Biometrics 63, 1–??
December 2007
DOI: 10.1111/j.1541-0420.2005.00454.x

Using Dirichlet mixture model to detect concomitant changes in allele frequencies

Wen J. Shi[1,*] , Jan Hannig[1,] , and Corbin D. Jones[2,***]**

[1]Department of Statistics and Operations Research, UNC at Chapel Hill, North Carolina, U.S.A.

[2]Department of Biology and Carolina Center for Genome Science, UNC at Chapel Hill, North Carolina, U.S.A.

*email: wjshi@live.unc.edu

**email: jan.hannig@unc.edu

***email: cdjones@email.unc.edu

Summary: RNA viruses are challenging for protein and nucleotide sequence based methods of molecular evolutionary analysis because of their high mutation rates and complex secondary structures. With new DNA and RNA sequencing technologies, viral sequence data from both individuals and populations are becoming easier and less expensive to obtain. Population genomic and evolutionary studies of viruses and their hosts using these new technologies are becoming increasingly common. For these studies there is a critical need for methods that can identify alleles whose frequencies change over time or due to a treatment. We have developed a novel statistical approach for identifying evolved nucleotides and/or amino acids in a viral genome without relying on sequence annotation or the nature of the change. Instead our method identifies nucleotides that have similar patterns of change over time. Our approach models allelic variance under a Bayesian Dirichlet mixture distribution. We have developed an efficient multi-stage clustering procedure that distinguishes treatment causal changes from variation within viral populations. After validation using simulations and a well characterized HIV-1 data set, we applied our method to a longitudinal time-sampled influenza A H1N1 virus strain in either the absence of presence of oseltamivir. Along with the most common H1N1 oseltamivir resistance mutation H274Y on segment 6, we found another genomic location on segment 8 with strong evidence of treatment effect and a list of sites with high allelic variation in the untreated environment. We believe that our approach can be broadly applied and is particularly useful for the data from genomes that are recalcitrant to traditional sequence analysis.

Key words: Objective Bayes; Markov chain Monte Carlo methods; Parallel computing; High-throughput sequencing.

This paper has been submitted for consideration for publication in *Biometrics*

1. Introduction

RNA viruses and retroviruses, such as SARS, influenza, hepatitis C, polio, and HIV, use RNA as their genetic material. The RNA polymerases of these viruses lack the proof-reading ability of DNA polymerases, which results in a high rate of mutation and of genome evolution. This rapid rate of evolution can be advantageous for the virus as it can evolve away from immune system pressure and evolve resistance to antiviral drugs (Rambaut et al., 2004; Boutwell et al., 2010; Jabara et al., 2014).

Phylogenetic and molecular evolutionary methods are commonly applied to viral genes or genomes to investigate RNA virus evolution at a molecular level (Norstrm et al., 2012). However, the high mutation rate and the complex secondary structures of RNA viruses genomes often compromise these sequence based methods (Simmonds and Smith, 1999; Damgaard et al., 2004; Watts et al., 2009; Cuevas et al., 2012). These aspects of viral biology make it hard for current methods to tease apart the evolutionary signal of adaptation, such as evolution of drug resistance, from the signal of neutral evolutionary processes, such as genetic drift. Further complicating analysis are compensatory mutations that offset structural defects and other pleiotropic costs of adaptive alleles, which often arise and sweep to fixation in viral populations (Knies et al., 2008). Thus there is a clear need for analytical methods that are robust to these complications, make minimal assumptions as to how the virus should evolve, and can identify regions of the viral genome that have changed over time in response to treatment.

Recent advances in sequencing technology have produced a wealth of new viral sequence data, which has amplified the need for new analytical tools, e.g. (Jabara et al., 2011; Shao et al., 2013). Increasingly, populations of thousands of viruses are sampled and sequenced from an infected individual. This approach captures a snapshot of the viral genetic variation within that individual. A few studies have combined this approach with traditional passage

experiments or with sampling during the course of an infection (Leitner et al., 1993; Eriksson et al., 2008; Kuroda et al., 2010; Wright et al., 2010). This powerful experimental design reveals how a population of viruses genomically responds to evolutionary pressure. With the ever-decreasing cost of sequencing, these studies are expected to become commonplace. These studies, however, have technical limits (Beerenwinkel et al., 2012). High sequencing error rate, variable sequencing depth, PCR bias, sequencing bias, etc. complicate analysis. Methods of analysis must be robust to these technical issues.

Our motivating dataset was a study of influenza A H1N1 viruses (IVA) response to an inhibitor of neuraminidase, oseltamivir (a.k.a. Tamiflu). Oseltamivir has been used both for prevention and treatment of influenza viruses. It prevents the virus from budding from the host cell, thereby slowing viral reproduction. How the IVA respond to oseltamivir at the genomic level has not been fully understood, thus our goal was to find the genomic regions of the virus that evolved in response to oseltamivir. The dataset contains replicate populations of IVA sampled over many generations ("passages") in the presence and absence oseltamivir (Ghedin et al., 2012; Renzette et al., 2014). The IVA were first adapted from chicken eggs to Madin-Darby canine kidney (MDCK) cells for three passages. Then the samples were serially passaged in MDCK cells in either the absence or presence of oseltamivir in replicated experiments (Figure 1). At the end of each passage, whole-genome high throughput sequencing data were collected from the viral populations (Renzette et al., 2014).

[Figure 1 about here.]

RNA viruses evolve rapidly even within the untreated group. It is important to distinguish genetic changes selected for by the inhibitor from those that arise due to other population genetic forces. The time series data and control-treatment setup provides multiple samples for the virus populations with and without the administration of oseltamivir. The two biological replicates allow us to crosscheck sites for drug resistance. We take advantage of the replicated

longitudinal data to develop a novel statistical approach for identifying evolved nucleotides in a viral genome without relying on sequence annotation or the nature of the change (non-synonymous or synonymous; transition or transversion).

Our approach analyzes multiple time-sampled observations simultaneously, models viral sequence position indices under a Bayesian Dirichlet mixture distribution, performs a series of clustering algorithms, and then identifies treatment causal substitution sites via comparing the before and after treatment posterior distributions for the corresponding regions on the viral genome. Our algorithm also allows us to identify genomic locations that have similar patterns of change.

We first validated our approach with synthetic test data. We then used a well-studied HIV-1 data set (Jabara et al., 2011) as a positive control. We showed that our approach correctly identifies key changes that have been experimentally shown to be important to drug resistance in HIV-1. Finally, we applied our method to the longitudinal time-sampled IVA data in the absence and presence of oseltamivir (Renzette et al., 2014; **?**). We identified two genome sites (S6-822 & S8-80) that presented the greatest evidence of drug resistance along with a set of locations might have been affected by adaptation to the host or genetic drift.

The next section, Section 2 describes the viral genome data and the Bayesian framework used to model the viral populations. Section 3 introduces a three-step sequential approach we have developed to identify treatment causal substitutions. Simulation results are presented in Section 4. Section 5 presents application of our method to a well described HIV-1 dataset as a proof of concept, followed by the analysis of the IVA dataset. Section 5 concludes the chapter with a few remarks and a discussion.

2. Parametric Bayesian Mixture Framework

2.1 *Sequencing Data*

Advances in high-throughput whole genome shotgun sequencing allow deep genome sequencing of viral populations within a host (Muers, 2011). This technology produces millions of short DNA or RNA sequences. These sequences are aligned to a reference genome and differences between the reference and sequenced population are noted. With this advanced shotgun sequencing method, we are able to combine the reads from each individual and work with data with the following form:

[Table 1 about here.]

Letters A, C, G, T, M stand for five possible read types in this toy example: Adenine, Cytosine, Guanine, Thymine, Missing/deleted data, respectively. Left-hand side of Table 1 illustrates a high through-put sequencing alignment result. Its read-specific compressed view is shown in the right panel of Table 1. Counts of each read type (A, C, G, T, M) at the i^{th} position are recorded as $Y_i = (y_i^1, y_i^2, y_i^3, y_i^4, y_i^5)$.

2.2 *Dirichlet Mixture Model*

To describe the genomic site specific variation residing within a viral population we constructed a parametric Bayesian mixture model based on observed nucleotide read counts. Assume that total number of read types is J. Given the probability parameters, the collection of different read counts at each genomic site is assumed to follow a J-dimensional multinomial distribution. For an arbitrary i^{th} position on the sequence, the probabilities of having each of the J read types are denoted as $P_{c_i} = (p_{c_i}^1, p_{c_i}^2, \cdots, p_{c_i}^J)$. Every p_{c_i} lies between 0 and 1; their sumation $\sum_{j=1}^{J} p_{c_i}^j = 1$. We assume a finite collection of K possible probability parameters, $\boldsymbol{P} = \{P_1, \cdots, P_K\}$, each genomic site could take on, i.e. every P_{c_i} is a member of $\boldsymbol{P}$. The subscript c_i is an assignment indicator denoting which probability parameter in the set $\boldsymbol{P}$ the

i^{th} genomic site is associated with, $c_i \in \{1, \cdots, K\}$. The number of elements in $\boldsymbol{P}$, K, is the number of mixture components in the Bayesian mixture framework. Because many sites in the genome sequence share the same tendencies of having certain kinds of genetic vairation (as captured by the reads), it is intuitive that K is much smaller than the length of the viral sequence of interest, N. Furthermore, a weakly informative symmetric Dirichlet prior is applied to all the elements of $\boldsymbol{P}$ to ensure probability properties of P_k's, $k = 1, \cdots, K$. With total J possible read types, a corrected Perks prior, Dirichlet $(\frac{1}{J^2}, \frac{1}{J^2}, \cdots, \frac{1}{J^2})$ is chosen for the multinomial parameters. The corrected Perks prior reduces the prior strength (concentration) by a factor proportional to the number of categories of the multinomial to ensure that the Bayesian estimator is preferred to maximum likelihood estimators for the parameters (Walley, 1996; de Campos and Benavoli, 2011). With an additional assumption that there is an equal chance of getting any P_k in $\boldsymbol{P}$, we construct the following hierarchical Dirichlet mixture model:

$$\begin{aligned} Y_i|c_i, \boldsymbol{P} &\overset{indep}{\sim} \textit{Multinomial}(m_i; P_{c_i}) \\ c_i|\boldsymbol{P} &\overset{iid}{\sim} \textit{Uniform Discrete}\left(\frac{1}{K}\right) \\ P_k &\overset{iid}{\sim} \textit{Dirichlet}\left(\frac{1}{J^2}, \frac{1}{J^2}, \cdots, \frac{1}{J^2}\right) \end{aligned}$$

where m_i indicates the total number of reads observed at the i^{th} position, i.e. $\sum_{j=1}^{J} y_i^j = m_i$. Component number K is some fixed unknown integer. Integrating the posterior density $\pi(c_1, \cdots, c_N, \boldsymbol{P}|Y_1, \cdots, Y_N)$ over $\boldsymbol{P}$, the marginal posterior for the assignments given reads on the sequences is

$$\pi(c_1, \cdots, c_N|Y_1, \cdots, Y_N) = \frac{1}{h(Y_1, \cdots, Y_N)} \prod_{k=1}^{K} \frac{\prod_{j=1}^{J} \Gamma\left(\sum_{i=1}^{N} y_i^j \mathbf{1}_{\{c_i=k\}} + \frac{1}{J^2}\right)}{\Gamma\left(\sum_{i=1}^{N} m_i \mathbf{1}_{\{c_i=k\}} + \frac{1}{J}\right)}, \tag{1}$$

where $h(Y_1, \cdots, Y_N)$ is the normalizing constant.

