ISSN 2250-0987
Neeshu Sharma et al, UNIASCIT, Vol 3 (1), 2013, 353-358

HMM AND PROTEIN PROFILE ANALYSIS: A STATISTICAL PERSPECTIVE

Er. Neeshu Sharma[1], Er. Reet Kamal Kaur[2] , Er. Manpreet Kaur

[1] RIMT-Maharaja Agarssen Engg. College, Mandi Gobindgarh

[2] RIMT-Maharaja Agarssen Engg. College, Mandi Gobindgarh

[3]RIMT-Maharaja Agarssen Engg. College, Mandi Gobindgarh

***Abstract*----** HMM has found its application in almost every field. Applying HMM to biological sequences has its own advantages. HMM's being more systematic and specific, yield a result better than consensus techniques. In this research work we have applied HMM to profile analysis. With our technique it was found that HMM outperformed Dot plots, local alignment and global alignment techniques considerably.

Keywords:, Profile Analysis, HMM, Profile HMM

1. Introduction

Bioinformatics, the application of computational techniques to analyze the information associated with biomolecules on a large-scale, has now firmly established itself as a discipline in molecular biology, and encompasses a wide range of subject areas from structural biology, genomics to gene expression studies The present role of bioinformatics is to aid biologists in gathering and processing genomic data to study protein function.

Proteins are complex organic compounds that consist of amino acids joined by peptide bonds. Proteins are essential to the structure and function of all living cells and viruses. Many proteins function as enzymes or form subunits of enzymes. and they were discovered by the Swedish scientist, Jons Jakob Berzelius in 1838.

Hidden Markov models are sophisticated and flexible statistical tool for the study of protein models. Using HMMs to

analyze proteins is part of a new scientific field called bioinformatics, based on the relationship between computer science, statistics and molecular biology. Hidden Markov models (HMMs) offer a more systematic approach to estimating model parameters. The HMM is a dynamic kind of statistical profile. HMMs are hidden because only the symbols emitted by system are observable, not the underlying walks between states

2. HMM Framework

This model makes use of alignment as the profiling method and uses conceptual alignment as the type of alignment.

Alignment can be considered the most important unsupervised learning problem. The diagram shows the general architecture of an instantiated HMM

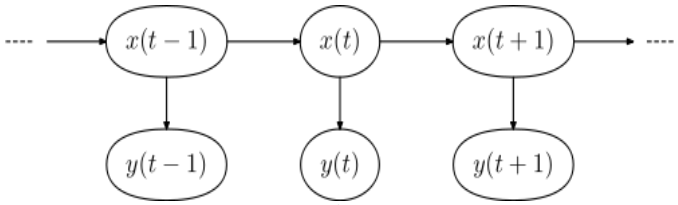

Figure 1.1: Architecture of Hidden Markov Model

Each oval shape represents a random variable that can adopt any of a number of values. The random variable x(t) is the hidden state at time t (with the model from the above diagram, x(t) ε { x1, x2, x3 }). The random variable y(t) is the observation at time t (with y(t) ε { y1, y2, y3, y4 }). The arrows in the diagram (often called a trellis diagram) denote conditional dependencies [30].

2.1 Mathematical Perspective

Hidden Markov models are widely used in science, engineering and many other areas (speech recognition, optical character recognition, machine translation, bioinformatics, computer vision, finance and economics, and in social science).

Definition: The Hidden Markov Model (HMM) is a variant of a *finite state machine* having a set of hidden *states*, Q, an output *alphabet* (observations), O, transition probabilities, A, output (emission) probabilities, B, and initial state probabilities, Π. The current state is not observable. Instead, each state produces an output with a certain probability (B). Usually the states, Q, and outputs, O, are understood, so an HMM is said to be a triple, (A, B, Π).

Formal Definition:

Hidden states Q = { qi }, i = 1, . . . , N .

Transition probabilities A = {aij = P(qj at t +1 | qi at t)}, where P(a | b) is the conditional probability of a given b, t = 1, . . . , T is time, and qi in Q. Informally, A is the probability that the next state is qj given that the current state is qi.

