

RESEARCH ARTICLE

Open Access

# Detection of viral sequence fragments of HIV-1 subfamilies yet unknown

Thomas Unterthiner<sup>1</sup>, Anne-Kathrin Schultz<sup>1</sup>, Jan Bulla<sup>2</sup>, Burkhard Morgenstern<sup>1</sup>, Mario Stanke<sup>3</sup> and Ingo Bulla<sup>1,3\*</sup>

## Abstract

**Background:** Methods of determining whether or not any particular HIV-1 sequence stems - completely or in part - from some unknown HIV-1 subtype are important for the design of vaccines and molecular detection systems, as well as for epidemiological monitoring. Nevertheless, a single algorithm only, the Branching Index (BI), has been developed for this task so far. Moving along the genome of a query sequence in a sliding window, the BI computes a ratio quantifying how closely the query sequence clusters with a subtype clade. In its current version, however, the BI does not provide predicted boundaries of unknown fragments.

**Results:** We have developed *Unknown Subtype Finder* (USF), an algorithm based on a probabilistic model, which automatically determines which parts of an input sequence originate from a subtype yet unknown. The underlying model is based on a simple profile hidden Markov model (pHMM) for each *known* subtype and an additional pHMM for an *unknown* subtype. The emission probabilities of the latter are estimated using the emission frequencies of the known subtypes by means of a (position-wise) probabilistic model for the emergence of new subtypes. We have applied USF to SIV and HIV-1 sequences formerly classified as having emerged from an unknown subtype. Moreover, we have evaluated its performance on artificial HIV-1 recombinants and non-recombinant HIV-1 sequences. The results have been compared with the corresponding results of the BI.

**Conclusions:** Our results demonstrate that USF is suitable for detecting segments in HIV-1 sequences stemming from yet unknown subtypes. Comparing USF with the BI shows that our algorithm performs as good as the BI or better.

## Background

An accurate and reliable classification of viral sequences data for human immunodeficiency virus-1 (HIV-1) and some other viruses of interest is important for epidemiological studies. It facilitates the understanding of the influence of genetic diversity on host immune response and provides therapeutic decision support [1-3]. As HIV-1 is, however, one of the genetically most variable viruses and genomic recombinations are frequent in HIV-1 [4], the task of classifying corresponding viral sequence data is a challenging one.

HIV-1 is classified into three main phylogenetic groups (M, N, and O), introduced into humans by separate zoonotic events (all stemming from simian immunodeficiency viruses (SIVs) in chimpanzees [5]. The M

group is responsible for the HIV pandemic, and it is divided into nine subtypes, with subtype A and F being subdivided into subsubtypes [6]. Inter-subtype recombination occurs very frequently among HIV-1 subtypes [7]: So far, 48 circulating recombinant forms have been reported [8].

Up to now, about fifty tools for classification of HIV genomes, recognition of recombinants, and breakpoint detection have been developed. Examples are the REGA HIV-1 Subtyping Tool [9], the Recombinant Identification Program (RIP) [10], the jumping profile Hidden Markov Model (jpHMM) [11,12], the Recco [13], and the oligonucleotide-based method introduced in [14]. Nevertheless, to our knowledge, however, only one algorithm, called the Branching Index (BI), has been developed for deciding whether an HIV-1 sequence in question stems - completely or in part - from a subtype still unknown [15,16]. Notice that it is impossible to deduce unknown sequence segments using an existing subtype classification method,

\* Correspondence: [ingobulla@gmail.com](mailto:ingobulla@gmail.com)

<sup>1</sup>Institute of Microbiology and Genetics, University of Göttingen, Goldschmidtstr. 1, 37077 Göttingen, Germany

Full list of author information is available at the end of the article

based on a probabilistic model such as jpHMM, and to identify regions of low a posteriori probabilities for all of the well known subfamilies (see paragraph 'Discussion and conclusions - Miscellaneous').

In view of the large and rapidly growing quantity of sequence data, the need for a fully automatic tool for pinning down boundaries of unknown fragments is increasing. Since the BI is based on a sliding window approach, it only provides a visualization of the breakpoint positions, but no report of their exact position. We have addressed this problem by developing a model-based algorithm, which automatically detects those boundaries by taking a multiple sequence alignment (MSA) grouped into subfamilies as a basis.

A comparison of our algorithm with the BI, regarding scope and performance, is carried out in the section 'Results - Comparison'.

## Methods

The main input into our algorithm consists of i) an MSA representing the known sequences, with its sequences grouped into subfamilies, ii) a query sequence, iii) a classification of the query sequence with respect to the subfamilies (i.e. each position of the query sequence has to be assigned to a subfamily from the MSA) as main input.

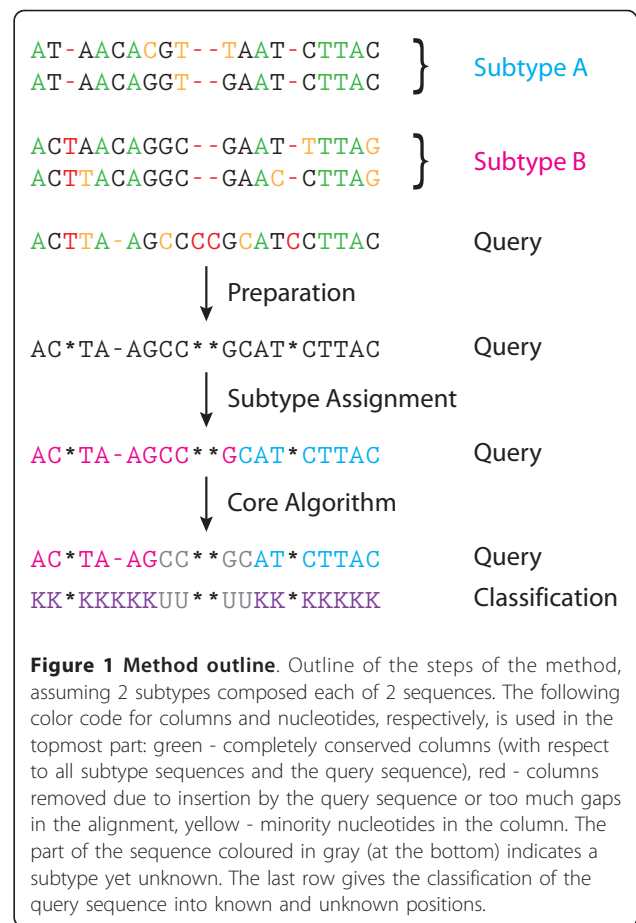
We use jpHMM in order to obtain the subfamily-wise classification [17]. For each position of the sequence in question, the algorithm then provides a mapping which determines whether the assignment to the subfamily is justified or whether it has to be classified as belonging to a subfamily yet unknown. In the first case, we refer to the position as 'known' (sometimes abbreviated by K), in the second one as 'unknown' (sometimes abbreviated by U). We shall refer to the mapping as the 'U/K-classification'. The work flow of the core algorithm and the preparatory step (described in the next subsection) is illustrated in Figure 1. The subfamily assigned to position  $i$  of the sequence under discussion is denoted by  $F_i$ .

### Preparatory step

Before the core algorithm is carried out, we take a preparatory step, allowing for the input of unaligned query sequences. More precisely, we align the query sequence to the given alignment with ClustalW [18], and remove columns which i) constitute insertions in the alignment by the query sequence, or ii) contain too many gaps in the alignment (we use a threshold of 10% gaps).

### Core algorithm

The main idea of our algorithm (in the following referred to as Unknown Subtype Finder or USF) is that of constructing two simple pHMMs (allowing neither deletions nor insertions): The first pHMM models the sequence of predicted subtypes  $F_i$  for each position (in



**Figure 1 Method outline.** Outline of the steps of the method, assuming 2 subtypes composed each of 2 sequences. The following color code for columns and nucleotides, respectively, is used in the topmost part: green - completely conserved columns (with respect to all subtype sequences and the query sequence), red - columns removed due to insertion by the query sequence or too much gaps in the alignment, yellow - minority nucleotides in the column. The part of the sequence coloured in gray (at the bottom) indicates a subtype yet unknown. The last row gives the classification of the query sequence into known and unknown positions.

the following pHMM K) and the second pHMM models an unknown subtype (in the following pHMM U). That is, for the example given in Figure 1, the first ten positions of pHMM K are modelled on the basis of the nucleotide frequencies of Subtype B at those positions and the last eight positions on the basis of the frequencies of Subtype A. In addition to the transitions within these two pHMMs, we allow for jumps between them.

### pHMMs K and U

pHMMs are widely used for modelling nucleotide and protein sequence families for the purpose of database searching (see [19,20]). In particular, they are used to model the position-wise nucleotide distribution in an MSA. Standard pHMMs also allow for the modelling of insertions and deletions in the query sequence. But we do not use insertion or deletion states, as the sequences are already aligned (The high conservation of HIV-1 sequences allows for this approach). Hence, except for the initial and final states, our pHMMs are composed of so-called match states only. For decoding the most probable path through our model, we use the Viterbi algorithm [21].