Furthermore, if both read counts and assignments are given for the entire sequence sample, we have

$$P_k|c_1, \cdots, c_N, Y_1, \cdots, Y_N \overset{indep}{\sim} \textit{Dirichlet}\left(\alpha_k^1, \alpha_k^2, \cdots, \alpha_k^J\right), \tag{2}$$

where $\alpha_k^j = \sum_{i=1}^{N} y_i^j \mathbf{1}_{\{c_i=k\}} + \frac{1}{J^2}$, for $j = 1, 2, \cdots, J$; and $k = 1, 2, \cdots, K$.

In the methodology section we will introduce a sequence of efficient Markov chain Monte Carlo (MCMC) procedures used to cluster the genome sequence positions and generate assignment labels c_i's for each viral genome site. Notice that the posterior distribution (1) is defined for a fixed mixture component number K. One may choose K *ad hoc*, however, if the chosen K is smaller than the real number of mixture components, at least one resulting cluster contains members from multiple true clusters ; if the chosen K is too large, the clustering procedure can be infeasible due to the high dimensionality of most genome sequence data. At every iteration of the MCMC updating step, one coordinate or a class of coordinates will be altered into one of the K possible assignments. As K increases, the probability of assigning the correct label to each position decreases. Equation (1) naturally places an AIC-like penalty on non-empty clusters. It encourages empty groups by scaling the marginal posterior $\pi(c_1, \cdots, c_N|Y_1, \cdots, Y_N)$ by $\left[\Gamma\left(\frac{1}{J^2}\right)\right]^J$. This shrinkage property allows our algorithm to start with a liberal upper bound of component number instead of the truth and naturally reduces it to a close upper bound of K. In Section 3 we will introduce a tree-like MCMC step that provides the liberal upper bound and a block-MCMC procedure that produces a reasonably close upper bound of K. In Section 4 we will show through a simulation study that with the close upper bound of K, our algorithm correctly identifies the genomic regions with evolutionary changes.

3. Methodology

Consider the toy example where three data collections, baseline (t_1), pre-treatment (t_2), and post-treatment (t_{3D}), were obtained (Figure 2). To see if the ith genomic site has been affected by the treatment, we can compute the marginal posterior distributions for site i at all three time points: $\pi_i^{t_1}, \pi_i^{t_2}, \pi_i^{t_{3D}}$, perform pairwise comparison with a transformed Hellinger distance Ht (see supplementary), and check if the comparisons between the treated and

non-treated populations, $Ht(\pi_i^{t_1}, \pi_i^{t_{3D}})$ & $Ht(\pi_i^{t_2}, \pi_i^{t_{3D}})$, are much greater than the variation within the untreated group, $Ht(\pi_i^{t_1}, \pi_i^{t_2})$.

[Figure 2 about here.]

To perform the comparisons illustrated in Figure 2, we first need the group assignments $c_1, c_2, \cdots$ to compute the marginal posterior distributions. In general, we assume that the viral population was sampled and sequenced before and after the treatment. To see whether a genome site has been affected by the treatment, we cluster the genome sites, generate the assignment labels, derive its marginal posterior distribution for each site from each sample and compare the posteriors across time points and treatments. If a site shows significant change over time under treatment but not under control environment, it is identified as a substitution site due to treatment. The details of this procedure are described below.

3.1 *Preprocess*

Prior to the clustering procedure, we consolidate the count data by merging the genome sites with homogeneous read types. The detail of the preprocess step is listed in the supplementary.

3.2 *Processing*

After preprocessing the read counts, a series of MCMC methods are implemented to cluster the geomic locations and obtain the assignment labels $c_i's$ (Figure 3).

[Figure 3 about here.]

The first step is a "top down" hierarchical clustering with 2-means initial states (*hierarchical SCMH*) based on a two-component Single Coordinate updating Metropolis Hastings algorithm. Under the divisive hierarchical model, sibling nodes are mutually exclusive and complementary respect to their parent node. In the case that one child node is empty, that branch stops growing and its parent node is recorded as a leaf node. Eventually this branching process stops. A block Metropolis Hastings (*block MH*) step is then applied to the leaf nodes,

each treated as a block. After assessing convergence (e.g. Geweke diagnostic), T thinned-out iterations of the assignment labels are reserved. Finally, one run of a fixed scan Gibbs sampler is implemented on the joint data with the reserved assignment labels as initial states (*Gibbs*). At every stage of a MCMC, a label proposal for each site is given according to the posterior likelihood for the joint sequence if the label is assigned.

The divisive hierarichical clustering model allows us to avoid choosing a K, number of the mixture components. The total number of leaf nodes in the tree forms a reasonable upper bound for the number of mixture components regarding the entire joint data matrix. With sufficient number of observations at each site, m_i (*n.b.* most sequencing data have hundreds to thousands of sequencing reads), the Metropolis-Hasting splitting algorithm clusters correctly with probability one. This result is a direct consequence of Theorem 1 in supplementary.

Most genomes are lengthy, which results in high dimensional data. Direct application of multiple mixture component MCMC methods to this high dimensional data quickly becomes infeasible. Our hierarchical tree model enables efficient computing on this high dimesionality data. After the first split (at the root node) only a portion–typically only a small portion–of the original data set is analyze at a time. With this reduced input size, each Markov chain converges much faster. At the root node, although the entire joint sequence is used, there are only two possible values for each assignment label. Thus the Markov chain typically reaches convergence inexpensively. The *hierarchical SCMH* step is also easily parallelized for high performance computing systems. As a result, we have an efficient clustering procedure that automatically produces a component number upper bound and initial assignments for the block MH step.

In practice, the MCMC tree usually splits the data into too many groups. The *block MH* step, however, allows clusters to combine. The binary hierarchical clustering process therefore does not require the true number of components to be representable by binary clusters. The

shrinkage property of the marginal distribution (1) favors combining leaf nodes that belong to the same group. As a result of this natural penalty on non-empty groups, the number of distinct groups at the end of the *block MH* step is almost always much smaller than the total leaf number in the hierarchical tree. The *block MH* step in essence tunes the assignments for each genome site and reduces the total number of mixture components. As shown later in Section 4, our clustering algorithm with only the first two steps (*hierarchical SCMH* & *block MH*) produces reasonable results with slightly higher error rate, compared to the full algorithm.

Occasionally a few indices in some end nodes can be misplaced in the tree splitting step. Because all the indices in each leaf node are kept in the same cluster throughout the *block MH* process, those position indices do not get a chance to be moved to a different cluster. We solve this by adding a fixed scan Gibbs sampler step, *Gibbs*, that can modify the assignments for individual indices and move them to more appropriate clusters.

It is worth noting that when the preprocessed dataset is very length (i.e. many columns), even one scan of Gibbs can take a large amount of time to compute with standard methods. For this very reason, direct Gibbs sampler can become infeasible even if a the number of mixture components K is provided. Furthermore, one must choose such a K ad hoc and risk either having a small mixture number that always misgroups elements from multiple true clusters together or risk having a large mixture number that might make computation infeasible. Further, most sequencing data sets are large. Direct Gibbs sampler might be infeasible even with a small K. Because the *hierarchical SCMH* step enables parallel computing and the *block MH* step works with a dataset of smaller dimension than the original, the computational cost is much lower in comparison to direct MCMC approaches. In the simulation study section (Section 4), we will compare the result and computation time using our three-step clustering approach to a direct Gibbs sampler with several choices of K,

including the truth. We will show that our clustering is much more efficient, it outperforms the direct Gibbs sampler given the true K.

Alternatively, one can apply a Dirichlet process model to the joint dataset. A Dirichlet process model can be viewed as a Dirichlet mixture model with infinite number of components. With this framework, the point when cluster number stops growing depends heavily on the shrinkage power of the prior. Hence for a Dirichlet process model to work a more careful choice of prior is required. Intuitively, the computational time for the Dirichlet process is at best as good as a direct Gibbs sampler with the true K and a good initial state.

3.3 *Postprocess*

After implementing the three steps: *hierarchical SCMH*, *block MH*, and *Gibbs*, we obtain T running sets of assignment labels for the joint data. By reversing the preprocessing step each genome position gets an assignment label for each time point from each Gibbs result. The posterior distribution per genome position per time point can now be computed. For each position i, we use the transformed Hellinger distance, Ht, to compare posterior distributions before and after treatment. Given two time points t_{k_1}, t_{k_2}, a collection of Hellinger distance values are obtained from the clustering result for each location i. We take the median of those distance values and denote it as $Ht(\pi_i^{t_{k_1}}, \pi_i^{t_{k_2}})$. In principle, one can use another measure of center. We chose the median for its simplicity and straightforward interpretation. The summary statistic of the Hellinger distance between treated and non-treated times for location i is denoted as $Ht(D_i)$. Large values in $Ht(D_i)$ indicate evolutionary changes in the viral genome. Those changes can be caused by the treatment or non-treatment related reasons, such as genetic drift and adaptation to the host. To distinguish between these potential causes of changes, we denote a summary statistic $Ht(N_i)$ for the comparison between time points without treatment.

Exact form of the statistics $Ht(D_i)$ and $Ht(N_i)$ depends on the experimental design.

The basic idea is that, at genomic location i, $Ht(D_i)$ is the minimum change between the last treated time point and all pre-treatment times, while $Ht(N_i)$ is the maximum change among pairwise comparisons between untreated samples. At position i, if $Ht(D_i)$ is large, the last sampled population after treatment is significantly different from *all* samples before treatment; if $Ht(N_i)$ is large, *some* untreated sample is significantly different from *some* other untreated sample.

Intuitively, if the nucleotide read count distribution at site i has been affected by the treatment, $Ht(D_i)$ shall be *large*, relative to $Ht(N_i)$ and the comparisons for all the other sites that are not affected by treatment. How large is *large* will be determined by thresholding. For any given cutoff d, we define the following three sets:

$$S_1^d = \{i : \ Ht(D_i) > d\}, \quad S_2^d = \{i : \ Ht(N_i) > d \ \ \& \ \ Ht(N_i) > Ht(D_i)\}, \quad S_3^d = S_1^d \setminus S_2^d.$$

The first set, S_1^d, includes all the genomic locations that have large changes when comparing the treated and untreated groups. It is a *potential* set for substitutions. The second set concerns the large differences within the untreated group. It consists of all the genomic locations that have large variation which is not due to the treatment. The set S_2^d can be viewed as a *noise* set. Taking the set difference between the potential list and the noise list, the resulting set S_3^d is the list containing the *signals*. As d decreases, the sets S_1^d and S_2^d grow. We expect the growth of the sets accelerates as the cutoff moves from the signal to noise portion of the data. Therefore, the threshold d_0 used for inference is determined by the tipping point where the size of S_2^d starts to increase dramatically, as d decreases.

