

Metrics and similarity measures for hidden Markov models

Rune B. Lyngsø* and Christian N. S. Pedersen† and Henrik Nielsen‡

Abstract

Hidden Markov models were introduced in the beginning of the 1970's as a tool in speech recognition. During the last decade they have been found useful in addressing problems in computational biology such as characterising sequence families, gene finding, structure prediction and phylogenetic analysis. In this paper we propose several measures between hidden Markov models. We give an efficient algorithm that computes the measures for left-right models, e.g. profile hidden Markov models, and briefly discuss how to extend the algorithm to other types of models. We present an experiment using the measures to compare hidden Markov models for three classes of signal peptides.

Introduction

A hidden Markov model describes a probability distribution over a potentially infinite set of sequences. It is convenient to think of a hidden Markov model as generating a sequence according to some probability distribution by following a first order Markov chain of states, called the path, from a specific start-state to a specific end-state and emitting a symbol according to some probability distribution each time a state is entered. One strength of hidden Markov models is the ability efficiently to compute the probability of a given sequence as well as the most probable path that generates a given sequence. Hidden Markov models were introduced in the beginning of the 1970's as a tool in speech recognition. In speech recognition the set of sequences might correspond to digitised sequences of hu-

man speech and the most likely path for a given sequence is the corresponding sequence of words. Rabiner (Rabiner 1989) gives a good introduction to the theory of hidden Markov models and their applications to speech recognition.

Hidden Markov models were introduced in computational biology in 1989 by Churchill (Churchill 1989). Durbin et al. (Durbin *et al.* 1998) and Eddy (Eddy 1996; 1998) are good overviews of the use of hidden Markov models in computational biology. One of the most popular applications is to use them to characterise sequence families by using so called profile hidden Markov models introduced by Krogh et al. (Krogh *et al.* 1994). For a profile hidden Markov model the probability of a given sequence indicates how likely it is that the sequence is a member of the modelled sequence family, and the most likely path for a given sequence corresponds to an alignment of the sequence against the modelled sequence family.

An important advance in the use of hidden Markov models in computational biology within the last two years, is the fact that several large libraries of profile hidden Markov models have become available (Eddy 1998). These libraries not only make it possible to classify new sequences, but also open up the possibility of comparing sequence families by comparing the profiles of the families instead of comparing the individual members of the families. To our knowledge little work has been done in this area. In this paper we propose measures for hidden Markov models that can be used to address this problem. The measures are based on what we call the co-emission probability of two hidden Markov models. We present an efficient algorithm that computes the measures for profile hidden Markov models and observe that the left-right architecture is the only special property of profile hidden Markov models required by the algorithm. We briefly mention how to extend the algorithm to broader classes of models and how to approximate the measures for general hidden Markov models. The method can easily be adapted to various special cases, e.g. if it is required that paths pass through certain states.

As the algorithm we present is not limited to profile hidden Markov models, we have chosen to emphasise

* Department of Computer Science, University of Aarhus, Denmark. E-mail: rlyngsoe@daimi.au.dk. Work done in part while visiting the Institute for Biomedical Computing at Washington University, St. Louis.

† Basic Research In Computer Science, Centre of the Danish National Research Foundation, University of Aarhus, Denmark. E-mail: cstorm@brics.dk.

‡ Center for Biological Sequence Analysis, Centre of the Danish National Research Foundation, Technical University of Denmark, Denmark. E-mail: hnielsen@cbs.dtu.dk.

this generality by presenting an application to a set of hidden Markov models for signal peptides. These models do not strictly follow the profile architecture and consequently cannot be compared using profile alignment (Gotoh 1993).

The rest of the paper is organised as follows. We first discuss hidden Markov models in more detail and introduce the co-emission probability of two hidden Markov models. We formulate an algorithm for computing this probability of two profile hidden Markov models, observe that it applies to all left-right models and briefly discuss extensions to other types of models. Then we use the co-emission probability to formulate several measures between hidden Markov models. Finally we present an experiment using the method to compare three classes of signal peptides, and briefly discuss how to compute relaxed versions of the co-emission probability.

Hidden Markov models

Let M be a hidden Markov model that generates sequences over some finite alphabet Σ with probability distribution P_M , i.e. $P_M(s)$ denotes the probability of $s \in \Sigma^*$ under model M . Like a classical Markov model, a hidden Markov model consists of a set of interconnected states. We use $P_q(q')$ to denote the probability of a transition from state q to state q' . These probabilities are usually called *state transition probabilities*. The transition structure of a hidden Markov model is often shown as a directed graph with a node for each state, and an edge between two nodes if the corresponding state transition probability is non-zero. Figure 1 shows an example of a transition structure. Unlike a classical Markov model, a state in a hidden Markov model can generate or emit a symbol according to a local probability distribution over all possible symbols. We use $P_q(\sigma)$ to denote the probability of generating or emitting symbol $\sigma \in \Sigma$ in state q . These probabilities are usually called *symbol emission probabilities*. If a state does not have symbol emission probabilities we say that the state is a silent state.

