recently developed composition-based approach developed in our group (?), trains interpolated Markov models (IMMs) on known genomes in order to classify sequences. Alternatively, genome signatures can be used for clustering by learning the signatures from the set of sequences in an unsupervised fashion without the use of known genomes (?). While supervised learning has proven useful in practice, shortcomings exist. Methods trained on the genomes in GenBank make an implicit assumption that they are representative of microbes waiting to be found by metagenomics projects. This assumption is clearly violated by many if not most metagenomes samples. Supervised learning methods that tread carefully with respect to the potential biases caused by this assumption can still be useful analytical tools for many environments. But unsupervised learning approaches are required when publicly available genomes are a poor fit to the data.

As an alternative to oligonucleotide frequencies, Markov chain models have shown great promise for characterizing genomic content (?), and have been implemented for both supervised classification (?) and unsupervised clustering (?) methods. In this paper, we cluster sequences using interpolated Markov models (IMMs), a type of Markov chain model that adapts the model complexity to take advantage of variable amounts of training data. This strategy is well suited to metagenomics clustering problems, where the size of the data set and the relative abundances of the species in the mix can vary widely. Our clustering framework builds upon one used to cluster sequences using hidden Markov models (?) where optimization is performed iteratively by a relative of the k-means clustering algorithm. We refer to our method as SCIMM (Sequence Clustering with Interpolated Markov Models).

We test SCIMM on simulated metagenomic datasets of reads from mixtures of randomly selected genomes and demonstrate improvement on the performance of the previous metagenomic sequence clustering programs CompostBin (?) and LikelyBin (?). We also assess the limitations of unsupervised learning on complex data sets, and describe how a combination of SCIMM and Phymm, which we call PHYSCIMM, performs better for clustering when useful training data is available.

2 METHODS

Markov models have proven to be an invaluable tool for sequence analysis (?), including capturing genome signatures (?). Here we present a clustering algorithm called SCIMM in which we use interpolated Markov models (IMMs) to model clusters of sequences. Clustering of sequences is performed using an iterative variant of the k-means algorithm.

2.1 Interpolated Markov models

A fixed-order Markov chain is a model for generating a sequence of outputs (in this case, nucleotides in a DNA molecule) in which the i-th element in the sequence has a distribution that is conditional on the previous w elements. Thus, given a sequence s and a model m, we can compute the probability that s was generated by m by walking along the sequence and multiplying the conditional probabilities.

$$P(s|m) = \prod_{i=1}^{|s|} P_m(s_i | s_{i-w+1} \ldots s_{i-1} )$$

IMMs were first used for modeling DNA sequences as part of the Glimmer gene finding system (?). IMMs are variable-order Markov chains, and as such are a strict generalization of fixed-order Markov chains. The variant of IMMs used in our system, introduced in the Glimmer 2.0 gene finder (?), allow the nucleotide distributions to be conditional on a subset of indexes in the preceding size w window (see Figure 2). These indexes are chosen using a mutual information computation to be the most informative for the distribution of the next nucleotide. Past work has found that increasing the order of the Markov model (e.g., using a 5-th-order model instead of a 2-th-order model) usually leads to more accurate predictions, but the order of the model is limited by the amount of data available. IMMs dynamically adjust the size of the Markov model based on the data, which allows them to make the most of whatever information is available. This is particularly useful for clustering of metagenomic sequences where the amount of sequence from each species may differ widely due to differential abundance of organisms and the amount of sequencing performed on the sample. For full details of the training of IMMs and sequence likelihood computations, see the Glimmer description (?).

2.2 k-means clustering framework

The k-means algorithm is a widely used, simple and effective method for clustering data points. Points are modeled as having come from k sources, each represented by a cluster mean. The algorithm begins by initializing these cluster means, e.g., by randomly choosing k data points. Next, one repeats the following steps. First, compute the distance between all points and the k cluster means. Second, assign each point to its nearest cluster. Finally, recompute the cluster means using the current assignment of points to clusters. After a number of iterations, one arrives at a stable partitioning of data points that approximates the minimum sum of distances between data points and their assigned cluster means.

2.3 SCIMM

SCIMM uses the same general algorithm as k-means, where the data points are DNA sequences and the cluster models are IMMs. Here the goal is to find the IMMs that maximize the likelihood of generating the sequences. The algorithm begins by initializing k IMMs (discussed in detail below), then the following steps are repeated. First, for all sequences s and all IMMs m, compute the likelihood that s was generated by m. Second, assign each sequence to the cluster corresponding to the IMM that generates it with greatest likelihood. Finally, re-train the IMMs on the sequences currently assigned to their corresponding clusters. The algorithm halts when fewer than 0.1% of the sequences change clusters. This loop is depicted in Figure 2. Over the course of the iterations, the IMMs converge to a set that should represent the phylogenetic sources.

2.4 Initial partitioning

SCIMM inherits the simplicity and effectiveness of the k-means algorithm, but also its sensitivity to initial conditions. We found that the likelihood

![Fig. 1. Markov models.](image)

ACGTACCGTATC

<table>
<thead>
<tr>
<th>b</th>
<th>P(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>.4</td>
</tr>
<tr>
<td>C</td>
<td>.2</td>
</tr>
<tr>
<td>G</td>
<td>.2</td>
</tr>
<tr>
<td>T</td>
<td>.1</td>
</tr>
</tbody>
</table>

---

Kelley and Salzberg