To illustrate the derivation of $Ht(D_i)$ and $Ht(N_i)$, we continue with the toy example. It can be easily generalized to more than three time point collections, with duplicates, or with a treatment-control setup (see Sections 4 & 5). Return to the toy example (Figure 2); the treatment is administrated after t_2; by time t_3 the viral population have completely responded. For each genome site i, we compare its marginal posterior distributions (Eq 2) at

time t_1, t_2, and t_{3D}, denoted as $\pi_i^{t_1}$, $\pi_i^{t_2}$, $\pi_i^{t_{3D}}$, respectively, using the transformed Hellinger distance (see Supplementary). Taking the clustering results from parallel chains, for a site i, we may define the summary statistics $Ht(D_i)$ and $Ht(N_i)$ as

$$Ht(D_i) = \min\{Ht(\pi_i^{t_1}, \pi_i^{t_{3D}}), Ht(\pi_i^{t_2}, \pi_i^{t_{3D}})\}, \quad Ht(N_i) = Ht(\pi_i^{t_1}, \pi_i^{t_2}), \quad \forall i. \tag{3}$$

If the change of read count distribution is caused by the treatment, the posterior distribution $\pi_i^{t_{3D}}$ ought to be much different from $\pi_i^{t_1}$ and $\pi_i^{t_2}$. Consequently, both $Ht(\pi_i^{t_1}, \pi_i^{t_{3D}})$ and $Ht(\pi_i^{t_2}, \pi_i^{t_{3D}})$ result in large values. Large $Ht(D_i)$ value guarantees that both $Ht(\pi_i^{t_1}, \pi_i^{t_{3D}})$ and $Ht(\pi_i^{t_2}, \pi_i^{t_{3D}})$ are large, it is therefore sufficient to look at $Ht(D_i)$.

Depending on the noise level of the data, the boundary of noise and signal portions of the data can be approximated by the curvature of noise set size function as the cutoff d decreases. The algorithm for determining the threshold can be found in the supplementary.

4. Simulation Study

In this section, we use simulations test our algorithm, with and without the *Gibbs* modification step, and compare its efficiency with direct Gibbs samplers. Five substitution sites were implemented at sites 1, 21, 41, 61 81. The detailed setup is included in supplementary.

[Figure 4 about here.]

For a simulated dataset (Figure 4), we analyzed the clustering results from both without (top panels) and with (bottom panels) the Gibbs modification step. The left two panels in Figure 4 show the sizes of sets S_1^d, S_2^d, S_3^d (potential, noise, signal) as threshold d decreases (zoom-in view). The dashed and the dotted lines are the thresholds obtained using our method at two different choices of α $(0.5, 0.25)$. At either threshold, without or with *Gibbs*, both S_1^d and S_3^d have five elements; S_2^d is empty. It is clear that a wide range of α parameter would produce different threshold, yet the same results here. The right two panels present the summary statistics $Ht(D_i)$ (small green circle) and $Ht(N_i)$ (blue cross) at each genome

position. The horizontal dashed and dotted lines correspond to the thresholds chosen in the left panels. Above the dashed line, five large red circles highlights the $Ht(D_i)$ corresponding to the signal sites in right panels. They show much larger values than the rest and reveal clear separation between signals and noise. The potential set, noise set, and signal set are:

$$S_1^{d0} = \{1, 21, 41, 61, 81\}, \quad S_2^{d_0} = \emptyset, \quad S_3^{d0} = S_1^{d_0}. \tag{4}$$

Compared between without and with the *Gibbs* step, the inference results are equally good for this simulated dataset.

We repeat above data generating procedure and analysis 100 times. All of the 100 test sets precisely identified the five substitution sites with our full algorithm. Without the *Gibbs* step, 97 out of 100 simulated date sets were able to correctly identify the five signal sets robustly regarding to the choices of α parameter. The few tests that did not produce perfect result each included one false positive identification. The overall result with default parameter setting ($\Delta = 3,\ \alpha = 0.5$) is summarized in Table 2, along with results from direct Gibbs samplers with $K = 20, 40, 60, 80$. PR, FN, FP, FNP correspond to perfect results, only false negatives, only false positives, both false negatives and false positives, respectively. The numbers under each category are test counts. The *Gibbs* step improves the result obtained for the *block MH* step. The median of K derived from our algorithm is $K = 42$. Although the derived K is larger than the true K, the inference result is still 100% correct. This suggests that an overestimated K can still result in correctly identified signal sites. Similarly, direct Gibbs with larger K's show perfect result for all 100 tests. As the goal is to compare posterior distributions not to obtain precise cluster number, an upper bound of mixture component number suffices. Note that when using our algorithm, we are guaranteed to be working with an appropriate upper bound of K. This insurance does not exist if K is chosen ad hoc, as would be required for a direct Gibbs.

[Table 2 about here.]

To assess the efficiency of our sequential MCMC algorithm, we also compared the clustering time (measured in CPU time) of the methods discussed in Table 2 for the 100 synthetic data sets (Table 3). For each test, the reported time under our method was the waiting time for the processing step (with or without *Gibbs*); the reported time for the direct Gibbs samplers was the time needed for one chain completing with corresponding K-means initial states. The CPU time is based on compute nodes including 122 blade servers, each with 8-cores 2.80 GHz Intel processors, 2×4M L2 cache (Model X5560), and 48GB memory for a total of 976 processing cores, two similar 8 core blades with 96 GB memory, and three more blades with 192 GB memory and 24 total cores. Summaries including means and standard deviations of computing time are recored for each method. As shown below, our method takes only a fraction of the time needed for the direct Gibbs, even when the true K is given. The variation of clustering time among the 100 tests is also much smaller using our algorithm. As K increases, the processing time for the direct Gibbs grows rapidly. It is worthy noted that the general computational issue with Gibbs sampler also affects the modification step in our algorithm. Our algorithm without the *Gibbs* step does not suffer the same issue. At the price of slight higher error rate, it produces reasonable results promptly.

[Table 3 about here.]

5. Real Data Analysis

5.1 *Human immunodeficiency virus 1 (HIV-1)*

As a "positive control", we applied our approach to an experimentally well characterized HIV-1 dataset (Jabara et al., 2011). Viral RNA was extracted from three longitudinal blood plasma samples taken from one individual infected with subtype B HIV-1, participating in a protease inhibitor (ritonavir) efficiency trial (Cameron DW, 1998). 454 sequencing was used to survey the genetic variation at the protease (*pro*) gene within the viral population. This

population variation was surveyed twice, separated by six months, prior to ritonavir drug selection (t_1, t_2) and then once after the initiation of therapy (t_{3D}). HIV-1 is known to rapidly evolve resistance to ritonavir and several resistance mutations in the *pro* gene have already been identified and confirmed with *in vitro* experiments. Thus, if our method is efficacious we should recover these same sites through our analysis.

The length of the protease gene is 297. As in the toy example, there are five possible reads and the corresponding Hellinger summary statistics are

$$Ht(D_i) = \min\{Ht(\pi_i^{t_1}, \pi_i^{t_{3D}}), Ht(\pi_i^{t_2}, \pi_i^{t_{3D}})\}, \quad Ht(N_i) = Ht(\pi_i^{t_1}, \pi_i^{t_2}), \quad i = 1, \cdots, 297. \quad (5)$$

[Figure 5 about here.]

The inference results based on clustering without and with the *Gibbs* step are shown in the top and bottom panels of Figure 5 respectively. The left two panels set size plot of S_1^d, S_2^d, S_3^d as the threshold d decreases (zoom-in); the right are the summary statistics plots. The thresholds according to two α parameter levels (0.5, dashed line; 0.25, dotted line), are plotted as well. In the summary statistics panels, the large red circles highlight the signal with default $\alpha(= 0, 5)$. Observing the trajectory of the S_2^d size function in the top left panel, the default α appears to be too conservative, only site 245 was identified. The smaller α seems to be more appropriate, with which, three additional sites, 48, 55, 268, were added to the signal set. The two α levels considered in the full algorithm produce similar thresholds and the same inference result:

$$S_1^{d_0} = \{48, 55, 243, 245, 250, 264, 268\}, \quad S_2^{d_0} = \{70, 72, 168, 219, 289\}, \quad S_3^{d_0} = S_1^{d_0}. \quad (6)$$

Due to the noise level and limited time points of this dataset, the clustering without the *Gibbs* produced much more conservative results. In the detected signal set, sites 48, 55, 245, 250 268 correspond to known drug resistance mutations. The other two sites identified, positions 243 and 264, both correspond to synonymous amino acid variation prior to treatment that disappeared post treatment. Meanwhile, the corresponding amino acids to sites 70, 72,

168, 219, 289, in the noise set S_2^{d0}, were identified as high variability in the study of genetic variation in the untreated environment (Jabara et al., 2011).

As shown above, our method reveals not only the well-known drug resistant sites but also additional genomic locations that present clear evolutionary changes. The sites identified correspond to major resistance sites manually identified and curated from the literature in Jabara et al. Site 245, which corresponds to the major ritonavir resistant variant, V82A, shows a strong signal (Baldwin et al., 1995). Similar patterns are seen at other known resistance sites. These data suggest that our approach can identify biologically important genetic changes. We also note that in contrast to the earlier work, we were able to identify these sites with minimal *a priori* knowledge of genome. That is, we assumed nothing about where the genes were in the genome, if the change altered an amino acid or other structural element, etc. Thus, we can apply our method with confidence to viral genomes that are not nearly as well studied as HIV-1.

5.2 *H1N1 Influenza A (IVA)*

We applied our method to the whole-genome sequencing time series data of influenza A virus A/Brisbane/59/2007 strain (NIH Biodefencse and Emerging Infectious Research Resources Repository NIAID, NIH; NR-12282; lot 58550257). The data were collected from multiple passages in the presence and absence of an inhibitor of neuraminidase, oseltamivir, for a total of two biological replicates (E1 & E2) (see Figure 1). At the end of each passage, whole-genome high throughput sequencing data were collected. The read counts are unbalanced between the two experiments, as the first replicate, E1, consistently had more reads than the second one. There are four possible nucleotides: A, C, G, T, i.e. $J = 4$.

This IVA strain consists of 8 segments: PB2 (2313 nucleotides (nts)), PB1 (2301 nts), PA (2303 nts), HA (1775 nts), NP (1396 nts), NA (1426 nts), M1/2 (1005 nts), and NS1/2 (869 nts). To reduce computational intensity, we examined each segment per replicate separately.

Within each replicate, we analyzed the control and treatment groups over selected time points simultaneously. In particular, we chose five time points: 1, 3, 9, 12, and the end (13 and 18 for E1 and E2, respectively). As the first three passages were shared across groups, we analyzed total of 8 time-samples, three of which were treated, for each biological replicate. Denote the 8 collection times as $t_1, t_2, t_3, t_4, t_5, t_{3D}, t_{4D}, t_{5D}$. To allow additional response time for the drug, the comparisons to t_{3D} and t_{4D} were not directly included in $Ht(D_i)$. The summary statistics were then formulated as

$$\begin{aligned} Ht(D_i) &= \min\{Ht(\pi_i^{t_1}, \pi_i^{t_{5D}}), Ht(\pi_i^{t_2}, \pi_i^{t_{5D}})\} \\ Ht(N_i) &= \max\{Ht(\pi_i^{t_1}, \pi_i^{t_2}), Ht(\pi_i^{t_1}, \pi_i^{t_j}), Ht(\pi_i^{t_2}, \pi_i^{t_j}),\ j = 3, 4, 5\} \end{aligned} \tag{7}$$

Taking segment 6 as an example, per replicate we analyzed control and treatment groups simultaneously, without and with the *Gibbs* step. The result plots for E1 and E2 are presented in Figures 6 & 7, respectively. Both replicates revealed site 822 (S6-822). The clear separation between $Ht(D_{S6-822})$ and the rest indicates that there is strong signal attributable to the treatment for S6-822.

[Figure 6 about here.]
[Figure 7 about here.]

To further investigate S6-822, we plot the proportions of nucleotide type at each time point using the raw data (Figure 8). The two panels on top are based on E1, while the two on the bottom are based on E2. The controls are the left two panels; the treatment groups are the right two panels. The complete transition of the nucleotide type in the treated group and nearly no change in the control group indicates strong drug effect. The consistent behavior across replicates enables us to conclude that S6-822 is a substitution site due to the treatment. In fact, this is a known oseltamivir-resistant mutation, H274Y (Collins et al., 2008). The color tiles on the top of each panel indicates that the total read count at each time point varies.

