aBayesQR: A Bayesian method for reconstruction of viral populations characterized by low diversity

Soyeon Ahn* and Haris Vikalo

The University of Texas at Austin, Austin TX, USA,
soyeon.ahn@utexas.edu, hvikalo@ece.utexas.edu,

Abstract. RNA viruses replicate with high mutation rates, creating closely related viral populations. The heterogeneous virus populations, referred to as viral quasispecies, rapidly adapt to environmental changes thus adversely affecting efficiency of antiviral drugs and vaccines. Therefore, studying the underlying genetic heterogeneity of viral populations plays a significant role in the development of effective therapeutic treatments. Recent high-throughput sequencing technologies have provided invaluable opportunity for uncovering the structure of quasispecies populations (i.e., reconstruction of viral sequences and discovery of their relative frequencies). However, accurate reconstruction of viral quasispecies remains difficult due to limited read-lengths and presence of sequencing errors. The problem is particularly challenging when the strains in a population are highly similar, i.e., the sequences are characterized by low mutual genetic distances, and further exacerbated if some of those strains are relatively rare; this is the setting where state-of-the-art methods struggle. In this paper, we present a novel viral quasispecies reconstruction algorithm, aBayesQR, that employs a maximum-likelihood framework to infer individual sequences in a mixture from high-throughput sequencing data. The search for the most likely quasispecies is conducted on long contigs that our method constructs from the set of short reads via agglomerative hierarchical clustering; operating on contigs rather than short reads enables identification of close strains in a population and provides computational tractability of the Bayesian method. Results on both simulated and real HIV-1 data demonstrate that the proposed algorithm generally outperforms state-of-the-art methods; aBayesQR particularly stands out when reconstructing a set of closely related viral strains (e.g., quasispecies characterized by low diversity).

Keywords: viral quasispecies reconstruction, low diversity, bayesian method

1 Introduction

A number of potentially life-threatening infectious diseases are caused by RNA viruses, including human immunodeficiency virus (HIV), hepatitis C virus (HCV), influenza and Ebola. RNA viruses have a relatively high mutation rate due to both their error-prone replication process and the lack of sophisticated repair mechanisms [1]. Consequently, they rapidly evolve and exist as a set of non-identical but closely related genetic variants, known as a viral quasispecies. Viral populations can readily adapt to dynamic environments and develop resistance to antiviral drugs and vaccines, which makes the design of effective and long-lasting treatments for RNA viral diseases exceedingly difficult [2]. Determining the structure of viral populations helps the understanding of viral diseases and provides guidance in the development of effective medical therapeutics. Quasispecies spectrum reconstruction (QSR) aims to assemble individual haplotype sequences in a population and estimate their prevalence using sequencing reads generated from a sample containing a set of viral variants. High-throughput next-generation sequencing (NGS) technologies have enabled affordable acquisition of data needed to assemble quasispecies. However, relatively short length of the NGS reads and the presence of errors in sequencing data render the QSR problem difficult. The QSR problem is particularly challenging when the strains in a viral population are highly similar, i.e., the sequences are characterized by low mutual genetic distances, and further exacerbated if some of those strains are relatively rare [3].

Several software tools for solving the QSR problem by analyzing NGS data have been developed in recent years. ShoRAH [4], the earliest publicly available such software, was developed by combining a path cover based approach and probabilistic clustering in [5] and [6], respectively, and applied to analysis of HIV data [7]. Read-graph approach was the basis for ViSpA [8], developed as a variant of the network flow method proposed in [9]. [10], proposed a combinatorial method for QSR and the resulting software, QuRe, was provided by [11]. An approach that resulted in the software package PredictHaplo [12] relied on a Dirichlet Process mixture model and was developed specifically targeting HIV population reconstruction; QuasiRecomb [13] is based on a hidden Markov model that explicitly models recombination events. In [14], a benchmarking study that compares the performance of several publicly available quasispecies reconstruction softwares was presented. The study demonstrated that none of the tested methods could reconstruct populations characterized by low pairwise distance between the haplotype sequences. Following this study other softwares, including HaploClique [15], based on max-clique enumeration of a read alignment graph, and VGA [16], a graph-coloring based heuristic method, were developed. Most recently, a reference-assisted *de novo* assembly pipeline, ViQuaS, was proposed in [17]. ViQuaS extends an existing algorithm, QuRe [10], and outperforms various other techniques on a wide range of dataset. However, performance of these more recent methods deteriorates dramatically in the scenarios where the genetic diversity of a population is low [3].

Both [3] and [14] have pointed out that the existing methods for viral quasispecies reconstruction struggle in the scenarios where the populations are characterized by low diversity. This, in part, is due to the presence of relatively long genetic regions that are common to pairs of closely related viral sequences; clearly, this makes distinguishing different strains challenging. The problem becomes even more difficult when the frequency of one (or more) of the close strains is low; in such settings small genetic distances may be confused for sequencing errors and hence remain undetected. Such failures to detect may have serious consequences in antiviral treatment studies since undetected strains cannot be properly targeted for drug and vaccine design. It has been shown that even the viral strains existing at low frequencies can cause a drug treatment failure due to their resistance to the drug [18, 19]. Therefore, complete recovery of the composition of viral populations is of critical importance for effective antiviral therapies.

In this paper, we propose a novel QSR algorithm, aBayesQR (combining agglomerative hierarchical clustering and Bayesian inference), that overcomes limitations of the existing methods and reliably reconstructs quasispecies characterized by low diversity. The algorithm performs reconstruction of a quasispecies from next-generation sequencing (NGS) data in two stages. In the first stage, conflict-free short reads are hierarchically merged and assembled into longer sequences (contigs) which we refer to as super-reads. In the second stage, likelihoods of the probable quasispecies are computed using the assembled super-reads (rather than using the original set of short reads), and the most likely set of viral strains is selected. Note that the super-reads synthesized in the first stage of aBayesQR allow us to distinguish between closely related strains which share long genetic regions as well as reduce the search space and enable computational tractability of the Bayesian inference conducted in the second stage. The second stage of aBayesQR involves sequential pruning of the solution space; in particular, the likely set of partial viral strains comprising n single nucleotide variants (SNVs) is generated by extending previously inferred partial viral strains having $n-1$ SNVs. The number of sequences in a set (i.e., the size of a viral population) is dynamically updated at each step by evaluating quality of the set of partially reconstructed viral strains, and ultimately precisely inferred at the end of the search process. The relative frequencies of each strain are determined by counting the numbers of reads unambiguously associated with each of the reconstructed strains. Our tests on both simulated and experimental data demonstrate superior performance compared to state-of-the-art methods for quasispecies reconstruction. In particular, it is shown that unlike the competing methods, aBayesQR is capable of detecting and reliably reconstructing viral haplotypes having very small mutual genetic distances.

2 Proposed Method

Our algorithm for inferring spectrum of a viral population consists of the following two steps: (1) constructing super-reads by hierarchically clustering aligned

paired-end reads, (2) inferring the most likely quasispecies from the set of super-reads and estimating the frequencies of the strains in the quasispecies.

2.1 Super-reads construction via agglomerative clustering

In the first stage of aBayesQR, paired-end reads uniquely mapped to a reference genome are grouped into super-reads via agglomerative hierarchical clustering. This is facilitated by a weighted graph $G = (\mathcal{V}, \mathcal{E})$ which is constructed and recursively updated as the clustering proceeds. In particular, each vertex of G is associated with a cluster collecting reads that originated from a single strain in a quasispecies; we denote the set of reads in the ith cluster (i.e., the cluster associated with the ith vertex) as $V_i = \{v_i^j, j = 1, \cdots, |V_i|\}$. Let sr_i denote a consensus sequence (i.e., a super-read) constructed from the reads in V_i. The ith and j^{th} vertex of G are connected by an edge $e_{ij} \in \mathcal{E}$ if all the reads in V_i and V_j (or, equivalently, sr_i and sr_j) are conflict-free and an overlap criterion, specified later in this subsection, is satisfied. The weight w_{ij} of the edge e_{ij} is a measure of similarity between V_i and V_j at each step, the algorithm merges a pair of vertices connected by the edge having the largest weight to form a new vertex and agglomerates the corresponding clusters.

The alleles at homozygous sites, common to all the components of a quasispecies, are not utilized in the reconstruction procedure. Instead, we separate reads having originated from different strains by clustering them using heterogeneous sites with reliable SNV information. An SNV information is considered reliable if the relative abundance of the allele is above a pre-determined threshold, as in ([20]); alleles whose abundance is below the threshold are treated as sequencing errors and disregarded in the process of clustering. For convenience, let us denote the set of pre-processed paired-end reads by $R = \{r_i, i = 1, \cdots, |R|\}$. The agglomerative clustering is initialized with $|R|$ clusters, one for each read; in other words, we start with $V_1 = r_1, \cdots, V_{|R|} = r_{|R|}$, implying that $|\mathcal{V}| = \sum_{i=1}^{|\mathcal{V}|} |V_i| = |R|$, and proceed by sequentially merging judiciously chosen pairs of vertices (i.e., agglomerating the corresponding clusters). Intuitively, it is meaningful to reduce the number of vertices in the graph by merging those associated with conflict-free consensus sequences that have a large overlap. To formalize this, let $L_i = \{l_1, \cdots, l_{|L_i|}\}$ denote an index set of the SNV positions covered by sr_i, let $L_{i\cap j} = \{l_1, \cdots, l_{|L_{i\cap j}|}\}$ be the index set of SNV positions covered by both sr_i and sr_j, and let $L_{i\cup j} = \{l_1, \cdots, l_{|L_{i\cup j}|}\}$ be the index set of SNV positions covered by either sr_i or sr_j. Then the pairs of vertices (i, j) that we consider as candidates for merging and thus connect by an edge are those satisfying either

$$|L_{i\cap j}| \geq \theta \cdot |L_{i\cup j}| \quad \text{or} \quad |L_{i\cap j}| = \min(|L_i|, |L_j|),$$

where the 2nd condition promotes merger of short super-reads, and the choice of θ is discussed below. To quantify uncertainty inherent to a clustering solution

due to existence of non-overlapping positions among the reads in each cluster, we define a position-specific confidence score

$$score_i[l] = \frac{cr_i[l] - cr[l]}{1 - cr[l]}$$

where l denotes the position, $cr[\cdot]$ is the overall coverage rate, and $cr_i[\cdot]$ denotes cluster-specific coverage rate for V_i (i.e., $cr_i[l]$ is the fraction of reads in $V_i = \{v_i^j, j = 1, \cdots, |V_i|\}$ covering position l). On the one hand, this score is penalized at a site where the fraction of cluster members (short reads) covering the site is low; the score is negative if the cluster-specific coverage rate is below the global coverage rate which implies uncertainty of the clustering decision. On the other hand, positive scores indicate high confidence in the decision to group the reads into the same cluster. Note that the highest possible score of 1 at position l is achieved when all the reads in a cluster cover the lth position. Using the confidence scores, we define the weight w_{ij} assigned to an edge e_{ij} to quantify similarity between V_i and V_j as

$$w_{ij} = \frac{1}{|L_{i \cup j}|} \sum_{l \in L_{i \cup j}} score_{i \cup j}[l].$$