It is often convenient to think of a hidden Markov model as a generative model, in which a run generates or emits a sequence $s \in \Sigma^*$ with probability $P_M(s)$. A run of a hidden Markov model begins in a special start-state and continues from state to state according to the state transition probabilities until a special end-state is reached. Each time a non-silent state is entered, a symbol is emitted according to the symbol emission probabilities of the state. A run thus results in a Markovian sequence of states as well as a generated sequence of symbols. The name “hidden Markov model” comes from the fact that the Markovian sequence of states, also called the path, is hidden, while only the generated sequence of symbols is observable.

Hidden Markov models have found applications in many areas of computational biology, e.g. gene finding (Krogh 1997) and protein structure prediction (Sonnhammer, von Heijne, & Krogh 1998), but

probably the most popular use is as *profiles* for sequence families. A profile is a position-dependent scoring scheme that captures the characteristics of a sequence family, in the sense that the score peaks around members of the family. Profiles are useful when searching for unknown members of a sequence family and several methods have been used to construct and use profiles (Gribskov, McLachlan, & Eisenberg 1987; Luthy, Bowie, & Eisenberg 1992; Taylor 1986). Krogh et al. (Krogh *et al.* 1994) realized that simple hidden Markov models, which they called profile hidden Markov models, were able to capture all other profile methods.

The states of a profile hidden Markov model are divided into match-, insert- and delete-states. Figure 1 illustrates the transition structure of a simple profile hidden Markov model. Note the highly repetitive transition structure. Each of the repeated elements consisting of a match-, insert- and delete-state models a position in the consensus sequence for the sequence family. The silent delete-state makes it possible to skip a position while the self-loop on the insert-state makes it possible to insert one or more symbols between two positions. Another distinctive feature of the structure of profile hidden Markov models is the absence of cycles, except for the self-loops on the insert-states. Markov models with this property are generally referred to as left-right (Jelinek 1976) models, as they can be drawn such that all transitions go from left to right.

The state transition and symbol emission probabilities of a profile hidden Markov model (the parameters of the model) should be such that $P_M(s)$ is significant if s is a member of the sequence family. These probabilities can be derived from a multiple alignment of the sequence family, but more importantly, several methods exist to estimate them (or train the model) if a multiple alignment is not available (Baldi *et al.* 1994; Durbin *et al.* 1998; Eddy 1998).

Co-emission probability of two models

When using a profile hidden Markov model, it is sometimes sufficient just to focus on the most probable path through the model, e.g. when using a profile hidden Markov model to generate alignments. It is, however, well known that profile hidden Markov models possess a lot more information than the most probable paths, as they allow the generation of an infinity of sequences, each by a multitude of paths. Thus, when comparing two profile hidden Markov models, one should look at the entire spectrum of sequences and probabilities.

In this section we will describe how to compute the probability that two profile hidden Markov models independently generate the same sequence, that is for models M_1 and M_2 generating sequences over an alphabet Σ we compute

$$\sum_{s \in \Sigma^*} P_{M_1}(s)P_{M_2}(s). \quad (1)$$

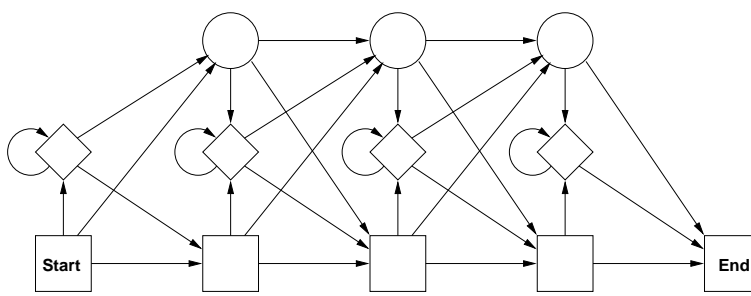


Figure 1: The transition structure of a profile hidden Markov model. The squares are the match-states, the diamonds are the insert-states and the circles are the silent delete-states.

We will call this the *co-emission probability* of the two models. The algorithm we present to compute the co-emission probability is a dynamic programming algorithm similar to the algorithm for computing the probability that a hidden Markov model will generate a specific sequence (Durbin *et al.* 1998, Chapter 3). We will describe how to handle the extra complications arising when exchanging the sequence with a profile hidden Markov model.

When computing the probability that a hidden Markov model M generates a sequence $s = s_1 \dots s_n$, a table indexed by a state from M and an index from s is usually built. An entry (q, i) in this table holds the probability of being in the state q in M and having generated the prefix $s_1 \dots s_i$ of s . We will use a similar approach to compute the co-emission probability. Given two hidden Markov models M_1 and M_2 , we will describe how to build a table, A , indexed by states from the two hidden Markov models, such that the entry $A(q, q')$ – where q is a state of M_1 and q' is a state of M_2 – holds the probability of being in state q in M_1 and q' in M_2 and having independently generated identical sequences on the paths to q and q' . We will denote this entry with $A(q, q')$. The entry indexed by the two end-states will then hold the probability of being in the end-states and having generated identical sequences, that is the co-emission probability.

To build the table, A , we have to specify how to fill out all entries of A . For a specific entry $A(q, q')$ this depends on the types of states q and q' . As explained in the previous section, a profile hidden Markov model has three types of states (insert-, match- and delete-states) and two special states (start and end). We postpone the treatment of the special states until we have described how to handle the other types of states. For reasons of succinctness we will treat insert- and match-states as special cases of a more general type we will call a *generate-state*; this type encompasses all non-silent states of the profile hidden Markov models.