[Figure 8 about here.]

In contrast, Segment 7 evinces a negative result. Figures 9 & 10 show result plots for E1 and E2, respectively. Top two panels were obtained without the *Gibbs* step. In the top right panel of Figure 9, site 503 (S7-503) was highlighted since it is above the threshold ($\Delta = 3,\ \alpha = 0.5$). However, it does not exceed the threshold in Figure 10. As we are initially interested in substitution sites that are not replicate specific, we are only looking for signals observed in both biological replicates. The one red circle above threshold (top right panel of Figure 9) corresponds to S7-503. It appears to be a signal site based on E1, without the *Gibbs*. However, in Seg7E2, $Ht(D_{S7-503})$ is below the threshold (top right panel of Figure 10). Hence, we conclude that S7-503 is not a substitution site based on clustering result without the modification step. Conservatively, our conclusion is based on the intersection of the findings from each experiment. Of course it is possible each replicate could evolve along its own evolutionary path and hence differ between replicates. However, with the *Gibbs*, the signal set was adjusted to be empty for Seg7E1 (see bottom right panel of Figures 9 & 10) and we arrive at the same negative conclusion.

[Figure 9 about here.]
[Figure 10 about here.]

There is one site, S7-91, that consistently presented large $Ht(D_i)$ and $Ht(N_i)$ values across the two replicate. With or without the treatment, a complete transversion from G to C presented in both replicates. The large values in $Ht(D_{S7-91})$ and $Ht(N_{S7-91})$ precisely capture the read type switch that is likely due to genetic drift or adaptation to the host cells–not the drug.

With multiple biological replicates and many time point collections, the algorithm without the *Gibbs* step produced reasonable results with much greater computational efficiency. As discussed in Section 3, the modification step can take a long time due to the pitfall of standard Gibbs sampler, in which case, one may skip the *Gibbs* step at the cost of a slightly higher error rate.

For the rest of the segments, we performed the same analysis. Comparing the sites identified based on each biological replicate, we conclude that positions S6-822, and S8-80 are drug resistant sites. In addition, the following sites present evolutionary changes that are likely due to genetic drift or adaptation to the hosts: S1-2299, S1-2303, S3-2193, S4-1210, S5-1103, S7-91, S8-819 (Table 4). All of our findings are supported by the raw nucleotide read proportion plots (Supplementary figures).

[Table 4 about here.]

As mentioned earlier, for lengthy preprocessed data, direct Gibbs sampler can be computational expensive. Although our last clustering step, the *Gibbs*, only includes one scan of Gibbs sampling, it can also suffer from the same computational issue. For each segment, the one scan Gibbs sampler took at least two weeks real time on a high performance computing cluster while the first two steps only took a day or two. Our algorithm with and without the *Gibbs* step present consistent result generally. This is partly because that multiple time points were incorporated in the clustering procedure, yet only the last treated time was used to define the control statistics. Furthermore, because there are two biological replicates, taking the intersection of discoveries between the two helped to tease out some noise within each replicate. Skipping the modification step leads to a lightly higher error rate, however, with multiple time points and replicates, the clustering result without the *Gibbs* leads to similar inference conclusion as basing on the full algorithm. When drawing inference without the modification step, we advise to double check the shift parameter α, as the default setting might not be the best for capturing the curvature of noise set size function.

We require that a "true" site be one that showed the same evolutionary behavior in both replicates. This approach is conservative as it requires that the same evolutionary path is taken by both viral populations, which may not necessarily be true. While at least two sites–including a known resistance variant–meet this strict criterion, there are several "signal" sites

in each replicate that do not. These are potentially replicate specific adaptations. Moreover, it is possible the same amino acid can evolve through different nucleotide substitutions. For example, on segment 2 positions 31 and 32 evolved in Seg2E1 and Seg2E2 respectively. These neighboring changes both affect the amino acid lysine coded for by the 10th codon of the protein. Similar pattern is seen at sites 1004, 1005 on segment 7.

The first 12 passages of the dataset were analyzed by Foll et al. from a population genetics and structural perspective (Foll et al., 2014). For a fairer comparison, we applied our method to the joint data from Passages 1, 3, 9, 12 (See supplementary for more details). Intriguingly, most sites identified in Foll et al. appear in our analysis to only have signal in E1. The exception, S6-822, has a strong signal in both replicates and regardless of end point generation analyzed (Figure 8). We speculate that the lack of consistent signal/false signal coming from the other sites is caused by the lower average read count per site for the second replicate compare to the first. The population genetic approach used in Foll et al. appears to be heavily influenced by the first replicate. This leads us to postulate that their result is adversely affected by the large imbalance in counts.

The additional sites identified in Table 4, S8-80, showed a more pronounced drug effect after the 12th passage in both replicate. We conclude that IVA may not have fully responded to the treatment by Passage 12, which was the final passage analyzed by Foll et al. Thus our previous analysis, which included the last time point collection, is likely more reliable for the identification of substitution sites.

6. Discussion

In this manuscript, we introduced a Dirichlet mixture model for detecting and clustering changes in allele frequencies in DNA or RNA sequence data from a population sampled at different time points. This genome annotation free approach is particularly suited for

RNA viruses and other organisms where the secondary structure of the RNA can influence evolution in ways not predicted by standard molecular evolutionary analysis methods.

To identify significant changes in allele frequency, our clustering algorithm uses a combination of *hierarchical SCMH*, *block MH*, and *Gibbs* procedures. This approach does not require a prior distribution on the number of mixture components. The *hierarchical SCMH* step automatically produces an upper bound for the number of mixture components, K, and fine clusters for the *block MH* step. The hierarchical tree structure enables parallel computing and overcomes the computational difficulties any direct Markov chain Monte Carlo method presents. The *block MH* step improves the upper bound for K and combines similar clusters. Last but not least, the *Gibbs* step modifies the clustering result. The threshold for identifying substitution sites is based on the posterior distribution comparison for the timepoints without treatment. It is chosen by examining the curvature in the graph of the number of members in the noise set instead of selecting an *ad hoc* cutoff.

With synthetic datasets we showed that our method with full clustering algorithm achieves results comparable to direct Gibbs without having to choose a K *ad hoc*. When the *Gibbs* step was skipped, we still achieved high identification rate with a greater gain in computation time. The *hierarchical SCMH* step enables parallel computing with partial data, which makes our clustering algorithm is much more efficient, even compared to the direct Gibbs with the true K. The last cluster step of our algorithm, the *Gibbs*, can take a long time if the consolidated dataset is very large. One may choose to skip this modification step at the price of a slightly higher error rate. It is advised to check the set size function plot and determine if the default parameters are appropriate.

As a positive control, we applied our method to a well described HIV-1 dataset. With minimal assumptions on gene annotation or the coding nature of the substitution, we suc-

cessfully identified known drug resistance alleles and a list of sites with significant allelic changes within untreated population.

For the IVA dataset that motivated this study, we analyzed multiple time points and treatment-control simultaneously. We identified two sites, S6-822 & S8-80, with strong evidence of evolution in response to inhibitor treatment and six locations with high variability not due to the inhibitor. We compared our findings to a previous analysis of the same dataset based on a population genetic approach. Noticing that most of the sites identified using the latter method only appear in the biological replicate with larger sample size, we suspect that the population genetic based approach is biased due to this imbalance. Our algorithm performs analysis on each biological replicate individually first and then aggregate the results across replicates. Therefore, our inference technique is not sensitive to the unbalanced nature of the data.

Here we applied our method to nucleotide count data produced by high-throughput sequencing. It can also be applied to other types of count data, such as amino acids or metabolite abundances. As the model requires minimum assumptions, it can be broadly applied. For example, this approach can be used to identify evolved sites in non-coding regions of the genome such as the regulatory regions of genes or in RNA genes such as ribosomal RNA and other long non-coding RNAs.

Acknowledgements

The authors would like to thank Dr. Dominik Reinhold for his insightful comments. This project is supported by University Cancer Research Fund (C.D.J.), NCBC grant (2013-MRG-1110 C.D.J.), and National Science Foundation (DMS-1016441 and DMS-1007543 to J.H.).

References

Baldwin, E., Bhat, T., Liu, B., Pattabiraman, N., and Erickson, J. (1995). Structural basis of drug resistance for the v82a mutant of hiv-1 proteinase. *Natrue Structural & Molecular Biology* **2,** 244 – 249.

Beerenwinkel, N., Günthard, H. F., Roth, V., and Metzner, K. J. (2012). Challenges and opportunities in estimating viral genetic diversity from next-generation sequencing data. *Frontiers in microbiology* **3,**.

Boutwell, C. L., Rolland, M. M., Herbeck, J. T., Mullins, J. I., and Allen, T. M. (2010). Viral evolution and escape during acute hiv-1 infection. *The Journal of Infectious Diseases* **202(Suppl 2),** S309–S314.

Cameron DW, e. a. (1998). Randomised placebo-controlled trail of ritonavir in advanced hiv-1 disease. the advanced hiv disease study group. *Lancet* **351(9102),** 543–549.

Collins, P., Haire, L., Liu, Y., Russell, R., Walker, R., Skehel, J., et al. (2008). Crystal structure of oseltamivir-resistant influenza virus neuraminidase mutants. *Nature* **453(7199),** 1258–61.

Cuevas, J., Domingo-Calap, P., and Sanjun, R. (2012). The fitness effects of synonymous mutations in dna and rna viruses. *Mol. Biol. Evol.* **29(1),** 17–20. doi: 10.1093/molbev/msr179.

Damgaard, C., Andersen, E., Knudsen, B., Gorodkin, J., and Kjems, J. (2004). Rnaintevolution in the 5' region of the hiv-1 genome. *J Mol. Biol.* **336(2),** 269–79.

de Campos, C. P. and Benavoli, A. (2011). Inference with multinomial data: why to weaken the prior strength.

Eriksson, N., Pachter, L., Mitsuya, Y., Rhee, S., Wang, C., Gharizadeh, B., et al. (2008). Viral population estimation using pyrosequencing. *PLoS Computational Biology* .

Foll, M., Poh, Y., Renzette, N., Ferrer-Admetlla, A., Bank, C., Shim, H., et al. (2014).

Influenza virus drug resistance: A time-sampled population genetics perspective. *PLOS Genetics* .

Ghedin, E., Holmes, E. C., DePasse, J. V., Pinilla, L. T., Fitch, A., Hamelin, M.-E., et al. (2012). Presence of oseltamivir-resistant pandemic a/h1n1 minor variants before drug therapy with subsequent selection and transmission. *Journal of Infectious Diseases* **206,** 1504–1511.

Jabara, C., Hu, F., Mollan, K., Willieford, S., Menezes, P., Yang, Y., et al. (2014). Hepatitis c virus (hcv) ns3 sequence diversity and antiviral resistance-associated variant frequency in hcv/hiv coinfection. *Antimicrobial agents and chemotherapy* **58,** 6079–6092.

Jabara, C., Jones, C., Roach, J., Anderson, J., and Swanstrom, R. (2011). Accurate sampling and deep sequencing of the hiv-1 protease gene usinga primer id. *Proceedings of the National Academy of Sciences of the United States of America* **108(50),** 20166–20171.