Given the weights w_{ij}, we can now specify the clustering procedure. In each step, the pair of vertices connected by the edge with maximum weight is merged; the newly constructed vertex inherits edges from the merged vertices and the weights on those edges are re-evaluated. A new (longer) consensus sequence is constructed by combining the two super-reads associated with the merged vertices; recall that there are no conflicts between the super-reads being merged. If after such an update step no edges connect the new vertex with the rest of the graph (because no inherited edges satisfy the connectivity condition), θ is decreased and the above process is repeated. We initially set θ to 0.9 and gradually decrease it by 0.1 while $\theta > 0$. The above procedure is repeated until no pairs of vertices satisfy the connectivity condition. By that point, a set of long consensus sequences (the final super-reads) has been formed from the clusters of reads associated with the nodes of the final graph. While the complexity of agglomerative clustering is, in general, $O(N^3)$ where N denotes the input data size ([21]), it has been shown that its time complexity can be reduced to $O(N^2)$ with accuracy equal to that of the brute-force method by using the partial maximum array technique [22]. We exploit this to efficiently construct super-reads. The algorithm for super-read construction is formalized as Algorithm 1.

2.2 ML reconstruction of quasispecies from super-reads

Here we describe how to reconstruct the most likely set of strains in a viral quasispecies using super-reads from Sect. 2.1 and their confidence scores. While in principle the method outlined in this section could be applied directly to the short reads provided by a sequencing platform, such an approach would in

Algorithm 1: Agglomerative clustering for super-reads construction

Input: Set of reads aligned to the reference genome
Output: Set of super-reads and the corresponding confidence scores
for $\theta > 0$ **do**
  Build a weighted graph $G = (\mathcal{V}, \mathcal{E})$
  while $\mathrm{E} \neq \emptyset$ **do**
    Merge two clusters connected with the largest weight
    Update $G = (\mathcal{V}, \mathcal{E})$ and weights using partial maximum array
  end while
  $\theta = \theta - 0.1$
end for

general not only be computationally prohibitive due to a very large number of short reads but also limit the ability of the algorithm to distinguish strains with small mutual genetic distances due to having long conserved regions. Relying on a relatively small number of long super-reads constructed from short reads circumvents both of these problems and makes the reconstruction more accurate and practically feasible. Note that sequencing errors may undesirably prevent clusters of reads from being merged with other clusters due to a violation of conflict-free requirement; consequently, a set of short reads in a small cluster is likely to have a disproportionate amount of sequencing errors. For this reason, we ignore clusters with very small memberships (in particular, those containing fewer than $0.001 \cdot |R|$ reads), which limits the detection of strains to those constituting more than 0.1% of the quasispecies.

Let $\mathcal{C} = \{C_m, m = 1, \cdots, M\}$ denote the collection of clusters that remain after deleting clusters having only few reads; moreover, for convenience let us re-label the reads in C_m as c_m^j, i.e, $C_m = \{c_m^j, j = 1, \cdots, |C_m|\}$ where $c_m^j \in R$. We organize the super-reads obtained by Algorithm 1 in Sect. 2.1 into the rows of an $M \times N$ matrix $\mathbf{S} = \{s_{mn}, m = 1, \cdots, M, n = 1, \cdots, N\}$ with entries $s_{mn} \in \{\mathsf{A}, \mathsf{C}, \mathsf{G}, \mathsf{T}, -\}$ where $-$ denotes a site not covered by a super-read and N denotes the total number of SNV sites in the strains of a quasispecies. A nucleotides in the (m, n) position of $\mathbf{S}$ is assigned confidence $score_m[n]$ defined in Sect. 2.1; the scores for the entire matrix are normalized so that they fall between 0 and 1 in order to use them in our Bayesian approach to assembly. Let ε_{mn} be the probability that s_{mn} was estimated erroneously due to either a sequencing error in reads on the nth SNV position or the uncertainty induced by reads not covering the nth SNV position. Note that negative scores indicates low confidence resulting from insufficient cluster-specific coverage rate while positive scores imply relatively confident information. In order to map $score_m[n] \in (-\infty, 1]$ to the set $[0, 1]$, we set $\varepsilon_{mn} = 1 - e^{score_m[n]}$ for $score_m[n] < ln(1 - \epsilon)$, where ϵ denotes the error rate of a sequencing platform. Otherwise, we set $\varepsilon_{mn} = \epsilon$.

Let $Q = \{q_k, k = 1, \cdots, K\}$ denote the set of K strains of a viral quasispecies. The goal in the second stage of our method is to determine Q from the super-reads matrix $\mathbf{S}$ using a probabilistic framework. An exhaustive search over the entire solution space is computationally intractable even for small $\mathbf{S}$; instead, we

reconstruct the set of K viral strains sequentially, extending partially estimated strains one SNV position at each step. Since maintaining and extending all possible partial strains inevitably increases their number exponentially, unlikely sets of candidate strains are pruned in each step. Each step consists of three basic parts: (a) extension of the partially reconstructed strains, (b) selection of probable sets comprising K strains chosen among those generated in step (a), and (c) evaluation of the quality of the selected sets of strains and an update of K. The sequential Bayesian inference procedure in step t is illustrated in Fig. S1 in Appendix A.

Extending partially reconstructed strains. Let $F_{1:t\text{-}1} = \{f^i_{1:t\text{-}1}, i = 1, \cdots, |F_{1:t\text{-}1}|\}$ be the collection of partially reconstructed strains covering the first $t-1$ SNV sites and let $B_t = \{b^j_t, j = 1, \cdots, |B_t|\}$ be the lists of distinct bases in the tth column of $\mathbf{S}$, where $b^i_t \in \{\mathsf{A}, \mathsf{C}, \mathsf{G}, \mathsf{T}\}$ and $2 \leq |B_t| \leq 4$. Then, all the possible extensions of $f^i_{1:t-1}$ to the SNV site t can be enumerated as $\{[f^i_{1:t\text{-}1}, b^1_t], \cdots, [f^i_{1:t\text{-}1}, b^{|B_t|}_t]\}$. Let $S^i_{1:t\text{-}1} = \{s^{i_{c'}}_{1:t\text{-}1}, c' = 1, \cdots, |S^i_{1:t\text{-}1}|\}$ be the collection of super-reads covering some of the first t SNV sites which are consistent with $f^i_{1:t\text{-}1}$ (ignoring "$-$" in $s^{i_{c'}}_{1:t\text{-}1}$) where $\{i_{c'}\}$ denote indices of rows of $\mathbf{S}$ that are placed in $S^i_{1:t\text{-}1}$, and let $S^i_t = \{s^{i_c}_t, c = 1, \cdots, |S^i_t|\}$ denote the collection of nucleotides ($s^{i_c}_t \in \{\mathsf{A}, \mathsf{C}, \mathsf{G}, \mathsf{T}\}$, not "$-$") observed at the tth SNV site of the super-reads in $S^i_{1:t\text{-}1}$ where $\{i_c\}$ denote the indices of rows in $\mathbf{S}$ that contribute to S^i_t. Given $S^i_{1:t\text{-}1}, S^i_t$ and $f^i_{1:t\text{-}1}$, the probability of b^j_t being the true extension of $f^i_{1:t\text{-}1}$ is given by

$$P(S^i_t | b^j_t, S^i_{1:t\text{-}1}, f^i_{1:t\text{-}1}) = \prod_{c=1}^{|S^i_t|} P(s^{i_c}_t | b^j_t),$$

$$P(s^{i_c}_t | b^j_t) = \begin{cases} 1 - \varepsilon_{i_c t}, \text{ if } b^j_t = s^{i_c}_t, \\ \frac{\varepsilon_{i_c t}}{|B_t|}, \quad \text{otherwise.} \end{cases}$$

We extend $f^i_{1:t\text{-}1}$ to $[f^i_{1:t\text{-}1}, b^j_t] \in F_{1:t\text{-}1,t}$ by appending the $b^j_t \in B_t$ which satisfies $\frac{P(S^i_t | b^j_t, S^i_{1:t\text{-}1}, f^i_{1:t\text{-}1})^{\frac{1}{|S^i_t|}}}{\sum_{B_t} P(S^i_t | b^j_t, S^i_{1:t\text{-}1}, f^i_{1:t\text{-}1})^{\frac{1}{|S^i_t|}}} \geq \delta_0$, where the exponent ensures proper normalization and is needed since the number of super-reads, $|S^i_{1:t\text{-}1}|$, varies for each $\{f^i_{1:t\text{-}1}, i = 1, \cdots, |F_{1:t\text{-}1}|\}$. For $f^i_{1:t\text{-}1}$ which has no matched super-reads, i.e., $|S^i_{1:i\text{-}1}| = 0$, we keep all of $|B_t|$ possible extensions of $f^{i\text{-}1}_{1:t}$. By collecting probable extensions for each $f^i_{1:t\text{-}1} \in F_{1:t\text{-}1}$, we obtain the set of partial strains stretching over the first t SNV sites, $F_{1:t\text{-}1,t}$. This procedure is formalized as function *ExtendFrag* in Appendix A.

