Hidden Markov Models

Rushali Chopra

26.06.2023

Abstract

This Report provides a comprehensive explanation of a statistical model called the Hidden Markov Model, elucidating its theoretical background as well as practical applications.

1 Introduction

Andrey Markov, a pioneer who popularized them in the early 1900s, is responsible for giving Markov models their name. These models, which fall under the genre of probabilistic models, are useful for predicting a system's future state while taking into account its current state. The Hidden Markov Model, often known as HMM, is a particular kind of Markov model. It initially made it's debut in the field of speech recognition and later found effective applications in the analysis of biological sequences, starting in the late 1980s. HMMs are statistical models created for situations in which the system under study involves hidden states that cannot be seen immediately. When used, they model a sequence as the result of a discrete stochastic process that progresses through a number of states that are hidden from the observer. Each of these hidden states emits a symbol that corresponds to an elementary component of the modelled data. For instance, these emitted symbols could correspond to individual amino acids in case of a protein sequence. In the following sections, the discussion will revolve around the concepts of Markov chains and HMMs. The emphasis will be on presenting some of the algorithms that drive the functionality of HMMs. Additionally, the problem of identifying methylated and un-methylated regions in DNA (CpG-islands finding) is also discussed.

2 Markov Chains

Markov chains can be thought of as a system or mechanism that switches between various states S where $S = \{s_1, \ldots, s_n\}$, frequently forming a sequential chain. These chains follow the Markov property, which asserts that the probability of transitioning to a certain state i.e s_j next depends only on the present state i.e s_i and is independent of earlier states. A Markov chain is characterized by three essential components:

1. **Initial Probability Distribution**: This initial probability distribution, denoted as π_i, signifies the likelihood that the Markov chain commences in a specific state, i. For certain states, say j, where π_j equals zero ($\pi_j = 0$), it implies that these states cannot serve as initial states.

2. **One or More States**: The Markov chain comprises a set of states S, which can represent various conditions, scenarios, or positions within the system.

3. **Transition Probability Distribution**: The transition probability matrix, represented as A, encapsulates the probabilities of transitioning from one state, i , to another state, j. Each element a_{ij} in this matrix denotes the probability of moving from state i to state j.

3 Hidden Markov Models(HMMs): A Deeper Look

There is a concept known as a 'hidden state' in the world of HMMs. This hidden state denotes a variable that is not visible to the naked eye. However, a technique known as the Markov assumption allows us to establish accurate estimations about it by analysing one or more observable states. Essentially, the Markov assumption asserts that, in particular with regard to hidden states, the subsequent event heavily depends on the preceding event. The key distinction between Markov chains and HMMs is that HMMs do not have a one-to-one correlation between states and symbols.

Imagine, for instance, that one of our tasks is to predict anything, like the weather for tomorrow. Although humans are unable to directly detect every component influencing the weather, we can nevertheless anticipate future circumstances by looking at past weather patterns (observable states) and using the Markov assumption.

3.1 Components of HMMs:-

A Hidden Markov Model (HMM) comprises five key components:

1. **Initial State Probabilities (π_i)**: These represent the likelihood that the HMM starts in each hidden state, denoted as π_i. In some cases, certain states (j) may have $\pi_j = 0$, meaning they cannot serve as initial states. These initial probabilities define the starting conditions of each hidden variable at time $t = 0$.

2. **Hidden States**: HMMs involve a set of hidden states, which represent different situations or conditions within the system.

3. **Transition Probability Matrix (A)**: This matrix provides the probabilities of transitioning from one hidden state (state i) to another (state j). It serves as a roadmap for how hidden states evolve from one to another.

$$\Pr(s_{n+1} = i_{n+1} \mid s_0 = i_0, s_1 = i_1, \ldots, s_n = i_n) = p_{i_n i_{n+1}}(t_{n+1} - t_n) \tag{1}$$

   where s_{n+1} represents the state at time t_{n+1} , $s_0 = i_0, s_1 = i_1$ represents the state sequence being observed from time t_0 to t_n and $p_{i_n i_{n+1}}(t_{n+1} - t_n)$ represents the transition probability.

4. **Sequence of Observations**: While hidden states are not directly observable, we observe a sequence of outcomes over time.

