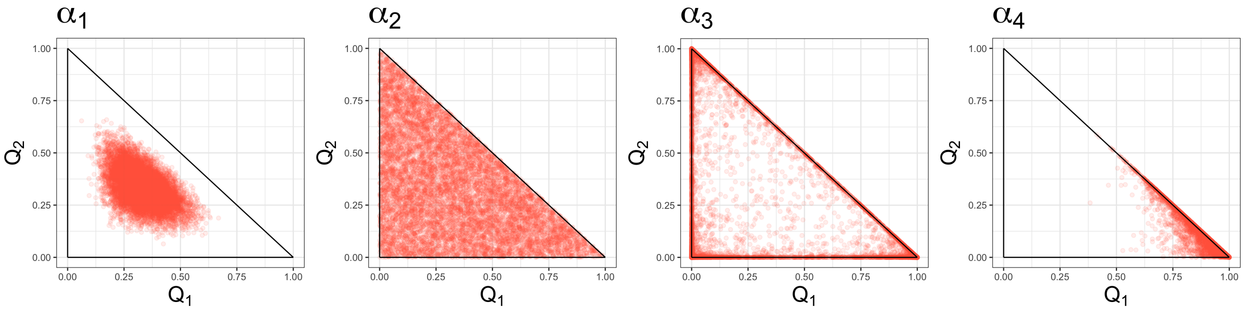


Figure 3: Examples of typical random samples from the four different $\boldsymbol{\alpha}$-prototypes. As can be seen, only $\boldsymbol{\alpha}_2$ and $\boldsymbol{\alpha}_3$ approximately obey the “anchor-individuals" condition.

When we produced datasets with $d > 3$, we extended the prototypes in the natural way; for example for $d = 6$, the $\boldsymbol{\alpha}_4$ is becomes $(10, 10, 1, 1, 0.1, 0.1)$. Table 1 lists all of the parameters we used to generate data under the Dirichlet model, for a total of 96 distinct combinations.

The parameters of the Balding-Nichols distributions from which rows of the $\boldsymbol{P}$ matrix were drawn were taken from real data, following the same strategy taken in GOPALAN *et al.* (2016). Specifically, F_i and p_i were estimated for each SNP in the Human Genome Diversity Project (HGDP) dataset (CAVALLI-SFORZA, 2005). Then for each simulated dataset, m random samples are taken (with replacement) from the HGDP parameter estimates.

In addition to simulating $\boldsymbol{Q}$ matrices from the classical Dirichlet$(\boldsymbol{\alpha})$ distribution with many different parameters $\boldsymbol{\alpha}$, we also simulated data from the a spatial model of admixture developed in OCHOA and STOREY (2016). We deliberately chose to study this model because it is ill-suited for ALStructure; while ALStructure relies on the estimation of the d-dimensional linear subspace $\langle \boldsymbol{Q} \rangle$, the columns of $\boldsymbol{Q}$ produced under the spatial model lie on a one-dimensional curve within $\langle \boldsymbol{Q} \rangle$. Despite this fundamentally challenging scenario, we see that ALStructure is still often capable of recovering an accurate approximation.

3.2 Results from the PSD model

In order to give a representative picture of the relative performance of ALStructure against existing algorithms, we first plot the fits of all of the algorithms for two particular data sets out of the total 96 model data sets: (i) the data set in which ALStructure performs the best and (ii) the data set in which ALStructure performs the worst. We measure the quality of fit as the mean absolute error between the $\hat{\boldsymbol{Q}}$ and $\boldsymbol{Q}$:

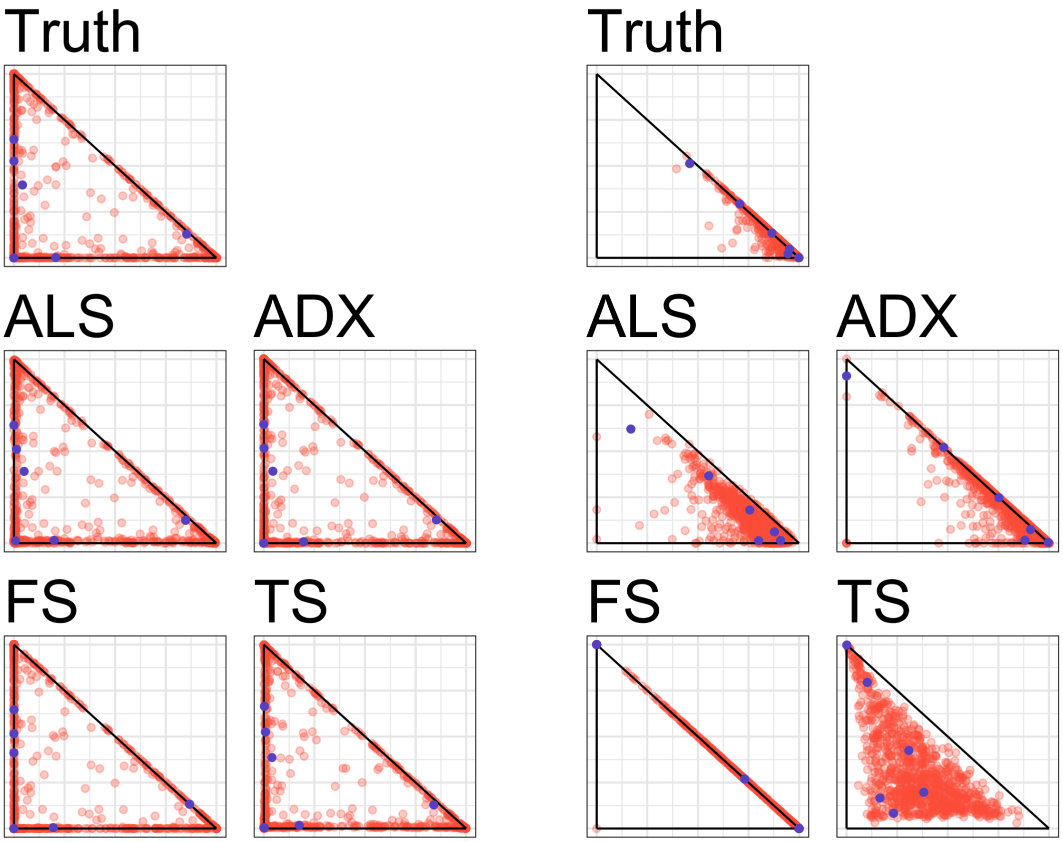
$$\frac{1}{dn} \sum_{k=1}^{d} \sum_{j=1}^{n} |\hat{q}_{kj} - q_{kj}|$$

In Fig. 4a, we see that all four algorithms perform very well for the data set in which ALStructure performs best, which comes from the $\boldsymbol{\alpha}_1$-prototype. In Fig. 4b, the dataset was generated from the $\boldsymbol{\alpha}_4$-prototype. We see that while ALStructure certainly deviates substantially from the truth, so does every algorithm. Both fastSTRUCTURE and terastructure provide results that are qualitatively very different from the truth; where fastSTRUCTURE compresses all columns of $\boldsymbol{Q}$ onto a single edge of the simplex, terastructure spreads them out through the interior of the simplex. Both Admixture and ALStructure provide solutions qualitatively similar to the truth. While the points in the Admixture solution extend much further along the edge of the simplex than the true model, the ALStructure solution spreads into the interior of the simplex more than the true model.

Fig. 8a and 8b summarize the performance of ALStructure against the existing algorithms on all simulated datasets. Fig. 8a shows the distributions of quality of fit (measured by the mean absolute difference between $\boldsymbol{Q}$ and $\hat{\boldsymbol{Q}}$) for each algorithm on all modeled datasets. Fig. 8b shows the distributions of run times for each algorithm on all modeled datasets. It is clear that ALStructure is competitive with respect to both model fit and time. While Admixture is closest to ALStructure in accuracy, terastructure is closest to ALStructure in speed; no single competitor algorithm is competitive with ALStructure in both accuracy and time. Furthermore, ALStructure has both the smallest median error and run time among all methods.