Observations (symbols) O = { ok }, k = 1, . . . , M .

Emission probabilities B = { bik = bi(ok) = P(ok | qi) }, where ok in O. Informally, B is the probability that the output is ok given that the current state is qi.

Initial state probabilities Π = {pi = P(qi at t = 1)

The model is characterized by the complete set of parameters: Λ = {A, B, Π }.

CANONICAL PROBLEMS

There are 3 canonical problems to solve with HMMs:

1. Given the model parameters, compute the probability of a particular output sequence. This problem is solved by the Forward and Backward algorithms (described below).
2. Given the model parameters, find the most likely sequence of (hidden) states which could have generated a given output sequence. Solved by the Viterbi algorithm and Posterior decoding.
3. Given an output sequence, find the most likely set of state transition and output probabilities. Solved by the Baum-Welch algorithm

3. MATHEMATICAL VIEW OF FORWARD VITERBI,AND BACKWARD:

3.1 FORWARD ALGORITHM

Let αt(i) be the probability of the partial observation sequence Ot = {o(1), o(2), ... , o(t)} to be produced by all possible state sequences that end at the i-th state.
αt(i) = P(o(1), o(2), ... , o(t) | q(t) = qi).
Then the unconditional probability of the partial observation sequence is the sum of αt(i) over all N states.

Formal Definition

Initialization:

$$\alpha_1(i) = p_i \, b_i(o(1)) \, , \; i = 1, \ldots , N$$

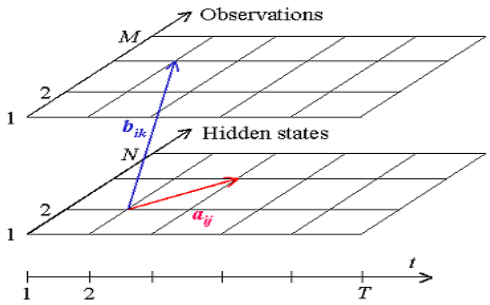

Fig 1.2: Representing HMM States

Recursion:

$$\alpha_{t+1}(i) = \left[\sum_{j=1}^{N} \alpha_t(j) \, a_{ji} \right] b_i\left(o(t+1)\right)$$

here $i = 1, \ldots , N$, $t = 1, \ldots , T - 1$

Termination:

$$P(o(1)o(2)\ldots o(T)) = \sum_{j=1}^{N} \alpha_T(j)$$

3.2 BACKWARD ALGORITHM

In a similar manner, we can introduce a symmetrical backward variable βt(i) as the conditional probability of the partial observation sequence from o(t+1) to the end to be produced by all state sequences that start at i-th state (3.13).

βt(i) = P(o(t+1), o(t+2), ... , o(T) | q(t) = qi).

The Backward Algorithm calculates recursively backward variables going backward along the observation sequence. The Forward Algorithm is typically used for calculating the probability of an observation sequence to be emitted by an HMM, but, as we shall see later, both procedures are heavily used for finding the optimal state sequence and estimating the HMM parameters.

Formal Definition

Initialization:

$$\beta_T(i) = 1 \, , \; i = 1, \ldots , N$$

According to the above definition, $\beta_T(i)$ does not exist. This is a formal extension of the below recursion to $t = T$.

Recursion:

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j\left(o(t+1)\right) \beta_{t+1}(j)$$

here $i = 1, \ldots , N$, $t = T - 1, \; T - 2 , \ldots , 1$

Termination:

$$P(o(1)o(2)\ldots o(T)) = \sum_{j=1}^{N} p_j b_j(o(1))\beta_1(j)$$

Obviously both Forward and Backward algorithms must give the same results for total probabilities $P(O) = P(o(1),$

3.3 VITERBI ALGORITHM

The Viterbi algorithm chooses the best state sequence that maximizes the likelihood of the state sequence for the given observation sequence.

