Gene Finding Using Hidden Markov Model

Dyotana Das(19223) Vidushi Dwivedi (19606)
April, 2022

1. Introduction

Gene finding refers to finding genes within a DNA sequence and labeling them as coding, intergenic, introns etc. This can be done using the statistical model called Hidden Markov Model, which is considered to be a Markov chain with hidden or unobservable states.
Gene finding application using a hidden markov model involves three basic problems of evaluation, decoding and learning. Evaluation involves finding the probability of a certain HMM generating a given sequence , decoding step finds the most likely hidden state and learning step is responsible for finding the most optimal model by adjusting and tuning parameters.
The main aim of the problem is to find generative models to describe sequences.

2. Motivation

It is an important task in bioinformatics to deduce a DNA protein sequence and compare it with a database of protein sequences. HMM is used in both of the aforementioned tasks.
Determining DNA and RNA sequences rather than protein sequences is cheaper and thus if preferred over the latter. Gene finding is one of the first and most important steps in understanding the genome of a species once it has been sequenced which is important to develop understanding needed for curing diseases. Gene discovery helps in describing individual genes in terms of their functions.

3. Hidden Markov Model

Hidden Markov Model (HMM) is a statistical Markov model in which the system being modeled is assumed to be a Markov process — call it X — with unobservable ("hidden") states. As part of the definition, HMM requires that there be an observable process Y whose outcomes are "influenced" by the outcomes of X in a known way. Since X cannot be observed directly, the goal is to learn about X by observing Y. HMM has an additional requirement that the outcome of Y at time t=t_0 may be "influenced" exclusively by the outcome of X at t=t_0 and that the outcomes of X and Y at t<t_0 must not affect the outcome of Y at t=t_0. [6]

3.1. Problem 1: Evaluation

Evaluation is finding how probable is a given sequence under a given HMM λ. More technically it is finding P(O/λ) where O is observation sequence and λ is the HMM where P(O/λ) = Σ P(O,q/λ) i.e. sum of probabilities of the observed sequence under all possible hidden state sequences. The naive way to approach this task is computationally extensive and requires $O(TN^T)$ for T time instances and N states. Thus we move to Forward algo and Backward algo which are a better way to approach the problem $O(N^2T)$. The two algorithms are:

Backward Algorithm:

- Define a backward variable $\beta_t(i)$ [for partial sequence from t upto time instance T, probability of stopping at ith state at time instance t]

Initialization:

$$\beta_T(j) = 1$$

Recursion:

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

Final Result:

$$p(O|\lambda) = \sum_{i=1}^{N} \beta_1(i) \qquad t = T-1,\ldots,1$$

Forward Algorithm:

- Define a forward variable $\alpha_t(i)$ [for partial sequence upto time instance t, probability of stopping at ith state at time instance t].
- The algorithm is:

Initialize:

$$\alpha_1(i) = \pi_i b_i(o_1)$$

Recursion:

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] b_j(o_{t+1})$$

Final Result:

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

3.2. Problem 2: Decoding

Decoding is finding the best state sequence for a given observation sequence. This too is highly computationally extensive and requires $O(TN^T)$ for T time instances and

N states. Thus we se Viterbi algorithm to reduce time complexity (keeps best state sequence at each instance). The algorithm is:
(Define variable δ such that (where δ(i) is the probability of most probable path ending at state q_t = i)):

$$\delta_t(i) = \max_q P(q_1, q_2, ..., q_t = i, o_1, o_2, ...o_t \mid \lambda)$$

Recurrent property:

$$\delta_{t+1}(j) = \max_i(\delta_t(i)a_{ij})b_j(o_{t+1})$$

Algorithm:

1. Initialise:

$$\delta_1(i) = \pi_i b_i(o_1) \qquad 1 \le i \le N$$

$$\psi_1(i) = 0$$

2. Recursion:

$$\delta_t(j) = \max_{1 \le i \le N}(\delta_{t-1}(i)a_{ij})b_j(o_t)$$

$$\psi_t(j) = \arg\max_{1 \le i \le N}(\delta_{t-1}(i)a_{ij}) \qquad 2 \le t \le T, 1 \le j \le N$$

3. Termination:

$$P^* = \max_{1 \le i \le N} \delta_T(i)$$

$$q_T^* = \arg\max_{1 \le i \le N} \delta_T(i)$$

4. Backtracking state sequence

$$q_t^* = \psi_{t+1}(q_{t+1}^*) \qquad t + T - 1, T - 2, ..., 1$$

3.3. Problem 3: Learning

The third, and by far the most difficult, problem of HMMs is to determine a method to adjust the model parameters (A, B, π) to satisfy a certain optimization criterion. There is no known way to analytically solve for the model parameter set that maximizes the probability of the observation sequence in a closed-form. We can, however, choose A = (A, B, π) such that its likelihood, P(O|λ), is locally maximized using an iterative procedure such as the Baum-Welch method (also known as the EM (expectation-maximization) method), or using gradient techniques. We discussed one iterative procedure, based primarily on the classic work of Baum - welch, for choosing the maximum likelihood (ML) model parameters.