3.3 Results from the spatial model

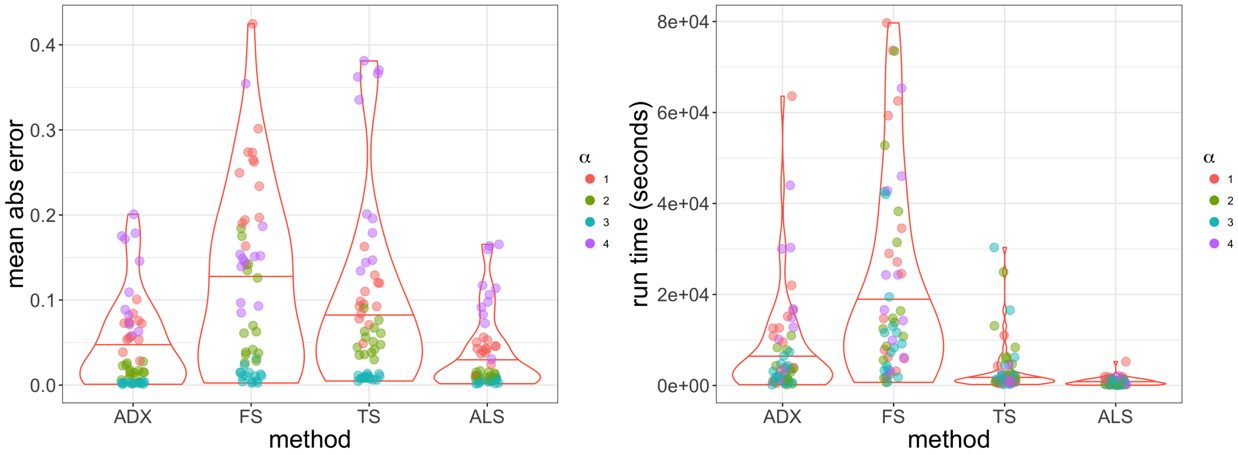
As a challenge to ALStructure, we simulate data from a model developed in OCHOA and STOREY (2016), which we will refer to as the *spatial model*. This model mimics an admixed population that was generated by a process of diffusion in a one-dimensional environment. There are d unmixed ancestral populations equally spaced at positions $\{x_0, x_0 + 1, \ldots, x_0 + d - 1\}$ on an infinite line. If all populations



(a) Dataset with best `ALStructure` fit.

(b) Dataset with worst `ALStructure` fit.

Figure 4: Model fits by `ALStructure`, `Admixture`, `fastSTRUCTURE`, `terastructure` on the two particular simulated datasets. Each point represents a column of the Q matrix and is plotted by the first and second coordinates. Blue points are plotted as a visual aid and delineate a common subset of individuals.



(a) Comparison of quality of fit

(b) Comparison of run time

Figure 5: Summary of performance of ALStructure and existing algorithms. The points are colored by $\boldsymbol{\alpha}$-prototype.

begin to diffuse at time $t = 0$ at the same diffusive rate, then population i will be distributed as a Gaussian with mean $\mu_i = x_0 + i - 1$ and standard deviation σ. Therefore, under the Spatial model, an individual sampled from position x will have admixture proportions:

$$(q_1(x), q_2(x), \ldots, q_d(x)) = \left(\frac{f_{(\mu_1,\sigma)}(x)}{\sum_{i=1}^{d} f_{(\mu_i,\sigma)}(x)}, \frac{f_{(\mu_2,\sigma)}(x)}{\sum_{i=1}^{d} f_{(\mu_i,\sigma)}(x)}, \ldots, \frac{f_{(\mu_d,\sigma)}(x)}{\sum_{i=1}^{d} f_{(\mu_i,\sigma)}(x)} \right) \tag{20}$$

where $f_{(\mu,\sigma)}$ denotes the Gaussian distribution with parameters (μ, σ).

Although this is just a special case of the admixture model, one would expect the spatial model to be particularly challenging for ALStructure because the admixture proportions belong to a one-dimensional curve parameterized by x, and ALStructure necessitates the estimation of a d-dimensional linear subspace in $\mathbb{R}^n$. The challenge is much more pronounced when the populations are highly admixed (large σ). Fig. 6 shows the model fits provided by ALStructure. Indeed, for large values of σ ($\sigma = 2$), ALStructure fails to correctly capture the admixture proportions. However, for smaller values of σ ($\sigma = \{1, 0.5\}$), it can be seen that the fits provided by ALStructure are excellent. In all simulations $m = 10^5$, $n = 10^3$, and $d = 3$.

We note that Gopalan *et al.* (2016) tested Admixture, fastSTRUCTURE, and terastructure on data drawn from the spatial model (which they refer to as "Scenario B"). They found this model to pose a significant challenge for all three methods, but found terastructure to perform the best.