Inferring likely sets of K partial strains. Having generated the probable partial strains $F_{1:t\text{-}1,t}$, we denote the set of all its possible subsets of K

strains (i.e., the quasispecies population candidates) as $\mathcal{Q}_{1:t\text{-}1,t} = \{Q^i_{1:t\text{-}1,t}, i = 1, \cdots, \binom{|F_{1:t\text{-}1,t}|}{K}\}$ where $Q^i_{1:t\text{-}1,t} = \{q^i_{kn}, k = 1, \cdots, K, n = 1, \cdots, t\}$ and $q^i_{kn} \in F_{1:t\text{-}1,t}$. The log-likelihoods of $Q^i_{1:t\text{-}1,t}$ can be expressed as

$$\ln P(\mathbf{S}|Q^i_{1:t\text{-}1,t}) = \sum_{m=1}^{M} \ln P(s_{m\cdot}|Q^i_{1:t\text{-}1,t}),$$
$$P(s_{m\cdot}|Q^i_{1:t\text{-}1,t}) = \frac{1}{K}\left(\sum_{k=1}^{K}\left(\prod_{n=1}^{t} P(s_{mn}|q^i_{kn})\right)\right),$$

where $s_{m\cdot}$ denotes the mth row vector of the matrix of super-reads $\mathbf{S}$ and

$$P(s_{mn}|q^i_{kn}) = \begin{cases} 1-\varepsilon_{mn}, \text{ if } q^i_{kn} = s_{mn}, \\ \frac{\varepsilon_{mn}}{|B_n|}, \quad \text{ if } q^i_{kn} \neq s_{mn} \quad \text{for} \quad s_{mn} \neq -. \end{cases}$$

Let $Q^{max}_{1:t} = \max_{Q^i_{1:t\text{-}1,t} \in \mathcal{Q}_{1:t\text{-}1,t}} P(\mathbf{S}|Q^i_{1:t\text{-}1,t})$. Among the $\binom{|F_{1:t\text{-}1,t}|}{K}$ sets in $\mathcal{Q}_{1:t\text{-}1,t}$, we keep only those that satisfy $P(\mathbf{S}|Q^i_{1:t\text{-}1,t}) > \delta_1 \cdot Q^{max}_{1:t}$ while the others are discarded; let us denote the collection of candidate sets that pass this test as $\mathcal{Q}_{1:t}$. For practical feasibility of the scheme, the collection of partially reconstructed strains $F_{1:t\text{-}1,t}$ is trimmed by excluding from it all the strains that are not part of at least one of the sets in $\mathcal{Q}_{1:t}$; we denote the resulting collection of partial strains by $F_{1:t} \in F_{1:t\text{-}1,t}$ and use it when extending the strains onto the $t+1$ SNV site. The described procedure is formalized as function *InferQuasi* in Appendix A.

Determining the number of strains K in a quasispecies. In this step, we assess appropriateness of K used in the inference of $\mathcal{Q}_{1:t}$ and update it if necessary. To this end, we rely on the minimum error correction (MEC) score which has previously been broadly used as a criterion in the design of methods for haplotype assembly ([23] and [24]). In the context of polyploid haplotype assembly, the MEC score is defined as the smallest number of nucleotides that needs to be changed in data (i.e., in observed reads) so that the corrected reads are consistent with having originated from K haplotypes. Let $HD_t(\cdot,\cdot)$ denote the Hamming distance between two sequences counted over the observed nucleotides in the first t SNV positions.[1] Then the MEC score of the most likely set $Q^{max}_{1:t}$of K viral strains evaluated on the first t SNVs is

$$\text{MEC}_t(K) = \sum_{m=1}^{M} \min_{k \in \{1,\cdots,K\}} \sum_{j=1}^{|C_m|} HD_t(c^j_m, q^{max}_{k\cdot}),$$

where $q^{max}_{k\cdot}$ is the kth row vector of $Q^{max}_{1:t}$. Let N_t be the total number of nucleotides observed in the first t SNV positions of all the reads of the dataset.

[1] If either of the two sequences has a gap "−" in a position, that position is ignored in the computation of the aforementioned Hamming distance.

Note that the smaller the MEC scores, the higher the accuracy of a clustering. If $\mathrm{MEC}_t(K)/N_t < 2\epsilon$, we use the same value K in the next step where the likely set of viral strains stretching over the first $t+1$ SNV positions is inferred. Otherwise, we increase K by 1, repeat the estimation of $\mathcal{Q}_{1:t}$, and evaluate the improvement rate of MEC score as

$$MECimpr(K) = \frac{\mathrm{MEC}_t(K) - \mathrm{MEC}_t(K+1)}{\mathrm{MEC}_t(K)}.$$

The reason for selecting K based on the MEC improvement rate ($MECimpr$) is that the MEC score drops significantly once K matches the actual number of clusters; our scheme attempts to detect that change in order to infer population size. If $MECimpr(K) > \eta$, where η denotes a pre-specified threshold, the number of species is updated as $K \leftarrow \min\{K+n, |F_{1:t\text{-}1,t}|\}$ where n is the smallest integer number such that $MECimpr(K+n) < \eta$. If $MECimpr(K) < \eta$, we update the number of species as $K \leftarrow \max\{K-n, 2\}$ where n is the smallest integer such that $MECimpr(K-n) \geq \eta$. The choice of threshold η is discussed in the Appendix B. The updated value of K is used for the inference of $\mathcal{Q}_{1:t+1}$. Note that the probable set of viral strains, $\mathcal{Q}_{1:t}$, is stored for each K to avoid performing redundant $MECimpr(\cdot)$ calculations.

Once we obtain the most likely set of K viral sequences covering N SNVs, $Q^{max}_{1:N}$, the full-length K quasispecies strains are reconstructed by inserting the consensus nucleotides observed in R into the non-SNV sites. We estimate relative frequencies p_k, $1 \leq k \leq K$, of quasispecies strains based on the Hamming distance between super-reads and the reconstructed sequences. In particular, for each super-read sr_i we determine the nearest assembled strain q_j where $j = \arg\min_{k\in\{1,\cdots,K\}} HD(sr_i, q_k)$ and the number of reads involved in constructing the super-read sr_i is counted towards p_j. The entire scheme proposed in this subsection is summarized as Algorithm 2.

3 Results and Discussion

3.1 Performance comparison on simulated data

To evaluate performance of the proposed method for quasispecies reconstruction, we use metrics *Recall*, *Precision*, *Predicted Proportion*, and *Reconstruction Rate*. *Recall* is defined as the ratio of the number of correctly reconstructed strains to the total number of true strains in the quasispecies, i.e., $Recall = \frac{TP}{TP+FN}$, while *Precision* is defined as the fraction of correctly reconstructed strains among all the assembled sequences, i.e., $Precision = \frac{TP}{TP+FP}$. Noting that *Precision* usually reports high scores when the number of strains is underestimated while penalizing overestimation of the population size, we also report the ratio of the number of reconstructed sequences to the true population size, *Predicted Proportion*. The closer *Predicted Proportion* to 1, the more accurate the number of reconstructed strains. Moreover, to assess the degree of reconstruction accuracy, we define $Reconstruction\ Rate = \frac{1}{K}\sum_{k=1}^{K}\left(1 - \frac{HD(q_k, \hat{q_k})}{G}\right)$, where G is the

Algorithm 2: Sequential Bayesian Inference for quasispecies reconstruction

Input: Set of super-reads and the corresponding confidence scores
Output: Set of K strains of a viral quasispecies
Initial $K \leftarrow 2$, $F_{1:1} \leftarrow B_1$
for t $\in \{2, \cdots, N\}$ **do**
  $F_{1:t\text{-}1,t}$ = ***ExtendFrag***$(F_{1:t\text{-}1}, t, \delta_0)$
  $\mathcal{Q}_{1:t}$ = ***InferQuasi***$(F_{1:t\text{-}1,t}, K, \delta_1)$
  $K^* \leftarrow K$, $\mathcal{Q}^*_{1:t} \leftarrow \mathcal{Q}_{1:t}$
  if $\mathrm{MEC}_t(K)/N_t \geq 2\epsilon$ and $K < |F_{1:t\text{-}1,t}|$ **do**
    $\mathcal{Q}_{1:t}$ = ***InferQuasi***$(F_{1:t\text{-}1,t}, K{+}1, \delta_1)$
    if $MECimpr(K) < \eta$ **do**
      while $MECimpr(K) < \eta$ and $K > 2$
        $\mathcal{Q}^*_{1:t} \leftarrow \mathcal{Q}_{1:t}$, $K^* \leftarrow K$, $K \leftarrow K - 1$
        $\mathcal{Q}_{1:t}$ = ***InferQuasi***$(F_{1:t\text{-}1,t}, K, \delta_1)$
      end while
    else do
      while $MECimpr(K) \geq \eta$ and $K < |F_{1:t\text{-}1,t}|$
        $\mathcal{Q}^*_{1:t} \leftarrow \mathcal{Q}_{1:t}$, $K^* \leftarrow K$
        $\mathcal{Q}_{1:t}$ = ***InferQuasi***$(F_{1:t\text{-}1,t}, K{+}1, \delta_1)$
      end while
    end if
  end if
  $K \leftarrow K^*$, $\mathcal{Q}_{1:t} \leftarrow \mathcal{Q}^*_{1:t}$
  Get $F_{1:t}$ by pruning $F_{1:t\text{-}1,t}$ based on $\mathcal{Q}_{1:t}$
end for
Reconstruct full-length quasispecies Q from $Q^{max}_{1:N} \in \mathcal{Q}_{1:t}$ and R
Estimate frequencies of each strain $q_k \in Q$ based on $HD(sr_i, q_k)$ and $|C_i|$

length of a genome, K is the number of strains in a quasispecies and q_k and $\hat{q_k}$ denote the kth true strain and its nearest sequence among the K estimated ones, respectively. To assess the accuracy of estimated frequencies, we use Jensen-Shannon divergence (JSD) which quantifies similarity between two distributions. Given a true distribution P and its approximation Q, the Kullback-Leibler (KL) divergence $D(P||Q) = \sum_{i=1}^{n} P(i)\log\frac{P(i)}{Q(i)}$ is undefined when $Q(i) = 0$. JSD, a symmetrized and smoothed version of the KL divergence, circumvents this problem by defining similarity of P and Q as $JSD(P||Q) = \frac{1}{2}D(P||M) + \frac{1}{2}D(Q||M)$, where M is defined as $M = \frac{1}{2}(P + Q)$.