To describe the procedure for reestimation (iterative update and improvement) of HMM parameters, we first define $\xi_t(i,j)$, , the probability of being in state i at time t, and
state j at time r+1, given the model and the observation sequence.

$$\xi_t(i,j) = P(q_t = i,\ q_{t+1} = j | \mathbf{O}, \lambda).$$

From the definition of forward and backward variable we can rewrite this as :

$$\begin{aligned}\xi_t(i,j) &= \frac{P(q_t = i,\ q_{t+1} = j,\ \mathbf{O} \mid \lambda)}{P(\mathbf{O} \mid \lambda)} \\ &= \frac{\alpha_t(i)\, a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{P(\mathbf{O}|\lambda)} \\ &= \frac{\alpha_t(i)\, a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i)\, a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}.\end{aligned}$$

As we have define $\gamma_t(i)$ earlier, so we can relate it with the equation :

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j).$$

If we sum $\gamma_t(i)$ over the time index t, we get a quantity that can be interpreted as the expected (over time) number of times that state ‘i’ is visited, or equivalently, the expected number of transitions made from state ‘i’.

Similarly, summation of $\xi_t(i,j)$ over t (from 1 tor= T-1) can be interpreted as the expected number of transitions from state ‘i’ to state ‘j’.

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{expected number of transitions from state } i \text{ in } \mathbf{O}$$

$$\sum_{t=1}^{T-1} \xi_t(i,j) = \text{expected number of transitions from state } i \text{ to state } j \text{ in } \mathbf{O}.$$

Using the above formulas (and the concept of counting event occurrences), we can give a method for re-estimation of the parameters of an HMM. A set of reasonable reestimation formulas for π, A, and B is -

$$\bar{\pi}_i = \text{expected frequency (number of times) in state } i \text{ at time } (t = 1) = \gamma_1(i)$$

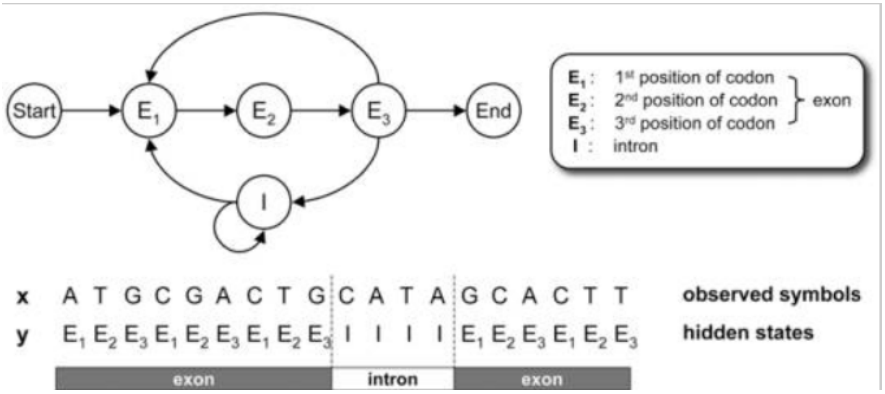
$$\bar{a}_{ij} = \frac{\text{expected number of transitions from state } i \text{ to state } j}{\text{expected number of transitions from state } i} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$\bar{b}_j(k) = \frac{\text{expected number of times in state } j \text{ and observing symbol } \mathbf{v}_k}{\text{expected number of times in state } j} = \frac{\sum_{t=1 \text{ s.t. } \mathbf{o}_t = \mathbf{v}_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}.$$

3.4. Gene finding using HMM

HMMs can be effectively used for representing biological sequences. Considering an HMM that models protein-coding genes in eukaryotes. It is well known that many protein-coding regions display codon bias. The non-uniform usage of codons results in different symbol statistics for different codon positions, and it is also a source of the period-3 property in the coding regions. These properties are not observed in introns, which are not translated into amino acids. Therefore, it is important to incorporate these codon statistics when modeling protein-coding genes and building a gene-finder.HMM tries to capture the statistical differences in exons and introns. The HMM has four states, where E1, E2, and E3 are used to model the base statistics in exons. Each Ek uses a different set of emission probabilities to reflect the symbol statistics at the kth position of a codon. The state I is used to modeling the base statistics in introns. This HMM can represent genes with multiple exons, where the respective exons can have a variable number of codons, and the introns can also have variable lengths.

If x is a protein-coding gene, how can we predict the locations of the exons and introns in the given sequence? by computing the observation probability of x based on the given HMM that models coding genes. If this probability is high, it implies that this DNA sequence is likely to be a coding gene. Otherwise, we may conclude that x is unlikely to be a coding gene.



A simple HMM for modeling eukaryotic genes. [7]

The second question is about predicting the internal structure of the sequence, first predict the state sequence y in the HMM that best describes x. Once we have inferred the best y, it is straightforward to predict the locations of the exons and introns.

4. Experiments

Detailed study of HMM was done to fully understand the problem. We then implemented Forward, Backward, Viterbi and Baum Welch algorithms from scratch and tested it on audio data(speech and music classes). The code, data and other reference materials are uploaded on the GitHub repository: https://github.com/vidushid/HiddenMarkovModel

5. Refrences

1.https://www.cs.jhu.edu/~langmea/resources/lecture_notes/22_markov_chains_v2.pdf

2.https://www.cs.jhu.edu/~langmea/resources/lecture_notes/23_hidden_markov_models_v2.pdf

3. Burge, C.B. and S. Karlin, 1998. Finding the genes in genomic DNA. Curr. Opin. Struct. Biol., 8: 346-354

4. Birney, E., 2001. Hidden markov models in biological sequence analysis. IBM J. Res. Dev., 45: 449-454

5. https://scialert.net/fulltext/?doi=jas.2012.1518.1525#936002_ja

6. https://en.wikipedia.org/wiki/Hidden_Markov_model#:~:text=Hidden%20Markov%

20Model%20(HMM)%20is,of%20in%20a%20known%20way.

7.https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2766791/

8. Fundamentals of speech recognition by Lawrence Rabiner