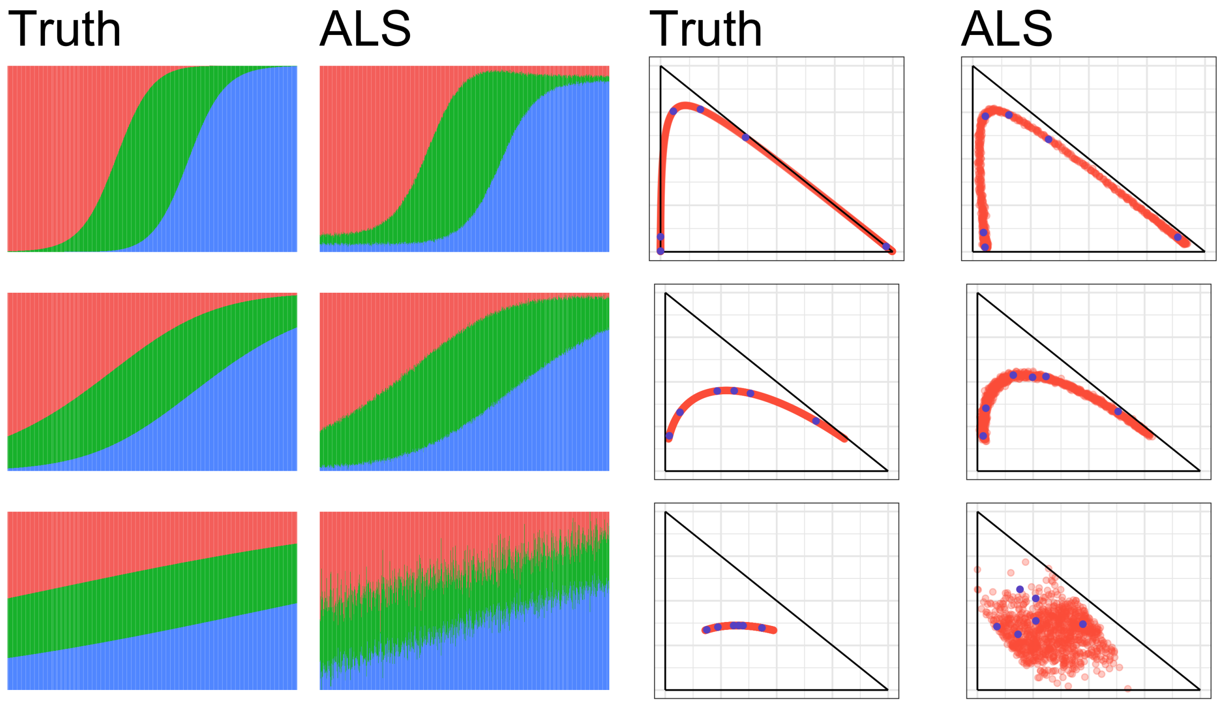


Figure 6: `ALStructure` of datasets from the Spatial model. (left) Stacked barplots of `ALStructure` fits. (right) Bi-plots of `ALStructure` fits. The parameter σ was set to 2, 1, and 0.5 for the top, middle and bottom rows, respectively. Blue points are plotted as a visual aid and delineate corresponding columns of Q and $\hat{Q}$.

4 Applications to global human studies

Here we apply ALStructure and existing methods to three globally sampled human genotype datasets: the Thousand Genomes Project (TGP), Human Genome Diversity Project (HGDP), and Human Origins (HO) datasets (CAVALLI-SFORZA, 2005; LAZARIDIS *et al.*, 2014; THE 1000 GENOMES PROJECT CONSORTIUM, 2015). Table 2 summarizes several basic parameters of each of the datasets and Appendix A details the procedures used for building each dataset. Although we recommend using sHWE from HAO and STOREY (2017) for choosing d, here we take directly from GOPALAN *et al.* (2016) the number of ancestral populations d so that our results can be compared to those.

Dataset	m	n	d	$m \times n$
TGP	520036	1716	8	$\sim 8.9 \times 10^8$
HGDP	550303	940	10	$\sim 5.2 \times 10^8$
HO	372446	2248	14	$\sim 8.4 \times 10^8$

Table 2: Dataset parameters.

Fig. 7 shows scatterplots of the first two rows of $\hat{\boldsymbol{Q}}$ for each of the three datasets provided by each of the four fits. To disambiguate the inherent non-identifiability (see section 2.5), we ordered the rows of the fits $\hat{\boldsymbol{Q}}$ by decreasing variation explained: $s_i^2 = ||\boldsymbol{X}\hat{\boldsymbol{q}}_{i\cdot}^T||^2$. Perhaps the most striking aspect of Fig. 7 is the difference between the fits produced by each method. With the notable exception that Admixture and ALStructure have similar fits for the TGP and HGDP datasets, every pair of comparable scatterplots (i.e., within a single row of Fig. 7) are qualitatively different.

Next we compare the performance of ALStructure to existing methods both in terms of efficiency and accuracy. Unlike in the case of simulated datasets where the ground truth is known, here we cannot directly compare the quality of model fits across methods. Instead, we assess the quality of each method by its performance on data simulated from real data fits. For concreteness, we briefly outline the process below:

(i) Fit each dataset with each of the four methods to obtain 12 model fits.

(ii) Simulate datasets from the admixture model using parameters obtained in the previous step.