We model pHMM K in the conventional way: For each position  $i$  in the alignment, we model the emission

probabilities  $\bar{p}$  of the  $i$ -th state of the pHMM K on the basis of the nucleotide frequencies of  $F_i$ . To this end, choosing a Bayesian approach to model the emission frequencies, we assume that the a priori distribution of  $\bar{p}$  is a Dirichlet distribution (see [22]), with parameter  $\bar{\alpha}$  (estimated in [17]). The parameter may be interpreted as pseudo counts which are added to the nucleotide frequencies. The emission probabilities then are the corresponding relative frequencies of these modified nucleotide frequencies.

For pHMM U, we have to choose another approach, as the empirical nucleotide frequencies of an unknown subtype are not available. Hence, we try to deduce reasonable emission probabilities of an unknown subtype on the basis of the nucleotide frequencies of the known subtypes. For more details, see the paragraph 'Emission probabilities of pHMM U' in this subsection.

#### Jumps between pHMMs

As in the jpHMM, we allow for jumps between the pHMMs K and U. If a given path contains a jump, that jump represents a breakpoint between a known and an unknown segment. In our model, we distinguish two kinds of jumps (passing from left to right): (i) jumps from K to U with the path not having entered any state of pHMM U up to the current position, and (ii) all other jumps between K and U (see Figure 2 for examples for the determination of the jump probabilities). The probability of the first type of jumps is denoted by  $p_1$ , the probability of the second type by  $p_2$ . By modeling jumps in this way, we account for the fact that HIV-1 recombination events usually imply the occurrence of multiple breakpoints (cf. [8]). That is, traversing an HIV-1 genome from left to right, it is much more probable to revisit a particular subtype than it is to visit

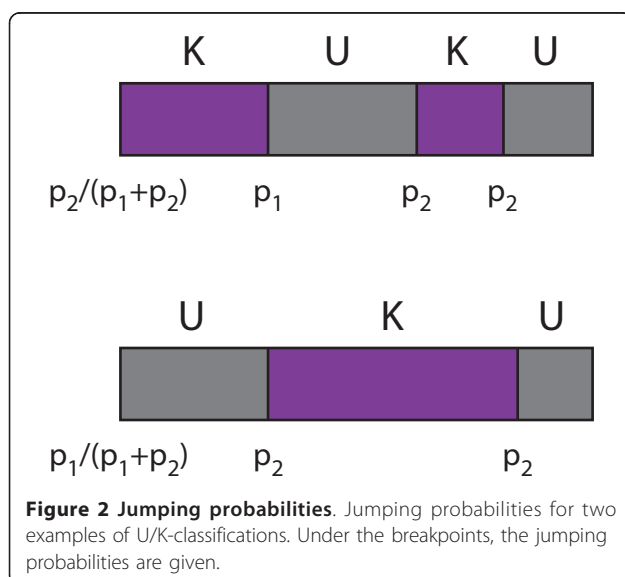
it for the first time ever. So, a realistic model should allow for choosing  $p_1 \ll p_2$ . To cover the case where the first position is classified as unknown, a jump from the initial state to pHMM U is less probable than a jump to pHMM K by the factor  $p_2/p_1$ .

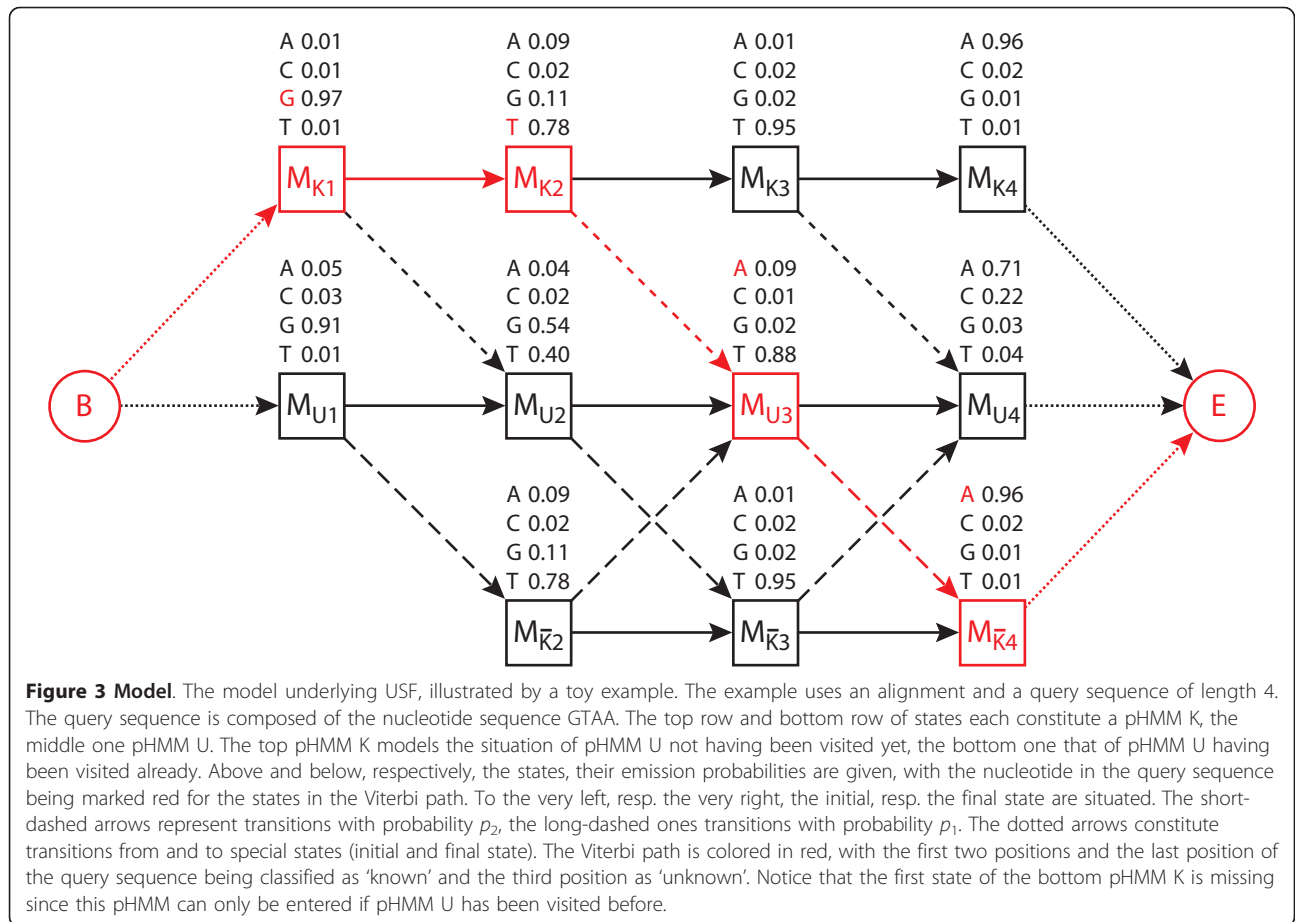
In order to be able to model these two jump probabilities, we have to incorporate the pHMM K in our model twice: Both model states represent the assignment of a position to be known, with one of them being used if no position has been assigned as unknown so far, and the other being applied if some position has been assigned to pHMM U already. Figure 3 shows a toy example of our model.

#### Emission probabilities of pHMM U

In order to model the emission probabilities of pHMM U, we rely on the observation that for almost all sites for HIV-1 at least some of the subtypes share the same emission probabilities. In fact, for the majority of sites, it would be most plausible to assign equal emission probabilities to all subtypes. Neglecting the trivial case of all subtypes having the same emission probability assigned to, the phenomenon that some but not all of the subtypes show equal emission probabilities could be explained biologically as follows: If a site allows for more than one nucleotide to be present (i.e., if at least two alleles are observed), there are very few, discrete characteristics of the virus which determine the fitness of the virus, depending on the nucleotide present at the respective site. As the characteristics at a particular site are small in number and discrete, the number of corresponding nucleotide distributions is also small. To clarify that, let us assume that for a site  $i$  the dependence of the virus fitness on the nucleotide at site  $i$  is determined by a binary characteristics (values 0 and 1) of the virus. Then i) for the value 0, it might be that the virus can only survive if adenine is present at site  $i$  (leading to a nucleotide distribution where adenine has a probability very near to one), ii) for the value 1 the virus can survive if cytosine is present, with a significant disadvantage with respect to its fitness (leading to a nucleotide distribution where adenine has a probability of, say, about 90% and cytosine one about 10%). In the following we will call the different nucleotide distributions (resp. emission probabilities) at the site "sources". In the example just given there are two sources.