Let δ t(i) be the maximal probability of state sequences of the length t that end in state i and produce the t first observations for the given model.

$$\delta\, t(i) = \max\{P(q(1), q(2), \ldots, q(t-1)\ ; o(1), o(2), \ldots , o(t) \mid q(t) = qi\ ).\}$$

The Viterbi algorithm is a dynamic programming algorithm that uses the same schema as the Forward algorithm except for two differences:

1. It uses maximization in place of summation at the recursion and termination steps.
2. It keeps track of the arguments that maximize δ t(i) for each t and i, storing them in the N by T matrix ψ. This matrix is used to retrieve the optimal state sequence at the backtrack

Initialization:

$$\delta_1(i) = p_i\, b_i(o(1))$$
$$\psi_1(i) = 0\ ,\ i = 1, \ldots , N$$

According to the above definition, βT(i) does not exist. This is a formal extension of the below recursion to .

Recursion:

$$\delta t\ (\ j) = \max i\ [\delta t - 1(i)\ aij]\ b\ j\ (o(t))$$
$$\psi t(\ j) = \arg\max i\ [\delta t - 1(i)\ aij]$$

Termination:

$$p^* = \max i\ [\delta T(\ i\ )]$$
$$q^*T = \arg\max i\ [\delta T(\ i\ )]$$

Path (state sequence) backtracking:

$$q^*t = \psi t+1(\ q^*t+1)\ ,\ t = T - 1,\ T - 2\ , \ldots , 1$$

Path (state sequence) backtracking:

$$q^*t = \psi t+1(\ q^*t+1)\ ,\ t = T - 1,\ T - 2\ , \ldots , 1$$

4. Criteria of profile Analysis

Basically, a profile is a description of the consensus of a multiple sequence alignment. It uses a position-specific scoring system to capture information about the degree of conservation at various positions in the multiple alignments. This makes it a much more sensitive and specific method for database searching than pair wise method Following are the steps followed in this research work:

Align the sequences in the family: Initially, we will assume that there are no gaps in the alignment. We look at the alignment of N sequences of l positions as follows:

Sequence	Position					
	1	2	3	4	…	l
1	a11	a12	a13	…	…	a1l
2	a21	a122	a23	…	…	a2l
3	a31					
-						
-						
N	aN1	aN2	aN3	…	…	aNl

Table 1: Alignment of sequences

where aij denotes the amino acid from the ith sequence at the jth position.

Use the alignment to create a profile: We build the profile as follows. We compute:

fij = % of column j that is amino acid i

bi = % of background which is amino acid i

The background" can be computed, for example, from a large sequence database, or from a genome, or from some particular protein family.

Now compute the 20 x l array Pij , where

Pij = fij/bi

Intuitively, Pij is the "propensity" for amino acid i in the j position in the alignment.

This gives us the following table:

Sequence	Position						
	1	2	3	4	5	…	L
L	PL1	PL2	PL3	…	…		PLl
V	PV1	PV2	PV3	…	…		PVl
F	PF1						
.							
.							
.							

Table 2: Alignment to compute the Profile

And we use this table to compute:
Scoreij = log(Pij)
Test new sequences against the profile: To use the profile to score a new sequence, we do the following:
Slide a window of width l over the new sequence.
The score of the window equals the sum of the scores of each position in the window.
If the score of the window is higher than the cut off, which is determined empirically, we can conclude that the window is a member of the family.
In addition, the higher the score, the more confident the prediction.

5. METHODOLOGY OF WORK

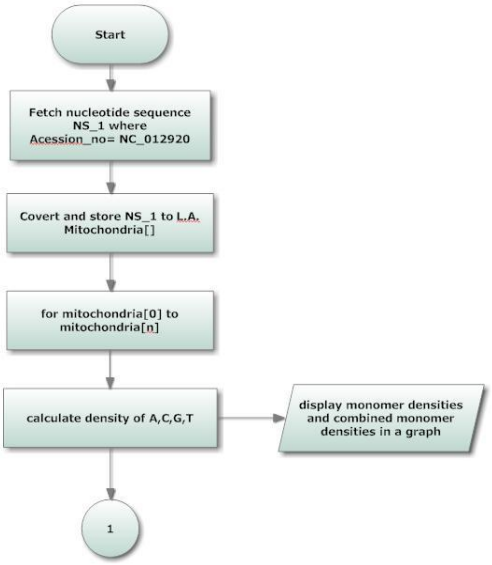

Fig 1.3: Flowchart for accessing a sequence and plotting Monomer and combined Monomer densities