(iii) Fit each of the 12 simulated datasets with each of the four datasets (48 fits) and compute error measures.

The process above treats each of the four methods symmetrically, evaluating each method based on its ability to fit data simulated from both its own model fits as well as every other methods' model fits.

Fig. 8a shows that every method performs similarly on datasets simulated from real data fits, however it is notable that ALStructure has the smallest error in terms of minimum, median, and maximum. Fig. 8b demonstrates that ALStructure performs significantly faster than preexisting methods

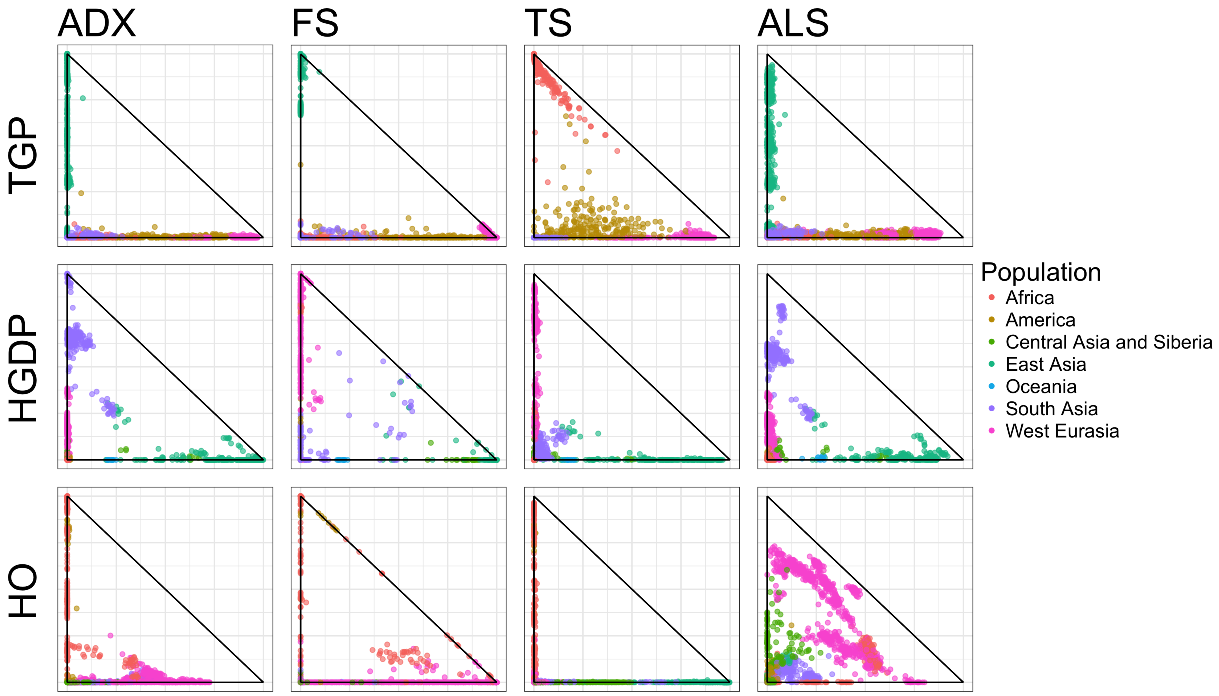
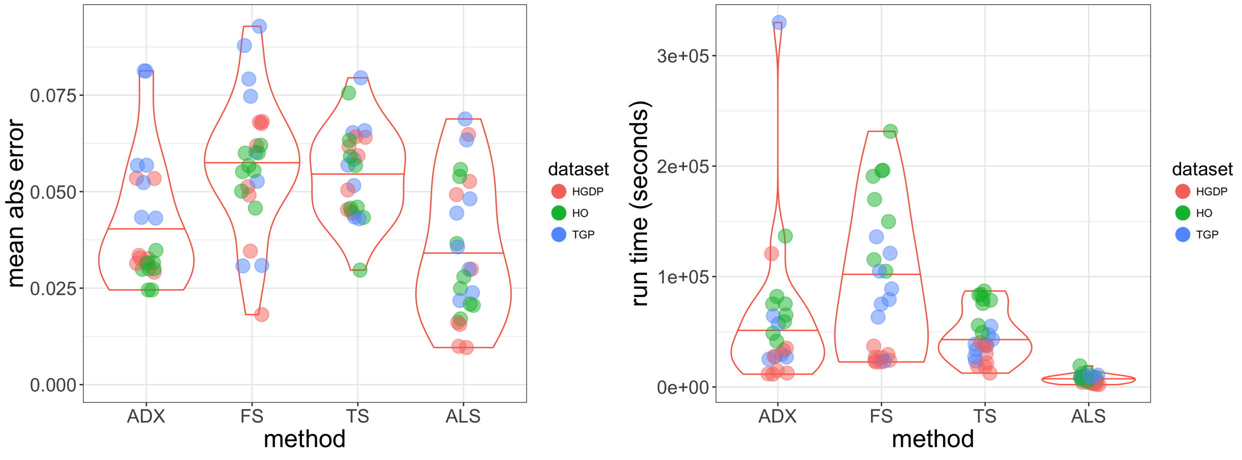