In view of such considerations, we model the emission probabilities of the subtypes jointly (see Table 1 for examples). Notice that a related approach was used in [23] for an automatic classification of protein sequences. The model for the emission probabilities of an unknown subtype is illustrated in Figure 4. It is composed of two parts: The part on the left refers to the case in which the unknown subtype is related to a group of known subtypes (or a single one) sharing the same emission





probability at the respective site. The part on the right concerns the case of an unknown subtype with characteristics leading to emission probabilities (at the respective sites) yet unobserved (among the known subtypes).

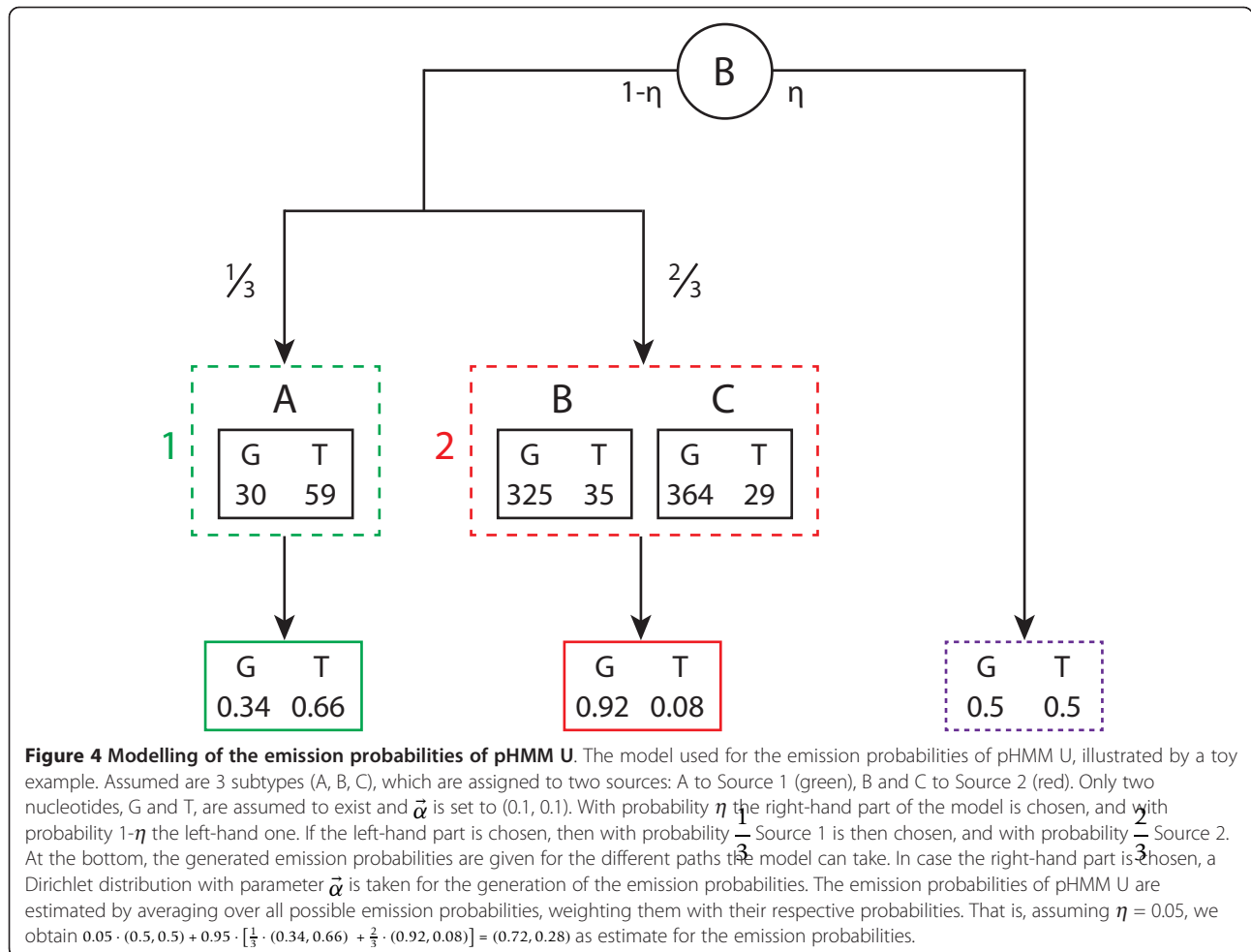
To construct the left-hand part of the model, we use a Bayesian approach to determine position-wise an optimal number of sources and how the subtypes should be assigned to the sources. For each source the emission

probabilities are estimated on the basis of the emission frequencies of the subtypes assigned to the source. The probability, with which a source is chosen, is proportional to the number of subtypes assigned to it. The right-hand part is modelled by a Dirichlet distribution with the same value for the parameter  $\bar{\alpha}$  as in paragraph 'pHMMs K and U' of this subsection. We denote the a priori probability of a source involved, but yet unknown,

**Table 1 Examples of calculation of emission probabilities**

Pos.	Sub./Src.	A		B		C		D		1		2		3	
		Nucl.	G	T	G	T	G	T	G	T	G	T	G	T	
1	freq	89	0	360	0	393	0	3	0	846	0				
	p	0.9989	0.0011	0.9997	0.0003	0.9997	0.0003	0.969	0.031	0.9999	0.0001				
2	freq	65	24	<b>355</b>	<b>5</b>	<b>382</b>	<b>11</b>	<b>3</b>	<b>0</b>	65	24	<b>740</b>	<b>19</b>		
	p	0.73	0.27	0.986	0.014	0.972	0.028	0.969	0.031	0.73	0.27	0.975	0.025		
3	freq	30	59	<b>325</b>	<b>35</b>	<b>364</b>	<b>29</b>	0	3	30	59	<b>689</b>	<b>64</b>	0	3
	p	0.34	0.66	0.903	0.097	0.926	0.074	0.0031	0.969	0.34	0.66	0.915	0.085	0.0031	0.969

Simplified example of position- and subtype-wise nucleotide frequencies of HIV. For three sites the subtype-wise nucleotide frequencies for subtypes A, B, C, and D are given on the left side of the table. Below them the emission probabilities estimated on the basis of only on the frequencies of the respective subtypes (using  $\bar{\alpha} = (0.1, 0.1)$ ) are shown. The different typefaces (regular, bold, italic) indicate which subtypes should be jointly modelled (i.e. belong to the same source). On the right-hand side of the table, the nucleotide frequencies of the sources (i.e. the aggregated frequencies of the subtypes belonging to it) and the emission probabilities estimated on the basis of these frequencies are given (using the same  $\bar{\alpha}$ ). For the sake of simplicity, only the nucleotides G and T are assumed to exist. Apart from this simplification and the restriction to 4 subtypes, the example is taken from actual HIV-1 sequences.



by  $\eta \ll 1$ . The estimates of the emission probabilities of the unknown subtype are obtained by averaging over both parts of the model, i.e. we use the expectations corresponding to the emission probabilities under the model. The details of the estimation procedure are given in the next subsection.

### Details of pHMM U

#### Notation

Let  $1, \dots, S$  be the subtype indices. If sources have been assigned to all subtypes, we speak of a source combination. The individual sources in a combination of  $r$  sources are indexed by  $1, \dots, r$ . The space of all source combinations is denoted by  $Q$ , the source of subtype  $i$  by  $q_i$ . For each source  $j$  of a source combination  $\vec{q} = \{q_1, \dots, q_S\}$ , we denote the subtypes assigned to source  $j$  by  $\{i_1^{(j)}, \dots, i_{m_j}^{(j)}\}$ . That is, if  $S = 4$  and the subtypes 1, 2, and 4 are assigned to Source 1, and the Subtype 3 to Source 2, we have  $m_1 = 3$ ,  $m_2 = 1$ ,  $\{i_1^{(1)}, i_2^{(1)}, i_3^{(1)}\} = \{1, 2, 4\}$ , and  $\{i_1^{(2)}\} = \{3\}$  (Notice that  $r$  and the  $\{i_1^{(j)}, \dots, i_{m_j}^{(j)}\}$  are defined for a particular source

combination, but that for the sake of readability we do not identify that source by an additional index, in case several sources are considered). The number of nucleotides, generally denoted by  $N$ , is equal to 5 in this case (We treat gaps as ordinary nucleotides). The nucleotide frequencies of subtype  $i$  at a fixed position of the genome are denoted by  $\vec{n}_i$ .