5. **Emission Probabilities**: These probabilities, often referred to as emission probabilities, describe the likelihood of generating each observation (o_i) from a specific hidden state. They indicate how probable it is to see a particular observable output for each hidden state at the beginning of the process.

$$e_k(b) = P(o_i = b \mid \pi_i = k) \tag{2}$$

where $e_k(b)$ represents the emission probability for b when in state k and $P(o_i = b \mid \pi_i = k$ is the conditional probability of observing symbol o_i at time i given π_i as hidden state.

In simplest terms, an HMM starts in one of multiple potential initial states, switches between these states with specific probabilities, and produces observable outcomes along the way. These results offer clues as to the concealed situations it is experiencing, and the emission probabilities indicate how likely each clue is in each situation. Even when the background is unknown, these elements work together to understand the sequence of events.

4 Algorithms and Implementation

HMMs involve several algorithms that are crucial for their implementation and application. The Viterbi algorithm, which is one of the crucial algorithms, is important for decoding sequences and figuring out the most likely hidden state sequence if a set of observations is given . Then there is the Forward method for probabilistic computation and the other is the forward-backward algorithm which is essentially crucial for model training.

4.1 Decoding: Viterbi Algorithm

The purpose of the Viterbi Algorithm is to predict the hidden states sequence given the observed data and trained model. It is a dynamic programming algorithm in determining the most optimal path by considering the maximum probabilities of state transitions and observations at each step. It is being widely used in field of speech recognition, pattern recognition etc.

$$\pi^* = argmax P(x, \pi) \tag{3}$$

where π^* is the state sequence to be found , argmax is to maximize the probability and $P(x, \pi)$ is the joint probability of observing x where the state sequence is π.

Here's a simplified step-by-step explanation of the Viterbi algorithm:

1. **Initialisation**: The matrix is being initialised in which each cell corresponds to hidden state at specific time step whereas the first column is initialised with the product of initial state probabilities and emission probabilities of the corresponding observations.

2. **Recursion**: Probabilities of being in each hidden state at each time step based on the previous time step's probabilities and the transition probabilities are calculated. Out of all, maximum probability is taken into account.

$$v_l(i) = e_l(x_i) \max_k (v_k(i-1)a_{kl}) \tag{4}$$

where $v_l(i)$ is the viterbi variable for state l at time i, $e_l(x_i)$ is emission probability for observing x_i at time i given the model is in state l and a_{kl} is the transition probability from state k to state l.

3. **Backtracking**: Starting from the last time step and tracing back by choosing the hidden state with the highest probability in the previous time step will give the sequence of hidden states.

It considers the transitions between states and the emission probabilities of observations, effectively navigating through the hidden state space to identify the optimal path.

4.2 Likelihood Computation: Forward Algorithm

It is also a dynamic programming algorithm similar to Viterbi algorithm. Essentially, it deals in computing the likelihood of the observation sequence given the model λ by summing over all the probabilities of all possible state paths.

It involves the following steps:

1. **Initialisation**: The product of initial state probability π_i and corresponding emission probability of the very first observation $b_i(o_1)$ is calculated which is basically referred to as initialisation of the vector N , where N is the number of states in HMM. $\alpha[i]$ is the forward variable represented for state i at time step 1.

$$\alpha[i] = \pi_i \cdot b_i(o_1), \quad for\ i = 1toN \tag{5}$$

2. **Recursion**: The forward probabilities for each state S_i at time t are calculated.

$$\alpha[i] = \sum_{j=1}^{N} \alpha[j] \cdot a_{ji} \cdot b_i(o_t) \tag{6}$$

where $\sum_{j=1}^{N} \alpha[j]$ represents the summation over all possible states , $b_i(o_t)$ is the emission probability of observation o_t and a_{ji} is the transition probability.

3. **Termination**: The final probability for the entire sequence is calculated by adding all the probabilities for all states at time T.

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha[i] \tag{7}$$

where $P(O|\lambda)$ is the probability of observing emissions sequence O given the model λ.