Figure 7: Bi-plots of the first two rows of Q (ranked by variation explained) of the fits of the TGP (top), HGDP (middle), and HO (bottom) datasets for each algorithm. Individuals are colored by coarse subpopulation from which they are sampled.



(a) Comparison of quality of fit

(b) Comparison of run time

Figure 8: Summary of performance of `ALStructure` and preexisting algorithms on datasets simulated from the fits of real datasets. Horizontal bar represents median value.

5 Discussion

In this work we have introduced ALStructure, a new method to fit the admixture model from observed genotypes. Our method attempts to find common ground between two previously distinct approaches to understanding genetic variation: likelihood-based approaches and PCA-based approaches. ALStructure features important merits from both. Like the likelihood-based approaches, ALStructure is grounded in the probabilistic admixture model and provides full estimates of global ancestry. However, operationally the ALStructure method closely resembles PCA-based approaches. In particular, ALStructure's estimates of global ancestry are derived from a PCA-based estimate of the underlying low-dimensional latent subspace of the data. In this way, ALStructure can be considered a unification of likelihood-based and PCA-based methods.

Because ALStructure is operationally similar to PCA-based methods, it is computationally efficient. Specifically, the only computationally expensive operations required by the ALStructure algorithm are singular value and QR decompositions. Both of these computations have been extensively studied and optimized. Although ALStructure already performs favorably compared to preexisting algorithms in computational efficiency, it is likely that by applying more sophisticated eigendecomposition techniques ALStructure may see significant improvements in speed.

The high-level strategy of ALStructure is to take advantage of the inherent nonidentifiability of the admixture model to obviate many computationally expensive procedures of likelihood-based approaches. As such, rather than searching for over a rough likelihood landscape for optimal solutions, the ALStructure algorithm is based on a pursuit of feasible solutions subject to a set of optimal constraints. It is a simple, but central observation of the strategy of ALStructure that any element of the derived feasible set is a risk-minimizing solution up to the inherent non identifiability of the admixture model; consequently, any further optimization is unnecessary. Perhaps more important than its appealing theoretical and computational properties, ALStructure typically outperforms preexisting algorithms both in terms of accuracy and time. This observation holds under a wide array datasets, both simulated and real.

Finally, the basic approach is quite general. In particular, the set of models that satisfy the underlying assumptions of LSE is large, subsuming the admixture model as well as many other probabilistic models with low intrinsic dimensionality. Consequently, we expect that the ALStructure method can be trivially altered to apply to many similar problems beyond the estimation of global ancestry.

Acknowledgements

This research was supported in part by NIH grant HG006448.

Software

An R package implementing the method proposed here is available at https://github.com/StoreyLab/ALStructure.

References

ALEXANDER, D. H., J. NOVEMBRE, and K. LANGE, 2009 Fast model-based estimation of ancestry in unrelated individuals. Genome Research **19**: 1655–1664.

ARORA, S., R. GE, Y. HALPERN, D. MIMNO, A. MOITRA, *et al.*, 2013 A practical algorithm for topic modeling with provable guarantees. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*. PMLR, Atlanta, Georgia, USA, 280–288.

BALDING, D. J., and R. A. NICHOLS, 1995 A method for quantifying differentiation between populations at multi-allelic loci and its implications for investigating identity and paternity. Genetica **96**: 3–12.

BERRY, M. W., M. BROWNE, A. N. LANGVILLE, V. P. PAUCA, and R. J. PLEMMONS, 2007 Algorithms and applications for approximate nonnegative matrix factorization. Computational Statistics & Data Analysis **52**: 155–173.

BOYD, S., and L. VANDENBERGHE, 2009 *Convex Optimization*. Cambridge University Press.

BRISBIN, A., K. BRYC, J. BYRNES, F. ZAKHARIA, L. OMBERG, *et al.*, 2012 PCAdmix: Principal components-based assignment of ancestry along each chromosome in individuals with admixed ancestry from two or more populations. Human Biology **84**: 343–364.