#### Prior probability of number of sources

We denote the probability of a given number of sources by  $P(r = r_0) = \rho_{r_0}$ . It is estimated as follows: We compile an alignment of all available HIV-1 sequences of complete length, classified as a pure subtype in the LANL HIV database (i.e. not being identified as recombinant or unknown). Hereby, we discard all sites at which the sequences of at least one subtype have only gaps. Then we determine, site-wise for each number of sources, the most probable source combination yielding the number of sources under consideration. For that we need the likelihood of  $\vec{q} \in Q$ , which is given by

$$P(\vec{q}) = \prod_{k=1}^r P(\vec{n}_{i_1^{(k)}}, \dots, \vec{n}_{i_{m_k}^{(k)}}). \quad (1)$$

The probabilities on the right hand side of (1) can be calculated as described in the following. For the next step we restrict ourselves to the case that  $\{i_1^{(k)}, \dots, i_m^{(k)}\} = \{1, \dots, m\}$  for notational convenience and make use of the equations

$$P(\vec{n}|\vec{p}) = \Gamma(|\vec{n}| + 1) \prod_{j=1}^N \frac{p_j^{n_j}}{\Gamma(n_j + 1)} \quad (2)$$

and

$$P(\vec{p}) = \frac{\Gamma(|\vec{\alpha}|)}{\Gamma(\prod_{j=1}^N \alpha_j)} \prod_{j=1}^N p_j^{\alpha_j - 1} \quad (3)$$

as well as

$$\int_{\vec{p}} \prod_{j=1}^N p_j^{\beta_j - 1} d\vec{p} = \frac{\prod_{j=1}^N \Gamma(\beta_j)}{\Gamma(|\vec{\beta}|)}, \quad (4)$$

With  $\beta_1, \dots, \beta_N \geq 0$ . Here,  $\vec{\alpha}$  denotes the parameter of the Dirichlet distribution introduced in the paragraph ‘Methods - Core algorithm - pHMMs K and U’. Thus, we obtain

$$\begin{aligned} &P(\vec{n}_1, \dots, \vec{n}_m) \\ &= \int_{\vec{p}} P(\vec{n}_1, \dots, \vec{n}_m | \vec{p}) P(\vec{p}) d\vec{p} \\ &= \int_{\vec{p}} \left( \prod_{i=1}^m P(\vec{n}_i | \vec{p}) \right) P(\vec{p}) d\vec{p} \\ &= \int_{\vec{p}} \left[ \prod_{i=1}^m \left( \Gamma(|\vec{n}_i| + 1) \prod_{j=1}^N \frac{p_j^{n_{i,j}}}{\Gamma(n_{i,j} + 1)} \right) \right] \times \\ &\quad \left[ \frac{\Gamma(|\vec{\alpha}|)}{\Gamma(\prod_{j=1}^N \alpha_j)} \prod_{j=1}^N p_j^{\alpha_j - 1} \right] d\vec{p} \\ &= \left( \prod_{i=1}^m \frac{\Gamma(|\vec{n}_i| + 1)}{\prod_{j=1}^N \Gamma(n_{i,j} + 1)} \right) \frac{\Gamma(|\vec{\alpha}|)}{\Gamma(\prod_{j=1}^N \Gamma \alpha_j)} \times \\ &\quad \frac{\prod_{j=1}^N \Gamma(\sum_{i=1}^m n_{i,j} + \alpha_j)}{\Gamma(\sum_{i=1}^m |\vec{n}_i| + |\vec{\alpha}|)}. \end{aligned}$$

Using (1) and the AIC (Akaike Information Criterion [24]), we deduce the most plausible source combination for each site and with that the most plausible number of sources. Estimating the  $\rho_j$  as the empirical frequencies of the number of sources (considering all eligible sites), we obtain the values  $(\rho_j)_{j=1,2,3} = (0.85, 0.09, 0.06)$ . For the sake of computational efficiency, we restrict the number of sources to values lower or equal to 3. Notice that the number of sources to which one can restrict the algorithm depends on the scale of the intersubtype variation of the virus genome at the informative sites of the genome.

### Estimation of emission probabilities

Using

$$P(\vec{q} | \vec{\rho}) = \rho_r \prod_{k=1}^r P(\vec{n}_{i_1^{(k)}}, \dots, \vec{n}_{i_m^{(k)}}),$$

we deduce the most likely source combination. Then, for a given source combination  $\vec{q} \in Q$ , we can estimate the emission probability of a nucleotide  $v$  for a particular source (assuming, for notational convenience, that the source under consideration is composed of the subtypes 1, ...,  $m$ ) by

$$\hat{p}_v = \int_{\vec{p}} p_v P(\vec{p} | \vec{n}_1, \dots, \vec{n}_m) d\vec{p}. \quad (5)$$

Using (2) and (3), we get

$$\begin{aligned} &P(\vec{p} | \vec{n}_1, \dots, \vec{n}_m) \\ &= \frac{P(\vec{n}_1, \dots, \vec{n}_m | \vec{p}) P(\vec{p})}{P(\vec{n}_1, \dots, \vec{n}_m)} \\ &= \frac{\prod_{i=1}^m P(\vec{n}_i | \vec{p}) P(\vec{p})}{P(\vec{n}_1, \dots, \vec{n}_m)} \\ &= \left[ \prod_{i=1}^m \left( \Gamma(|\vec{n}_i| + 1) \prod_{j=1}^N \frac{p_j^{n_{i,j}}}{\Gamma(n_{i,j} + 1)} \right) \right] \times \\ &\quad \frac{\Gamma(|\vec{\alpha}|)}{\Gamma(\prod_{j=1}^N \alpha_j)} \prod_{j=1}^N p_j^{\alpha_j - 1} \\ &\quad / \left[ \left( \prod_{i=1}^m \frac{\Gamma(|\vec{n}_i| + 1)}{\prod_{j=1}^N \Gamma(n_{i,j} + 1)} \right) \times \right. \\ &\quad \left. \frac{\Gamma(|\vec{\alpha}|)}{\Gamma(\prod_{j=1}^N \alpha_j)} \frac{\prod_{j=1}^N \Gamma(\sum_{i=1}^m n_{i,j} + \alpha_j)}{\Gamma(\sum_{i=1}^m |\vec{n}_i| + |\vec{\alpha}|)} \right] \\ &= \frac{\Gamma(\sum_{i=1}^m |\vec{n}_i| + |\vec{\alpha}|)}{\prod_{j=1}^N \Gamma(\sum_{i=1}^m n_{i,j} + \alpha_j)} \prod_{j=1}^N p_j^{\sum_{i=1}^m n_{i,j} + \alpha_j - 1}. \end{aligned}$$

Consequently, we can transform (5) into

$$\begin{aligned} \hat{p}_v &= \frac{\Gamma(\sum_{i=1}^m |\vec{n}_i| + |\vec{\alpha}|)}{\prod_{j=1}^N \Gamma(\sum_{i=1}^m n_{i,j} + \alpha_j)} \times \\ &\quad \int_{\vec{p}} p_v \prod_{j=1}^N p_j^{\sum_{i=1}^m n_{i,j} + \alpha_j - 1} d\vec{p} \end{aligned}$$

Finally, by using (4) we obtain the simple formula

$$\hat{p}_v = \frac{\sum_{i=1}^m n_{i,v} + \alpha_v}{\sum_{i=1}^m |\vec{n}_i| + |\vec{\alpha}|}.$$

## Results

In this section, we present the results of i) the calibration of USF on a) artificial HIV-1 recombinants and b) non-recombinant HIV-1 sequences designated as having emerged from a known subtype, ii) the application of USF to a) SIV sequences and b) sequences designated as unknown in the LANL HIV database (in the following called “Subtype U” sequences), and iii) the comparison of USF and BL.