We compare our algorithm with publicly available ShoRAH [4], PredictHaplo [12], and ViQuaS [17]. Since ViQuaS is an extension of the algorithm in [10, 11], and was shown to have superior performance compared to its predecessor, we omit the comparison with the software QuRe in [10, 11]. It is worth pointing out that for the synthetic data sets we study, ShoRAH could not reconstruct strains in the regions where the simulated sequencing coverage is relatively low compared to the average, resulting in reconstruction of strains that are shorter than the true length G. To facilitate a fair comparison with ShoRAH, we aligned

its reconstructed strains to the reference genome and completed missing sites with bases from the reference. ViQuaS, on the other hand, tends to reconstruct many more strains than actually present; thus we followed ViQuaS's authors recommendation and retained only those having frequencies greater than f_{min} when calculating *Precision*. Finally, not all of the synthetic data sets could be processed with PredictHaplo, preventing us from reporting its performance in some of the scenarios.

Table 1. Performance comparison of different methods for varied diversities (*div*) on simulated data. Performance comparison of aBayesQR, ShoRAH, ViQuaS and PredictHaplo in terms of *Recall*, *Precision*, *Predicted Proportion* (*PredProp*), *Reconstruction Rate* (*ReconRate*) and *JSD* on the simulated data with $err = 0.1\%$ and $cov = 500\times$ vs. *div* for a mixture of 5 and 10 viral strains. Averaged PredictHaplo results are reported if it provides answers for more than 50% of data sets. Boldface values indicate the best performance for each *div*(%).

		5 strains					10 strains				
	div(%)	1	2	3	4	5	1	2	3	4	5
Recall	aBayesQR	**0.7080**	**0.7120**	**0.6840**	0.6560	0.6320	**0.5810**	**0.6380**	**0.6080**	**0.5860**	**0.5550**
	ShoRAH	0.1920	0.1600	0.1300	0.1060	0.0780	0.0150	0.0380	0.0740	0.0640	0.0930
	ViQuaS	0.3700	0.5240	0.6040	0.6360	0.5960	0.0980	0.1700	0.3730	0.4720	0.5050
	PredictHaplo	-	-	-	**0.6918**	**0.6808**	-	-	0.1021	0.1550	0.2010
Precision	aBayesQR	**0.7113**	**0.7130**	**0.6826**	0.6447	0.6319	**0.6210**	**0.6881**	**0.6610**	**0.6373**	0.6140
	ShoRAH	0.1062	0.1418	0.1240	0.1078	0.0790	0.0050	0.0170	0.0498	0.0506	0.0824
	ViQuaS	0.1960	0.3206	0.4559	0.4982	0.5298	0.0485	0.1079	0.2973	0.4690	0.5596
	PredictHaplo	-	-	-	**0.9373**	**0.8822**	-	-	0.4509	0.6000	**0.6833**
PredProp	aBayesQR	**1.0180**	**1.0120**	**1.0120**	**1.0360**	**1.0140**	**0.9680**	**0.9440**	**0.9240**	**0.9240**	**0.9100**
	ShoRAH	1.9660	1.2200	1.0780	1.0000	1.0180	3.2000	2.9100	1.6710	1.3520	1.1860
	ViQuaS	2.1100	1.7220	1.4080	1.3340	1.2180	2.0860	1.8580	1.5450	1.2320	1.0730
	PredictHaplo	-	-	-	0.7388	0.7737	-	-	0.1947	0.2430	0.2890
ReconRate	aBayesQR	**0.9990**	**0.9982**	**0.9971**	**0.9961**	**0.9953**	**0.9975**	**0.9967**	**0.9952**	**0.9942**	**0.9924**
	ShoRAH	0.9948	0.9903	0.9891	0.9851	0.9827	0.9941	0.9900	0.9899	0.9897	0.9911
	ViQuaS	0.9963	0.9949	0.9917	0.9936	0.9897	0.9944	0.9910	0.9899	0.9881	0.9858
	PredictHaplo	-	-	-	0.9906	0.9896	-	-	0.9850	0.9797	0.9747
JSD	aBayesQR	**0.0022**	**0.0008**	**0.0008**	0.0014	**0.0008**	**0.0043**	**0.0026**	**0.0023**	**0.0023**	**0.0025**
	ShoRAH	0.0762	0.0174	0.0047	**0.0009**	0.0012	0.1390	0.1110	0.0422	0.0238	0.0109
	ViQuaS	0.0651	0.0255	0.0222	0.0097	0.0180	0.0993	0.0747	0.0495	0.0469	0.0454
	PredictHaplo	-	-	-	0.1020	0.1036	-	-	0.1971	0.1636	0.1312

We generated synthetic datasets by emulating high-throughput sequencing of a viral population consisting of a number of closely related viral genomes having length of 1300bp; this particular length was chosen to coincide with the longest region of the HIV *pol* gene. Quasispecies sequences are generated by introducing independent mutations at uniformly random locations along the length of a randomly generated reference genome so as to obtain a predefined level of diversity (*div*%), i.e., a predefined average Hamming distance between quasispecies strains. Simulating Illumina's MiSeq data, 2×250bp-long paired-end reads are sampled uniformly from each viral strain with a mean coverage of $cov\times$ per strain. Inserts of the paired-end reads are on average 150bp long with

standard deviation of 30. In our benchmarking tests, we focus on exploring the effects of diversity ($div\%$) on the accuracy of the quasispecies reconstruction. Two sets of viral populations are considered: (1) a mix of 5 viral strains with abundance levels $50\%, 30\%, 15\%, 4\%$ and 1%; and (2) a mix of 10 strains with abundance levels $36\%, 24\%, 16\%, 8\%, 5.5\%, 4\%, 3\%, 2\%, 1\%$ and 0.5%. Note that the abundances are chosen to approximately follow geometric distribution and that the populations include low abundant strains. For each combination of the parameters, 100 data sets were generated and the reported results were obtained by averaging over those data instances. For PredictHaplo, which did not produce results in each instance, the averaged results are reported if more than 50 instances were successfully processed.

In all of the following experiments, potential SNVs are called if their abundance is higher than 1%, which is set relatively high to avoid false positives (FPs); FPs prevent reads to be merged with existing clusters in Sect. 2.1. We execute the function ***ExtendFrag*** with parameter $\delta_0 = 0.1$. Parameter δ_1 in function ***InferQuasi*** is initially set to 0.001, but adaptively increases if the number of combinations of partially reconstructed strains exceeds 10000; this is done to limit the number of likelihood calculations performed in each run of ***InferQuasi***. The recommended value of η, a threshold used to determine population size K based on *MECimpr(·)*, is discussed in Appendix B.

We compare performances of aBayesQR, ShoRAH, ViQuaS and PredictHap when applied to the reconstruction of a quasispecies spectrum with diversity levels varying between 1% and 5% (i.e., $div \in \{1\%, 2\%, 3\%, 4\%, 5\%\}$). To test the ability of different methods to reconstruct quasispecies with low diversity, we assume low sequencing error rate of $err = 0.1\%$ (median mismatch error rates for 454 Life Sciences and Illumina platforms are 0.1% and 0.12%, respectively ([25])). Coverage per strain $cov\times$ is set to $500\times$, implying total coverage of $2500\times$ and $5000\times$ for the 5-strain and 10-strain population, respectively; strains having frequencies 0.23% or higher in the 5-strain case and those with frequencies 0.46% or higher in the 10-strain case are covered with probability 0.99 ([5]).

Table 1 demonstrates that the proposed aBayesQR algorithm outperforms existing schemes. In terms of *Recall* and *Precision*, aBayesQR exhibits exceptionally good performance compared to competing methods when reconstructing quasispecies strains with diversity $div < 4\%$. The performance of ViQuaS deteriorates at low diversities in terms of most of the criteria (i.e., *Recall*, *Precision*, *Predicted Proportion* and *JSD*). PredictHaplo could not perform reconstruction in most of the low diversity instances yet it overall achieves the highest *Precision* because it typically underestimates the number of strains as shown by *Predicted Proportion* (e.g., estimating only 2-3 out of 10 strains), which is in agreement with the results reported by a previous study [14]. Among all methods, ShoRAH has the lowest performance in terms of *Recall* and *Precision*. As indicated by *Predicted Proportion*, aBayesQR is the most accurate method in terms of estimating the population size although it often misses a strain with the lowest frequency when applied to reconstruction of a quasispecies consisting of 10 strains. ViQuaS and ShoRAH typically overestimates the number of strains especially at low di-

versity levels. aBayesQR is the best method in terms of *Reconstruction Rate* at all levels of diversity. In terms of frequency estimation, aBayesQR overall outperforms all the other methods whereas PredictHaplo shows the highest *JSD* due to its drawback of underestimating the number of strains. Note that both ViQuaS and ShoRAH exhibit significantly increased (i.e., deteriorated) *JSD* at low diversity levels. This fact, along with the low *Recall* and *Precision* scores they have in low diversity settings, indicates that state-of-the-art methods experience major difficulties when attempting to reconstruct viral quasispecies in those settings, as also observed in [5, 14] and [17].

We further study the effects that sequencing error rate ($err\%$) and coverage per strain ($cov\times$) have on the performance of the algorithms. Those results are reported in Table S2 and S3 in the Appendix C, demonstrating superiority of aBayesQR as compared to the competing methods. The runtimes of the tested algorithms are shown in Table S4 in the Appendix C.

3.2 Performance comparison on real HIV data

To further test the performance of our proposed method, we employ it for the analysis of the HIV 5-virus-mix dataset published in [20]. Specifically, we apply our algorithm to reconstruct an *in vitro* generated quasispecies population consisting of 5 known HIV-1 strains: HIV-1_{HXB2}, HIV-$1_{89.6}$, HIV-1_{JR-CSF}, HIV-1_{NL4-3} and HIV-1_{YU2}. Compared to the simulated data set, relative frequencies of the 5 HIV-1 strains are more evenly distributed (about $10\% - 30\%$) and the pairwise distances between strains are higher ($2.61\% - 8.45\%$) [20]. We use the 2×250bp-long paired-end reads provided by Illumina's MiSeq Benchtop Sequencer. The reads are aligned to the HIV-1_{HXB2} reference genome; the reads shorter than 150nt and those having bases with quality scores less than a PHRED threshold of 60 are discarded. We compare the performance of our method applied to gene-wise quasispecies reconstruction of the above described HIV data with that of the competing techniques. Since the current version of ViQuaS software does not support specifying genomic regions, we could not use it in this experiment. When running aBayesQR, we set the parameter η to 0.09 (the setting recommended in Appendix B). Other parameters are set to the same values as the ones used in Sect. 3.1.