Knies, J., Dang, K., Vision, T., Hoffman, N., Swanstrom, R., and Burch, C. (2008). Compensatory evolution in rna secondary structures increases substitution rate variation among sites. *Mol. Biol. Evol.* **25(8),** 1778–87. doi: 10.1093/molbev/msn130.

Kuroda, M., Katano, H., Nakajima, N., Tobiume, M., Ainai, A., Hasegawa, H., et al. (2010). Characterization of quasispecies of pandemic 2009 infulenza a virus (a/h1n1/2009) by de novo sequencing using a next-generation dna sequencer. *PLoS ONE* **5(4),** e10256.

Leitner, T., Halapi, E., Scarlatt, G., Rossi, P., Albert, J., Feny, E., et al. (1993). Analysis of heterogeneous viral populations by direct dna sequencing. *Biotechniques* **15(1),** 120–7.

Muers, M. (2011). Technology: Getting moore from dna sequencing. *Nature Reviews Genetics* **12.9,** 586–587.

Norstrm, M., Karsson, A., and Salemi, M. (2012). Towards a new paradigm linking virus molecular evolution and pathegenesis: experimental design and phylodynamic inference. *New Microbiol.* **35(2),** 101–11.

Rambaut, A., Posada, D., Crandall, K., and Holmes, E. (2004). The causes and consequences of hiv evolution. *Nature Reviews Genetics* **5,** 52–61.

Renzette, N., Caffrey, D., Zeldovich, K., Liu, P., Gallagher, G., Aiello, D., et al. (2014). Evolution of the influenza a virus genome during development of oseltamivir resistance in vitro. *Journal of Virology* **88,** 272–281.

Shao, W., Boltz, V., Spindler, J., Kearney, M., Maldarelli, F., Mellors, J., et al. (2013). Analysis of 454 sequencing error rate, error sources, and artifact recombination for detection of low-frequency drug resistance mutations in hiv-1 dna. *Retrovirology* **10,** 1–16.

Simmonds, P. and Smith, D. (1999). Structral constraints on rna virus evolution. *J Virol.* **73(7),** 5787–94.

Walley, P. (1996). Inferences from multinomial data: Learning about a bag of marbles (with discussion). *Journal of the Royal Statistical Society* **Vol. 58, No. 1,** 3–57.

Watts, J., Dang, K., Gorelick, R., Leonard, C., Bess, J. J., Swanstrom, R., et al. (2009). Architecture and secondary structure of an entire hiv-1 rna genome. *Nature* **460(7256),** 711–6. doi:10.1038/natrue08237.

Wright, C., Morelli, M., Thbaud, G., Knowles, N., Herzyk, P., Paton, D., et al. (2010). Beyond the consensus: dissecting within-host viral population diversity of foot-and-mouth disease virus by using next-generation genome sequencing. *J Virol* **85(5),** 2266–75.

Received April 2007. *Revised April* 2007. *Accepted April* 2007.

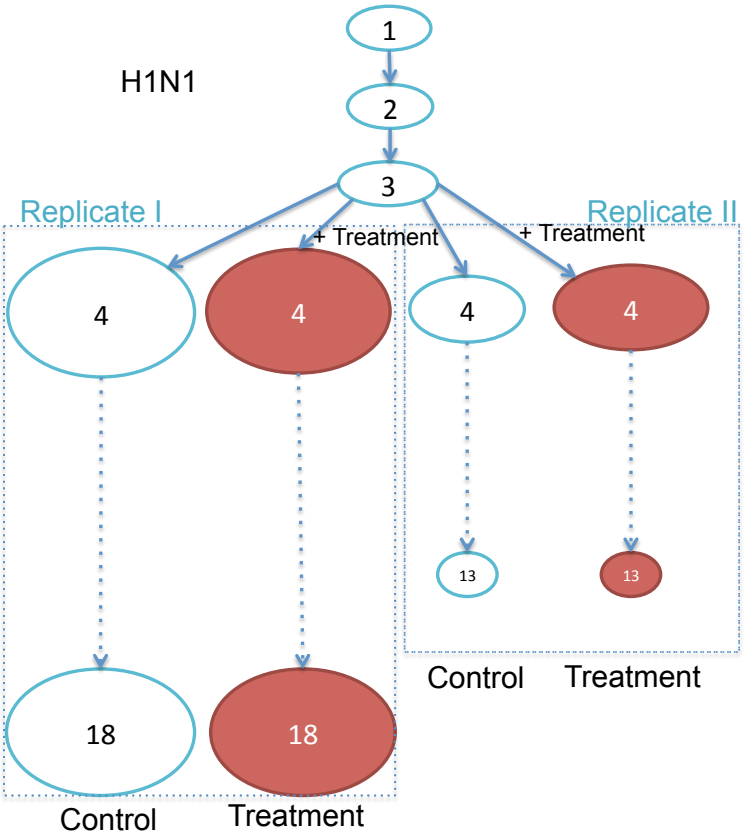

Figure 1. IVA adapted from chicken egg to MDCK cells for passages 1-3, then serially passaged in either absence (white) or presence (red) of oseltamivir environments. There are two biological replicates. The size of the oval corresponds to the average total read count per site. The number in the oval corresponds to the generation.

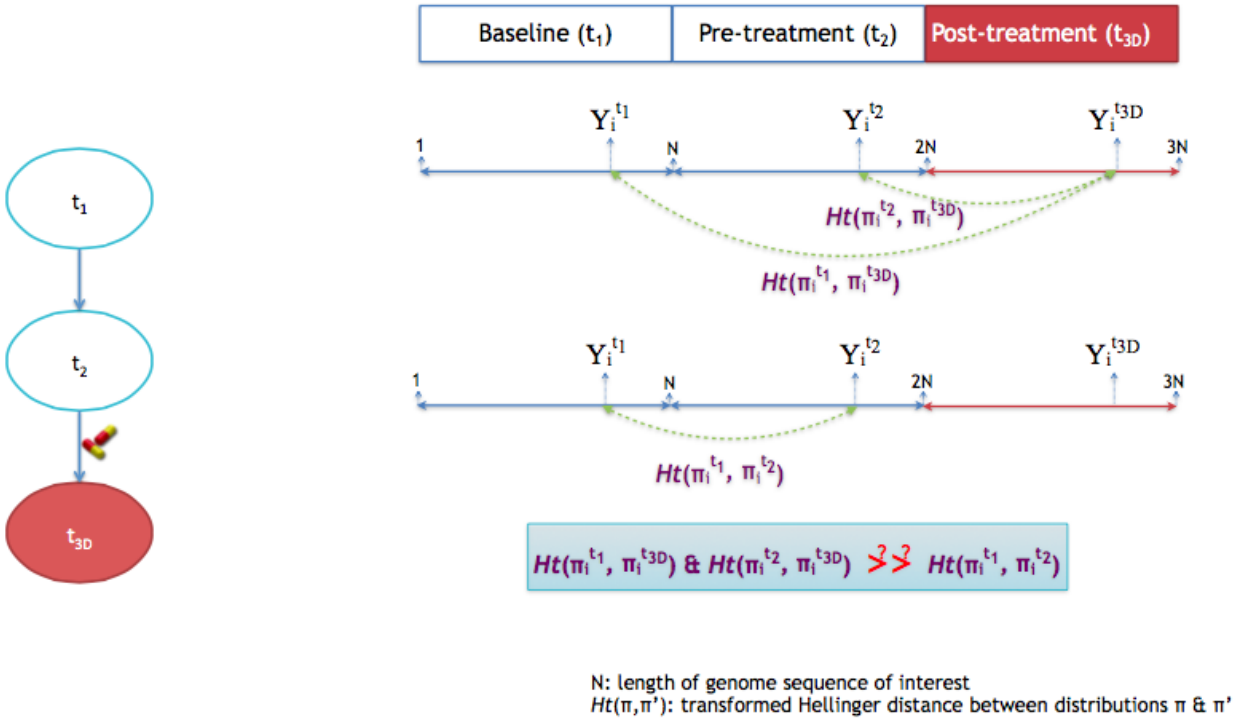

Figure 2. The transformed Hellinger distance is used to compare the marginal posterior distributions for the same genomic site across time. If the comparison between the treated and non-treated groups are much larger than the variation within drug-free environment, then the site is identified to be affected by the treatment.

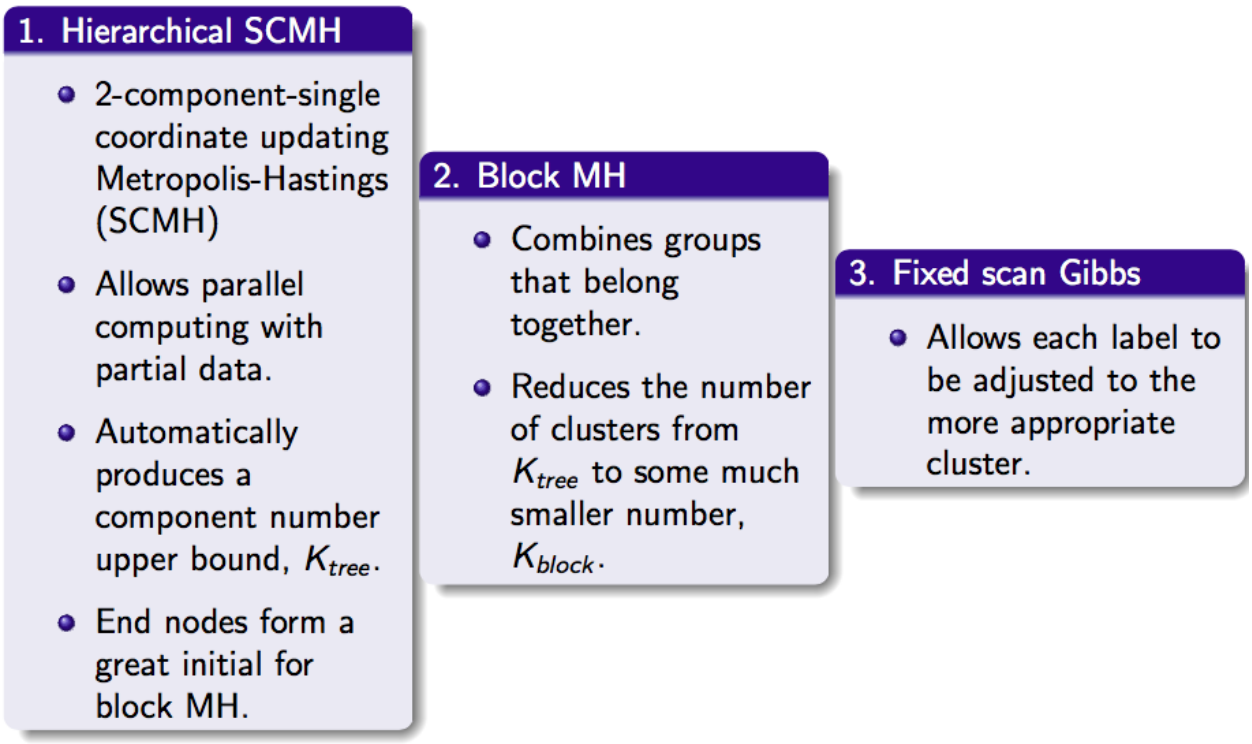

Figure 3. Three steps clustering procedure. It automatically produces an upper bound from K, assigns cluster labels to genomic sites at each time point, and allows parallel computing.