The generate-state will be a merging of match-states and insert-states, thus both allowing a transition to itself and having a transition from the previous insert-state; a match-state can thus be viewed as a generate-state with probability zero of choosing the transition to

itself, and an insert-state can be viewed as a generate-state with probability zero of choosing the transition from the previous insert-state. Note that this merging of match- and insert-states is only conceptual; we do not physically merge any states, but just handle the two types of states in a uniform way. This leaves two types of states and thus four different pairs of types. This number can be reduced to three, by observing that the two cases of a generate/delete-pair are symmetric, and thus can be handled the same way.

The rationale behind the algorithm is to split paths up in the last transition(s)¹ and all that preceded this. We will thus need to be able to refer to the states with transitions to q and q' . In the following, m , i and d will refer to the match-, insert- and delete-state with a transition to q , and m' , i' and d' to those with a transition to q' . Observe that if q (or q') is an insert-state, then i (or i') is the *previous* insert-state which, by the generate-state generalisation, has a transition to q (or q') with probability zero.

delete/delete entry Assume that q and q' are both delete-states. As these states don't emit symbols, we just have to sum over all possible combinations of immediate predecessors of q and q' , of the probability of being in these states and having independently generated identical sequences, multiplied by the co-emission probability of independently choosing the transitions to q and q' . For the calculation of $A(q, q')$ we thus get the equation

$$\begin{aligned}
 A(q, q') = & \\
 & A(m, m')P_m(q)P_{m'}(q') + A(m, i')P_m(q)P_{i'}(q') \\
 & + A(m, d')P_m(q)P_{d'}(q') + A(i, m')P_i(q)P_{m'}(q') \quad (2) \\
 & + A(i, i')P_i(q)P_{i'}(q') + A(i, d')P_i(q)P_{d'}(q') \\
 & + A(d, m')P_d(q)P_{m'}(q') + A(d, i')P_d(q)P_{i'}(q') \\
 & + A(d, d')P_d(q)P_{d'}(q').
 \end{aligned}$$

delete/generate entry Assume that q is a delete-state and q' is a generate-state. Envision paths leading to q and q' respectively while independently gen-

¹In some of the cases explained below, we will only extend the path in one of the models with an extra transition, hence the unspecificity.

erating the same sequence. As q does not emit symbols while q' does, the path to q 's immediate predecessor (that is, the path to q with the actual transition to q removed) must also have generated the same sequence as the path to q' . We thus have to sum over all immediate predecessors of q , of the probability of being in this state and in q' and having generated identical sequences, multiplied by the probability of choosing the transition to q . For the calculation of $A(q, q')$ in this case we thus get the following equation

$$A(q, q') = A(m, q')P_m(q) + A(i, q')P_i(q) + A(d, q')P_d(q). \quad (3)$$

generate/generate entry Assume that q and q' are both generate-states. The last character in sequences generated on the paths to q and q' are generated by q and q' respectively. We will denote the probability that these two states independently generate the same symbol by p , and it is an easy observation that

$$p = \sum_{\sigma \in \Sigma} P_q(\sigma)P_{q'}(\sigma). \quad (4)$$

The problem with generate/generate entries is that the last transitions on paths to q and q' might actually come from q and q' themselves, due to the self-loops of generate states. It thus seems that we need $A(q, q')$ to be able to compute $A(q, q')$!

So let us start out by assuming that at most one of the paths to q and q' has a self-loop transition as the last transition. Then we can easily compute the probability of being in q and q' and having independently generated the same sequence on the paths to q and q' , by summing over all combinations of states with transitions to q and q' (including combinations with either q or q' but not both) the probabilities of these combinations, multiplied by p (for independently generating the same symbol at q and q') and the joint probability of independently choosing the transitions to q and q' . We denote this probability by $A_0(q, q')$, and by the above argument the equation for computing it is

$$\begin{aligned} A_0(q, q') = & p(A(m, m')P_m(q)P_{m'}(q') + A(m, i')P_m(q)P_{i'}(q') \\ & + A(m, d')P_m(q)P_{d'}(q') + A(m, q')P_m(q)P_{q'}(q') \\ & + A(i, m')P_i(q)P_{m'}(q') + A(i, i')P_i(q)P_{i'}(q') \\ & + A(i, d')P_i(q)P_{d'}(q') + A(i, q')P_i(q)P_{q'}(q') \quad (5) \\ & + A(d, m')P_d(q)P_{m'}(q') + A(d, i')P_d(q)P_{i'}(q') \\ & + A(d, d')P_d(q)P_{d'}(q') + A(d, q')P_d(q)P_{q'}(q') \\ & + A(q, m')P_q(q)P_{m'}(q') + A(q, i')P_q(q)P_{i'}(q') \\ & + A(q, d')P_q(q)P_{d'}(q')). \end{aligned}$$