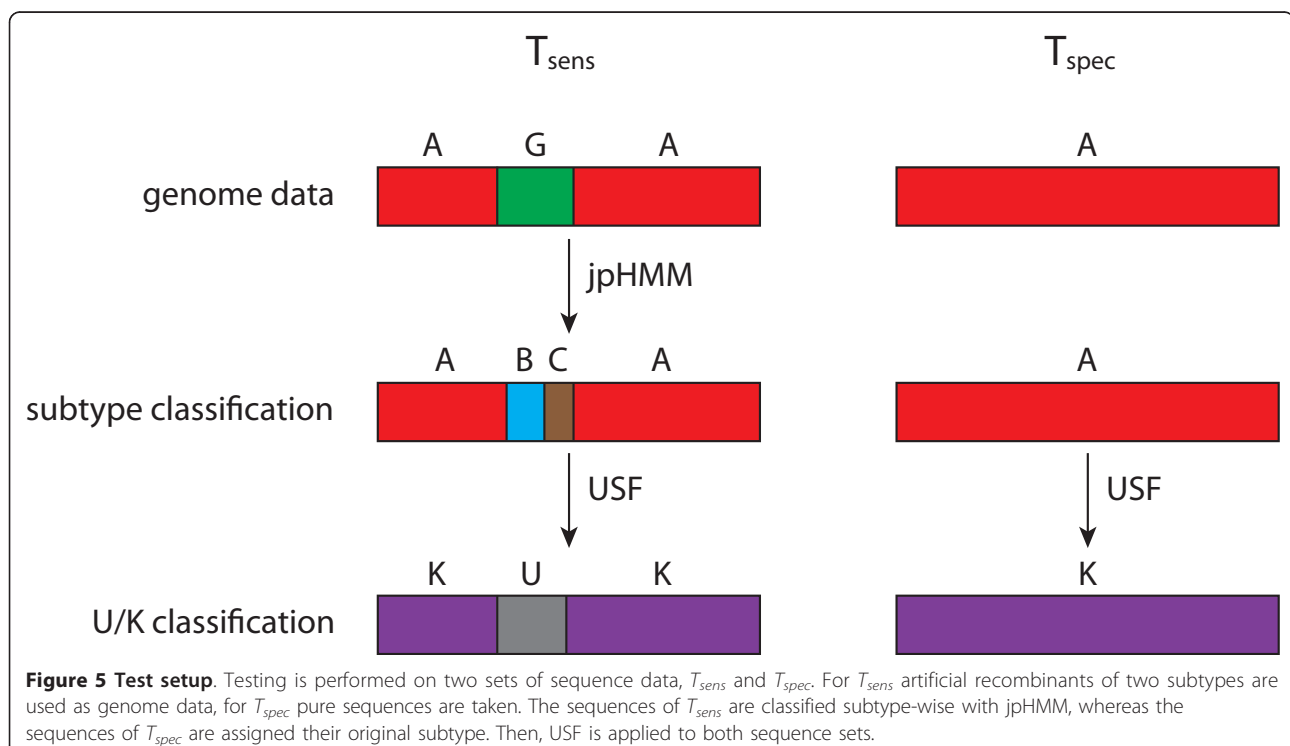
### Calibration

In order to calibrate USF and to investigate its behaviour in dependence of the choice of the parameters  $\eta$ ,  $p_1$  and  $p_2$ , we use two test settings, one of them suitable to assess the sensitivity of the algorithm, the other one the specificity. For the sensitivity, we remove one subtype from the MSA and consider it as unknown. Then we generate artificial recombinants of sequences from the “known” subtypes and the “unknown” subtype. For the specificity, we simply check whether sequences from the MSA are classified correctly. In both cases, we do not use the test data as training data for the emission probabilities of the HMMs. The testing setup is sketched in Figure 5. The MSA consists in all full-length HIV-1 Group M sequences, designated as stemming from a pure subtype in the LANL database, downloaded on 9th of July 2010.

### Test data

More precisely, we generate the following two sets of test sequences: (i) A set  $T_{sens}$  for measuring the

sensitivity with respect to the ability of the algorithm to detect genome segments stemming from an unknown subtype, and (ii) a set  $T_{spec}$  for measuring the specificity. The set  $T_{sens}$  is composed of 229 sequences generated by taking a sequence from subtypes A-D and F-G and replacing a segment of this sequence by a segment of a sequence from some other subtype. We call the subtype of the major part of the genome the ‘base subtype’ and the subtype of the inserted part of the genome the ‘insertion subtype’. A preliminary analysis shows that in case the subtypes H, J, or K have been assigned to the query sequence (or a part of it), USF is not suitable for a reliable detection of unclassifiable genome parts. Hence, for the role of a base subtype, those subtypes are excluded from our analysis. Nevertheless, segments of them may play the role of insertion subtypes. Segments of the subtypes B and D may not be combined, due to the small phylogenetic distance of those subtypes. Moreover, the replaced segments have a length of 1000 positions and their position has been chosen randomly.  $T_{spec}$  is composed of 265 sequences sampled from the genome-length sequences being classified as subtype A-D or F-G in the LANL HIV database (50 for all subtypes except for the subtypes F and G, for which only 35 and 30, respectively, sequences were available). For  $T_{spec}$  the sequences were assigned to the subtype they stem from according to their LANL HIV database designation. Therefore, if classified correctly, the complete sequence is classified as known. Any detected unknown regions



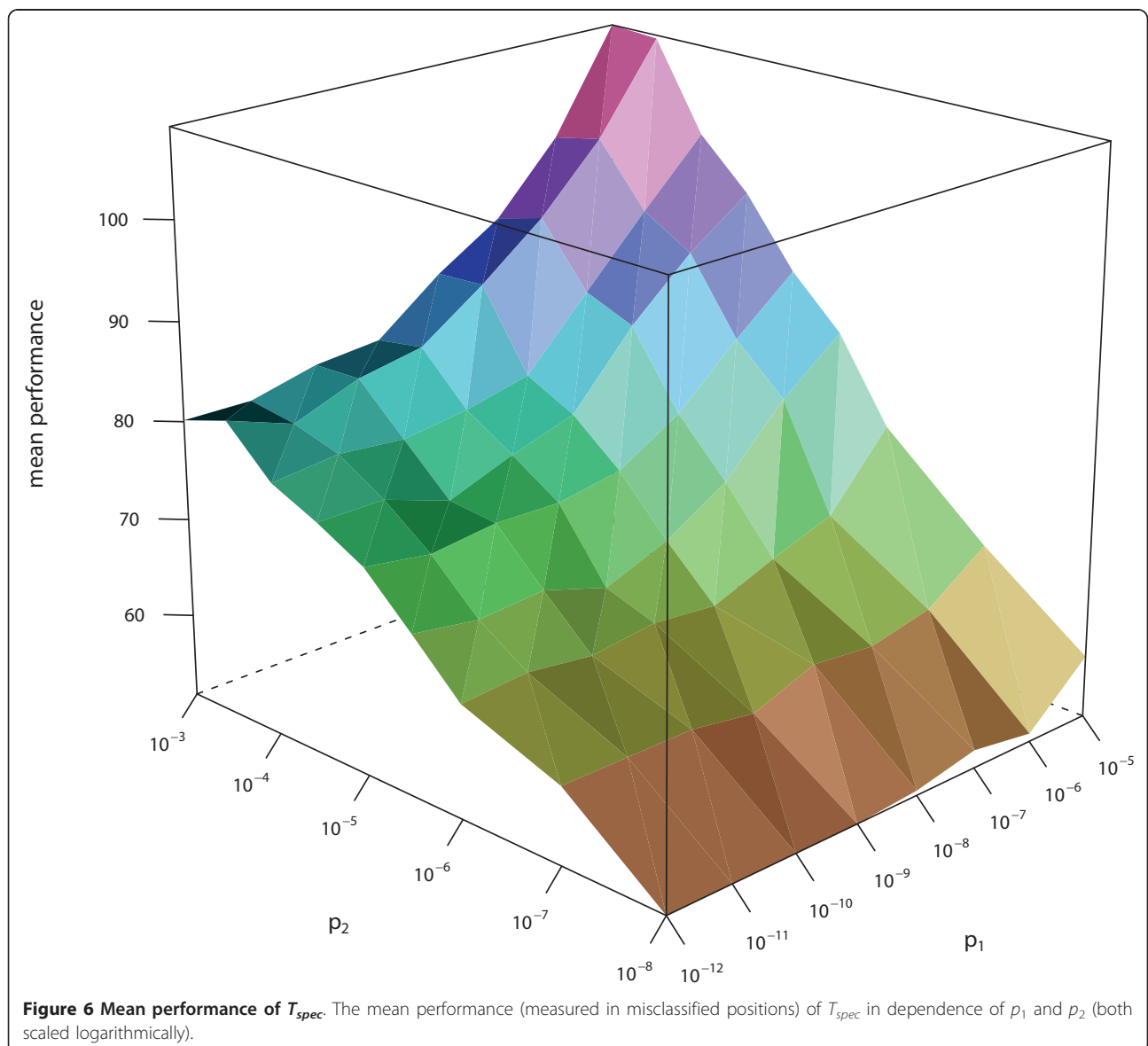
are counted as false positives. For  $T_{sens}$  we determine the subtype classification using the jpHMM, excluding the subtypes H, J, and K from the assignable subtypes.

