

Title of publication: Encyclopedia of Life Sciences

Article title: Hidden Markov Models (ID: A22825)

Author details:

Teresa M. Przytycka

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health

Bethesda, MD 20894, U.S.A.

Jie Zheng

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health

Bethesda, MD 20894, U.S.A.

Hidden Markov Models

Advanced Article

Abstract

A hidden Markov model (HMM) is a statistical approach that is frequently used for modelling biological sequences. In applying it, a sequence is modelled as an output of a discrete stochastic process, which progresses through a series of states that are 'hidden' from the observer. Each such hidden state emits a symbol representing an elementary unit of the modelled data, for example, in case of a protein sequence – an amino-acid. The parameters of a hidden Markov model can be estimated by learning from training data. Efficient algorithms are available to infer the most likely paths of states for given sequence data, which often lead to biological predictions and interpretations. Thanks to the well developed theories and algorithms, hidden Markov models have found wide applications in diverse areas of computational molecular biology.

Keywords: HMM; gene finding; profile HMM; training HMM; protein structure prediction; epigenetics; phylogenetics

Key Concepts

- Hidden Markov model is a statistical approach for modelling sequences with broad applications in computational biology.
- In an HMM, a biological sequence is modelled as being generated by a stochastic process moving from one state to the next state, where each state emits one element of the sequence according to some emission probability distribution which, in general, is different in different states.
- Training of an HMM is a process in which the parameters of the model are computed based on a training set of representative examples
- Overfitting/Overtraining the model occurs when model parameters correctly represent the training set but the model cannot generalize the training data to a larger set.
- Gene finding is a process of computational identification of genes, including exon/intron structure, in a genome.

Introduction

A hidden Markov model (HMM) is a statistical model, initially developed for speech recognition (Rabiner 1989), which has subsequently been used in numerous biological

sequence analysis applications. Classic applications of HMMs in computational biology include, among others, modelling of protein families (Krogh et al. 1994a, Bateman et al. 2002), gene finding (Krogh, Mian and Haussler 1994b, Burge and Karlin 1997, Henderson, Salzberg and Fasman 1997, Lukashin and Borodovsky 1998, Salzberg et al. 1998), predicting transmembrane helices (Krogh et al. 2001) and tertiary structure prediction (Di Francesco, Garnier and Munson 1997a, Bystroff, Thorsson and Baker 2000). More recent applications include modelling of epigenetic signals, copy number variations and molecular evolution, which are described in the last section.

In biological applications, HMMs are most often used to model sequence data such as protein, DNA or RNA sequences, and chromatin structure. The natural ordering of elements of sequences allows modelling them as an output of a stochastic process progressing through discrete time steps where at each time step, the process generates (emits) a symbol (an amino acid, a nucleotide, epigenetic state etc.). Note that properties of a sequence might be different at different positions. For example, in a family of homologous proteins some positions might be conserved, other might be fully random, yet other might be biased towards a particular group of amino-acids. To account for such variability, an HMM uses a finite number of states, where each state defines a specific emission distribution. The transition between states is described as a Markov process. Thus the two main components of an HMM are the emission probability (defined for each state) and the transition probability describing movements between states. HMMs are typically built based on a training set of examples as described in the section Construction and Training of an HMM.

Conceptually, HMMs are used in two somewhat different settings each requiring specific algorithmic tools. First, given an HMM model that describes a sequence family, say a DNA binding domain, one can use it to test if a given query sequence is a member of such family. That is, given an HMM M and a sequence S , the question is whether S has the property modelled by M . Other example of this type, discussed later in the Applications section, is protein fold recognition. To answer such question one needs to compute the probability $P[S|M]$ of sequence S being generated by M . The log of the ratio of $P[S|M]$ to the probability of generating S by chance is usually used as a scoring function in assessing whether S has the model property.

The second fundamental application of HMMs to annotate a biological sequence with features such as CpG islands, epigenetic structure, intron/exon structure etc. (see Applications section). In such cases, an HMM is typically designed so that the states correspond to the modelled features. To annotate the sequence with the features included in the model, one needs to establish which states are most likely to generate which position of the sequence.

The subsequent text provides a formal definition of an HMM, describes fundamental algorithms used to answer above mentioned questions, methods for designing HMMs, and finally surveys most recent applications of HMMs in molecular biology.