We evaluate and report the *Predicted Proportion* (i.e., the fraction of correctly estimated strains as defined in Sect. 3.1) and *Reconstruction Rate* in Table 2. On this real HIV-1 data set which (as pointed above) has different properties than the simulated data set in Sect. 3.1, aBayesQR is the most accurate among the considered methods in terms of *Predicted Proportion*. PredictHaplo underestimates the population size and reconstructs three or four strains in the 8 considered genes and ShoRAH greatly overestimates the population size for all 13 genes of the HIV-1 data set (e.g., it reconstructs 119 strains in gp120), which is consistent with our simulation results as well as with the results in [14]. aBayesQR and PredictHaplo are tied for the number of genes where all the strains are perfectly reconstructed (5 each); for the remaining genes, PredictHaplo provides a larger number of perfectly reconstructed strains. However, it is

Table 2. Performance comparisons on a real HIV-1 5-virus-mix data set. *Predicted Proportion* (PredProp) and *Reconstruction Rate* (RR) for aBayesQR, ShoRAH and PredictHaplo applied to reconstruction of HIV-1_{HXB2}, HIV-$1_{89.6}$, HIV-$1_{\text{JR-CSF}}$, HIV-$1_{\text{NL4-3}}$ and HIV-1_{YU2} for all 13 genes of the HIV-1 dataset. (note: RR are expressed in percentages.) Boldface values indicate the genes where all the strains are perfectly reconstructed. The inferred frequencies are shown in Table S5 in Appendix C.

		p17	p24	p2-p6	PR	RT	RNase	int	vif	vpr	vpu	gp120	gp41	nef
aBayesQR	PredProp	**1**	1	**1**	**1**	1	**1**	1	1	**1.2**	1	0.8	0.8	1.2
	RR_{HXB2}	**100**	99.4	**100**	**100**	98.5	**100**	99.9	100	**100**	99.6	98	0	95.8
	$RR_{89.6}$	**100**	98.7	**100**	**100**	98.6	**100**	100	100	**100**	92	96.5	98.9	95.5
	$RR_{\text{JR-CSF}}$	**100**	99.6	**100**	**100**	99	**100**	100	100	**100**	98.8	97.7	99.1	98.2
	$RR_{\text{NL4-3}}$	**100**	100	**100**	**100**	98.9	**100**	100	99.8	**100**	100	96.3	98.8	100
	RR_{YU2}	**100**	99.7	**100**	**100**	99.2	**100**	99.5	99.7	**100**	100	0	98.6	99.2
ShoRAH	PredProp	13	16.4	13.8	8.8	21.8	11.8	13.6	12.8	7.8	4	23.8	19.8	17.4
	RR_{HXB2}	100	96.7	100	100	98.2	100	97.5	100	100	100	97.7	98.4	98.2
	$RR_{89.6}$	100	99.7	100	100	98.6	100	98.9	99.8	100	93.6	96.1	98.6	98.9
	$RR_{\text{JR-CSF}}$	100	100	100	100	99.8	96.4	99	100	100	98	96.9	96.3	94.7
	$RR_{\text{NL4-3}}$	100	99.1	97.3	100	98.9	99.2	99.3	99.3	100	100	96.1	98.5	98.6
	RR_{YU2}	94.2	99	100	98.3	98	94.5	98.6	95	93.2	90.8	97	95.4	97.9
PredictHaplo	PredProp	**1**	0.6	**1**	**1**	**1**	0.8	0.8	0.8	**1**	0.8	0.8	0.8	0.8
	RR_{HXB2}	**100**	0	**100**	**100**	**100**	98.9	100	100	**100**	93.17	0	0	0
	$RR_{89.6}$	**100**	100	**100**	**100**	**100**	100	99.8	100	**100**	0	97.8	100	98.87
	$RR_{\text{JR-CSF}}$	**100**	100	**100**	**100**	**100**	100	100	100	**100**	100	99.7	100	100
	$RR_{\text{NL4-3}}$	**100**	99.1	**100**	**100**	**100**	100	100	100	**100**	100	100	100	100
	RR_{YU2}	**100**	0	**100**	**100**	**100**	0	0	0	**100**	100	98.6	100	100

worth pointing out that PredictHaplo, designed for identification of HIV haplotypes, missed at least one strain in each of the remaining 8 genes while aBayesQR reconstructed most of the strains on all but two genes, gp120 and gp41. ShoRAH did not perfectly reconstruct any of the 13 genes, which is consistent with the simulation results. Moreover, overestimating the number of strains negatively affects the accuracy of ShoRAH's frequency estimation; for instance, the sum of frequencies corresponding to the most abundant 5 strains does not exceed 50% in 9 out 13 genes (71% is the largest such sum, on *vpu*) (see Table S5 in the Appendix C).

To complement the gene-wise quasispecies reconstruction study with that of a global reconstruction, we consider the HIV-1 *gap-pol* region spanning 4307bp. To efficiently process 355241 paired-end reads that remain after applying a quality filter, we organize the region into a sequence of windows of length 400bp where the consecutive windows overlap by 150bp and run aBayesQR on those windows. The entire region is assembled by connecting strains in the consecutive windows while testing consistency in the overlapping intervals. The number of strains retrieved in the global reconstruction is decided by majority voting of the number of strains obtained in each window. The frequencies are estimated by counting reads nearest (in terms of Hamming distance) to each of the reconstructed strains. Following this procedure, both aBayesQR and PredictHaplo could re-

construct all 5 HIV-1 strains in the *gap-pol* region correctly, i.e., they both achieved *Reconstruction rate* of 100 for all 5 strains and *Predicted Proportion* of 1. The frequencies estimated by aBayesQR are 15.21%, 19.34%, 25.56%, 27.61% and 12.27% while those estimated by PredictHap are 13.21%, 13.56%, 25.67%, 19.69% and 27.86%. ShoRAH highly overestimated the number of strains and reported *Predicted Proportion* of 41.8; its five most abundant strains estimated are reported to have frequencies 8.51%, 5.04%, 3.41%, 3.24% and 3.09%.

4 Conclusions

In this paper, we presented a novel maximum-likelihood based approximate algorithm for reconstructing viral quasispecies from high-throughput sequencing data. aBayesQR assembles paired-end short reads into longer fragments based on similarity of the read overlaps and the uncertainty level of non-overlapping regions. The probable sets of partially reconstructed strains are inductively searched and a subset of those strains is extended to efficiently deduce the most likely set of strains in a quasisepcies. Detection of the population size is embedded into the algorithm and is empirically shown to be very accurate; the number of strains is dynamically adjusted based on the reliability of the partially assembled quasispecies in each extension step. Performance of the developed method is tested on both synthetic datasets and a real HIV-1 dataset. In both settings, the new algorithm outperforms existing techniques in terms of accuracy of the quasispecies size estimation, perfect reconstruction of strains, proportion of correct bases in each reconstructed strain and the estimation of their abundance. A particularly high accuracy is observed in estimating the population size (i.e., the number of strains) and their relative abundance. Tests on synthetic datasets demonstrates that aBayesQR is capable of reconstructing quasispecies at low diversity, showing superior performance in those settings compared to state-of-the-art algorithms. Furthermore, the study on a real HIV-1 dataset demonstrates that our proposed algorithm outperforms or has performance comparable to that of the existing methods in the general setting of viral quasispecies reconstruction.

aBayesQR can be extended and applied to the problem of estimating the population size and the degree of variation among the constituent species in related fields such as immunogenetics. On a related note, bacterial populations are characterized by having relatively lower mutation rates than viral and thus typically have fewer segregating sites on the sequences in a population. The ability of our method to perform highly accurate reconstruction in such settings should be further investigated.

A software aBayesQR is available at https://github.com/SoyeonA/aBayesQR.

Acknowledgements

This work was funded by the National Science Foundation under grants CCF 1507998 and CCF 1618427.

References

1. Duarte, E., Novella, I., Weaver, S., Domingo, E., Wain-Hobson, S., Clarke, D., Moya, A., Elena, S., De La Torre, J., Holland, J.: Rna virus quasispecies: significance for viral disease and epidemiology. Infectious agents and disease **3**(4), 201–214 (1994)
2. Lauring, A.S., Andino, R.: Quasispecies theory and the behavior of rna viruses. PLoS Pathogens **6**(7) (2010)
3. Posada-Cespedes, S., Seifert, D., Beerenwinkel, N.: Recent advances in inferring viral diversity from high-throughput sequencing data. Virus Research (2016)
4. Zagordi, O., Bhattacharya, A., Eriksson, N., Beerenwinkel, N.: Shorah: estimating the genetic diversity of a mixed sample from next-generation sequencing data. BMC bioinformatics **12**(1), 119 (2011)
5. Eriksson, N., Pachter, L., Mitsuya, Y., Rhee, S.Y., Wang, C., Gharizadeh, B., Ronaghi, M., Shafer, R.W., Beerenwinkel, N.: Viral population estimation using pyrosequencing. PLoS Comput Biol **4**(5), e1000,074 (2008)
6. Zagordi, O., Geyrhofer, L., Roth, V., Beerenwinkel, N.: Deep sequencing of a genetically heterogeneous sample: local haplotype reconstruction and read error correction. Journal of computational biology **17**(3), 417–428 (2010)
7. Zagordi, O., Klein, R., Däumer, M., Beerenwinkel, N.: Error correction of next-generation sequencing data and reliable estimation of hiv quasispecies. Nucleic acids research **38**(21), 7400–7409 (2010)
8. Astrovskaya, I., Tork, B., Mangul, S., Westbrooks, K., Măndoiu, I., Balfe, P., Zelikovsky, A.: Inferring viral quasispecies spectra from 454 pyrosequencing reads. BMC bioinformatics **12**(6), 1 (2011)
9. Westbrooks, K., Astrovskaya, I., Campo, D., Khudyakov, Y., Berman, P., Zelikovsky, A.: Hcv quasispecies assembly using network flows. In: Bioinformatics Research and Applications, pp. 159–170. Springer (2008)
10. Prosperi, M.C., Prosperi, L., Bruselles, A., Abbate, I., Rozera, G., Vincenti, D., Solmone, M.C., Capobianchi, M.R., Ulivi, G.: Combinatorial analysis and algorithms for quasispecies reconstruction using next-generation sequencing. BMC bioinformatics **12**(1), 1 (2011)
11. Prosperi, M.C., Salemi, M.: Qure: software for viral quasispecies reconstruction from next-generation sequencing data. Bioinformatics **28**(1), 132–133 (2012)
12. Prabhakaran, S., Rey, M., Zagordi, O., Beerenwinkel, N., Roth, V.: Hiv haplotype inference using a propagating dirichlet process mixture model. IEEE/ACM Trans. on Comput. Biol. Bioinform. (TCBB) **11**(1), 182–191 (2014)
13. Töpfer, A., Zagordi, O., Prabhakaran, S., Roth, V., Halperin, E., Beerenwinkel, N.: Probabilistic inference of viral quasispecies subject to recombination. Journal of Computational Biology **20**(2), 113–123 (2013)
14. Schirmer, M., Sloan, W.T., Quince, C.: Benchmarking of viral haplotype reconstruction programmes: an overview of the capacities and limitations of currently available programmes. Briefings in bioinformatics p. bbs081 (2012)
15. Töpfer, A., Marschall, T., Bull, R.A., Luciani, F., Schönhuth, A., Beerenwinkel, N.: Viral quasispecies assembly via maximal clique enumeration. PLoS Comput Biol **10**(3), e1003,515 (2014)
16. Mangul, S., Wu, N.C., Mancuso, N., Zelikovsky, A., Sun, R., Eskin, E.: Accurate viral population assembly from ultra-deep sequencing data. Bioinformatics **30**(12), i329–i337 (2014)