#### Test results

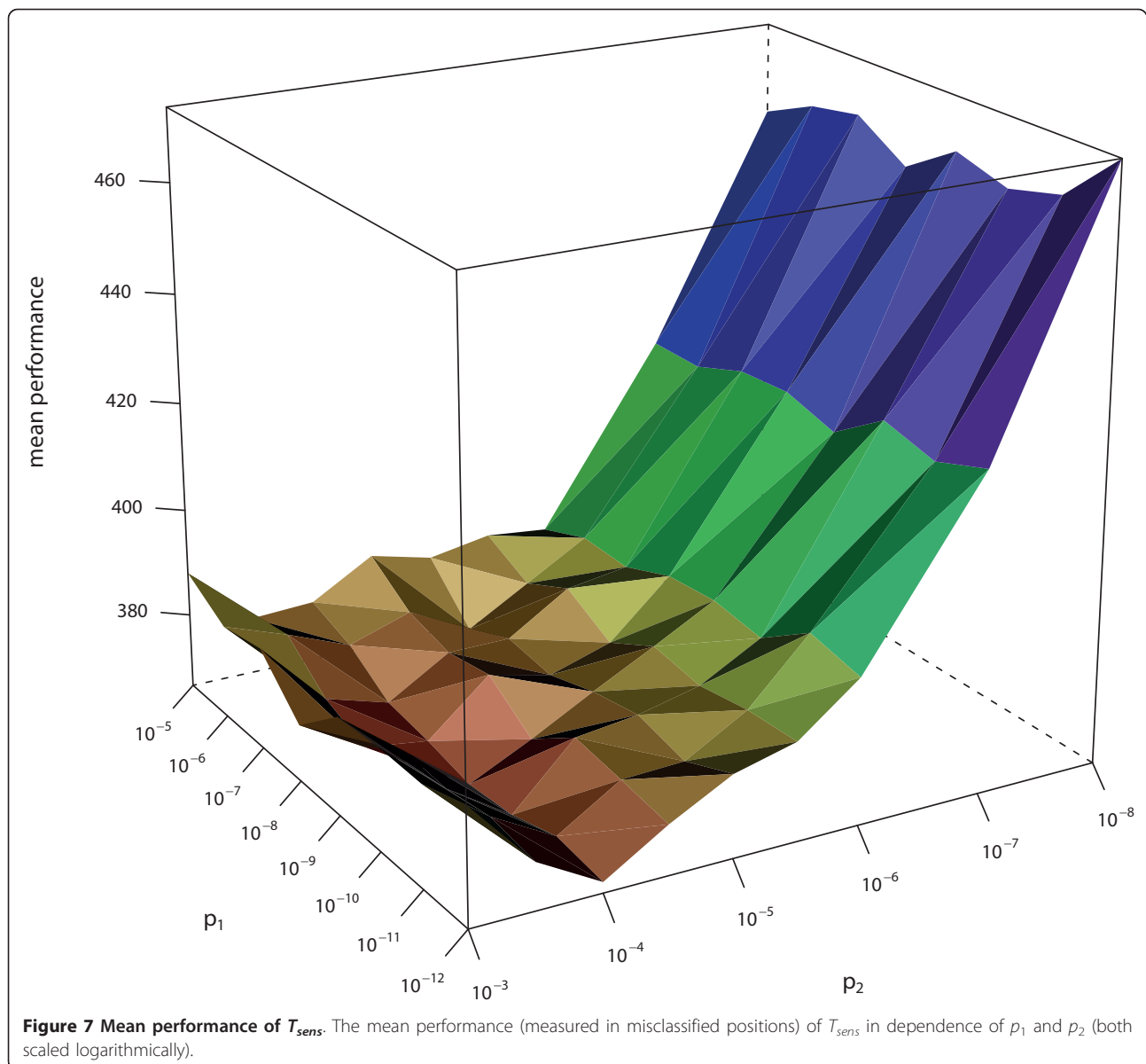
We measure the performance by counting how many positions in a sequence have been misclassified. Setting  $p_1 = 10^{-7}$  and  $p_2 = 10^{-4}$  (which seem to be reasonable values, in view of our experience gathered when applying the jpHMM to HIV), we determine  $\eta = 0.05$  as leading to the best tradeoff between sensitivity and specificity. With that choice of  $\eta$ , we evaluate the performance with respect to specificity and sensitivity on a grid for different choices of  $p_1$  and  $p_2$  (see Figure 6 and 7). From those data, we would recommend to choose  $p_1 = 10^{-9}$  and  $p_2 = 10^{-5}$ . In case a user has a different

priority with respect to specificity and sensitivity, he can adapt the values to his purpose. To achieve a higher sensitivity or specificity,  $p_1$  and  $p_2$  have to be increased or decreased, respectively. Increasing  $p_1$  merely results in a higher probability of finding any Subtype U fragments in the query sequence at all, whereas increasing  $p_2$  also leads to a higher number of Subtype U fragments to be found.

In Figure 8, resp. 9, the performance of the algorithm for  $T_{spec}$  resp.  $T_{sens}$  is displayed stratified by the assigned subtype  $T_{sens}$ , resp. the subtypes used for generating the artificial recombinants. Among the 6 sequences from  $T_{spec}$ , which yield the most misclassified positions, there are all 4 sequences of Subsubtype F2 and the sequence from Subsubtype F1, which cluster most closely to







Subsubtype F2 in a phylogenetic tree (using FastTree [25] and FigTree [26]).

To facilitate the testing technically, we restrict our analysis to the positions 808 to 8781 with respect to HXB [27]. Covering this part of the genome, we analyse the performance of USF in relatively conserved regions, as well as highly variable ones and we do not have to cope with the low number of sequences available for covering the boundary parts of the genome.

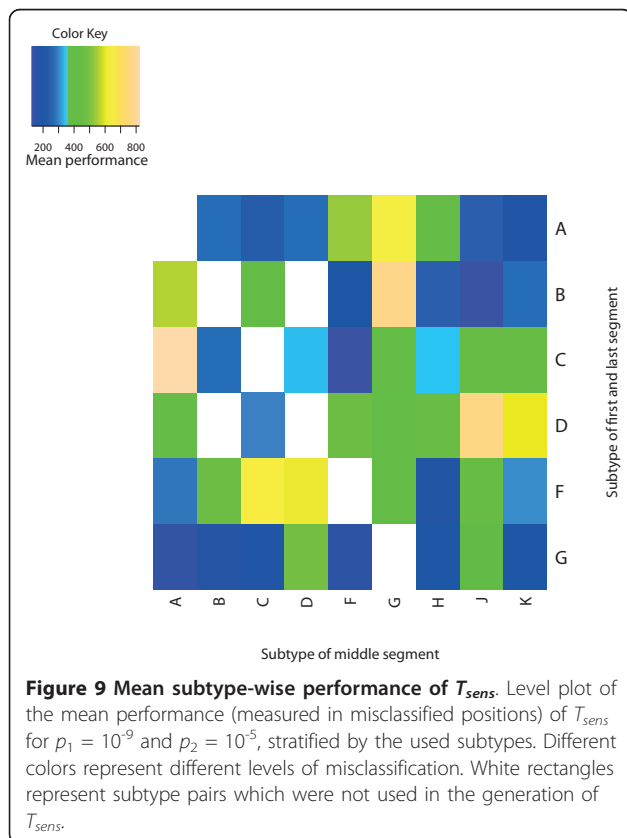
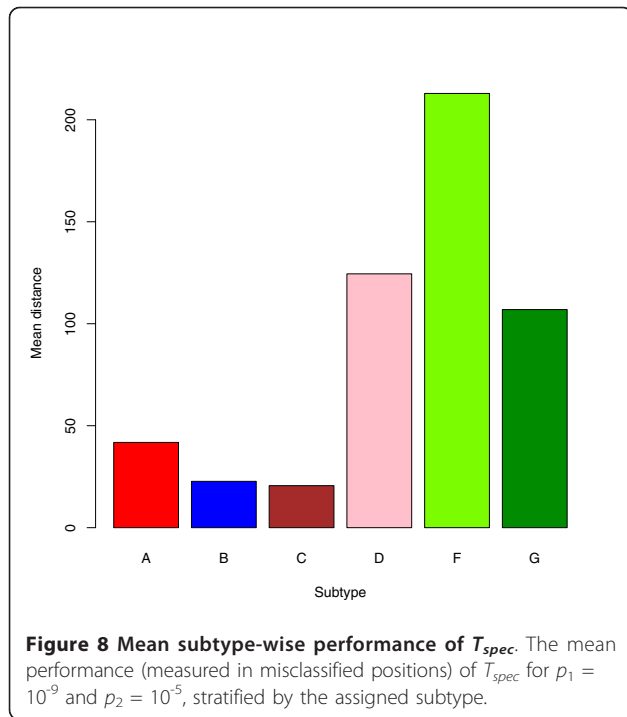
#### Theoretical determination of $\eta$

We have tried also to determine  $\eta$  by a theoretical approach. More precisely, we have simulated unknown subtypes by excluding a subtype from the data based on which the emission probabilities of pHMM U were

estimated. We then have chosen  $\eta$  such that the emission frequencies of the excluded subtype is estimated best (with respect to maximum likelihood). Unfortunately, this approach has failed to values of  $\eta$  smaller by an entire order of magnitude than the values found by means of the calibration described above. Consequently, we refrain from using this theoretical approach.