Definitions

A first-order HMM is defined formally as a 5-tuple $M=(Q, \Sigma, a, s, e)$, where $Q=\{1, \dots, n\}$ is a finite set of states; $\Sigma=\{\sigma_1, \dots, \sigma_m\}$ is the alphabet, that is, the set of output symbols; a is an $n \times n$ matrix of transition probabilities defined formally as

$a(i,j)=P[q_{t+1}=j|q_t=i]$, where q_t is the state visited at step t ; s is an n vector of start probabilities, that is, $s(i)=P[q_0=i]$; e is an $n \times m$ matrix of emission probabilities defined formally as $e(i,j)=P[o_t=\sigma_j|q_t=i]$, where $o_t \in \Sigma$ is the symbol outputted in step t .

It is often convenient to have distinguished 'start' and 'end' states (0 and $n+1$) that do not emit any symbols and remove vector s from the model definition. In the considerations below, we make this assumption. An HMM is usually visualized as a directed graph with vertices corresponding to the states and directed edges to the pairs of states i,j for which transition probability $a(i,j)$ is nonzero. A simple HMM is shown in Figure 1.

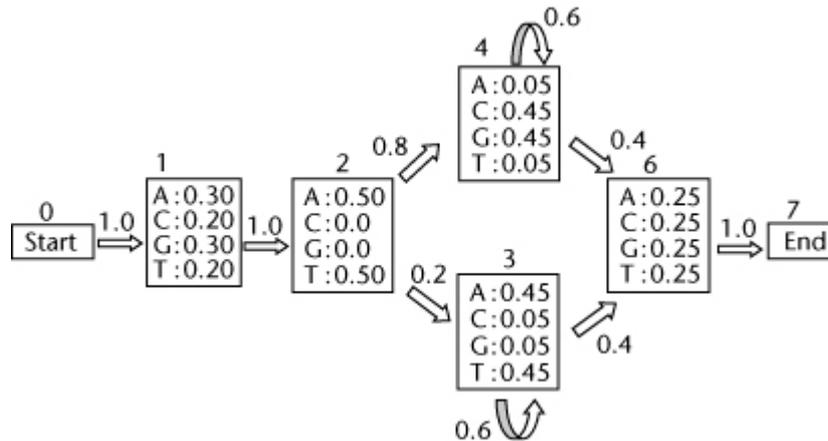


Figure 1. A simple hidden Markov model. The boxes correspond to states where the emission probabilities for each state are given inside each box. The transition probabilities are given above the corresponding arrows. Note that there are two state paths that can be used to generate the sequence GAGCGCT: 0,1,2,4,4,4,4,6,7 and 0,1,2,3,3,3,3,6,7. The probability of generating the sequence using the first path is 1.06×10^{-4} and using the second path is 1.35×10^{-7} . The probability of generating the sequences by the model is the sum of these probabilities.

In a k th order model, the transition and emission probabilities depend on k previous steps. Consequently, matrix a is of size n^{k+1} and matrix e is of size $n^k m$.

An HMM may generate the same sequence following different state paths (see Figure 1). Given an HMM M , sequence $S=o_1, \dots, o_T$ and a path of states $p=q_0 q_1 \dots q_m q_{T+1}$, where $q_0=0$ and $q_{T+1}=n+1$, the probability of generating S using path p in model M , $P[S, p|M]$, is equal to the following product:

$P[S, p|M] = P[p|M]P[S|p, M]$ where $P[p|M]$ is the probability of selecting the path p and $P[S|p, M]$ the probability of generating sequence S assuming path p .

$$P[p|M] = a(q_0, q_1)a(q_1, q_2)a(q_2, q_3)\dots a(q_T, q_{T+1})$$

$$P[S|p, M] = e(q_1, o_1)e(q_2, o_2)\dots e(q_T, o_T)$$

The most likely path of a sequence S in model M is the path p_{\max} that maximizes $P[S, p|M]$. Thus although the states of an HMM generating given sequence data are not directly observable, the most likely path provides information about the likely sequence of such 'hidden' states.

Finally, the probability $P[S|M]$ of generating sequence S by an HMM M is defined as

$$P[S|M] = \sum_p P[S|M, p]$$

In practical applications, probability values P are replaced with $-\log P$ scores. This avoids producing numbers that are too small to be represented by a computer.

Basic Algorithms

Given an HMM M and a sequence $S = o_1, \dots, o_T$ of length T , and assuming that at step $T+1$ the process is in the stop state ($n+1$) generating empty symbol, the values p_{\max} and $P[S|M]$ can be computed using a dynamic programming method.

Let $v_k(i)$ be the most probable path that generates o_1, \dots, o_i and ends in state k at step i . Obviously,

$$p_{\max} = v_{n+1}(T+1)$$

The recurrence for computing $v_k(i)$ is given by the following formula:

$$v_k(i) = e(k, o_i) \max_j x_j v_j(i-1) a(j, k)$$

With appropriate initial conditions, the above recurrence provides the basis for an $O(n^2 T)$ -time dynamic programming algorithm known as the *Viterbi algorithm*. Since the number of states n is fixed for the model, the running time of the algorithm depends linearly on the length of the input sequence.