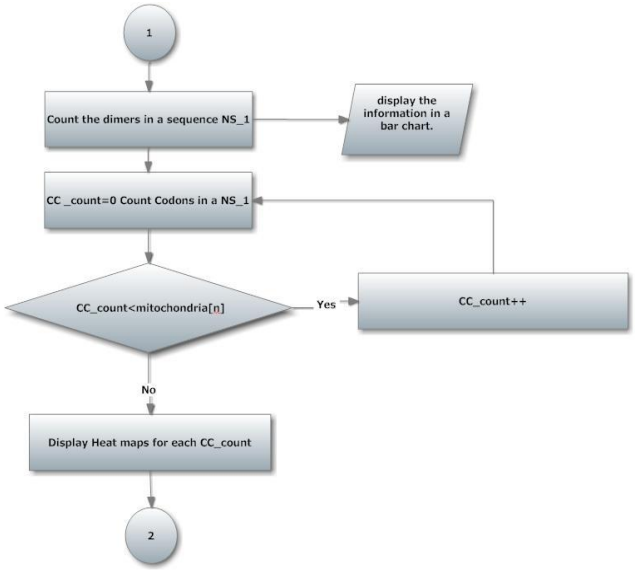

Fig 1.4: Counting Dimmers And plotting Heat maps

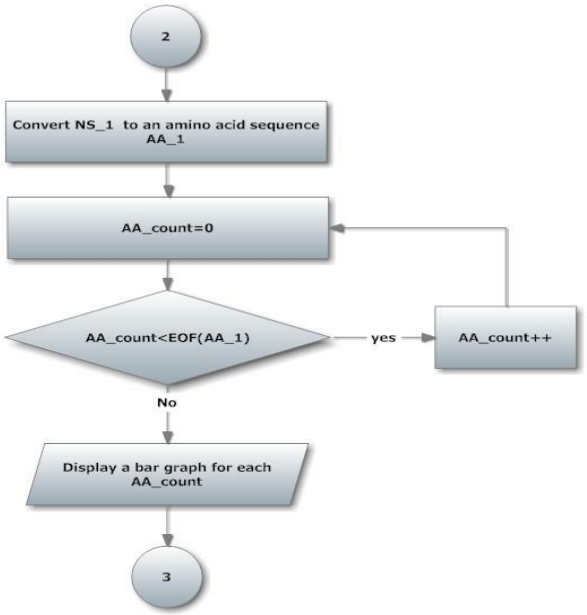

Fig 1.5: Converting to amino acid sequence and representing each amino acid count

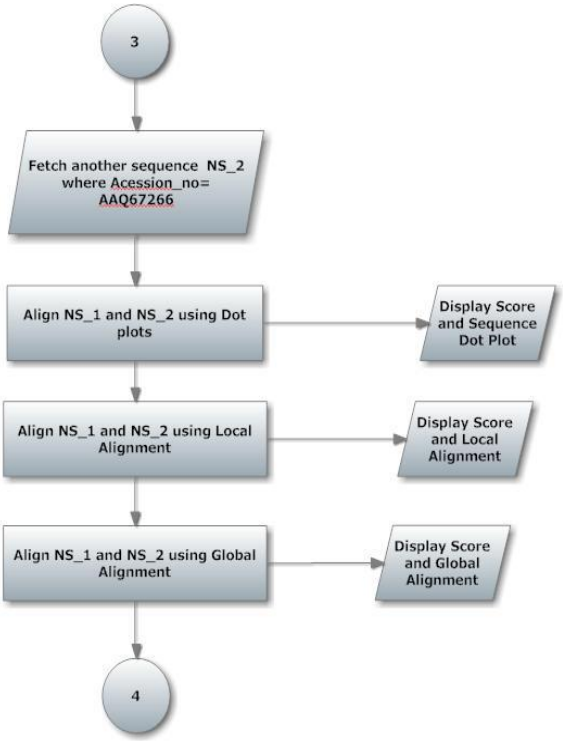

Fig 1.6: Aligning the sequences using Dot plots, Local alignment and Global Alignment