Now let us cautiously proceed, by considering a pair of paths where one of the paths has exactly one self-loop transition in the end, and the other path has at least one self-loop transition in the end. The probability – that we surprisingly call $A_1(q, q')$ – of getting

to q and q' along such paths while generating the same sequences is the probability of getting to q and q' along paths that do not both have a self-loop transition in the end, multiplied by the joint probability of independently choosing the self-loop transitions, and the probability of q and q' emitting the same symbols. But this is just

$$A_1(q, q') = rA_0(q, q'), \quad (6)$$

where

$$r = pP_q(q)P_{q'}(q') \quad (7)$$

is the probability of independently choosing the self-loop transitions and emitting the same symbols in q and q' . Similarly we can define $A_k(q, q')$, and by induction it is easily proven that

$$A_k(q, q') = rA_{k-1}(q, q') = r^k A_0(q, q'). \quad (8)$$

As any finite path ending in q or q' must have a finite number of self-loop transitions in the end, we get

$$\begin{aligned} A(q, q') &= \sum_{k=0}^{\infty} A_k(q, q') \\ &= \sum_{k=0}^{\infty} r^k A_0(q, q') \quad (9) \\ &= \frac{1}{1-r} A_0(q, q'). \end{aligned}$$

Despite the fact that there is an infinite number of cases to consider, we observe that the sum over the probabilities of all these cases comes out as a geometric series that can easily be computed.

Each of the entries of A pertaining to match- insert- and delete-states can thus be computed in constant time using the above equations. As for the start-states (denoted by s and s') we initialise $A(s, s')$ to 1 (as we have not started generating anything and the empty sequence is identical to itself). Otherwise, even though they do not generate any symbols, we will treat the start-states as generate states; this allows for choosing an initial sequence of delete-states in one of the models. The start-states are the only possible immediate predecessors for the first insert-states, and together with the first insert-states the only immediate predecessors of the first match- and delete-states; the equations for the entries indexed by any of these states can trivially be modified according to this. The end-states (denoted by e and e') do not emit any symbols and are thus akin to delete-states, and can be treated the same way.

The co-emission probability of M_1 and M_2 is the probability of being in the states e and e' and having independently generated the same sequences, and can be found by looking up $A(e, e')$. In the rest of this paper we will use $A(M_1, M_2)$ to denote the co-emission probability of M_1 and M_2 . As all entries of A can be computed in constant time, we can compute the co-emission probability of M_1 and M_2 in time $O(n_1 n_2)$ where n_i denotes the number of states in M_i . The straightforward space requirement is also $O(n_1 n_2)$ but can be reduced to $O(n_1)$ by a standard trick (Gusfield 1997, Chapter 11).

One can observe that the method described here is not limited to profile hidden Markov models, but can be applied to all left-right hidden Markov models. If we replace generate-state with non-silent state and delete-state with silent state in the above description, we simply have to sum over all pairs of predecessors in the two cases of identical types of states, and over all predecessors of the silent state in the case where we are computing the entry for a silent/non-silent pair of states. The time required to compute the co-emission probability becomes $O(m_1 m_2)$ where m_i is the number of transitions in M_i , as each pair of transitions is considered for at most one entry of A .

Actually we can relax the requirement from requiring that the models being left-right models having only self-loops as cycles, to just demanding that each state takes part in at most one cycle. With some caution, the technique described above can be extended to handle this situation with no further increase in the time complexity (Lyngsø, Pedersen, & Nielsen 1999).

If we extend the type of models even further to unrestricted hidden Markov models, cycles can be intertwined, and the method of recognising the geometric series in the summation for the probability of all finite sequences is no longer available. By iterating the computation of co-emission probabilities at all pairs of states, thus finding the co-emission probabilities for longer and longer paths, the total co-emission probability can however be approximated efficiently (Lyngsø, Pedersen, & Nielsen 1999).

Measures on hidden Markov Models

Based on the co-emission probability we define two metrics that hopefully, to some extent, express how similar the family of sequences represented by two hidden Markov models are. A problem with the co-emission probability is that the models having the largest co-emission probability with a specific model, M , usually will not include M itself, as shown by the following proposition.

Proposition 1 *Let M be a hidden Markov model and $p = \max\{P_M(s) \mid s \in \Sigma^*\}$. The maximum co-emission probability with M attainable for any hidden Markov model is p . Furthermore, the hidden Markov models attaining this co-emission probability with M , are exactly those models, M' , for which $P_{M'}(s) > 0 \Leftrightarrow P_M(s) = p$ for all $s \in \Sigma^*$.*

Proof. Let M' be a hidden Markov model with $P_{M'}(s) > 0 \Leftrightarrow P_M(s) = p$. Then

$$\sum_{s \in \Sigma^*, P_M(s)=p} P_{M'}(s) = 1 \quad (10)$$

$$\sum_{s \in \Sigma^*} P_M(s) P_{M'}(s) = \sum_{s \in \Sigma^*, P_M(s)=p} P_M(s) P_{M'}(s) = p. \quad (11)$$

Now let M' be a hidden Markov model with $P_{M'}(s') = p' > 0$ for some $s' \in \Sigma^*$ with $P_M(s') = p'' < p$. Then the co-emission probability of M and M' is

$$\begin{aligned} \sum_{s \in \Sigma^*} P_M(s) P_{M'}(s) &= p' p'' + \sum_{s \in \Sigma^* \setminus \{s'\}} P_M(s) P_{M'}(s) \\ &\leq p' p'' + (1 - p') p \\ &< p. \end{aligned} \quad (12)$$

This proves that a hidden Markov model, M' , has maximum co-emission probability, p , with M , if and only if the assertion of the proposition is fulfilled. \square

Proposition 1 indicates that the co-emission probability of two models not only depends on how alike they are, but also on how ‘self-confident’ the models are, that is, to what extent the probabilities are concentrated to a small subset of all possible sequences.