Replacing maximum with summation in the recursive formula for $v_k(i)$ yields the recurrence for $P[S|M]$. Namely, let $f_k(i)$ denote the probability of generating subsequence o_1, \dots, o_i using a path that ends in state k at step i . Then,

$$f_k(i) = e(k, o_i) \sum_j f_j(i-1) a(j, k)$$

and

$$P[S|M] = f_{n+1}(T+1)$$

Variable $f_k(i)$, called the *forward variable*, is also used for computing the probability of state k at step i , $P[q_i=k|S, M]$. To compute the last probability, a similar *backward variable* $b_k(i)$ is also used. Formally, $b_k(i)$ is the probability of generating the subsequence o_{i+1}, \dots, o_T using state k as the starting state and the usual 'end' state $n+1$. The backward variable is computed similar to the forward variable, but the algorithm is executed in the 'backward' direction: using the 'end' state in the place of the 'begin' state. By definitions of $f_k(i)$ and $b_k(i)$ it follows that

$$P[q_i = k | S, M] = (f_k(i) b_k(i)) / P[S|M]$$

Construction and Training of an HMM

There are two basic steps in building an HMM: designing the directed graph that describes the topology of the model (number of states, connections between states); and assigning transition and emission probabilities.

The topology of an HMM is usually designed in an ad hoc way, based on the designer's understanding of the modelled sequence. Frequently, such a sequence can be described by a 'grammar'. For example, a simple grammar for a prokaryotic gene can be given as $S(C)^n.E$, where S is the start codon, C is any codon different from an end codon, E is an end codon and C is repeated n times. In this case, it is natural to design the topology of an HMM in a way that follows the grammatical description. In the prokaryotic gene example, a topology implied by the simple grammar is shown in Figure 2. The grammar, and subsequently a corresponding HMM, for the eukaryotic gene is far more complicated. It needs to describe a gene sequence as an interleaving sequence of exons and introns taking into account that the splicing can occur at any codon position.

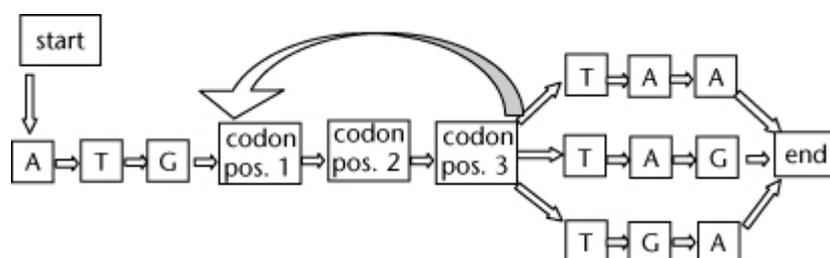


Figure 2. Topology of a simple HMM for prokaryotic gene recognition. In practice, the topology is more complex (e.g. (Krogh et al. 1994b, Henderson et al. 1997)).

A different approach is taken in designing the so-called profile HMMs for protein families (Krogh, Brown et al. 1994). Namely, a universal topology is used and a correct setting of parameters elucidates the variations between families. The design includes 'match' states, 'insert' states and silent 'delete' states (Figure 3).

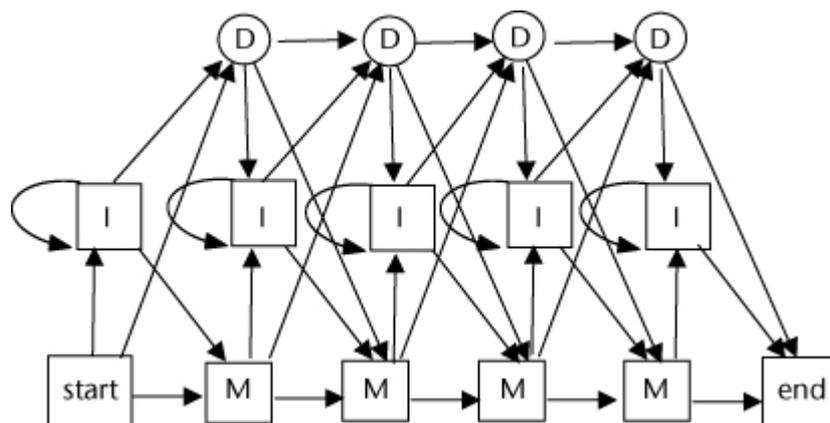


Figure 3. Topology of a profile HMM for a sequence family. The states labeled with M correspond to matches, the states labeled with I correspond to insertions and (silent) circle states correspond to deletions. (Adopted with permission from (Durbin et al. 1998))

In the second phase of the construction, the transition and emission probabilities are assigned to the model. This is done automatically, based on a representative sample of sequences called the training set. The computational problem is described formally as follows.