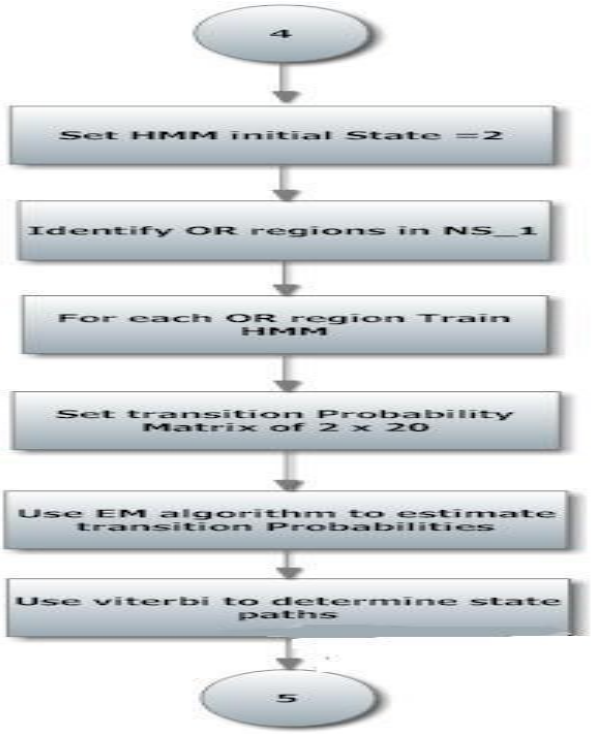

Fig 1.7: Creating and training HMM

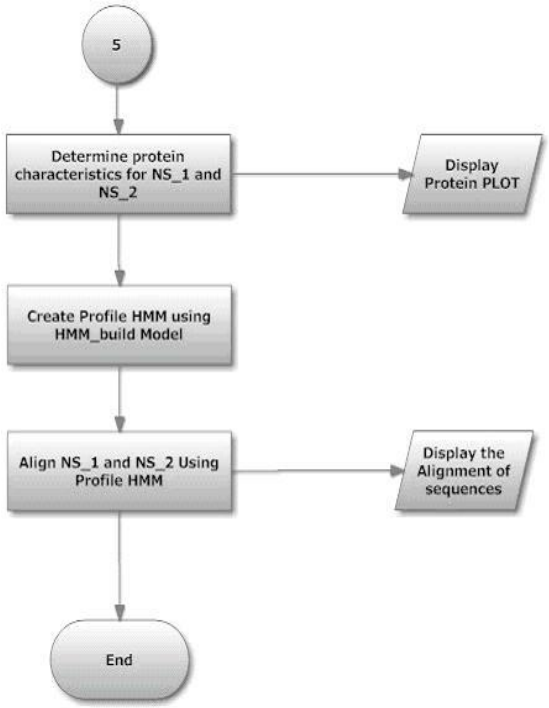

Fig1.8: Plotting protein Properties, Identifying Profiles and Aligning using HMM

6. CONCLUSION AND FUTURE WORK

Profile is used to search a target sequence for possible matches to the profile using the scores in the table to evaluate the likelihood at each position.
The research can be extended to:

1. Real user interface.
2. Provision to include other sequences (i.e. with different accession numbers and their supported files) automatically.
3. Provision to access the data from a database.
4. Provision for choice of alignment technique
5. Provision to incorporate various input formats

REFERENCES

[1] Andrew E. Teschendorff1, Ali Naderi1, Nuno L. Barbosa-Morais, and Carlos Caldas(2006) " *Protein homology detection by HMM–HMM comparison",* Department of Oncology , Vol. 22, pp.2269–2275.

