We thank the reviewers for their feedback. In the following, we first address questions brought up by multiple reviewers.

Q1: Why does CAECseq perform better than GAEseq? As mentioned in the paper, there are three reasons: (1) Convolutional layers allow CAECseq to capture spatial relationships (correlations) between SNPs. (2) Mini-batch stochastic gradient descent helps CAECseq escape local optima while full gradient descent may cause GAEseq to get stuck in local optima. (3) Nucleotide representation used by CAECseq (one-hot encoding) means the distances between nucleotides are symmetric while the representation via integers used by GAEseq leads to unjustified asymmetry.

Q2: Where does the KL loss under the p distribution come from? How can it indirectly minimize the MEC loss while allowing for minibatch optimization? The KL loss, $L_c = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$, is equivalent to categorical cross entropy; here $[p_{ij}]$ form a k-dimensional standard unit vector and $\sum_j q_{ij} = 1$. As explained in the paper, p_{ij} are obtained by reassigning read origins to minimize the MEC loss at each epoch (using all the reads) while q_{ij} is acquired from the clustering layer at each iteration (using mini-batch of reads). Therefore, mini-batch optimization enables updating q_{ij} at each iteration, and minimizing the KL loss indirectly minimizes the MEC loss.

Reviewer 1 Q3: What happens if the clusters don't get initialized using k-means? We will use the p17 region of HIV-1 as an example. In the paper, we reported that on this region CAECseq achieves the MEC score and CPR of 34036 and 100%, respectively. Without the k-means initialization, these deteriorate to 115134 and 54.2%, respectively. Q4: [To verify that CAECseq captures spatial relationships between SNPs, run a simple experiment] on a training / test set in which the SNP matrices all have their column indices shuffled consistently. We did so on a randomly shuffled SNP fragment matrix of the p17 region in HIV-1; the resulting MEC score and CPR are 206475 and 34.6%, respectively. Therefore, random shuffling destroyed the spatial relationship between SNPs that the convolutional layers originally captured. Q5: What are additional example tasks? Is there a potential application for this method to GWAS data? An additional example task is anomaly detection, e.g., in an application to viral sequence classification. Applications to GWAS data are certainly possible with appropriately formatted input data and may be a part of our future work.

Reviewer 2 Q6: The main weakness is the limited methodological novelty of the proposed method. The paper presents the first ever *deep learning* architecture (autoencoder with a clustering layer) for a pair of challenging problems in bioinformatics. The method incorporates domain knowledge in a novel and unique way and enables unprecedented accuracy. We anticipate it will be very valuable in practice as it outperforms classical approaches by orders of magnitude and is orders of magnitude faster than the only other existing neural network based method (GAEseq, a shallow architecture). Q7: The AE seems to take aligned reads as inputs, but the text only mentions reads (which could also be unaligned). We explicitly state that the reads are mapped to a known reference genome (please see Section 2.1, line 128). Q8: An empty cell in the example read in Fig. 2 is confusing. The empty space models gaps in coverage of paired-end reads. Q9: What is the impact of the gamma parameter? We tested CAECseq for γ varying from 0.01 to 0.99. A large γ distorts the feature space by reducing the ability of the convolutional AE to learn salient features of reads while a small γ implies that the model does not put much effort in the reconstruction task. Q10: Would the results be very different using only the k-means step for clustering? Yes. E.g., on p17 region of HIV-1, k-means achieve the MEC score and CPR of 152464 and 46.6%, while CAECseq on the same task achieves 34036 and 100%, respectively.

Reviewer 3 Q11: In what sense is the low-dimensional embedding "stable"? We refer to a low-dimensional embedding as being stable if it helps minimize the MEC score when the clustering layer is employed. Q12: A key methodological innovations here is the inclusion of a clustering layer. This should be explained [...] A cite should be given for PReLU. Thanks, we are committed to making these updates! Q13: I could not understand this sentence (line 170-171). A clarification: if the dimension of the learned features is larger than the length of haplotypes, the auto-encoder only learns to copy the input (i.e., $f(x) = x$, thus learning non-informative features). Q14: "The reported results were obtained on test data" (line 207) is insufficiently clear. We tuned the hyper-parameters and validated them on ten simulated tetraploid datasets, and then applied them to all the datasets in the paper. Therefore, all the datasets in the paper are test data. We do not split datasets into training, validation and testing parts because such splitting reduces sequencing coverage and thus reduces reconstruction accuracy. Q15: Why the GAEseq method couldn't be trained using minibatches. Calculation of the MEC score, which GAEseq aims to directly minimize, requires using all the reads at each iteration. Q16: "enables the proposed method to distinguish reads obtained from highly similar genomic components" (line 76-77) is not very clear and is not supported by any evidence. We compared the performance of CAECseq with other SOTA methods on simulated viral quasispecies data with diversity from 1% to 10% in Supplementary Document D.

Reviewer 4 Please see our answers to questions Q1, Q2 and Q14. Q17: Clarify the choices of parameters. Mapping quality scores 40, 60 and read length 150 bp are standard in literature and practice of haplotype assembly. For viral data, higher mapping quality score (60) is needed since the assembly task is generally more challenging. Q18: Why did the authors choose the subset they did to compare against? Extensive literature review reveals that GAEseq, HapCompass, H-PoP, AltHap, TenSQR are state-of-the-art methods that outperform other existing techniques in terms of accuracy; moreover, while other methods are restricted to bi-allelic diploid data, these can handle multi-allelic polyploid data.