Given a training set S_1, \dots, S_n and a topology of HMM M , find emission and transition probabilities that maximize the likelihood that S_1, \dots, S_n are generated by the model.

The usual assumption is that S_1, \dots, S_n are generated independently and therefore

$$P(S_1, \dots, S_n | M) = \prod_i P(S_i | M)$$

And replacing the probability with $-\log$ score we have

$$\text{Score}(S_1, \dots, S_n | M) = \sum_i \text{Score}(S_i | M)$$

The training step is straightforward if for each training sequence S_i , the paths of states, which the model uses to generate S_i , are known. In this case, the training step reduces to collecting transition and emission frequencies along these paths. The training step becomes more sophisticated if the state paths are unknown. The main strategy in this case is to start with some initial probability distribution and then iteratively improve the model using the training set. For example, one frequently used method, the Expectation Maximization method, approaches this problem as follows:

1. Assign some initial values to parameters (say uniform probability distribution).
2. For each sequence in the training set, compute the expected number of times each transition/emission is used. This can be done efficiently using the algorithms described in the previous section.
3. Estimate new values of the parameters of the model based on the expected values from step 2.

Repeat steps 2 and 3 until some convergence criterion is reached. It can be shown that Expectation Maximization method converges to a local maximum. Other training methods include the gradient descent method and simulated annealing.

One of the fundamental questions that one needs to consider during the training process is whether the training set contains enough data to estimate correctly the transition and emission probabilities. Lack of data leads to overfitting of the model – the model cannot generalize the training data to a larger set. In particular, the question of sufficient data needs to be examined when deciding on the order of the model. In principle, a higher order model should be more accurate. For example, gene recognition models often are of fifth order. (This is the equivalent of keeping memory of two codons.) The number of parameters that need to be estimated grows exponentially with the order of the model and the possibility of overfitting increases.

Applications

In the previous sections, we illustrated the concept of HMM using two prominent biological applications – modelling of sequence families and gene finding. However applications of this modelling technique extend to many other areas of bioinformatics

that are briefly surveyed in this section. In particular, the Illumina's sequencing technology combined with Chromatin immunoprecipitation technology has led to burgeoning of experimental methods for genome-wide detection of diverse DNA properties, often naturally modelled with HMM which we outline below. We also discuss applications related to uncovering DNA copy number variations, evolutionary biology and protein structure.

HMM- based modelling of chromatin structure and properties of DNA *in vivo*

The chromatin immunoprecipitation ("*ChIP*") followed by microarray technology (Chip-chip) and the more recent Chromatin immunoprecipitation followed by high-throughput sequencing (ChIP-seq) provide experimental methods for performing genome scale surveys of chromosome and chromatin properties. The applications of these technologies are growing but a general strategy starts with a protein, P, that recognizes (or is otherwise associated with) a genomic feature of interest, for example a transcription factor recognizing DNA binding site. This protein is cross-linked with the DNA site it binds to, the cells are lysed, the DNA is sheared, and the genomic positions of DNA fragments bound to P identified. While specific procedures are different between the two technologies and additionally vary between specific applications, the end result of these and related experiments is a mapping of DNA fragments associated with the specific signal onto the genome. HMM are increasingly used to interpret such mappings.

For example, an important application of ChIP-chip and ChIP-seq technologies is identification of transcription factor binding sites. Such data can be modelled with simple 2-state model where the two states correspond to "signal enrichment" and "background" (Figure 4a) (Li, Meyer and Liu 2005, Humburg, Bulger and Stone 2008, Qin et al. 2010). In the genome wide analysis of PRC1 and PRC2 occupancy, Ku et al. (Ku et al. 2008) used four states (masked, low density, medium density, and high density). It is important to mention that, unlike profile HMM discussed in the previous section, where at each state the HMM is emitting a symbol from a finite alphabet, in ChIP-seq experiments the read patterns are considered to be a continuous observation and thus the emission probabilities are defined by a probability density function.