17. Jayasundara, D., Saeed, I., Maheswararajah, S., Chang, B., Tang, S.L., Halgamuge, S.K.: Viquas: an improved reconstruction pipeline for viral quasispecies spectra generated by next-generation sequencing. Bioinformatics p. btu754 (2014)
18. Le, T., Chiarella, J., Simen, B.B., Hanczaruk, B., Egholm, M., Landry, M.L., Dieckhaus, K., Rosen, M.I., Kozal, M.J.: Low-abundance hiv drug-resistant viral variants in treatment-experienced persons correlate with historical antiretroviral use. PloS one **4**(6), e6079 (2009)
19. Simen, B.B., Simons, J.F., Hullsiek, K.H., Novak, R.M., MacArthur, R.D., Baxter, J.D., Huang, C., Lubeski, C., Turenchalk, G.S., Braverman, M.S., et al.: Low-abundance drug-resistant viral variants in chronically hiv-infected, antiretroviral treatment–naive patients significantly impact treatment outcomes. Journal of Infectious Diseases **199**(5), 693–701 (2009)
20. Di Giallonardo, F., Töpfer, A., Rey, M., Prabhakaran, S., Duport, Y., Leemann, C., Schmutz, S., Campbell, N.K., Joos, B., Lecca, M.R., et al.: Full-length haplotype reconstruction to infer the structure of heterogeneous virus populations. Nucleic acids research **42**(14), e115–e115 (2014)
21. Sasirekha, K., Baby, P.: Agglomerative hierarchical clustering algorithm–a review. International Journal of Scientific and Research Publications **3**(3) (2013)
22. Jung, S.Y., Kim, T.S.: An agglomerative hierarchical clustering using partial maximum array and incremental similarity computation method. In: Data Mining, 2001. ICDM 2001, Proc. IEEE Int. Conf. on, pp. 265–272. IEEE (2001)
23. Lancia, G., Bafna, V., Istrail, S., Lippert, R., Schwartz, R.: Snps problems, complexity, and algorithms. In: Algorithms—ESA 2001, pp. 182–193. Springer (2001)
24. Lippert, R., Schwartz, R., Lancia, G., Istrail, S.: Algorithmic strategies for the single nucleotide polymorphism haplotype assembly problem. Briefings in bioinformatics **3**(1), 23–31 (2002)
25. Archer, J., Baillie, G., Watson, S.J., Kellam, P., Rambaut, A., Robertson, D.L.: Analysis of high-depth sequence data for studying viral diversity: a comparison of next generation sequencing platforms using segminator ii. BMC bioinformatics **13**(1), 47 (2012)
26. Quail, M.A., Smith, M., Coupland, P., Otto, T.D., Harris, S.R., Connor, T.R., Bertoni, A., Swerdlow, H.P., Gu, Y.: A tale of three next generation sequencing platforms: comparison of ion torrent, pacific biosciences and illumina miseq sequencers. BMC genomics **13**(1), 1 (2012)
27. Ross, M.G., Russ, C., Costello, M., Hollinger, A., Lennon, N.J., Hegarty, R., Nusbaum, C., Jaffe, D.B.: Characterizing and measuring bias in sequence data. Genome Biol **14**(5), R51 (2013)

Appendix

Appendix A

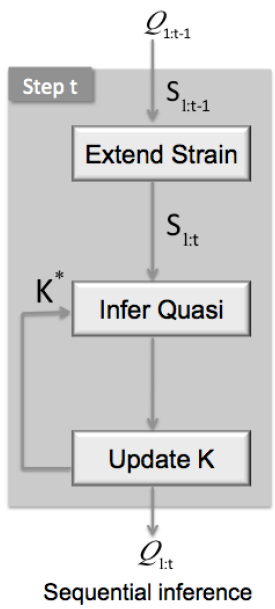

Fig. S1. Procedure of sequential Bayesian inference in step t

function ***ExtendFrag***$(F_{1:t\text{-}1}, t, \delta_0)$: Extend $F_{1:t\text{-}1}$ to $F_{1:t\text{-}1,t}$
Input: $F_{1:t\text{-}1}, t, \delta_0$
Output: $F_{1:t\text{-}1,t}$
for $f^i_{1:t\text{-}1} \in F_{1:t\text{-}1}$ **do**
  for $b^j_t \in B_t$ **do**
    if $\frac{P(S^i_t|b^j_t, S^i_{1:t\text{-}1}, f^i_{1:t\text{-}1})^{\frac{1}{|S^i_t|}}}{\sum_{B_t} P(S^i_t|b^j_t, S^i_{1:t\text{-}1}, f^i_{1:t\text{-}1})^{\frac{1}{|S^i_t|}}} \geq \delta_0$ **then**
      $F_{1:t\text{-}1,t} \leftarrow F_{1:t\text{-}1,t} \cup \{[f^i_{1:t\text{-}1}, b^j_t]\}$
    end if
  end for
end for

function ***InferQuasi***$(F_{1:t\text{-}1,t}, K, \delta_1)$: Infer likely sets of K strains $\mathcal{Q}_{1:t}$ from $F_{1:t\text{-}1,t}$
Input: $F_{1:t\text{-}1,t}, K, \delta_1$
Output: $\mathcal{Q}_{1:t}$
Enumerate $\mathcal{Q}_{1:t\text{-}1,t}$ from $F_{1:t\text{-}1,t}$ and K
for $Q^i_{1:t\text{-}1,t} \in \mathcal{Q}_{1:t\text{-}1,t}$ **do**
  if $P(\mathbf{S}|Q^i_{1:t\text{-}1,t}) > \delta_1 \cdot Q^{max}_{1:t}$ **then**
    $\mathcal{Q}_{1:t} \leftarrow \mathcal{Q}_{1:t} \cup \{Q^i_{1:t\text{-}1,t}\}$
  end if
end for

Appendix B: Guideline for choosing parameter η

Here we discuss the choice of parameter η, a threshold used for assessing value of the metric $MECimpr$ in the process of determining the number of sequences K in a quasispecies. Let $Q = \{q_k, k = 1, \cdots, K\}$ denote a viral population consisting of K strains. The set of reads of length L, $R = \{r_i, i = 1, \cdots, |R|\}$, is generated by a sequencing platform having error rate ϵ and aligned to the reference genome of length G. The MEC score characterizing accuracy of the assembly of strains in Q from reads in R is calculated as

$$\mathrm{MEC}(K) = \sum_{m=1}^{M} \min_{k \in \{1, \cdots, K\}} \sum_{j=1}^{|R|} HD(r_i, q_k).$$

Assume that the sequencing errors are independent and identically distributed across all reads. When the quasispecies recovery is perfect, i.e., when all of the K strains are reconstructed correctly and the relative frequencies of strains are estimated accurately, the MEC score is $|R|L\epsilon$. If a strain with the smallest frequency (f_{min}) is not recognized as a distinct mixture component while the other $K-1$ sequences are correctly reconstructed, the incorrect clustering of $|R|f_{min}$ reads generated from the rarest quasispecies component induces an extra contribution to the MEC score. The MEC score obtained following perfect reconstruction of $K-1$ strains and misclassification of the Kth strain is

$$\mathrm{MEC}(K-1) = |R|L\epsilon + |R|Lf_{min}d(1 - \frac{15}{16}\epsilon),$$

where d is the average diversity rate. The MEC improvement rate achieved by increasing the number of viral strains from $K-1$ to K (and thus reducing the MEC score thanks to a correct clustering of the rarest strain) is

$$\begin{aligned} MECimpr(K-1) &= \frac{\mathrm{MEC}(K-1) - \mathrm{MEC}(K)}{\mathrm{MEC}(K-1)} \\ &= \frac{f_{min}d(1 - \frac{15}{16}(1-\epsilon))}{f_{min}d(1 - \frac{15}{16}(1-\epsilon)) + \epsilon} \\ &\equiv \eta_1. \end{aligned}$$

While η_1 is a potential choice for the threshold, it is beneficial to soften it and allow that $MECimpr(\cdot)$ takes on values slightly below it. To this end, let us also consider the scenario where in addition to the perfect recovery of K strains, an extra strain is erroneously inferred by misclassifying reads that in fact should have been placed in the cluster associated with the most abundant strain (i.e., the one having frequency f_{max}). This reduces evaluated MEC score to an unrealistically low value given by

$$\mathrm{MEC}(K+1) = |R|L\epsilon - |R|Lf_{max}\frac{\epsilon^2}{3}.$$

Improvement of the MEC score due to having an extra (unnecessary) cluster can be expressed as

$$\begin{aligned} MECimpr(K) &= \frac{\mathrm{MEC}(K) - \mathrm{MEC}(K+1)}{\mathrm{MEC}(K)} \\ &= \frac{\epsilon}{3} f_{max} \\ &\equiv \eta_2 \end{aligned}$$

We note that for typical parameter values, $\eta 1 \gg \eta 2$; we choose the threshold η by taking a weighted geometric mean of η_1 and η_2,

$$\eta = (\eta_1^{w_1} \eta_2^{w_2})^{\frac{1}{w_1+w_2}}.$$

To avoid overestimation of the number of strains, we choose w_1 to be larger than w_2. In our experiments, the ratio $r = w_1/w_2$ was set to 5. We find that the results are fairly robust with respect to the choice of parameter η as demonstrated in Table S1.