#### SIV sequences and Subtype U sequences

In order to check whether USF correctly classifies very divergent sequences, we have applied it to five full-length SIV genomes (AF103818, DQ373063, EF394356, U42720, X52154) from different parts of the SIV clade. As before, we did not allow for assigning subtypes H, J,



and K in the subtype classification. In the same way we have tested the 8 full-length Subtype U genomes (AF286236, AF457101, AY046058, EF029066, EF029067, EF029068, EF029069, FJ388921). Except the Subtype U sequence AY046058, all sequences have been correctly identified as completely unknown (about 8% of the genome have not been classified as unknown).

### Comparison with the BI

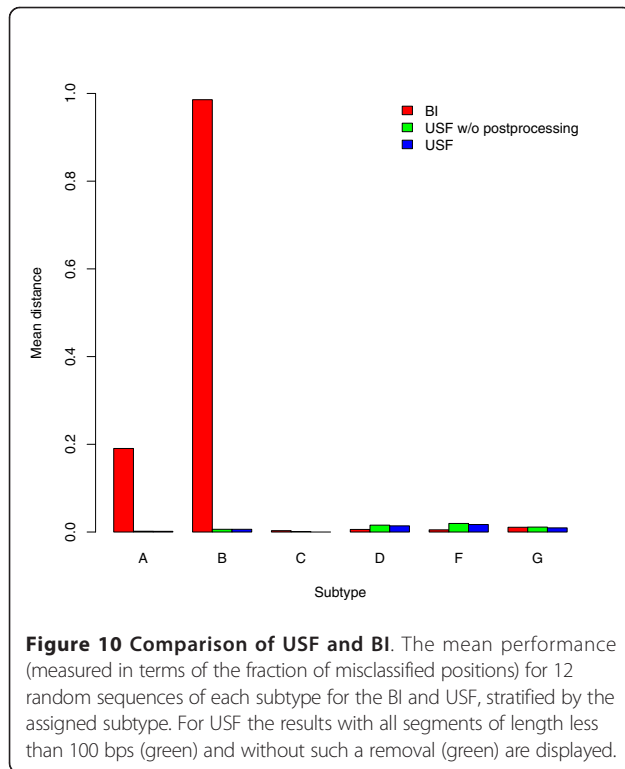
The BI is a method based on distance and phylogeny. It determines which parts of a query sequence should be classified among known sequences. Moving along the genome of a query sequence with a sliding window, the BI computes a ratio quantifying how closely the query sequence clusters with a subtype clade. On the basis of this quantity, it determines whether the respective part of a query sequence is unclassifiable with respect to the known subtypes.

We apply the BI to a subset of  $T_{spec}$ , as well as the SIV and Subtype U sequences used in the evaluation described in the subsection ‘Results - SIV sequences and Subtype U sequences’. As we had to carry out the testing manually, using the web interface of the BI [28], we had to confine ourselves to a limited number of sequences from  $T_{spec}$  and could not test the BI on  $T_{sens}$  at all. (For the purpose of the latter, it would have been necessary to reestimate the parameters of the BI after having removed a subtype from the training data. That, however, the web interface available does not allow.)

Application of the BI to the 5 SIV sequences and the 8 Subtype U sequences from the subsection ‘Results - SIV sequences and Subtype U sequences’ yields valid results in 3 and 4 cases, respectively. Out of these 7 sequences, all but one Subtype U sequence (AY046058) are classified correctly as completely unknown, with about 6% of the genome of AY046058 being misclassified.

Testing the BI on 12 sequences for each subtype from  $T_{spec}$ , yield the results illustrated in Figure 10. Since USF tends to misclassify very short segments as unknown for some subtypes, we also compare the BI with USF, removing all segments of length smaller than 100 bps from the outcome of USF.

Using the two-sided Wilcoxon signed-rank test, the version of USF without postprocessing performs significantly better (with respect to our position-wise measure) than the BI for the subtypes A and B. For the Subtype F, the BI is significantly better than USF ( $p = 0.05$ ). For the other subtypes, this test does not yield significant results. If USF is used in the version equipped with postprocessing, it yields significantly better results than the BI for the subtypes A and B, with the differences on the other subtypes being highly insignificant.



### Running time

Excluding the running time of ClustalW and jpHMM (described in [18,17]), the running time for a full length HIV-1 sequence is about 35 seconds on a Linux PC with 3 GHz and 4 GB RAM.

### Discussion and conclusions

We have presented USF, a tool for detection of unclassifiable segments in viral sequences. Using a probabilistic, model-driven approach, the tool is suitable in principle for all species (or other taxa) which are subdivided into subfamilies i) without too many indels separating the subfamilies and ii) where the phylogenetic distances between the subfamilies are not too inhomogeneous.

### Testing

We have applied USF to i) artificial recombinants of two subtypes (excluding one subtype from the training data to simulate an unknown subtype), ii) sequences designated (in the LANL HIV database) as originating from a pure subtype, iii) SIV sequences, and iv) Subtype U sequences. As far as feasible, we have compared our results with the only other tool available with the same scope, the Branching Index (BI).

### Performance of USF

Analyzing the performance of USF by subtype, one can see that it performs considerably better (with respect to

specificity) on the subtypes A-C than on D, F, and G, whereas it does not yield acceptable results for the subtypes H, J, and K. Its unsatisfactory performance on the last three subtypes does not come unexpectedly: The subtypes H, J, and K are composed of only 2 or 3 complete genome sequences, and that does not allow for a realistic modelling of the emission probabilities of a pHMM without using an information sharing protocol (see [23]). The weaker performance for subtypes D, F, and G might also be explicable by this effect, with the situation being obfuscated for the Subtype F by the fact that this subtype is divided into two subsubtypes.

The results of the application of USF to artificial recombinants can be explained in part also by the size of the involved subtypes: The poorest results are achieved when subtypes G or J, which both belong to the subtypes of smaller size, act as base subtype. Obviously, the size of the insertion subtype should not have any impact on the performance of USF (and the results also do not suggest that). Astonishingly, there does not seem to be a correlation between the phylogenetic distance of a pair of base and insertion subtypes and the performance of USF on the respective pair: Testing  $T_{sens}$  involves 46 pairs of subtypes. Considering the 13 pairs with the lowest phylogenetic distance, none of them is among the 3 poorest performing pairs and 3 are among the 7 poorest performing. As we have observed a poor performance of USF when the subtypes B and D are the base and insertion subtypes, we may conclude that, if the phylogenetic distance of the subtype pair is above a certain threshold, the performance of USF does not seem to depend on how remotely the subtypes are related exactly.

### Specificity of USF & BI

Comparing USF (employing the removal of very short segments in the outcome) with the BI with respect to specificity, USF, roughly speaking, performs better on some of the large size subtypes (A and B), whereas there are no significant differences on the large size Subtype C and the smaller size subtypes D, F, and G.

### Sensitivity of USF & BI

For a comparison of the sensitivities we had to restrict ourselves to the SIV and Subtype U sequences. In spite of the importance of the sensitivity to assess the performance of USF and BI, the analysis of this characteristic had to be carried out on quite a small test set, due to technical limitations in the implementation of the BI. Except for the Subtype U sequence AY046058, all SIV and Subtype U sequences were classified as unknown by USF as well as the BI. Since both tools detect the same sequence as not completely unknown (although different segments were detected as known), this might be a hint that the classification of AY046058 as a pure Subtype U

sequence is questionable. To conclude, our analysis does not reveal any significant differences between USF and the BI with respect to their sensitivity.

### Versatility of USF & BI

With respect to versatility, the BI seems to be slightly inferior to USF (at least in their current versions). As it is not possible to determine  $\eta$  by a theoretical approach (as described in paragraph 'Results - Calibration - Theoretical determination of  $\eta$ '), both methods require a parameter calibration on training data when applied to a new species, respectively taxon. Regarding breakpoint positions, the BI only provides a graph from which the user would have to deduce the breakpoint positions by visual inspection. Hence, it is not possible to run any automated procedures on the BI if breakpoint positions are required.

### Outlook

In the near future, we plan to incorporate our method in the jpHMM. This would lead to a tool capable not only of assigning the known subtypes of HIV-1 (or subfamilies of other viruses or species) to a query sequence (or parts of it) but also of detecting segments of the genome stemming from a subtype yet unknown. Moreover, we are currently working on the implementation of an information sharing protocol for the jpHMM, which then would attenuate the poor performance of USF when applied to the small size subtypes.

In addition, it has been discussed whether the core gene of some D/E-recombinants of Hepatitis B virus (HBV) might stem from a clade which became rare or extinct [29]. We will apply USF to HBV data in order to investigate this question.

Furthermore, it has been claimed that the HBV genotype G is a recombinant between i) an ancestor comparable in divergence to those between the genotypes A-E, contributing the S gene, and ii) an HBV variant which is much more divergent, contributing the rest of the genome [30]. In the face of this finding, we plan to incorporate more than one unknown subtype in our model so that different degrees of divergence can be modelled.