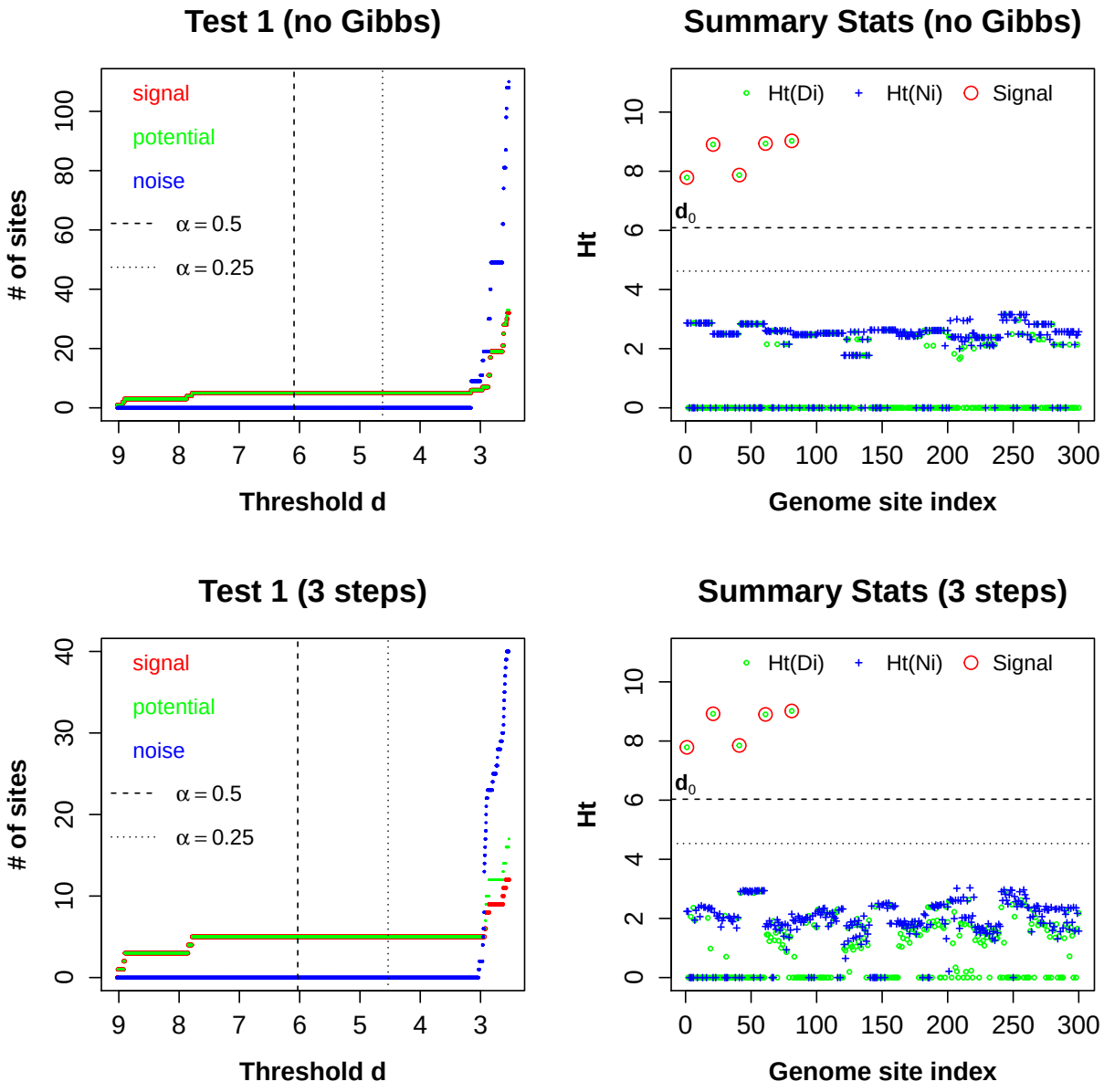

Figure 4. The result plots for Test 1 without (top) and with (bottom) the *Gibbs* step. The left two panels show the number of elements of S_1^d, S_2^d, S_3^d as the threshold d decreases with thresholds indicated in dashed ($\alpha = 0.5$) and dotted lines ($\alpha = 0.25$); the right two panels are the summary statistics plots with correspond thresholds to the left. The small green circles and blue pluses are the $Ht(D_i)$ values and $Ht(N_i)$ values, respectively. For genome site i that belongs to the signal set, its $Ht(D_i)$ value is highlighted in a large red circle. The five red circles on the top left correspond to the true substitution sites: 1, 21, 41, 61, 81. There is a clear separation between signals and the rest of the sites.

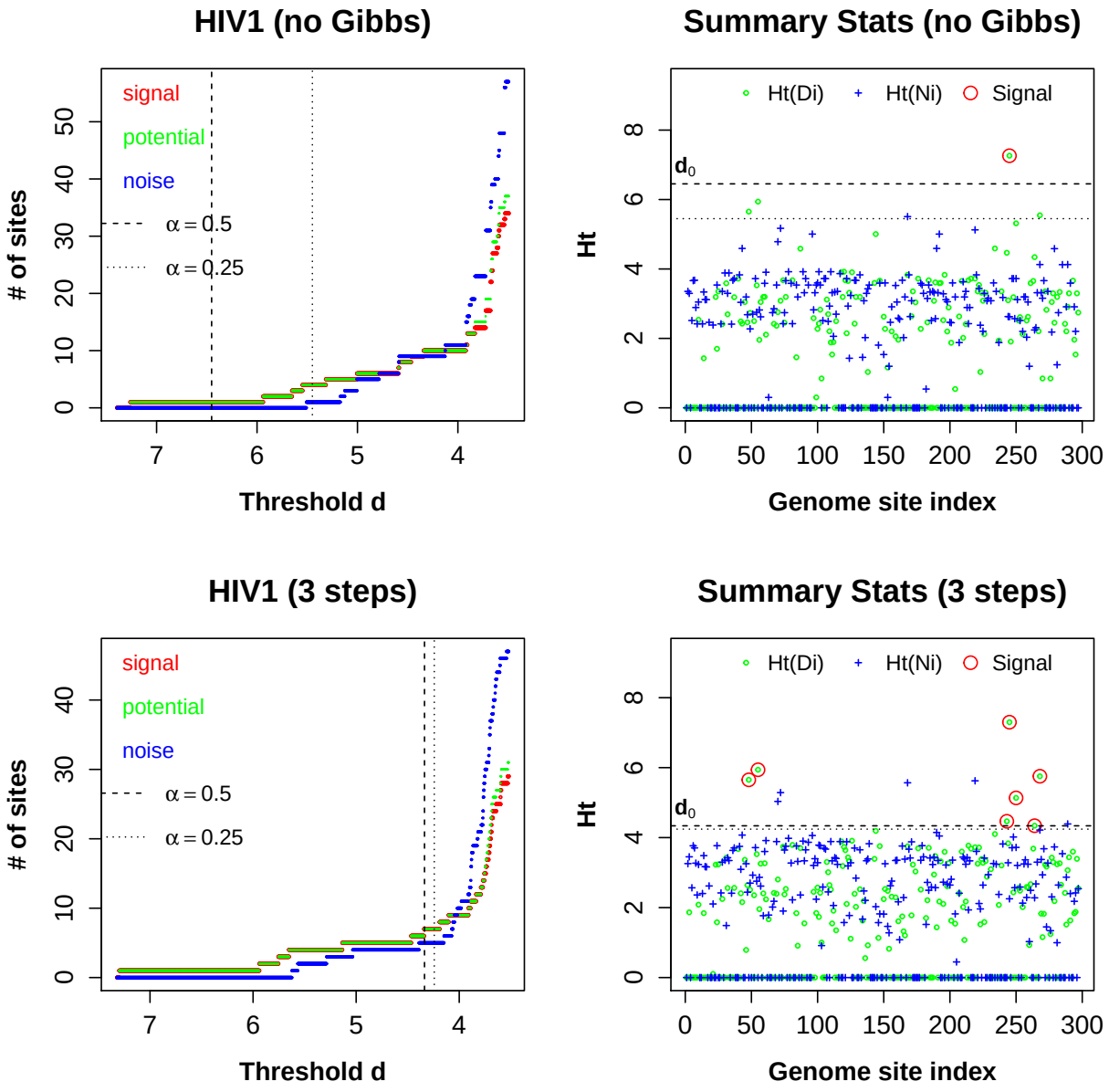

Figure 5. Results from the HIV-1 protease genome data set, without (top) and with (bottom) the *Gibbs* step. The left panels show the sizes of sets S_1^d, S_2^d, S_3^d as the threshold d decreases; the right panels are the summary statistics plots with signal identified (with default parameters) highlighted in red circles. Without the modification step, the default α appears to be too conservative. For the full algorithm result, either choice of α produced the same inference result with seven signal sites: 48, 55, 243, 245, 250, 264, 26, and five noise sites: 70, 72, 168, 219, 289.

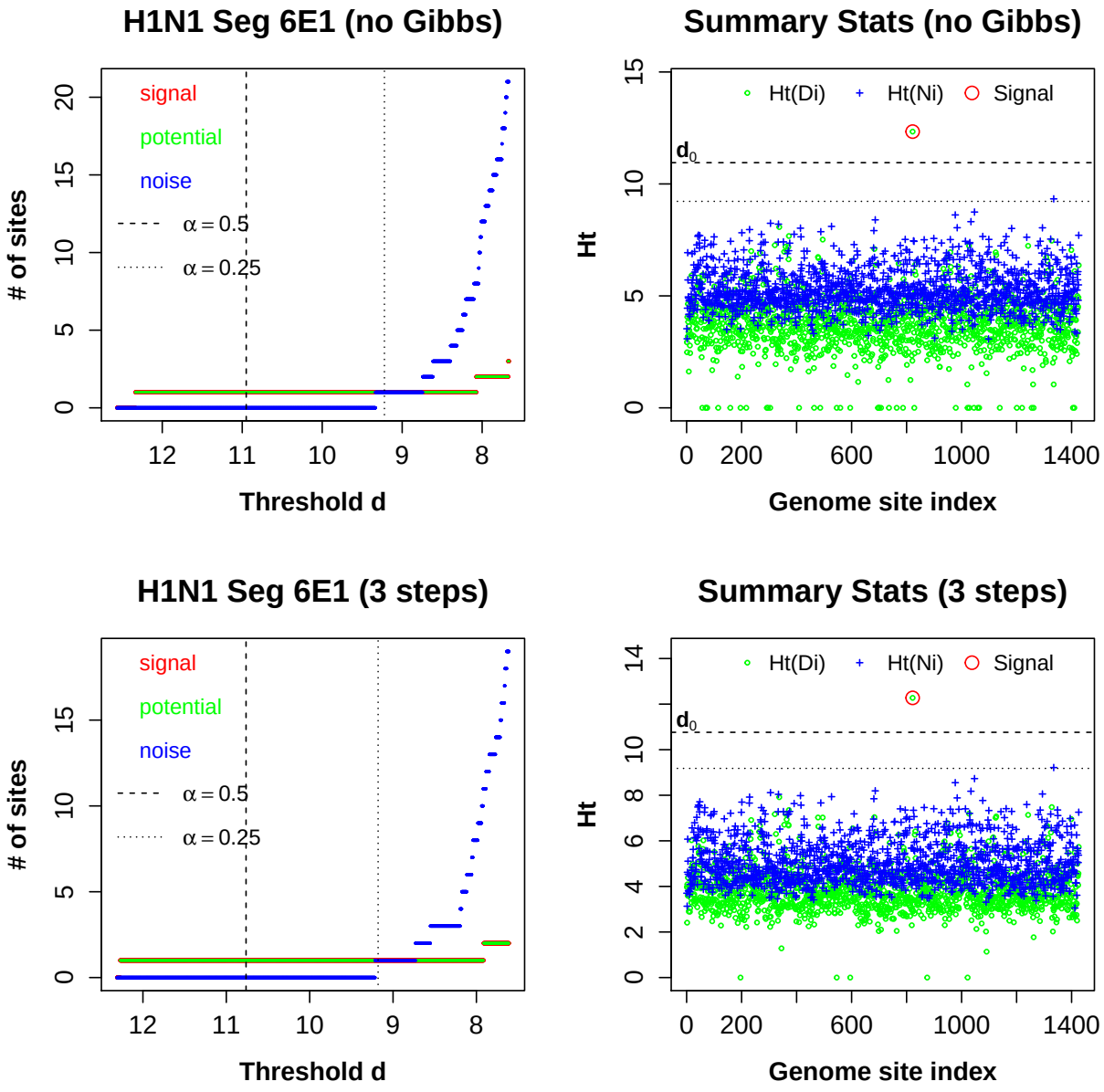

Figure 6. The results for H1N1 Seg6E1. Without (top panels) or with (bottom panels) the *Gibbs* step, our algorithm identified one signal site, site 822 (S6-822). It corresponds to a known oseltamivir-resistant mutation for H1N1. The inference result for H1N1 Seg6E1 is consistent even without the *Gibbs* step, and is robust to the choices of α parameter.