Another way to explain this undesirable property of the co-emission probability, is to interpret Markov models – or rather the probability distribution over finite sequences of Markov models – as vectors in the infinite dimensional space spanned by all finite sequences over the alphabet. With this interpretation the co-emission probability, $A(M_1, M_2)$, of two Markov models, M_1 and M_2 , simply becomes the inner product,

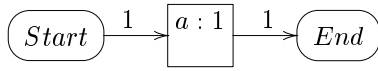
$$\langle M_1, M_2 \rangle = |M_1| |M_2| \cos v, \quad (13)$$

of the models. Here v is the angle between the models – or vectors – and $|M_i| = \sqrt{\langle M_i, M_i \rangle}$ is the length of M_i . One observes the direct proportionality between the co-emission probability and the length (or ‘self-confidence’) of the models being compared. If the length is to be completely ignored, a good measure of the distance between two Markov models would be the angle between them – two models are orthogonal, if and only if they can not generate identical sequences, and parallel (actually identical as the probabilities have to sum to 1) if they express the same probability distribution. This leads to the definition of our first metric on Markov models.

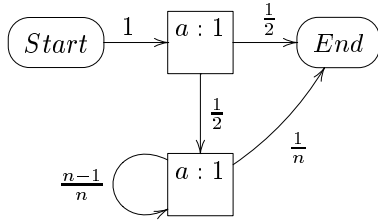
Definition 1 *Let M_1 and M_2 be two hidden Markov models, and $A(M, M')$ be the co-emission probability of M and M' . We define the angle between M_1 and M_2 as*

$$D_{\text{angle}}(M_1, M_2) = \arccos \left(A(M_1, M_2) / \sqrt{A(M_1, M_1) A(M_2, M_2)} \right).$$

Having introduced the vector interpretation of Markov models, another obvious metric to consider is



(a) Markov model M_1 with $P_{M_1}(a) = 1$.



(b) Markov model M_2 with $P_{M_2}(a) = 1/2$ and $P_{M_2}(a^k) = \frac{1}{2^n} \left(\frac{n-1}{n}\right)^{k-2}$ for $k > 1$.

Figure 2: Two distinctly different models can have an arbitrarily small distance in the D_{angle} metric. It is easy to see that $A(M_1, M_1) = 1$, $A(M_1, M_2) = 1/2$ and $A(M_2, M_2) = 1/4 + 1/(8n - 4)$; for $n \rightarrow \infty$ one thus obtains $D_{\text{angle}}(M_1, M_2) \rightarrow 0$ but $D_{\text{diff}}(M_1, M_2) \rightarrow 1/2$.

the standard metric on vector spaces, that is, the (euclidian) norm of the difference between the two vectors

$$|M_1 - M_2| = \sqrt{\langle M_1 - M_2, M_1 - M_2 \rangle}. \quad (14)$$

Considering the square of this, we obtain

$$\begin{aligned} |M_1 - M_2|^2 &= \langle M_1 - M_2, M_1 - M_2 \rangle \\ &= \sum_{s \in \Sigma^*} (P_{M_1}(s) - P_{M_2}(s))^2 \\ &= \sum_{s \in \Sigma^*} P_{M_1}(s)^2 + P_{M_2}(s)^2 - 2P_{M_1}(s)P_{M_2}(s) \\ &= A(M_1, M_1) + A(M_2, M_2) - 2A(M_1, M_2). \end{aligned} \quad (15)$$

Thus this norm can be computed based on co-emission probabilities, and we propose it as a second choice for a metric on Markov models.

Definition 2 Let M_1 and M_2 be two hidden Markov models, and $A(M, M')$ be the co-emission probability of M and M' . We define the difference between M_1 and M_2 as

$$D_{\text{diff}}(M_1, M_2) = \sqrt{A(M_1, M_1) + A(M_2, M_2) - 2A(M_1, M_2)}. \quad (16)$$

One problem with the D_{diff} metric is that $||M_1| - |M_2|| \leq D_{\text{diff}}(M_1, M_2) \leq |M_1| + |M_2|$. If $|M_1| \gg |M_2|$ we therefore get that $D_{\text{diff}}(M_1, M_2) \approx |M_1|$, and we basically only get information about the length of M_1 from D_{diff} .

The metric D_{angle} is not prone to this weakness, as it ignores the length of the vectors and focuses on the

sets of most probable sequences in the two models and their relative probabilities. But this metric can also lead to undesirable situations, as can be seen from figure 2 which shows that D_{angle} might not be able to discern two clearly different models. Choosing what metric to use, depends on what kind of differences one wants to highlight.

For some applications one might want a similarity measure instead of a distance measure. Based on the above metrics or the co-emission probability one can define a variety of similarity measures. We decided to examine the following two similarity measures.

Definition 3 Let M_1 and M_2 be two Markov models and $A(M, M')$ be the co-emission probability of M and M' . We define the similarity between M_1 and M_2 as

$$\begin{aligned} S_1(M_1, M_2) &= \cos(D_{\text{angle}}(M_1, M_2)) \\ &= A(M_1, M_2) / \sqrt{A(M_1, M_1)A(M_2, M_2)} \end{aligned}$$

and

$$S_2(M_1, M_2) = 2A(M_1, M_2) / (A(M_1, M_1) + A(M_2, M_2)).$$

One can easily prove that these two similarity measures possess the following nice properties.