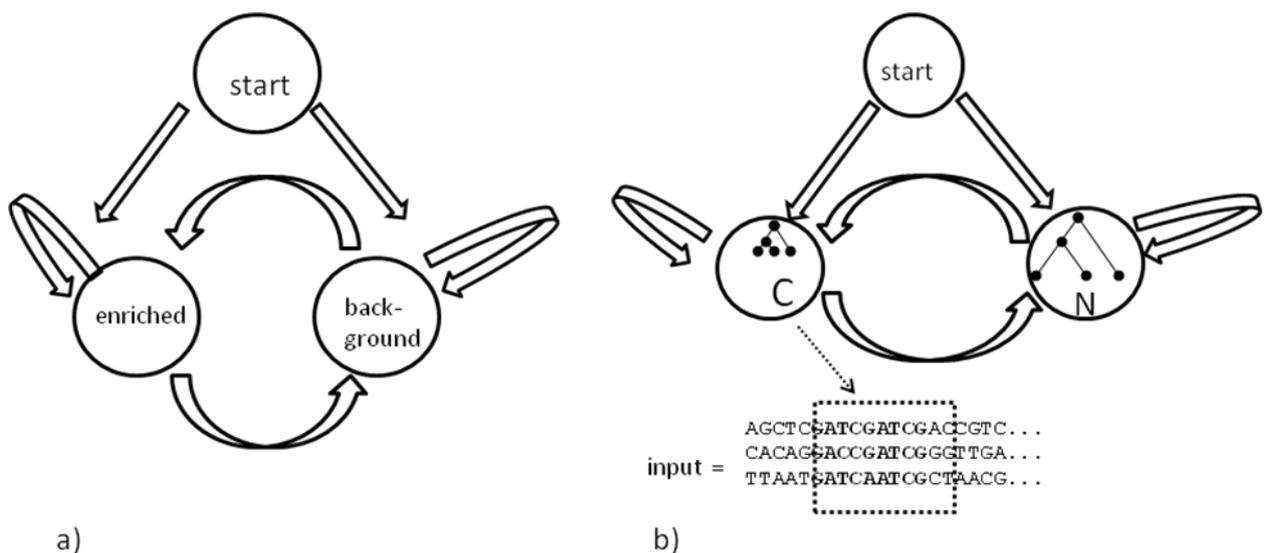


Figure 4. Two related HMMs modelling two different biological processes. a) Topology of an HMM for recognising of regions with epigenetic makers. The simple HMM model consist with the states: enriched region and background regions. B) Topology of a phylogenetic HMM (Siepel et al. 2005) for the prediction of conserved genomic elements. The states labeled with C and N corresponding to conserved and nonconserved regions respectively. The block of input alignment in the box illustrates a conserved region and the corresponding alignment columns are assumed to be emitted by state C.

HMMs are also used to uncover regions of characteristic chromatin structure. Namely, ChIP-seq technology is now routinely applied to survey DNA regions with particular modifications of histones. Designing an HMM so that each state emits signals of several types of histone modifications, Won et al. predicted genomic promoters and enhancers (Won et al. 2008) and, including PSSM binding pattern, transcription factor binding sites (Won, Ren and Wang 2010). In their model, they used HMM with left-to-right topology (Rabiner 1989) allowing them to limit possible transition to ensure specific order of states and to capture more complex signal patterns. Since chromatin structure is dynamic and depends on tissue and conditions, HMM-based methods have been also applied to genome-wide identification of such differences (Xu et al. 2008).

In the context of modelling of chromatin structure, HMM have also some shortcomings. Note that the length distribution of “enriched” and “background” intervals from model in Figure 4 is geometric, but it is not necessarily the case in real data. Furthermore some observations might be missing. Therefore Lian et al. in their approach to modeling chromatin structure used a generalization of HMM allowing for including length distribution in the model (Lian et al. 2008). Chen et al. (Chen et al. 2010) accounted for missing information by generalizing HMM to a Bayesian network model.

Detection of Copy Number Variation

Copy number variations (CNV) are duplication or deletion of a DNA segment compared to a reference genome, and have been found to be common in human genome. Such variations might be responsible for a significant proportion of phenotypic variations (Freeman et al. 2006). HMMs have been used in detecting CNV from single nucleotide polymorphism (SNP) genotyping data (Colella et al. 2007, Wang et al. 2007). In this case, the input data are two measures of genotype signal at each SNP: the log R Ratio (normalized total signal intensity) and the B Allele Frequency (normalized allelic intensity ratio). The hidden states are the unknown copy number at each SNP. The design of HMMs takes into account the state of homozygosity, distance between consecutive SNPs, etc. The most likely state path computed by the Viterbi algorithm implies the predicted regions of copy number gain or loss. Likewise, an HMM has been developed to detect CNVs from short read sequence data (Simpson et al. 2010).