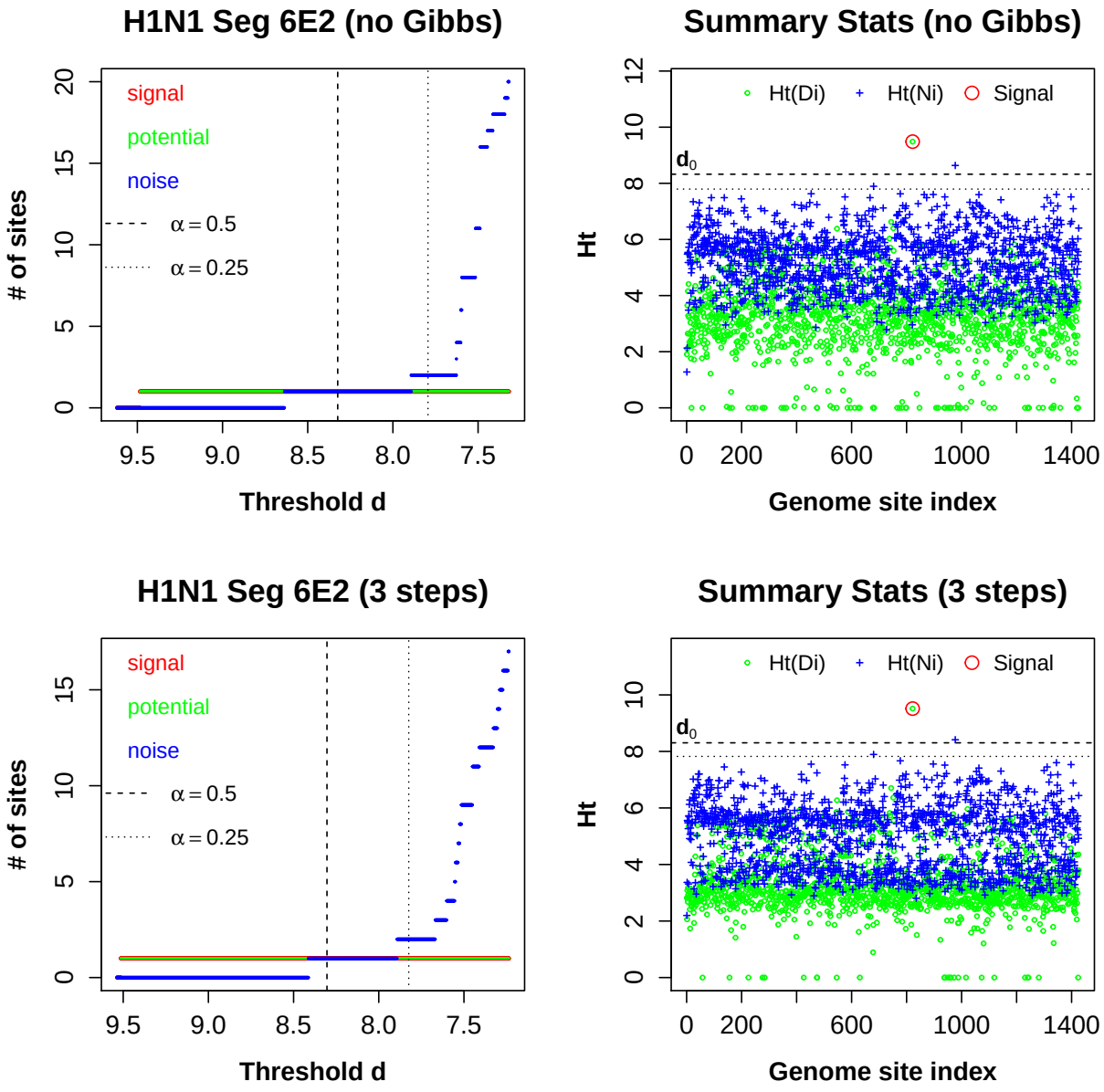

Figure 7. The result plots for H1N1 Seg6E2. Similar to Seg6E1, without (top panels) and with (bottom panels) the *Gibbs* step, our algorithm identified site 822 (S6-822). The inference result for H1N1 Seg6E1 is consistent even without the *Gibbs* step, and is robust to the choices of α parameter.

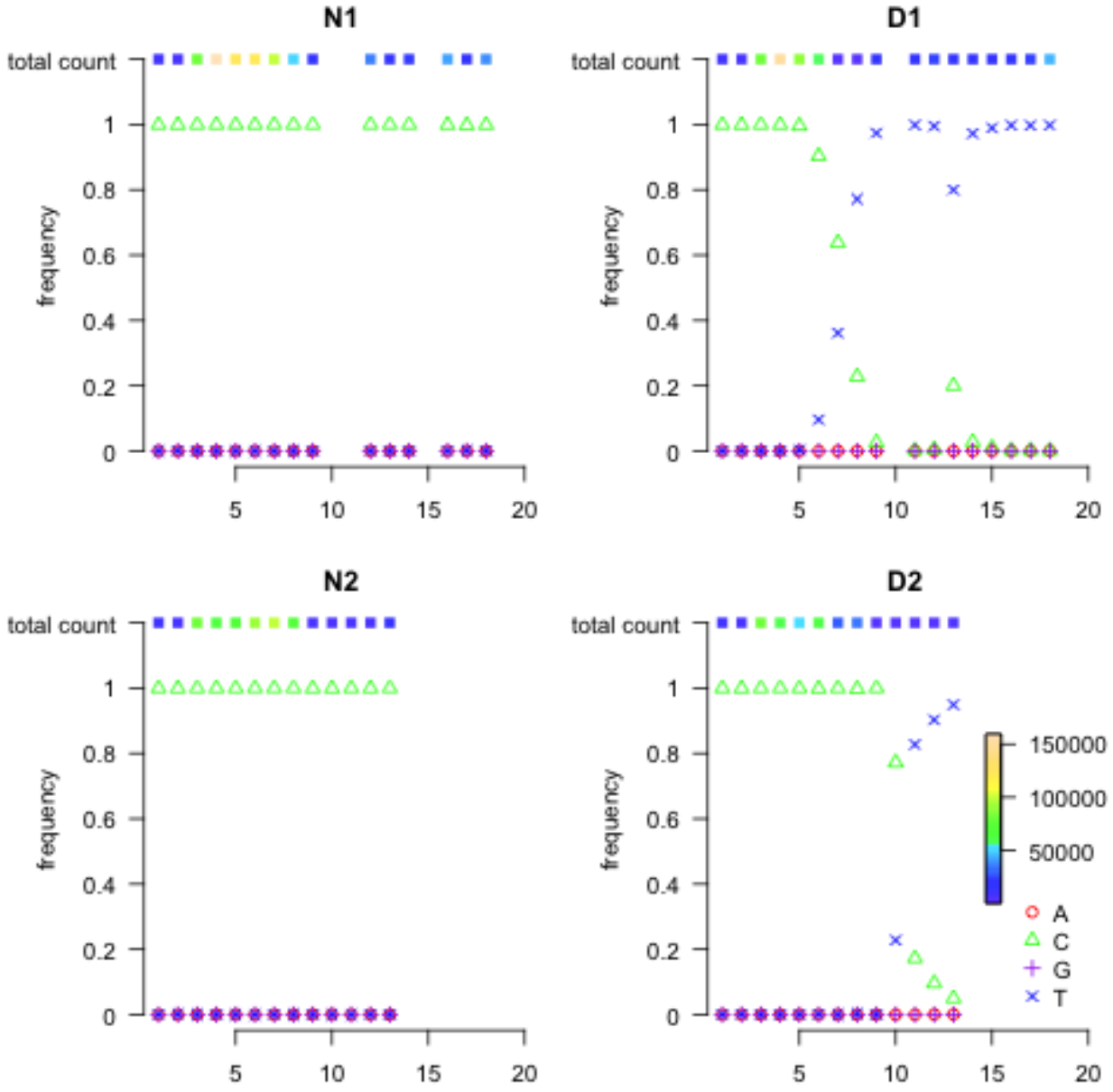

Figure 8. H1N1 nucleotide read count proportion and total count at position S6-822. The top and bottom rows are for Replicate I and Replicate II, respectively; the left and right panels are for control and treatment groups, respectively. For the treated groups, there is a complete transition from C to T due to the drug. The color tiles on the top of each panel indicates the total read count at each time point.