1. $0 \leq S_i(M_1, M_2) \leq 1$.
2. $S_i(M_1, M_2) = 1$ if and only if $\forall s \in \Sigma^* : P_{M_1}(s) = P_{M_2}(s)$.
3. $S_i(M_1, M_2) = 0$ if and only if $\forall s \in \Sigma^* : P_{M_i}(s) > 0 \Rightarrow P_{M_{3-i}}(s) = 0$, that is, there are no sequences that can be generated by both M_1 and M_2 .

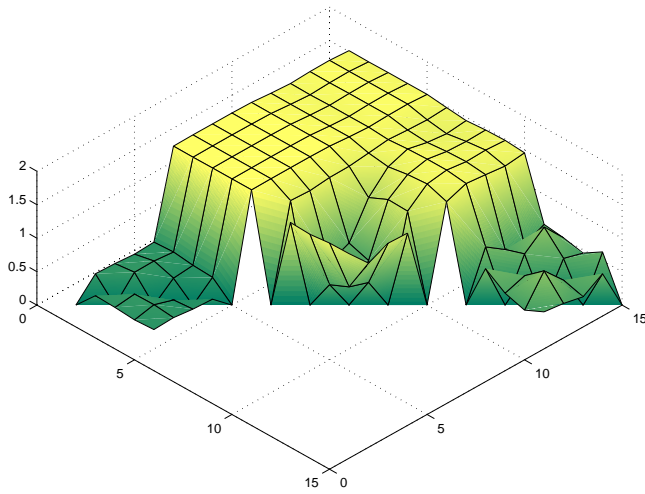
The only things that might not be immediately clear are that S_2 satisfies properties 1 and 2. This however follows from

$$\begin{aligned} A(M_1, M_1) + A(M_2, M_2) - 2A(M_1, M_2) &= \\ \sum_{s \in \Sigma^*} (P_{M_1}(s) - P_{M_2}(s))^2, \end{aligned} \quad (17)$$

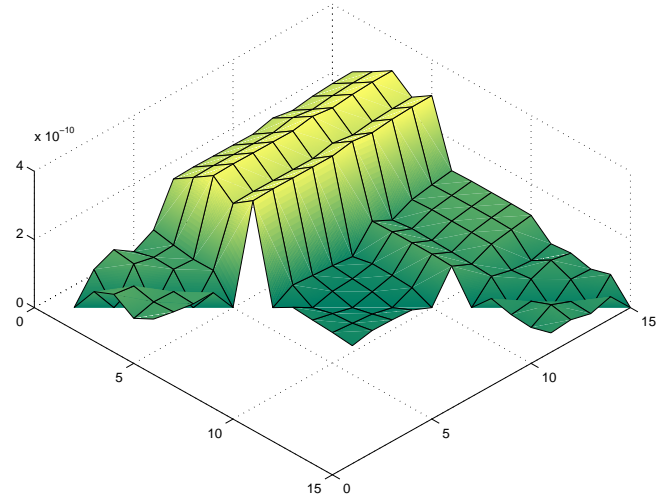
cf. equation 15, why $2A(M_1, M_2) \leq A(M_1, M_1) + A(M_2, M_2)$, and equality only holds if for all sequences their probabilities in the two models are equal.

Results

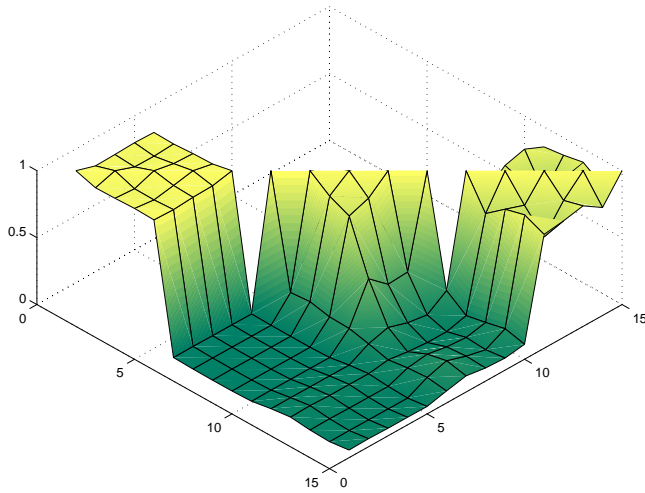
We have implemented the method described in the previous sections for computing the co-emission probabilities and derived measures of two left-right models, and the program is currently available at www.brics.dk/~cstorm/hmmcomp. The program was used to test the four measures in a comparison of Markov models for three classes of secretory signal peptides – cleavable N-terminal sequences which target secretory proteins for translocation over a membrane. Signal peptides do not have a well-defined consensus motif, but they do share a common structure: an N-terminal region with a positive charge, a stretch of