[2] *Anoop Kumar and Lenore Cowen(2010) "Recognition of beta-structural motifs using hidden Markov models trained with simulated evolution", Department of Computer Science,Vol 26,pp.i287-i293.*

[3] Can, T., Wang, Y. (2004*) "Automated Protein Classification using Consensus Decision",* Bioinformatics, vol. 12, pp. 317-327.

[4] Cheng BY, Carbonell JG, Klein-Seetharaman J.(2005*)"Protein classification based on text document classification techniques."*Journal of Bioinformatics, Volume 212, Pages 67-70.

[5] Christopher Tamas and Richard Hughey (1998)"Reduced space hidden Markov model training", Department of computer engineering, Jack Vol. 14 ,pp.401-406.

[6] Dariusz Mrozek†, Bożena Małysiak-Mrozek(2010) "An Improved Method for Protein

[7] Devos, D. and Valencia, A. (2000) "Practical Limits of Function Prediction", Protein Design Group, National Centre for Biotechnology, CNB-CSIC Madrid, EBaskin School engineering, University of California, USA, Spain, pp. 134-170.

[8] Erik L. L. Sonnhammer, Sean R. Eddy, Ewan Birney, Alex Bateman and Richard Durbin (1998) *"Pfam: multiple sequence alignments and HMM-profiles of protein domains",* Nucleic Acids Research vol. 26, No.1, pp. 320-322.

[9] Georgina Mirceva1 and Danco Davcev (2009) *"HMM based approach for classifying protein structures"* International Journal of Bio- Science and Bio- Technolog, vol. 1, no.1, pp. 37-46.

[10] Herbert Popp, Mona Singh and Johnson parker (2002) *"Topics in Computational Molecular Biology"* Lecture notes in bio computing, pp.1-11.

[11] Johanne Söding(2005) *"Protein homology detection by HMM and HMM comparison"* Department of Protein Evolution, Vol. 21 , pp. 951–960

[12] Jia Song, Chunmei Liu, Yinglei Song, Junfeng Qu, and Gurdeep S. Hura (2010*)" Alignment of multiple proteins with an ensemble of Hidden Markov Models"* ,Data mining in Bioinformatics, Vol 4,pp.60-71.

[13] N. von Öhsen, I. Sommer, R. Zimmer (2003) *"Profile-Profile Alignment: A Powerful Tool for Protein Structure Prediction"* Pacific Symposium on Biocomputing, Vol 8, pp 252-263.

[14] Park, C.Y., Park, S.H., Kim, D.H., Park, S.H. and Hwang, C.J. (2004) *"A new protein Classification method using dynamic classifier",* Bioinformatics, vol. 9, pp 32-35.

[15] Raninder Kaur, Shavinder Kaur, Reet Kamal Kaur and Amandeep Kaur (2010) *"Characterization of Parathyroid Hormone using HMM Framework"* International Journal of Computer Applications, vol. 1, no. 16, pp. 65-68.

[16] T. Plötz, and G.A. Fink, *"Pattern recognition methods for advanced stochastic protein sequence analysis using HMMs",* Pattern Recognition, vol. 39, 2006, pp. 2267-2280.

[17] Thakoor N, Gao J, Jung S.(2007) *"Hidden Markov model-based weighted likelihood discriminant for 2-D shape classification."* Online journal at Springerlink.com

[18] Tolga Can, Orhan C¸ amoglu, Ambuj K. Singh, Yuan-Fang Wang (2004) *"Automated Protein Classification Using Consensus Decision"* Journal of Molecular Biology, Volume 348, Issue 4, Pages 66-68.

[19] Usman Roshan and Dennis R. Livesay (2006) *"Probalign: multiple sequence alignment using partition function posterior probabilities"* Bioinformatics, Vol. 22, No. 22, pp 2715-2721.

[20] Valeria De Fonzo, Filippo Aluffi-Pentini and Valerio Parisi. (2009) *"Hidden Markov Models in Bioinformatics",* Current Bioinformatics, 2007, Vol. 2, No. 1, pp. 49-61.

[21] Wong, L., Chua, H., 17]W.R. Taylor, and C.A. Orengo, *"Protein structure alignment",* J. Mol. Biol., vol. 208, 1989, pp. 1-22.