Molecular Evolution

In the aforementioned applications of HMM, the input data are mostly single sequences, on which HMMs are used to predict the probabilistic distribution in space (along the

sequences). When applying HMMs to molecular evolution, we need to consider the dimension of time. A combination of HMMs and phylogenetic models, phylogenetic hidden Markov models (phylo-HMMs) were originally proposed to improve phylogenetic inference using HMMs to capture the variation of substitution rates among sites (Yang 1995, Felsenstein and Churchill 1996, Siepel and Haussler 2005). In a phylo-HMM, the input is a multiple sequence alignment; each state corresponds to a phylogenetic model, and at each time step it emits a new column in the input alignment, with probability determined by the associated phylogeny.

The phylo-HMMs have been used to identify conserved elements from multiple alignments of vertebrate genomes (Siepel et al. 2005) as shown in Figure 4b. The model consists of two states, for conserved and nonconserved regions, each associated with a phylogeny. The two phylogenies have identical topology but the conserved tree has shorter branches, which models the smaller substitution rate in conserved regions than the average rate in nonconserved regions. Other applications of phylo-HMM include comparative gene prediction (Pedersen and Hein 2003), detection of selection (Siepel, Pollard and Haussler 2006) and recombination (Husmeier and Wright 2001).

As a natural extension of phylo-HMM, a class of HMMs called population genetic hidden Markov models (popGenHMMs) has been developed (Kern and Haussler 2010). Input of popGenHMMs are sequence polymorphisms (say SNPs), and each hidden state corresponds to a population genetic model. The object emitted at a time step is the allele frequency at a SNP. Several popGenHMMs have been developed to detect genomic regions under selection (Boitard, Schlotterer and Futschik 2009, Kern and Haussler 2010).

Protein Structure Analysis

Following successful applications of HMMs in sequence analysis, this modelling technique has been also applied to recognizing and predicting protein 3D structures and/or motifs. One of the first applications of HMMs in the field of protein was to model transmembrane helices (Sonnhammer, von Heijne and Krogh 1998). This effort was quickly followed by HMM models to predict topology of transmembrane helical proteins (Zhou and Zhou 2003) and transmembrane β -barrels (Martelli et al. 2002, Bagos et al. 2004). In such models dedicated to predicting specific 3D structure, a state of the model often corresponds to a position of amino-acid in the structure.

HMM are also used for general structure prediction. For example, following previous HMM-based approaches incorporating prediction of secondary structure information (Di Francesco et al. 1997b, Karchin et al. 2003, Hargbo and Elofsson 1999), Karchin *et al.* proposed an HMM that emits a pair of symbols: one is an amino acid and the other a secondary structure assignment (Karchin et al. 2003). This approach and its subsequent refinements provided a series of successful fold prediction programs (e.g. (Karplus 2009)). HMMs have been also used to model structural biases of amino acids (Razzaki and Bukhari 1975, Li et al. 2008). A different idea has been explored by proteins structure prediction program HMMSTR (Bystroff et al. 2000). In the heart of this approach was designing a hidden Markov model by merging HMM models trained on structural motifs (I-sites), where each state contained information about the sequence and structure attributes of an individual position in the motif.

Given non-linear nature of protein 3D structure, topologies of HMMs designed to model protein structure tend to be more complicated than that of HMMs modelling sequence features. For example, HMMSTR has a highly branched topology.

Summary

Hidden Markov Models have proven to be widely applicable to modelling of diverse biological data. They are relatively simple, supported by well developed theory and algorithms and frequently lead to very intuitive and informative models.

Acknowledgments

This work was supported by the Intramural Research Program of the National Institutes of Health, National Library of Medicine.