CAVALLI-SFORZA, L. L., 2005 The human genome diversity project: past, present and future. Nature Reviews Genetics **6**: 333–340.

CAVALLI-SFORZA, L. L., A. PIAZZA, P. MENOZZI, and J. MOUNTAIN, 1988 Reconstruction of human evolution: Bringing together genetic, archaeological, and linguistic data. Proceedings of the National Academy of Sciences **85**: 6002–6006.

CHEN, X., and J. D. STOREY, 2015 Consistent Estimation of Low-Dimensional Latent Structure in High-Dimensional Data. ArXiv e-prints .

CHEN, Y., and X. YE, 2011 Projection Onto A Simplex. ArXiv e-prints .

ESTEBAN, J. P., A. MARCINI, J. AKEY, J. MARTINSON, M. A. BATZER, *et al.*, 1998 Estimating african american admixture proportions by use of population specific alleles. The American Journal of Human Genetics **63**: 839–851.

GOPALAN, P., W. HAO, D. M. BLEI, and J. D. STOREY, 2016 Scaling probabilistic models of genetic variation to millions of humans. Nature Genetics **48**: 1587–1592.

GRIPPO, L., and M. SCIANDRONE, 2000 On the convergence of the block nonlinear gauss-seidel method under convex constraints. Computational Statistics and Data Analysis **26**: 127–136.

HAO, W., M. SONG, and J. D. STOREY, 2016 Probabilistic models of genetic variation in structured populations applied to global human studies. Bioinformatics **32**: 713–721.

HAO, W., and J. D. STOREY, 2017 Extending a test of hardy-weinberg equilibrium to structured populations. bioRxiv .

JOLLIFFE, I. T., 2002 *Principal component analysis*. Springer Verlag.

KNOWLER, W. C., R. C. WILLIAMS, D. J. PETTITT, and A. G. STEINBERG, 1988 Gm3;5,13,14 and type 2 diabetes mellitus: an association in american indians with genetic admixture. The American Journal of Human Genetics **43**: 520–526.

LAZARIDIS, I., *et al.*, 2014 Ancient human genomes suggest three ancestral populations for present-day europeans. Nature **513**: 409–413.

LI, J. Z., D. M. ABSHER, H. TANG, A. M. SOUTHWICK, A. M. CASTO, *et al.*, 2008 Worldwide human relationships inferred from genome-wide patterns of variation. Science **319**: 1100–1104.

MARCHINI, J., L. R. CARDON, M. S. PHILLIPS, and P. DONNELLY, 2004 The effects of human population structure on large genetic association studies. Nature Genetics **36**: 512–517.

OCHOA, A., and J. D. STOREY, 2016 F_{ST} and kinship for arbitrary population structures II: Method of moments estimators. bioRxiv .

PAATERO, P., and U. TAPPER, 1994 Positive matrix factorization: a non-negative factor model with optimal utilization of error estimates of data values. Environmetrics **5**: 111–126.

PATTERSON, N., A. PRICE, and D. REICH, 2006 Population structure and eigenanalysis. PLoS Genetics **2**: e190.

PRICE, A., N. PATTERSON, R. PLENGE, M. WEINBLATT, N. SHADICK, *et al.*, 2006 Principal components analysis corrects for stratification in genome-wide association studies. Nature Genetics **38**: 904–909.

PRICHARD, J. K., M. STEPHENS, and P. DONNELLY, 2000 Inference of population structure using multilocus genotype data. Genetics **155**: 945–959.

RAJ, A., M. STEPHENS, and J. K. PRITCHARD, 2014 fastSTRUCTURE: Variational inference of population structure in large SNP data sets. Genetics **197**: 573–589.

SONG, M., W. HAO, and J. D. STOREY, 2015 Testing for genetic associations in arbitrarily structured populations. Nature Genetics **47**: 550–554.

TANG, H., J. PENG, P. WANG, and N. RISCH, 2005 Estimation of individual admixture: Analytical and study design considerations. Genet Epidemiol **28**: 289–301.

THE 1000 GENOMES PROJECT CONSORTIUM, 2015 A global reference for human genetic variation. Nature **526**: 68–74.

WEIR, B. S., and C. C. COCKERHAM, 1984 Estimating F-statistics for the analysis of population structure. Evolution **38**: 1358–1370.

ZHENG, X., and B. S. WEIR, 2016 Eigenanalysis of SNP data with an identity by descent interpretation. Theoretical Population Biology **107**: 65–76.