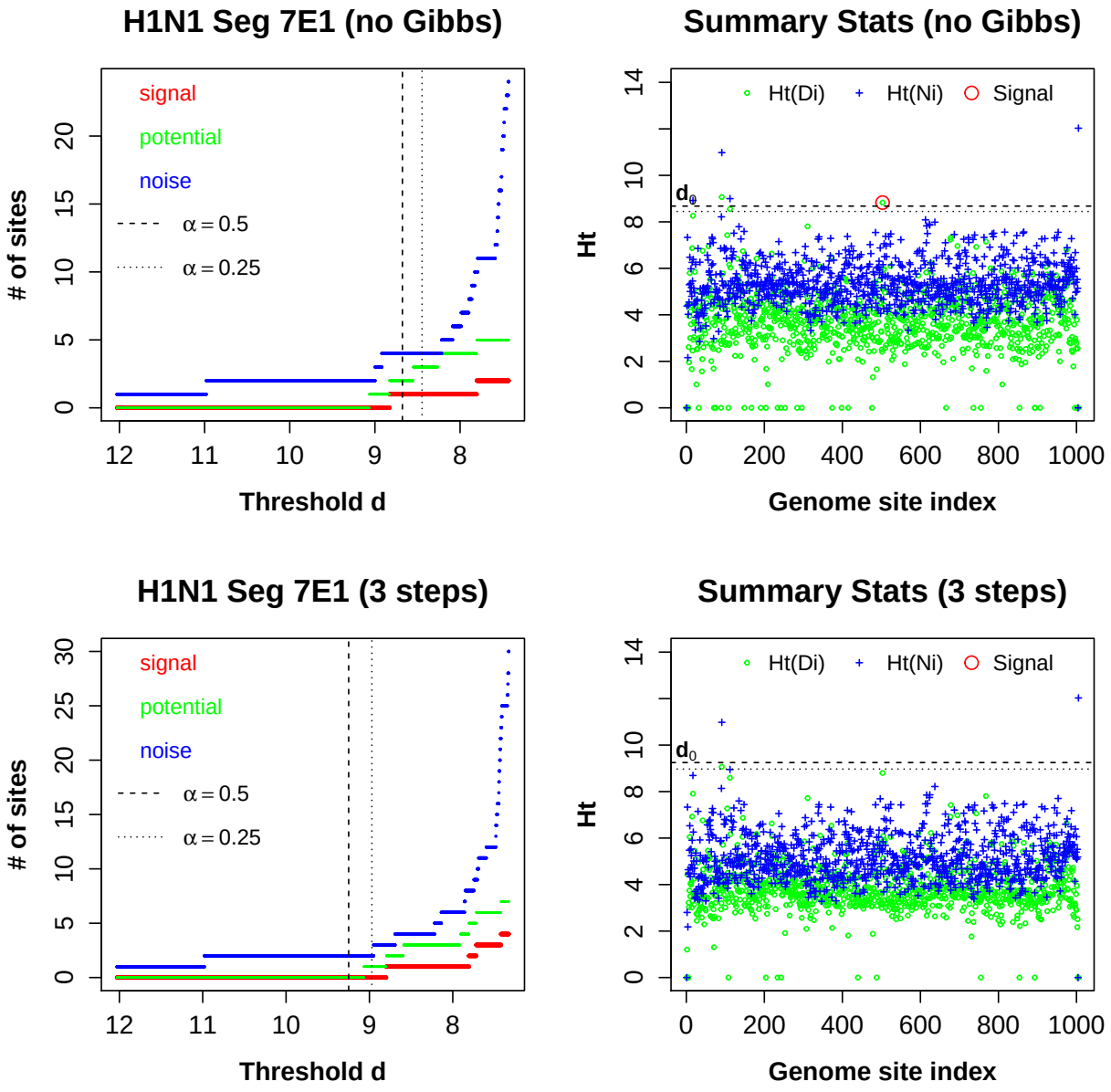

Figure 9. The results for H1N1 Seg7E1. Without the *Gibbs* step (top panels), S7-503 was highlighted. However, $Ht(D_{S7-503})$ does not exceed the threshold with the full algorithm(bottom right panel). Site 91 showed large summary statistic values $Ht(N_{S7-91})$ & $Ht(D_{S7-91})$ in both right panels. The control statistic value for S7-1005 is alarmingly high. We suspect that is the result of low alignment quality at the tail of the segment. The inference result for H1N1 Seg7E1 is robust to the choices of α parameter.

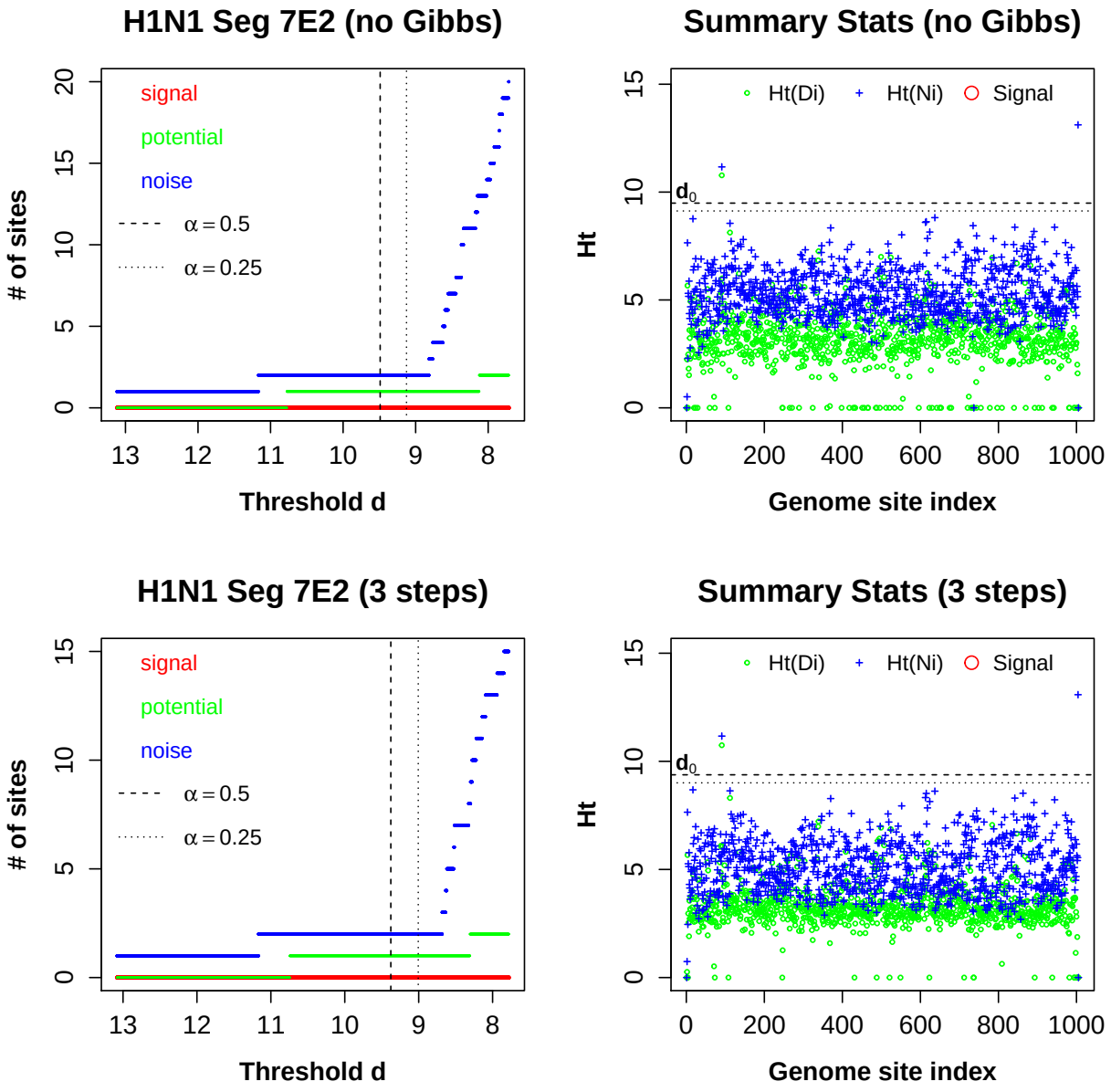

Figure 10. The result plots for H1N1 Seg7E2. The inference result for H1N1 Seg7E2 is consistent even without the *Gibbs* step, and is robust to the choices of α parameter. No site was identified as signal. Similar to Seg7E1, $Ht(N_{S7-91})$ & $Ht(D_{S7-91})$ exceeded the thresholds in both right panels. A site on the tail part of the segment, S7-1004, showed large control statistic values.

Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	$\cdots$
					C	A	T	$\cdots$
C	T	C	T	A	C	A		$\cdots$
		C	T	A	C	C	M	$\cdots$
				G	C	T	T	$\cdots$
	C	M	G	T	C	T		$\cdots$
	G	C	T	C				$\cdots$

$\Longrightarrow$

	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	$\cdots$
A	0	0	0	0	2	0	2	0	$\cdots$
C	1	1	3	0	1	5	1	0	$\cdots$
G	0	1	0	1	1	0	0	0	$\cdots$
T	0	1	0	3	1	0	2	2	$\cdots$
M	0	0	1	0	0	0	0	1	$\cdots$

Table 1

High through-put sequencing data from all samples are pooled and aligned (left panel) and then compressed into a five-row count matrix for the genome of interest (right panel).

Result	Our method		Direct Gibbs			
	w/ *Gibbs*	w/o *Gibbs*	K = 20	K = 40	K = 60	K = 80
PR	100	97	99	100	100	100
FN	0	0	1	0	0	0
FP	0	3	0	0	0	0
FNP	0	0	0	0	0	0

Table 2

Result comparison of our method, Gibbs with $K = 20$, Gibbs with $K = 40$, and Gibbs with $K = 60$ using a variety of thresholds. PR, FN, FP, FNP are the number of tests with perfect results, only false negatives, only false positives, both false negatives and false positives, respectively. All methods show good results. The Gibbs step improves the result over the block MH step alone.

Clustering Time	Our method		Direct Gibbs			
	w/ *Gibbs*	w/o *Gibbs*	K = 20	K = 40	K = 60	K = 80
Min	55.78	41.49	95.94	289.3	576.6	959.1
1st Quartile	60.96	48.83	104.9	320.7	647.2	1083
Median	67.61	56.04	124.7	371.0	721.4	1178
Mean	67.72	55.42	148.8	394.8	766.8	1241
3rd Quartile	73.46	61.01	171.4	423.0	805.7	1368
Max	86.20	74.86	422.8	706.6	1467	1961
Standard Deviation	7.996	7.747	63.22	94.92	166.8	208.6

Table 3

Clustering time comparison in CPU time. For each test set, the corresponding process time of the direct Gibbs was that of a single Markov chain with K-means initial state for Gibbs sampler given a pre-chosen number of clusters. For our method, the corresponding process time records the total waiting time (in CPU time) needed for the processing step to finish 100 parallel Markov chains for each test set. The medians, means, and standard deviations here are from all 100 test sets. Our algorithm shows clear advantage in computational efficiency.

Seg	E	Threshold d_0		Signal $S_3^{d_0}$		Noise $S_2^{d_0}$	
		w/ *Gibbs*	w/o *Gibbs*	w/ *Gibbs*	w/o *Gibbs*	w/ *Gibbs*	w/o *Gibbs*
1	1	8.756	10.59	1006, 1115, 1638, 1731, 1938, 2101	∅	33, 281, 311, 404, 632, 839, 926, 1542, 1889, 2290, 2298, 2299, 2303	∅
	2	9.401	9.483	∅	∅	2299, 2303	2299, 2303
2	1	9.782	9.86	∅	∅	224, 1118, 1499, 2066	224, 1118, 1499, 2066
	2	9.505	9.647	2099	2099	1036, 1675	1036, 1675
3	1	10.101	10.531	∅	∅	2193	2193
	2	9.655	8.937	174	174, 177	1850, 2193	79, 146, 153, 173, 176, 200, 203, 210, 1527, 1850, 2078, 2192, 2193, 2195
4	1	10.709	10.784	∅	∅	1210, 1394	729, 1210, 1394
	2	9.672	9.827	∅	∅	1210	638, 1210
5	1	10.352	9.98	∅	300	1103, 1395	24, 389, 1103, 1395
	2	8.626	8.566	300	300	24, 389, 1103	24, 389, 1103
6	1	10.763	10.912	822	822	∅	∅
	2	8.304	8.319	822	822	977	977
7	1	9.249	8.57	∅	503	91, 1005	17, 91, 112, 1005
	2	9.377	9.435	∅	∅	91, 1004	91, 1004
8	1	10.706	10.681	80	80, 848	385, 819, 848	385, 819, 848
	2	10.956	10.987	80	80	819	663, 819

Table 4

Result derived using Passages 1, 3, 9, 12, and the end time point. The table provides the thresholds and corresponding signal & noise sets for each segment according to each biological replicate with and without the Gibbs step. The sites identified as signal in both experiments are highlighted in red, the ones identified as noise in both experiments are highlighted in blue. The modification step took much more time comparing to the first two steps. The w/ Gibbs results was not finished for Seg1E1 and Seg2E1 in four weeks time with standard Gibbs. In comparison, the algorithm with or without the modification step produced similar final result after cross check the replicates.