4.3 Training : Forward-Backward Algorithm

This algorithm's primary goal is to determine model parameters from a given sequence of observations and collection of states. It is a particular instance of the Expectation-Maximization algorithm, or EM-algorithm. It involves initially estimating probability and subsequently improving these estimates incrementally. We need to establish Forward and Backward probabilities in order to comprehend the method.
Given the current state i at time t, backward probability, denoted as β, is the likelihood of observing the series of observations beginning at time t + 1 until end.

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \ldots, o_T | q_t = i, \lambda) \tag{8}$$

It is computed inductively in the similar manner to the Forward Algorithm. We then use the both the probabilities to estimate transition and emission probabilities and perhaps in re-estimating initial state probabilities. These updates are performed iteratively in order to converge to a set of parameters that maximizes the likelihood of the observations.

5 Applications of HMMs in Bioinformatics

Bioinformatics uses complex algorithms and Hidden Markov Models (HMMs) to discover the hidden information contained in biological sequences. For example, in gene prediction, viterbi algorithm helps in decoding DNA sequences thereby deciphering the intricate patterns of exons and introns in genes. Another application where HMMs are widely used is functional motif discovery to essentially find conserved patterns or domains within proteins , which uses forward-backward algorithm. Whether it is Multiple Sequence Alignment (MSA) , cancer genomics or epigenomics HMMs have had a significant impact.
Among the various applications mentioned before , finding CpG-islands in DNA regions will be specifically discussed.
CpG sites, which are DNA areas marked by the presence of cytosine followed by guanine, are usually discovered within CpG islands. Through epigenetic mechanisms, methylation of these CpG sites, which is frequently mediated by DNA methyltransferases, is important for the regulation of gene expression. In the majority of mammalian genomes, CpG islands are frequently linked to the gene's promoter regions, making their presence a significant marker for gene finding. Eight states comprise the model in this scenario, corresponding to four letters in DNA $\Sigma = \{A, C, G, T\}$, where the states might be in form A^+ or A^- and emitted symbols without any labels. The label + tells that a particular DNA sequence is a CpG-island while − signifies it opposite.
Given the DNA sequence , we first try to find the probability of it being a CpG-island and the probability of it not being a CpG-island. The next step is to find out log-odds ratios of these two probabilities for both the cases($+and-$). $P(x)$ for each markov chain is computed , denoted as $p(x|+)$ and $p(x|-)$ followed by finding log-odds ratio.

$$\log\left(\frac{p(x|+)}{p(x|-)}\right) \tag{9}$$

If $\log\left(\frac{p(x|+)}{p(x|-)}\right) > 0$, it means that the sequence belongs to a CpG-island.

Usually , both the markov chains (+) and (−) are combined to build a single HMM. It is possible due to the fact that there is no one-to-one correspondence between states and symbols of the given sequence.

6 Conclusion

To conclude, HMMs are employed for modeling and evaluating sequential data with hidden structures. They are fundamental in many disciplines because they are essentially based on probability theory while also managing discrete and continuous data. They facilitate various tasks in bioinformatics, including functional annotation of sequences and identifying phylogenetic relationships, among others. However, one of the drawbacks can be the estimation of parameters and the assumption of conditional independence of observations. They nevertheless continue to make a substantial contribution to biological research and discoveries.

References

[1] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Protein and Nucleic Acids*, Cambridge University Press, 1998, Chapter 3.

[2] Eisner, Jason. (2002). An interactive spreadsheet for teaching the forward-backward algorithm. Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching NLP and CL.

[3] Forney, Jr., Gerald D. (1973). The Viterbi algorithm. Proceedings of the IEEE, 61(3), 268–278.

[4] Franzese, M., Iuliano, A. (2018). Hidden Markov Models. Encyclopedia of Bioinformatics and Computational Biology, 753-762. https://doi.org/10.1016/B978-0-12-809633-8.20488-3

[5] CpG site. (2023, July 4). In Wikipedia. https://en.wikipedia.org/wiki/CpG_site

[6] Ajitesh Kumar .(2023).Hidden Markov Models: Concepts, Examples https://vitalflux.com/hidden-markov-models-concepts-explained-with-examples/

[7] OpenAI. (2023). ChatGPT (August 3 Version) [Large language model]. https://chat.openai.com