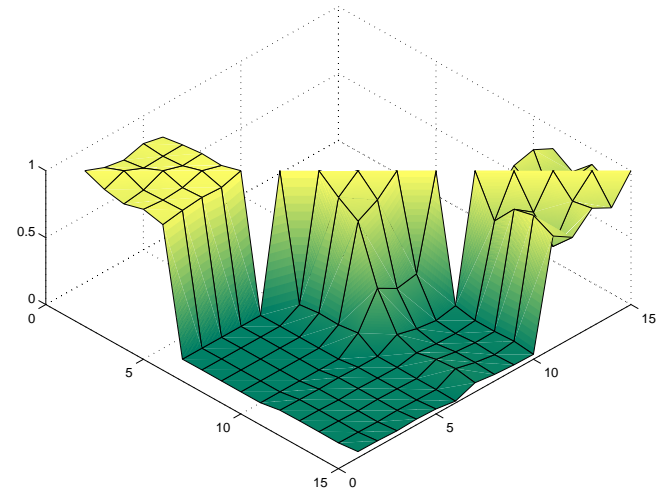
(a) Plot of D_{angle} values in radians



(b) Plot of D_{diff} values



(c) Plot of S_1 values



(d) Plot of S_2 values

Figure 3: Plots of the results obtained with the different measures. Models 1 through 5 are the models trained on eukaryotic sequences, models 6 through 10 are the models trained on Gram-positive bacterial sequences, and models 11 through 15 are the models trained on Gram-negative bacterial sequences. This gives 9 blocks, each of 25 entries, of different pairs of groups of organisms compared, but as all the measures are symmetric we have left out half the blocks showing comparisons between different groups of organisms. This should increase clarity, as no parts of the plots are hidden behind peaks.

	Euk	G _{pos}	G _{neg}
Euk	0.231	1.56	1.52
G _{pos}		0.864	1.47
G _{neg}			0.461

(a) Table of D_{angle} values

	Euk	G _{pos}	G _{neg}
Euk	$6.77 \cdot 10^{-11}$	$2.56 \cdot 10^{-10}$	$2.67 \cdot 10^{-10}$
G _{pos}		$1.95 \cdot 10^{-11}$	$9.09 \cdot 10^{-11}$
G _{neg}			$4.43 \cdot 10^{-11}$

(b) Table of D_{diff} values

	Euk	G _{pos}	G _{neg}
Euk	0.967		
G _{pos}	$1.06 \cdot 10^{-2}$	0.547	
G _{neg}	$4.74 \cdot 10^{-2}$	0.102	0.866

(c) Table of S_1 values

	Euk	G _{pos}	G _{neg}
Euk	0.955		
G _{pos}	$1.78 \cdot 10^{-3}$	0.511	
G _{neg}	$2.93 \cdot 10^{-2}$	$4.78 \cdot 10^{-2}$	0.839

(d) Table of S_2 values

Figure 4: Tables of the average values of each block plotted in figure 3. The empty entries corresponds to the blocks left out in the plots.

hydrophobic residues, and a region of more polar regions containing the cleavage site, where two positions are partially conserved (von Heijne 1985). There are statistical differences between prokaryotic and eukaryotic signal peptides concerning the length and composition of these regions (von Heijne & Abrahmsén 1989; Nielsen *et al.* 1997), but the distributions overlap, and in some cases, eukaryotic and prokaryotic signal peptides are found to be functionally interchangeable (Benson, Hall, & Silhavy 1985).

The Markov model used here is not a profile HMM, since signal peptides of different proteins are not necessarily related, and therefore do not constitute a sequence family that can be aligned in a meaningful way. Instead, the signal peptide model is composed of three region models, each having a characteristic amino acid composition and length distribution, plus seven states modelling the cleavage site – see Nielsen and Krogh (Nielsen & Krogh 1998) for a detailed description. A combined model with three branches was used to distinguish between signal peptides, signal anchors (a subset of transmembrane proteins), and non-secretory proteins; but only the part modelling the signal peptide plus the first few positions after the cleavage site has been used in the comparisons reported here.

The same architecture was used to train models of three different signal peptide data sets: eukaryotes, Gram-negative bacteria (with a double membrane), and Gram-positive bacteria (with a single membrane). For cross-validation of the predictive performance, each model was trained on five different training/test set partitions, with each training set comprising 80% of the data – i.e., any two training sets have 75% of the sequences in common.

The comparisons of the models are shown in figures 3 and 4. In general, models trained on cross-validation sets of the same group are more similar than models trained on data from different groups, and the two

groups of bacteria are more similar to one another than to the eukaryotes. However, there are some remarkable differences between the measures. According to D_{diff} , the two bacterial groups are almost as similar as the cross-validation sets, but according to D_{angle} and the similarity measures, they are almost as dissimilar as the bacterial/eukaryotic comparisons.

This difference actually reflects the problem with the D_{diff} measure discussed previously. The distribution of sequences for models trained on eukaryotic data are longer in the vector interpretation, i.e. the probabilities are more concentrated, than the distributions for models trained on bacterial data. What we mainly see in the D_{diff} values for bacterial/eukaryotic comparisons is thus the length of the eukaryotic models. This reflects two properties of eukaryotic signal peptides: they have a more biased amino acid composition in the hydrophobic region that comprises a large part of the signal peptide sequence; and they are actually *shorter* than their bacterial counterparts, thus raising the probability of the most probable sequences generated by this model.