Table S1. Performances comparison of aBayesQR with different parameter η for varied diversities *div* on simulated data. Performances of aBayesQR as a function of diversity with parameter η varied around the recommended value from -20% to $+20\%$; shown are *Recall*, *Precision*, *Predicted Proportion* (*PredProp*), *Reconstruction Rate* (*Recon-Rate*) and *JSD*. The data is generated synthetically, relevant parameters are $err = 0.1\%$ and $cov = 500\times$, simulated is a mixture of 5 and 10 viral strains.

		5 strains					10 strains				
	div(%)	1	2	3	4	5	1	2	3	4	5
Recall	0.8η	0.7020	0.7080	0.6840	0.6560	0.6320	0.5800	0.6390	0.6060	0.5810	0.5550
	0.9η	0.7060	0.7120	0.6840	0.6560	0.6300	0.5800	0.6390	0.6080	0.5850	0.5550
	η	0.7080	0.7120	0.6840	0.6560	0.6320	0.5810	0.6380	0.6080	0.5860	0.5550
	1.1η	0.7080	0.7120	0.6840	0.6560	0.6300	0.5780	0.6390	0.6100	0.5850	0.5580
	1.2η	0.7060	0.7120	0.6840	0.6560	0.6340	0.5780	0.6370	0.6100	0.5850	0.5590
Precision	0.8η	0.7019	0.7069	0.6811	0.6397	0.6265	0.6186	0.6874	0.6553	0.6273	0.6094
	0.9η	0.7083	0.7130	0.6811	0.6427	0.6301	0.6190	0.6892	0.6602	0.6325	0.6110
	η	0.7113	0.7130	0.6826	0.6447	0.6319	0.6210	0.6881	0.6610	0.6373	0.6140
	1.1η	0.7113	0.7144	0.6826	0.6470	0.6301	0.6177	0.6892	0.6637	0.6379	0.6238
	1.2η	0.7089	0.7144	0.6841	0.6448	0.6410	0.6181	0.6889	0.6660	0.6398	0.6265
PredProp	0.8η	1.0260	1.0160	1.0140	1.0400	1.0240	0.9720	0.9470	0.9300	0.9310	0.9180
	0.9η	1.0200	1.0120	1.0140	1.0380	1.0160	0.9710	0.9440	0.9250	0.9290	0.9140
	η	1.0180	1.0120	1.0120	1.0360	1.0140	0.9680	0.9440	0.9240	0.9240	0.9100
	1.1η	1.0180	1.0100	1.0120	1.0320	1.0160	0.9690	0.9440	0.9230	0.9210	0.9020
	1.2η	1.0200	1.0100	1.0100	1.0340	1.0060	0.9680	0.9410	0.9200	0.9180	0.8990
ReconRate	0.8η	0.9990	0.9980	0.9971	0.9962	0.9954	0.9975	0.9967	0.9952	0.9941	0.9924
	0.9η	0.9990	0.9982	0.9971	0.9962	0.9953	0.9975	0.9967	0.9952	0.9943	0.9924
	η	0.9990	0.9982	0.9971	0.9961	0.9953	0.9975	0.9967	0.9952	0.9942	0.9924
	1.1η	0.9990	0.9982	0.9971	0.9961	0.9953	0.5951	0.6621	0.6350	0.6094	0.5879
	1.2η	0.9990	0.9982	0.9971	0.9961	0.9951	0.9975	0.9967	0.9952	0.9942	0.9922
JSD	0.8η	0.0022	0.0016	0.0009	0.0010	0.0007	0.0043	0.0025	0.0021	0.0021	0.0023
	0.9η	0.0022	0.0008	0.0009	0.0014	0.0008	0.0043	0.0026	0.0022	0.0021	0.0024
	η	0.0022	0.0008	0.0008	0.0014	0.0008	0.0043	0.0026	0.0023	0.0023	0.0025
	1.1η	0.0022	0.0008	0.0008	0.0014	0.0008	0.0043	0.0026	0.0023	0.0023	0.0030
	1.2η	0.0022	0.0008	0.0008	0.0015	0.0010	0.0043	0.0026	0.0024	0.0024	0.0030

Appendix C: Supplementary results

Table S2. Performance comparison of different methods for varied error rates (*err*) on simulated data.Performance comparison of aBayesQR, ShoRAH, ViQuaS and PredictHaplo in terms of *Recall*, *Precision*, *Predicted Proportion* (*PredProp*), *Reconstruction Rate* (*ReconRate*) and *JSD* on the simulated data with $div = 3\%$ and $cov = 500\times$ vs. *err* for a mixture of 5 and 10 viral strains. Averaged PredictHaplo results are reported if it provides answers for more than 50% of data sets. Boldface value indicate the best performance for each *err*(%).

		5 strains					10 strains				
	err(%)	0.1	0.2	0.3	0.4	0.5	0.1	0.2	0.3	0.4	0.5
Recall	aBayesQR	**0.6840**	**0.5920**	**0.5840**	**0.5400**	**0.4840**	**0.6080**	**0.5150**	**0.4620**	**0.4200**	**0.3570**
	ShoRAH	0.1300	0.1420	0.1280	0.1300	0.1060	0.0740	0.0490	0.0570	0.0850	0.0880
	ViQuaS	0.6040	0.5300	0.4740	0.4160	0.3540	0.3730	0.2890	0.2200	0.1960	0.1590
	PredictHaplo	-	-	-	-	-	0.1021	0.1031	0.1072	0.1000	0.1194
Precision	aBayesQR	**0.6826**	**0.5954**	**0.6105**	**0.5565**	**0.4958**	**0.6610**	**0.5852**	**0.5377**	**0.5031**	0.4311
	ShoRAH	0.1240	0.1425	0.1274	0.1260	0.1057	0.0498	0.0318	0.0370	0.0516	0.0547
	ViQuaS	0.4559	0.3505	0.2757	0.2067	0.1453	0.2973	0.1889	0.1437	0.1126	0.0634
	PredictHaplo	-	-	-	-	-	0.4509	0.4287	0.4502	0.4325	**0.4787**
PredProp	aBayesQR	**1.0120**	1.0340	**0.9940**	**1.0000**	**1.0140**	**0.9240**	**0.8930**	**0.8780**	**0.8660**	**0.8650**
	ShoRAH	1.0780	**1.0160**	1.0360	1.0580	1.0520	1.6710	1.7460	1.7690	1.8500	1.7950
	ViQuaS	1.4080	1.6860	1.9700	2.3420	2.8280	1.5450	1.8430	1.9540	2.1550	2.5680
	PredictHaplo	-	-	-	-	-	0.1947	0.2000	0.2000	0.1970	0.2082
ReconRate	aBayesQR	**0.9971**	**0.9963**	**0.9957**	**0.9950**	**0.9937**	**0.9952**	**0.9941**	**0.9937**	**0.9925**	**0.9917**
	ShoRAH	0.9891	0.9884	0.9884	0.9879	0.9867	0.9899	0.9891	0.9891	0.9889	0.9887
	ViQuaS	0.9917	0.9923	0.9912	0.9829	0.9805	0.9899	0.9886	0.9879	0.9865	0.9860
	PredictHaplo	-	-	-	-	-	0.9850	0.9850	0.9848	0.9854	0.9848
JSD	aBayesQR	**0.0008**	**0.0018**	**0.0027**	**0.0028**	0.0045	**0.0023**	**0.0035**	**0.0041**	**0.0055**	**0.0060**
	ShoRAH	0.0047	0.0029	0.0038	0.0066	**0.0041**	0.0422	0.0499	0.0493	0.0522	0.0529
	ViQuaS	0.0222	0.0294	0.0384	0.0487	0.0703	0.0495	0.0574	0.0652	0.0696	0.0862
	PredictHaplo	-	-	-	-	-	0.1971	0.2024	0.1861	0.1973	0.1908

In Table S2, we report the results of the study of the effects sequencing errors have on the performance of aBayesQR, ShoRAH, ViQuaS and PredictHaplo. In particular, the sequencing error is varied from 0.1% to 0.5% (specifically, $err \in \{0.1\%, 0.2\%, 0.3\%, 0.4\%, 0.5\%\}$), reflecting the range of errors observed in the current and anticipated in future NSG technologies (e.g., the error rates of Illumina's Miseq have been reported to be below 0.4% in [26] and as high as 0.49 in [27]). We set *div* to be 3%[2] and set *cov* to be 500× which is the same as in the first experiment in 3.1. In this set of experiments, aBayesQR outperforms ShoRAH, ViQuaS and PredictHaplo over the considered range of *err* achieving the best scores overall for all 5 metrics. As expected, the performances of all methods deteriorate as *err* increases. Since PredictHaplo failed to generate results in most of the instances of the reconstruction problem involving a mixture of 5 strains, its results are not reported. For the problem involving a mixture

[2] This matches typical variations in the HIV *pol* gene which range between 3% and 5% ([5]).

with 10 strains, PredictHaplo did run successfully in most of the instances but significantly underdetermined the number of strains; on average, its *Predicted Proportion* is around 0.2 (while its *Precision*, as argued earlier in the paper, is somewhat misleadingly good). ViQuaS overestimates the number of strains in all instances; we observe that as *err* increases, ViQuaS generates increasingly more false negative viral strains which adversely affects *Precision* and *JSD*. Even though ShoRAH exhibits the lowest perfect reconstruction scores, it achieves better performance than ViQuaS in terms of frequency estimation and the number of reconstructed strains (i.e., *Predicted Proportion* and *JSD*) when applied to reconstruction of mixtures with 5 strains. For a mixture of 10 strains, however, ShoRAH overestimated the number of strains, which also leads to higher *JSD* scores.