Appendices

A HGDP, TGP, and HO dataset details

In Section 4 we analyze human genotype data from globally-sampled individuals. These data come from three public sources: HGDP (Cavalli-Sforza, 2005), TGP (The 1000 Genomes Project Consortium, 2015), HO (Lazaridis *et al.*, 2014). The various pre-processing steps are detailed below for each dataset:

TGP: The 1000 Genomes Project dataset (TGP) samples globally from 26 populations and is available here: `ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/supporting/hd_genotype_chip/`. Related individuals and SNPs with minor allele frequency $< 5\%$ are removed. The dimensions of this dataset are 1,716 individuals and 520,036 SNPs.

HGDP: The Human Genome Diversity Project dataset (HGDP) samples globally from 51 populations and is available here: `http://www.hagsc.org/hgdp/files.html`. Individuals with first- or second-degree relatives and SNPs with minor allele frequency $< 5\%$ are removed. The dimensions of this dataset are 940 individuals and 550,303 SNPs.

HO: The Affymetrix Human Origins dataset (HO) samples globally from 147 populations and is available here: `http://genetics.med.harvard.edu/reich/Reich_Lab/Datasets.html`. Nonhuman or ancient samples and SNPs with $< 5\%$ minor allele frequency are removed. The dimensions of this dataset are 2,248 individuals and 372,446 SNPs.

B Proof of sufficiency of anchors

Here we show that either a set of anchor SNPs or a set of anchor individuals is sufficient to specify a unique factorization $\boldsymbol{F} = \boldsymbol{PQ}$ up to the non identifiability associated with row permutations.

Proposition 1.

For a rank d matrix $\boldsymbol{F}$ with a factorization $\boldsymbol{F} = \boldsymbol{PQ}$, if there is a set S of d rows of $\boldsymbol{P}$ such that for each $i \in \{1, 2, \ldots, d\}$ there exists a row vector $\boldsymbol{p}_{i\cdot} \in S$ such that $\boldsymbol{p}_{i\cdot} = \delta_i \mathbf{e}_i$ for $\delta_i \neq 0$, then the factorization is unique up to permutation. When such a set S exists, we say that we have “anchor SNP’s."

Proof. Let us denote the matrix $\boldsymbol{D} = \text{diag}(\delta_1, \delta_2, \ldots, \delta_d)$. Without loss of generality, let us assume that S is the first d rows of P and are ordered such that

$$\boldsymbol{P} = \begin{pmatrix} \boldsymbol{D} \\ \boldsymbol{A} \end{pmatrix}$$

for some $(m-d)\times d$ matrix $\boldsymbol{A}$. Then there is a unique $\boldsymbol{Q}$ for this $\boldsymbol{F}$ matrix (up to permutation) which is

$$\boldsymbol{Q} = \boldsymbol{D}^{-1}\boldsymbol{F}_{1:d}$$

The matrix $\boldsymbol{A}$ is also uniquely determined by $\boldsymbol{F}$ once $\boldsymbol{Q}$ is fixed. To see this, note that

$$\boldsymbol{f}_{j\boldsymbol{\cdot}} = \boldsymbol{p}_{j\boldsymbol{\cdot}}\boldsymbol{Q}$$

where $\boldsymbol{f}_{j\boldsymbol{\cdot}}$ and $\boldsymbol{p}_{j\boldsymbol{\cdot}}$ denote the j row of $\boldsymbol{F}$ and $\boldsymbol{P}$ respectively. Since $\boldsymbol{f}_{j\boldsymbol{\cdot}}$ is fixed and $\boldsymbol{Q}$ is unique under the anchor SNP assumption, there is a unique solution for $\boldsymbol{p}_{j\boldsymbol{\cdot}}$ by the linear independence of the rows of $\boldsymbol{Q}$. □

The interpretation of the anchor SNP's assumption is that every ancestral population has at least one SNP that appears only in it. Presence of such an SNP is therefore a guarantee that the individual is a member of a particular population. Note that an identical argument could be made when we have a set S of d columns of $\boldsymbol{Q}$ that have exactly one nonzero entry at unique locations. When such a set exists, we say that we have "*anchor individuals*." Under the admixture model, the simplex constraint requires that the nonzero entry of each anchor genotype is exactly one. In this scenario, there exists at least one individual from each ancestral population whose entire genome was inherited by a single ancestral population. We summarize these results in the following corollary and visualize the anchor SNP and anchor genotype scenarios in Fig. 1.

Corollary 1. *Whenever a rank d matrix $\boldsymbol{F}$ admits a factorization $\boldsymbol{F} = \boldsymbol{PQ}$ such that there are either a set of anchor SNP's or a set of anchor genotypes, the factorization is unique up to permutation.*