### Miscellaneous

As mentioned in the section 'Background', it is not possible to find unknown sequence segments by identifying regions of small a posteriori probabilities for all of the known subfamilies when applying the jpHMM for example. That is easily exemplified as follows: Let us assume there were only two subtypes A and B and we examined a sequence stemming from an unknown subtype which is genetically considerably closer to Subtype A than to Subtype B. Then this sequence would achieve very large a posteriori probabilities for Subtype A and

very small ones for Subtype B. Thus, it would falsely be classified as known.

USF is implemented in C++ and the source code is freely available (see additional file 1).

### Additional material

**Additional file 1: Source code.** C++ implementation of USF.

### Acknowledgements

We would like to thank Thomas Leitner for the encouragement to develop USF and Heinrich Hering for proofreading. This work was supported by the Deutsche Forschungsgemeinschaft (STA 1009/5-1).

### Author details

<sup>1</sup>Institute of Microbiology and Genetics, University of Göttingen, Goldschmidtstr. 1, 37077 Göttingen, Germany. <sup>2</sup>LMNO, Université de Caen, CNRS UMR 6139, 14032 Caen Cedex, France. <sup>3</sup>Institut für Mathematik und Informatik, Walther-Rathenau-Straße 47, 17487 Greifswald, Germany.

### Authors' contributions

TU implemented and validated the algorithm. AKS carried out modifications on jpHMM. JB provided statistical expertise. MS and BM guided the project, MS contributed to the model development. IB conceived the approach, developed, implemented and tested the algorithm and supervised the program development. All authors read and approved the final manuscript.

Received: 28 September 2010 Accepted: 11 April 2011

Published: 11 April 2011

### References

1. Korber B, Gaschen B, Yusim K, Thakallapally R, Kesmir C, Detours V: **Evolutionary and immunological implications of contemporary HIV-1 variation.** *Br Med Bull* 2001, **58**:19-42.
2. Leitner T: *The molecular epidemiology of human viruses* Springer Berlin; 2002.
3. Hrabec P, Fischer W, Bruno W, Leitner T, Kuiken C: **Comparative analysis of hepatitis C virus phylogenies from coding and non-coding regions: the 5' untranslated region (UTR) fails to classify subtypes.** *Virology Journal* 2006, **3**:103.
4. Rhodes T, Wargo H, Hu WS: **High Rates of Human Immunodeficiency Virus Type 1 Recombination: Near-Random Segregation of Markers One Kilobase Apart in One Round of Viral Replication.** *J Virol* 2003, **77**(20):11193-11200.
5. Hahn BH, Shaw GM, De KM, Sharp PM: **AIDS as a Zoonosis: Scientific and Public Health Implications.** *Science* 2000, **287**(5453):607-614.
6. Robertson DL, Anderson JP, Bradac JA, Carr JK, Foley B, Funkhouser RK, Gao F, Hahn BH, Kalish ML, Kuiken C, Learn GH, Leitner T, McCutchan F, Osmanov S, Peeters M, Pieniazek D, Salminen M, Sharp PM, Wolinsky S, Korber B: **HIV-1 nomenclature proposal.** *Science* 2000, **288**:55-57.
7. Hoelscher M, Dowling WE, Sanders-Buell E, Carr JK, Harris ME, Thomschke A, Robb ML, Bix DL, McCutchan FE: **Detection of HIV-1 subtypes, recombinants, and dual infections in East Africa by a multi-region hybridization assay.** *AIDS* 2002, **16**:2055-2064.
8. **LANL HIV Databases: CRFs.** 2011 [http://www.hiv.lanl.gov/content/sequence/HIV/CRFs/CRFs.html].
9. de Oliveira T, Deforche K, Cassol S, Salminen M, Paraskevis D, Seebregts C, Snoeck J, van Rensburg EJ, Wensing AMJ, van de Vijver DA, Boucher CA, Camacho R, Vandamme AM: **An automated genotyping system for analysis of HIV-1 and other microbial sequences.** *Bioinformatics* 2005, **21**(19):3797-3800.
10. **Recombinant Identification Program Web Interface.** [http://www.hiv.lanl.gov/content/sequence/RIP/RIP.html].
11. Zhang M, Schultz AK, Calef C, Kuiken C, Leitner T, Korber B, Morgenstern B, Stanke M: **jpHMM at GOBICS: a web server to detect genomic recombinations in HIV-1.** *Nucleic Acids Res* 2006, **34**(S2):W463-465.

12. Schultz AK, Zhang M, Bulla I, Leitner T, Korber B, Morgenstern B, Stanke M: **jpHMM: Improving the reliability of recombination prediction in HIV-1.** *Nucl Acids Res* 2009, **37** Web Server: W647-651.
13. Maydt J, Lengauer T: **Recco: recombination analysis using cost optimization.** *Bioinformatics* 2006, **22**(9):1064-1071.
14. Pandit A, Sinha S: **Using genomic signatures for HIV-1 sub-typing.** *BMC Bioinformatics* 2010, **11**(Suppl 1):S26.
15. Wilbe K, Salminen M, Laukkanen T, McCutchan F, Ray SC, Albert J, Leitner T: **Characterization of novel recombinant HIV-1 genomes using the branching index.** *Virology* 2003, **316**:116-125.
16. Hraber P, Kuiken C, Waugh M, Geer S, Bruno WJ, Leitner T: **Classification of hepatitis C virus and human immunodeficiency virus-1 sequences with the branching index.** *J Gen Virol* 2008, **89**(9):2098-2107.
17. Schultz AK, Zhang M, Leitner T, Kuiken C, Korber B, Morgenstern B, Stanke M: **A jumping profile Hidden Markov Model and applications to recombination sites in HIV and HCV genomes.** *BMC Bioinformatics* 2006, **7**:265.
18. Chenna R, Sugawara H, Koike T, Lopez R, Gibson TJ, Higgins DG, Thompson JD: **Multiple sequence alignment with the Clustal series of programs.** *Nucl Acids Res* 2003, **31**(13):3497-3500.
19. Krogh A, Brown M, Mian IS, Sjölander K, Haussler D: **Hidden Markov Models in Computational Biology: Applications to Protein Modeling.** *Journal of Molecular Biology* 1994, **235**(5):1501-1531.
20. Eddy S: **Profile hidden Markov models.** *Bioinformatics* 1998, **14**(9):755-763.
21. Viterbi A: **Error bounds for convolutional codes and an asymptotically optimum decoding algorithm.** *Information Theory, IEEE Transactions* 1967, **13**(2):260-269.
22. Sjölander K, Karplus K, Brown M, Hughey R, Krogh A, Mian I, Haussler D: **Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology.** *Comput Appl Biosci* 1996, **12**(4):327-345.
23. Brown DP, Krishnamurthy N, Sjölander K: **Automated Protein Subfamily Identification and Classification.** *PLoS Comput Biol* 2007, **3**(8):e160.
24. Akaike H: **A new look at the statistical model identification.** *Automatic Control, IEEE Transactions* 1974, **19**(6):716-723.
25. Price MN, Dehal PS, Arkin AP: **FastTree: Computing Large Minimum Evolution Trees with Profiles instead of a Distance Matrix.** *Mol Biol Evol* 2009, **26**(7):1641-1650.
26. FigTree. [<http://tree.bio.ed.ac.uk/software/gtree/>].
27. Korber B, Foley B, Kuiken C, Pillai S, Sodroski J: **Numbering Positions in HIV Relative to HXB2CG.** *Human Retroviruses and AIDS 1998, Los Alamos, NM: Theoretical Biology and Biophysics Group, Los Alamos National Laboratory* 1998, 102-111.
28. **Branching Index Web Interface.** [<http://www.hiv.lanl.gov/content/sequence/phyloplace/PhyloPlace.html>].
29. Simmonds P, Midgley S: **Recombination in the Genesis and Evolution of Hepatitis B Virus Genotypes.** *J Virol* 2005, **79**(24):15467-15476.
30. Kato H, Orito E, Gish RG, Sugouchi F, Suzuki S, Ueda R, Miyakawa Y, Mizokami M: **Characteristics of Hepatitis B Virus Isolates of Genotype G and Their Phylogenetic Differences from the Other Six Genotypes (A through F).** *J Virol* 2002, **76**(12):6131-6137.

doi:10.1186/1471-2105-12-93

**Cite this article as:** Unterthiner et al.: Detection of viral sequence fragments of HIV-1 subfamilies yet unknown. *BMC Bioinformatics* 2011 12:93.

**Submit your next manuscript to BioMed Central and take full advantage of:**

- Convenient online submission
- Thorough peer review
- No space constraints or color figure charges
- Immediate publication on acceptance
- Inclusion in PubMed, CAS, Scopus and Google Scholar
- Research which is freely available for redistribution

Submit your manuscript at  
[www.biomedcentral.com/submit](http://www.biomedcentral.com/submit)

