

Lab 7 -HMM

Madhumita Krishnan

11/29/2018

1 Introduction

This report deals with the application of Viterbi Algorithm to find the most probable sequence of states that could have generated the observed sequence of events.

The algorithm is run through two inputs and in each case the most probable path and its corresponding probability is computed recursively.

2 Method

In a Hidden Markov Model the states are not directly visible. Only outputs or observations dependent on the state are visible. Each state has a probability distribution over the outputs. The Viterbi Algorithm works by finding the maximum probability over all possible state sequences.

The Hidden Markov Model considered in the report is composed of two states H and L . Each state emits four outputs each that can be observed. The model parameters namely prior probabilities, state transition probabilities and state conditional output probabilities for the system are known.

The probability of the most probable path ending in state L with observation i in position x given by Equation 1 where j precedes i in the output sequence.

$$P_L(i, x) = e_L(i) \max_K (P_K(j, x-1) * P_{KL}) \quad (1)$$

The first sequence given in this problem is GGCACTGAA. The probability of the most probable path ending in state H with observation C at 5th position is seen in Equation 2 and the probability of the most probable path ending in state L with observation C at 5th position is seen in Equation 3.

$$P_H(C, 5) = e_H(C) \max(P_L(A, 4) * P_{LH}, P_H(A, 4) * P_{HH}) \quad (2)$$

$$P_L(C, 5) = e_L(C) \max(P_L(A, 4) * P_{LL}, P_H(A, 4) * P_{HL}) \quad (3)$$

Iteratively $P_H(i, x)$ and $P_L(i, x)$ can be calculated for each element in the output sequence and the values are tabulated. The highest probability obtained in the end of the sequence is the probability of the most probable path. The most probable path can be retrieved by backtracking.

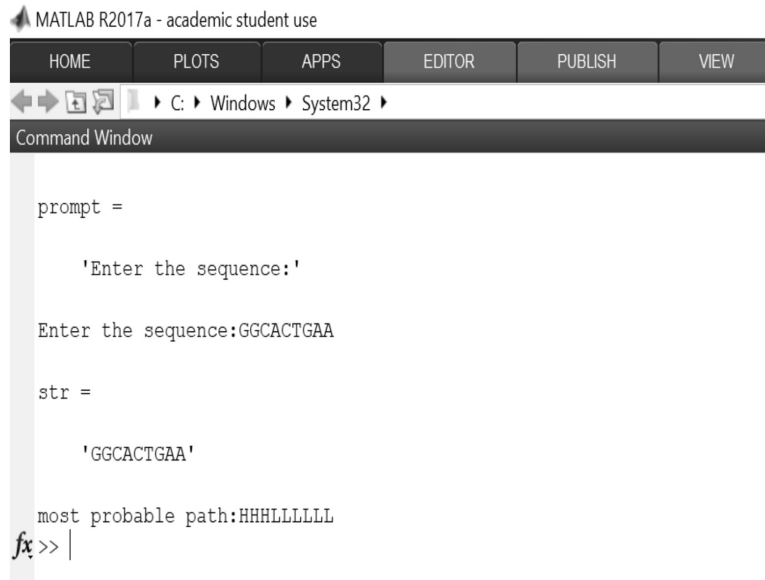
For calculations it is convenient and easier to use logarithm of the probabilities as this allows us to compute sums instead of products. The results can be extended to systems with any number of observations and states.

H	-2.736	-5.473	-8.210	-11.532	-14.006	-17.328	-19.539	-22.861	-25.657
L	-3.321	-6.058	-8.795	-10.947	-14.006	-16.480	-19.539	-22.013	-24.487