References

- Bagos, P. G., T. D. Liakopoulos, I. C. Spyropoulos & S. J. Hamodrakas (2004) A Hidden Markov Model method, capable of predicting and discriminating beta-barrel outer membrane proteins. *BMC Bioinformatics*, 5, 29.
- Bateman, A., E. Birney, L. Cerruti, R. Durbin, L. Ewinger, S. R. Eddy, S. Griffiths-Jones, K. L. Howe, M. Marshall & E. L. Sonnhammer (2002) The Pfam protein families database. *Nucleic Acids Res*, 30, 276-80.
- Boitard, S., C. Schlotterer & A. Futschik (2009) Detecting Selective Sweeps: A New Approach Based on Hidden Markov Models. *Genetics*, 181, 1567-1578.
- Burge, C. & S. Karlin (1997) Prediction of complete gene structures in human genomic DNA. *J Mol Biol*, 268, 78-94.
- Bystroff, C., V. Thorsson & D. Baker (2000) HMMSTR: a hidden Markov model for local sequence-structure correlations in proteins. *J Mol Biol*, 301, 173-90.
- Chen, X., M. M. Hoffman, J. A. Bilmes, J. R. Hesselberth & W. S. Noble (2010) A dynamic Bayesian network for identifying protein-binding footprints from single molecule-based sequencing data. *Bioinformatics*, 26, i334-42.
- Colella, S., C. Yau, J. M. Taylor, G. Mirza, H. Butler, P. Clouston, A. S. Bassett, A. Seller, C. C. Holmes & J. Ragoussis (2007) QuantiSNP: an Objective Bayes Hidden-Markov Model to detect and accurately map copy number variation using SNP genotyping data. *Nucleic Acids Research*, 35, 2013-2025.
- Di Francesco, V., J. Garnier & P. J. Munson (1997a) Protein topology recognition from secondary structure sequences: application of the hidden Markov models to the alpha class proteins. *J Mol Biol*, 267, 446-63.
- Di Francesco, V., V. Geetha, J. Garnier & P. J. Munson (1997b) Fold recognition using predicted secondary structure sequences and hidden Markov models of protein folds. *Proteins*, Suppl 1, 123-8.
- Durbin, R., S. R. Eddy, A. Krogh & G. Mitchison. 1998. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge, UK: Cambridge University Press.
- Felsenstein, J. & G. A. Churchill (1996) A Hidden Markov Model approach to variation among sites in rate of evolution. *Mol Biol Evol*, 13, 93-104.
- Freeman, J. L., G. H. Perry, L. Feuk, R. Redon, S. A. McCarroll, D. M. Altshuler, H. Aburatani, K. W. Jones, C. Tyler-Smith, M. E. Hurles, N. P. Carter, S. W. Scherer & C. Lee (2006) Copy number variation: New insights in genome diversity. *Genome Research*, 16, 949-961.
- Hargbo, J. & A. Elofsson (1999) Hidden Markov models that use predicted secondary structures for fold recognition. *Proteins*, 36, 68-76.
- Henderson, J., S. Salzberg & K. H. Fasman (1997) Finding genes in DNA with a Hidden Markov Model. *Journal of Computational Biology*, 4, 127-141.
- Humburg, P., D. Bulger & G. Stone (2008) Parameter estimation for robust HMM analysis of ChIP-chip data. *BMC Bioinformatics*, 9, 343.
- Husmeier, D. & F. Wright (2001) Detection of recombination in DNA multiple alignments with hidden Markov models. *Journal of Computational Biology*, 8, 401-427.
- Karchin, R., M. Cline, Y. Mandel-Gutfreund & K. Karplus (2003) Hidden Markov models that use predicted local structure for fold recognition: alphabets of backbone geometry. *Proteins*, 51, 504-14.
- Karplus, K. (2009) SAM-T08, HMM-based protein structure prediction. *Nucleic Acids Res*, 37, W492-7.
- Kern, A. D. & D. Haussler (2010) A Population Genetic Hidden Markov Model for Detecting Genomic Regions Under Selection. *Mol Biol Evol*, 27, 1673-1685.
- Krogh, A., M. Brown, I. S. Mian, K. Sjolander & D. Haussler (1994a) Hidden Markov models in computational biology. Applications to protein modeling. *J Mol Biol*, 235, 1501-31.