Table S3. Performance comparison of different methods for varied coverages (*cov*) on simulated data. Performance comparison of aBayesQR, ShoRAH, ViQuaS and PredictHaplo in terms of *Recall*, *Precision*, *Predicted Proportion* (*PredProp*), *Reconstruction Rate* (*ReconRate*) and *JSD* on the simulated data with $div = 3\%$ and $err = 0.3\%$ vs. *cov* for a mixture of 5 and 10 viral strains. Averaged PredictHaplo results are reported if it provides answers for more than 50% of data sets. Boldface value indicate the best performance for each $cov(\times)$.

		5 strains			10 strains		
	$cov(\times)$	250	500	750	250	500	750
Recall	aBayesQR	**0.4440**	**0.5840**	**0.5920**	**0.4450**	**0.4620**	**0.4450**
	ShoRAH	0.1820	0.1280	0.0900	0.0590	0.0570	0.3220
	ViQuaS	0.4220	0.4740	0.4840	0.1970	0.2200	0.2580
	PredictHaplo	-	-	-	0.1070	0.1072	0.1094
Precision	aBayesQR	**0.4231**	**0.6105**	**0.6393**	**0.5131**	**0.5377**	**0.5195**
	ShoRAH	0.1831	0.1274	0.0979	0.0544	0.0370	0.2136
	ViQuaS	0.2291	0.2757	0.3256	0.0965	0.1437	0.1900
	PredictHaplo	-	-	-	0.4558	0.4502	0.4635
PredProp	aBayesQR	1.0920	**0.9940**	**0.9400**	**0.8840**	**0.8780**	**0.8800**
	ShoRAH	**1.0140**	1.0360	1.0660	1.1200	1.7690	1.6440
	ViQuaS	1.9240	1.9700	1.6820	2.2940	1.9540	1.7180
	PredictHaplo	-	-	-	0.2030	0.2000	0.2031
ReconRate	aBayesQR	**0.9941**	**0.9957**	**0.9959**	**0.9939**	**0.9937**	**0.9930**
	ShoRAH	0.9906	0.9884	0.9854	0.9874	0.9891	0.9926
	ViQuaS	0.9905	0.9912	0.9918	0.9861	0.9879	0.9881
	PredictHaplo	-	-	-	0.9854	0.9848	0.9854
JSD	aBayesQR	0.0040	**0.0027**	**0.0026**	**0.0041**	**0.0041**	**0.0049**
	ShoRAH	**0.0021**	0.0038	0.0100	0.0051	0.0493	0.0330
	ViQuaS	0.0454	0.0384	0.0318	0.0839	0.0652	0.0555
	PredictHaplo	-	-	-	0.1946	0.1861	0.1930

In Table S3, we report the performance of the proposed algorithm for different coverage per strain, $cov \in \{250\times, 500\times, 750\times\}$, while fixing other parameters

Table S4. Running time comparisons (sec). Running time comparisons of aBayesQR, ShoRAH, ViQuaS and PredictHaplo on the simulated data with $cov \in \{250\times, 500\times, 750\times\}$, div=3% and err=0.3%, measured on a Linux OS desktop with 3.06GHz CPU and 8Gb RAM (Intel Core i7 880 processor). PredictHaplo results are shown if it provides answers for more than 50% of data sets.

	5 strains			10 strains		
$cov(\times)$	250	500	750	250	500	750
aBayesQR	96	113	236	451	739	1606
ShoRAH	559	2005	5265	2716	11473	22923
ViQuaS	93	337	718	360	1300	13350
PredictHaplo	-	-	-	93	143	187

– specifically, diversity is set to 3%, which is the same as in the second experiment in Table S2, and sequencing error rate is set to 0.3%, which emulates the error rates of Illumina's Miseq ($<$ 0.4% [26]). Performance of four algorithms as a function of coverage is compared in Table S3, demonstrating superiority of aBayesQR in all five metrics of interest.

Runtimes of each of the algorithms applied to this test set as a function of cov are shown in Table S4; note that this characterization of speed (i.e., complexity vs. cov) is the most meaningful one to study since the coverage is a main factor affecting the runtime of performing a reconstruction task. The speed is measured on a Linux OS desktop with 3.06GHz CPU and 8GbRAM (Intel Core i7 880 processor). When it completes the task and provides a solution, PredictHaplo is the most efficient among all schemes; however, this method fails to provide answers in most of instances on a mixture of 5 strains. Among the remaining 3 algorithms, our aBayesQR demonstrates the best time efficiency for $cov \geq 500$ while ShoRAH is the slowest one. ViQuaS is relatively fast at the low coverage $cov = 250$ but its time complexity appears to grow exponentially as cov increases, especially in the setting with a mixture of 10 viral strains.

Table S5. Performance comparisons on a real HIV-1 5-virus-mix data set. *Predicted Proportion* (PredProp), *Reconstruction Rate* (RR) and inferred frequencies (F) for aBayesQR, ShoRAH and PredictHaplo applied to reconstruction of HIV-1_{HXB2}, HIV-$1_{89.6}$, HIV-$1_{JR\text{-}CSF}$, HIV-$1_{NL4\text{-}3}$ and HIV-1_{YU2} for all 13 genes of the HIV-1 dataset (note: frequencies are reported in parenthesis, both RR and F are expressed in percentages).

		p17	p24	p2-p6	PR	RT	RNase	int	vif	vpr	vpu	gp120	gp41	nef
aBayesQR	PredProp	1	1	1	1	1	1	1	1	1.2	1	0.8	0.8	1.2
	RR(F)$_{HXB2}$	100(16.3)	99.4(21.1)	100(22.2)	100(12.5)	98.5(24.3)	100(16.1)	99.9(9.7)	100(9.2)	100(16.4)	99.6(17)	98(30.3)	0(0)	95.8(11.4)
	RR(F)$_{89.6}$	100(27.1)	98.7(17)	100(17.3)	100(17.3)	98.6(18.1)	100(19.7)	100(22.2)	100(20.6)	100(16.3)	92(10.4)	96.5(20.2)	98.9(23.7)	95.5(16.4)
	RR(F)$_{JR\text{-}CSF}$	100(31.3)	99.6(24.6)	100(25.8)	100(29.9)	99(21.5)	100(22.1)	100(20.8)	100(32.7)	100(27)	98.8(26.7)	97.7(21.4)	99.1(29.7)	98.2(21.1)
	RR(F)$_{NL4\text{-}3}$	100(12.9)	100(21.6)	100(25.6)	100(20.1)	98.9(17.7)	100(30)	100(39.5)	99.8(28.5)	100(23.2)	100(41.3)	96.3(28)	98.8(36.6)	100(31.8)
	RR(F)$_{YU2}$	100(12.4)	99.7(15.8)	100(9.2)	100 (20.3)	99.2 (18.5)	100(12.2)	99.5(7.9)	99.7(9)	100(17.1)	100(4.6)	0(0)	98.6(10.1)	99.2(14)
ShoRAH	PredProp	13	16.4	13.8	8.8	21.8	11.8	13.6	12.8	7.8	4	23.8	19.8	17.4
	RR(F)$_{HXB2}$	100(9.41)	96.7(6.3)	100(6.2)	100(7.8)	98.2(5.8)	100(8.9)	97.5(9.2)	100(5.9)	100(8)	100(8.2)	97.7(15.7)	98.4(4.6)	98.2(14)
	RR(F)$_{89.6}$	100(8.97)	99.7(10.7)	100(15.8)	100(14.5)	98.6(4.6)	100(11.3)	98.9(11.1)	99.8(12.5)	100(6.2)	93.6(14.6)	96.1(9)	98.6(8.4)	98.9(9.7)
	RR(F)$_{JR\text{-}CSF}$	100(26.2)	100(13.6)	100(10.7)	100(10.2)	99.8(7.2)	96.4(7.3)	99(7.6)	100(14.5)	100(16.9)	98(20.14)	96.9(9.8)	96.3(6.6)	94.7(4.3)
	RR(F)$_{NL4\text{-}3}$	100(9.4)	99.1(5.2)	97.3(7.3)	100(7.8)	98.9(4.4)	99.2(14.8)	99.3(4.8)	99.3(7.3)	100(22.9)	100(20.2)	96.1(4.4)	98.5(4.8)	98.6(1 7.1)
	RR(F)$_{YU2}$	94.2(6.1)	99(5.5)	100(8)	98.3(8.1)	98(7.2)	94.5(9.2)	98.6(6)	95(6.6)	93.2(7.6)	90.8(7.9)	97(4.7)	95.4(7.1)	97.9(5.1)
PredictHaplo	PredProp	1	0.6	1	1	1	0.8	0.8	0.8	1	0.8	0.8	0.8	0.8
	RR(F)$_{HXB2}$	100(17.8)	0(0)	100(18.7)	100(15.2)	100(12.2)	98.9(25.4)	100(12.1)	100(17.7)	100(10.2)	93.17(10.8)	0(0)	0(0)	0(0)
	RR(F)$_{89.6}$	100(19.9)	100(46.4)	100(21.7)	100(22.2)	100(19.4)	100(18.2)	99.8(27.6)	100(20.9)	100(22.1)	0(0)	97.8(20.7)	100(26.7)	98.87(20.7)
	RR(F)$_{JR\text{-}CSF}$	100(31.9)	100(21.8)	100(30.3)	100(26.9)	100(23.4)	100(23.2)	100(22.3)	100(24.9)	100(23.7)	100(34.1)	99.7(42.7)	100(28.9)	100(23.2)
	RR(F)$_{NL4\text{-}3}$	100(17)	99.1(31.8)	100(16.4)	100(20.9)	100(30.2)	100(33.2)	100(38.1)	100(36.6)	100(35.5)	100(47.1)	100(28.6)	100(32.7)	100(39.3)
	RR(F)$_{YU2}$	100(13.4)	0(0)	100(12.9)	100(14.8)	100(14.7)	0(0)	0(0)	0(0)	100(8.5)	100(7.9)	98.6(7.9)	100(11.7)	100(16.9)