Table 1: Probabilities emitted by states H and L

3 Results

For the first input sequence $S = \text{GGCACTGAA}$, Table 1 gives the logarithm of the probabilities $P_H(i, x)$ and $P_L(i, x)$ that observation i at position x has been emitted by states H and L respectively. The bold faced numbers show the maximum sequence probability ending in that state at each data index. We can see the most probable path for the sequence in Figure 1 and the corresponding maximum probability over all possible state sequences is $2^{-24.487} = 4.2515e - 08$.



```

MATLAB R2017a - academic student use
HOME PLOTS APPS EDITOR PUBLISH VIEW
C:\Windows\System32
Command Window

prompt =

    'Enter the sequence:'

Enter the sequence:GGCACTGAA

str =

    'GGCACTGAA'

most probable path:HHHLLLLLL
fx>>

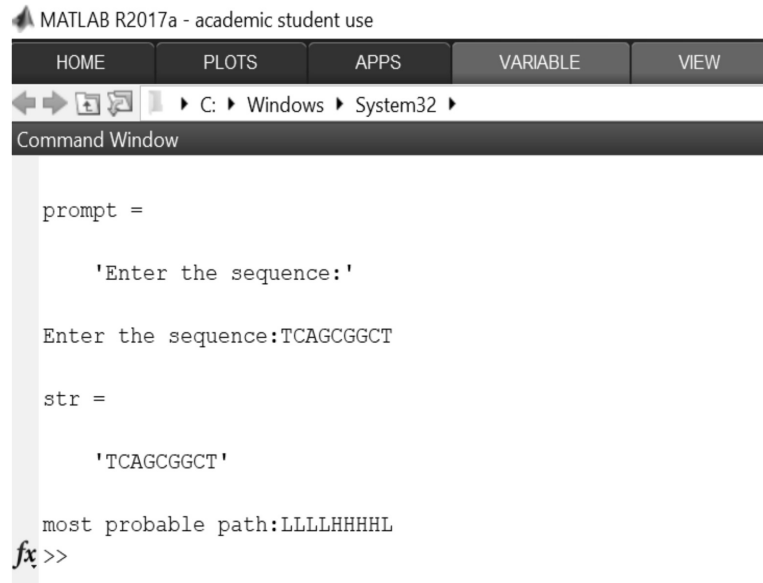
```

Figure 1: Most Probable Path for $S = \text{GGCACTGAA}$

Consider the second input sequence $S = \text{TCAGCGGCT}$, Table 2 gives the logarithm of the probabilities $P_H(i, x)$ and $P_L(i, x)$ emitted by states H and L respectively. The bold faced numbers show the maximum sequence probability ending in that state at each data index. We can see the most probable path for the sequence in Figure 2 and the corresponding maximum probability over all possible state sequences is $2^{-25.013} = 2.9524e - 08$.

H	-3.321	-5.795	-9.117	-11.328	-14.065	-16.802	-19.539	-22.276	-25.598
L	-2.736	-5.795	-8.269	-11.328	-14.387	-17.387	-20.124	-22.861	-25.013

Table 2: Probabilities emitted by states H and L



```
MATLAB R2017a - academic student use
HOME PLOTS APPS VARIABLE VIEW
C:\Windows\System32
Command Window

prompt =
    'Enter the sequence:'

Enter the sequence:TCAGCGGCT

str =
    'TCAGCGGCT'

most probable path:LLLLHHHHL
fx>>
```

Figure 2: Most Probable Path for S= TCAGCGGCT

4 Conclusion

The Viterbi Algorithm has become one of the most widely used algorithm in fields like digital communication systems, speech recognition, computational linguistics and bioinformatics. However its drawback is that for longer constraints, the computational costs grow exponentially as each state has to be enumerated. Other decoding methods such as BCJR or Fano sequential decoding algorithm are preferred in these cases.

5 Appendix

```
clc
clear
close all
prompt='Enter the sequence:'
str = input(prompt,'s')

%State Transitional Matrix
A=[0.5 0.5;0.4 0.6];

%Initial Probabilities
p=[0.5;0.5];

%Discrete Emission Probabilities
```

```

B=[0.2 0.3 0.3 0.2;0.3 0.2 0.2 0.3];

%Log of Probabilities
A=log2(A);
B=log2(B);
p=log2(p);

t=length(str);
%Initializing Best Probabilities
prob=zeros(1,t);
prob=zeros(2,t);
BestProb=zeros(1,t);
for t=1:t
    if strcmp(str(t),'A')
        Code(1,t)=1;
    elseif strcmp(str(t),'C')
        Code(1,t)=2;
    elseif strcmp(str(t),'G')
        Code(1,t)=3;
    elseif strcmp(str(t),'T')
        Code(1,t)=4;

    end

    if t==1
        prob(1,t)=p(1,1)+B(1,Code(1,t));
        prob(2,t)=p(2,1)+B(2,Code(1,t));
        BestProb(1,t)=max(prob(1,t),prob(2,t));

    else
        prob(1,t)=B(1,Code(1,t))+max(prob(1,t-1)+A(1,1),prob(2,t-1)+A(2,1));
        prob(2,t)=B(2,Code(1,t))+max(prob(1,t-1)+A(1,2),prob(2,t-1)+A(2,2));
        BestProb(1,t)=max(prob(1,t),prob(2,t));

    end

end

%Backtracking to find Most Probable Path
fprintf("most probable path:")

for t=t:-1:1
    if prob(1,t)>prob(2,t)
        State(1,t)="H";
    end
end

```

```
elseif prob(1,t)==prob(2,t)
    if prob(1,t-1)>prob(2,t-1)
        State(1,t)="H";
    else
        State(1,t)="L";
    end
else
    State(1,t)="L";
end
end

fprintf("%s",State)
fprintf("\n")
```