- Krogh, A., B. Larsson, G. von Heijne & E. L. Sonnhammer (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol*, 305, 567-80.
- Krogh, A., I. S. Mian & D. Haussler (1994b) A hidden Markov model that finds genes in *E. coli* DNA. *Nucleic Acids Res*, 22, 4768-78.
- Ku, M., R. P. Koche, E. Rheinbay, E. M. Mendenhall, M. Endoh, T. S. Mikkelsen, A. Presser, C. Nusbaum, X. Xie, A. S. Chi, M. Adli, S. Kasif, L. M. Ptaszek, C. A. Cowan, E. S. Lander, H. Koseki & B. E. Bernstein (2008) Genomewide analysis of PRC1 and PRC2 occupancy identifies two classes of bivalent domains. *PLoS Genet*, 4, e1000242.
- Li, S. C., D. Bu, J. Xu & M. Li (2008) Fragment-HMM: a new approach to protein structure prediction. *Protein Sci*, 17, 1925-34.
- Li, W., C. A. Meyer & X. S. Liu (2005) A hidden Markov model for analyzing ChIP-chip experiments on genome tiling arrays and its application to p53 binding sequences. *Bioinformatics*, 21 Suppl 1, i274-82.
- Lian, H., W. A. Thompson, R. Thurman, J. A. Stamatoyannopoulos, W. S. Noble & C. E. Lawrence (2008) Automated mapping of large-scale chromatin structure in ENCODE. *Bioinformatics*, 24, 1911-6.
- Lukashin, A. V. & M. Borodovsky (1998) GeneMark.hmm: new solutions for gene finding. *Nucleic Acids Res*, 26, 1107-15.
- Martelli, P. L., P. Fariselli, A. Krogh & R. Casadio (2002) A sequence-profile-based HMM for predicting and discriminating beta barrel membrane proteins. *Bioinformatics*, 18 Suppl 1, S46-53.
- Pedersen, J. S. & J. Hein (2003) Gene finding with a hidden Markov model of genome structure and evolution. *Bioinformatics*, 19, 219-227.
- Qin, Z. S., J. Yu, J. Shen, C. A. Maher, M. Hu, S. Kalyana-Sundaram & A. M. Chinnaiyan (2010) HPeak: an HMM-based algorithm for defining read-enriched regions in ChIP-Seq data. *BMC Bioinformatics*, 11, 369.
- Rabiner, L. R. (1989) A Tutorial on Hidden Markov-Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77, 257-286.
- Razzaki, T. & A. I. Bukhari (1975) Events following prophage Mu induction. *J Bacteriol*, 122, 437-42.
- Salzberg, S. L., A. L. Delcher, S. Kasif & O. White (1998) Microbial gene identification using interpolated Markov models. *Nucleic Acids Res*, 26, 544-8.
- Siepel, A., G. Bejerano, J. S. Pedersen, A. S. Hinrichs, M. M. Hou, K. Rosenbloom, H. Clawson, J. Spieth, L. W. Hillier, S. Richards, G. M. Weinstock, R. K. Wilson, R. A. Gibbs, W. J. Kent, W. Miller & D. Haussler (2005) Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. *Genome Research*, 15, 1034-1050.
- Siepel, A. & D. Haussler. 2005. Phylogenetic Hidden Markov Models. In *Statistical Methods in Molecular Evolution*, ed. R. Nielsen, 325 - 351. Springer.
- Siepel, A., K. S. Pollard & D. Haussler (2006) New methods for detecting lineage-specific selection. *Research in Computational Molecular Biology, Proceedings*, 3909, 190-205.
- Simpson, J. T., R. E. McIntyre, D. J. Adams & R. Durbin (2010) Copy number variant detection in inbred strains from short read sequence data. *Bioinformatics*, 26, 565-567.
- Sonnhammer, E. L., G. von Heijne & A. Krogh (1998) A hidden Markov model for predicting transmembrane helices in protein sequences. *Proc Int Conf Intell Syst Mol Biol*, 6, 175-82.
- Wang, K., M. Y. Li, D. Hadley, R. Liu, J. Glessner, S. F. A. Grant, H. Hakonarson & M. Bucan (2007) PennCNV: An integrated hidden Markov model designed for high-resolution copy number variation detection in whole-genome SNP genotyping data. *Genome Research*, 17, 1665-1674.
- Won, K. J., I. Chepelev, B. Ren & W. Wang (2008) Prediction of regulatory elements in mammalian genomes using chromatin signatures. *BMC Bioinformatics*, 9, 547.

- Won, K. J., B. Ren & W. Wang (2010) Genome-wide prediction of transcription factor binding sites using an integrated model. *Genome Biol*, 11, R7.
- Xu, H., C. L. Wei, F. Lin & W. K. Sung (2008) An HMM approach to genome-wide identification of differential histone modification sites from ChIP-seq data. *Bioinformatics*, 24, 2344-9.
- Yang, Z. (1995) A space-time process model for the evolution of DNA sequences. *Genetics*, 139, 993-1005.
- Zhou, H. & Y. Zhou (2003) Predicting the topology of transmembrane helical proteins using mean burial propensity and a hidden-Markov-model-based method. *Protein Sci*, 12, 1547-55.

Further Reading

Clote P and Backofen R (2000). *Computational Molecular Biology: An Introduction*. Chichester, UK: John Wiley & Sons.

Durbin, R., S. R. Eddy, Krogh, A and Mitchison, G (1998). [Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids](#). Cambridge, UK: Cambridge University Press.

Rasmus Nielsen (2005). *Statistical Methods in Molecular Evolution*. New York: Springer Verlag.

See also

[Gene Feature Identification](#), [Neural Networks](#), [Pattern Searches](#), [Profile Searching](#), and [Sequence Similarity](#)