D_{angle} also shows that the differences within groups are larger in the Gram-positive group than in the others. This may simply reflect the smaller sample size in this group (172 sequences vs. 356 for the Gram-negative bacteria and 1137 for the eukaryotes).

The values of D_{angle} in between-group comparisons are quite close to the maximal $\pi/2$. Thus the distributions over sequences for models of different groups are close to being orthogonal. This might seem surprising in the light of the reported examples of functionally interchangeable signal peptides; but it does not mean that no sequences can be generated by both eukaryotic and bacterial models, only that these sequences have low probabilities compared to those that are unique for one group. In other words: if a random sequence is generated from one of these models, it may with a high probability be identified which group of organisms it belongs to.

Discussion

Remember that the co-emission probability is defined as $\sum_{s_1, s_2 \in \Sigma^*} P_{M_1}(s_1)P_{M_2}(s_2)$ where $s_1 = s_2$. One problem with the co-emission probability – and measures based on it – is that it can be desirable to allow sequences to be slightly different. One might thus want to loosen the restriction of “ $s_1 = s_2$ ” to, e.g., “ s_1 is a substring (or subsequence) of s_2 ,” or even “ $|s_1| = |s_2|$ ” ignoring the symbols of the sequences and just comparing the length distributions of the two models.

Another approach is to take the view that the two Markov models do not generate independent sequences, but instead generates alignments with two sequences. Inspecting the equations for computing the co-emission probability, one observes that we require that when one model emits a symbol the other model should emit an identical symbol. This corresponds to only allowing columns with identical symbols in the produced alignments. A less restrictive approach would be to allow other types of columns, i.e. columns with two different symbols or a symbol in only one of the sequences, and weighting a column according to the difference it expresses. The modifications proposed in the previous paragraph can actually be considered special cases of this approach. Our method for computing the co-emission probability can easily be modified to encompass these types of modifications.

Acknowledgements This work was inspired by a talk by Xiaobing Shi on his work with David States on aligning profile hidden Markov models. The authors would like to thank Bjarne Knudsen, Jotun Hein and the reviewers for their valuable suggestions. Finally we would like to thank Anders Krogh for his help with the software used to train and parse the models. Christian N. S. Pedersen and Henrik Nielsen are supported by the Danish National Research Foundation.

References

- Baldi, P.; Chauvin, Y.; Hunkapiller, T.; and McClure, M. A. 1994. Hidden markov models of biological primary sequence information. In *Proceedings of the National Academy of Science, USA*, volume 91, 1059–1063.
- Benson, S. A.; Hall, M. N.; and Silhavy, T. J. 1985. Genetic analysis of protein export in *Escherichia coli* K12. *Annual Review of Biochemistry* 54:101–134.
- Churchill, G. A. 1989. Stochastic models for heterogeneous DNA sequences. *Bulletin of Mathematical Biology* 51:79–94.
- Durbin, R.; Eddy, S. R.; Krogh, A.; and Mitchison, G. 1998. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press.
- Eddy, S. R. 1996. Hidden markov models. *Current Opinion in Structural Biology* 6:361–365.
- Eddy, S. R. 1998. Profile hidden markov models. *Bioinformatics* 14:755–763.

Gotoh, O. 1993. Optimal alignment between groups of sequences and its application to multiple sequence alignment. *Computer Applications in the Biociences* 9(3):361–370.

Gribskov, M.; McLachlan, A. D.; and Eisenberg, D. 1987. Profile analysis: Detection of distantly related proteins. In *Proceedings of the National Academy of Science, USA*, volume 84, 4355–4358.

Gusfield, D. 1997. *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge University Press.

Jelinek, F. 1976. Continuous speech recognition by statistical methods. In *Proceedings of the IEEE*, volume 64, 532–536.

Krogh, A.; Brown, M.; Mian, I. S.; Sjolander, K.; and Haussler, D. 1994. Hidden markov models in computational biology: Applications to protein modeling. *Journal of Molecular Biology* 235:1501–1531.

Krogh, A. 1997. Two methods for improving performance of an HMM and their application for gene finding. In *Proceedings of the 5th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, 179–186.

Luthy, R.; Bowie, J. U.; and Eisenberg, D. 1992. Assessment of protein models with three-dimensional profiles. *Nature* 356:83–85.

Lyngsø, R. B.; Pedersen, C. N. S.; and Nielsen, H. 1999. Measures on hidden Markov models. Technical Report RS-99-6, BRICS.

Nielsen, H., and Krogh, A. 1998. Prediction of signal peptides and signal anchors by a hidden Markov model. In *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, 122–130.

Nielsen, H.; Brunak, S.; Engelbrecht, J.; and von Heijne, G. 1997. Identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Protein Engineering* 10:1–6.

Rabiner, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, 277–286.

Sonnhammer, E. L. L.; von Heijne, G.; and Krogh, A. 1998. A hidden Markov model for predicting transmembrane helices in protein sequences. In *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, 175–182.

Taylor, W. R. 1986. Identification of protein sequence homology by consensus template alignment. *Journal of Molecular Biology* 188:233–258.

von Heijne, G., and Abrahmsén, L. 1989. Species-specific variation in signal peptide design. *FEBS Letter* 244:439–446.

von Heijne, G. 1985. Signal sequences. The limits of variation. *Journal of Molecular Biology* 184:99–105.