

Abin assignment -4

1) well commented code submitted

output->

```
Menu:
1: Run Q1 Algorithm
2: Exit
Enter your choice: 1

a) Probability of Hidden States for all three steps:
Healthy: [0.3, 0.084, 0.00588]
Fever: [0.04000000000000001, 0.027, 0.01512]

b) Highest probability at step 3 from where we start backtracking:
0.01512

c) Most probable set of hidden states for the above-given input:
['Healthy', 'Healthy', 'Fever']

Menu:
1: Run Q1 Algorithm
2: Exit
Enter your choice: 2
Exiting program.
```

Viterbi algorithm is used to find the most likely sequence of hidden states (in this case, "Healthy" or "Fever") given a sequence of observations (like "normal," "cold," "dizzy"). It does this in four main steps:

Initialization: It starts with the initial probabilities of being in each state and computes the probability of the first observation for each state.

Recursion: It iteratively calculates the highest probability for each subsequent state based on the previous state's probability, the transition probability between states, and the emission probability of the observation.

Termination: It identifies the state with the highest probability at the final step.

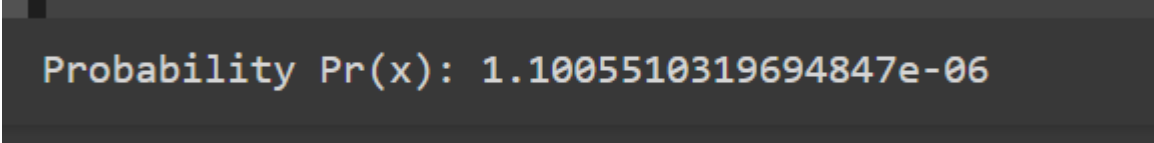
Path Backtracking: It traces back the path of states that led to the highest probability.

When you run the provided code and choose to run the algorithm, it will display:

The probabilities of being in each state at each step.

The overall highest probability at the last step.

The sequence of hidden states that is most likely given the observations (the Viterbi path).



```
Probability Pr(x): 1.1005510319694847e-06
```

2)

The `hmm_forward_probability` function calculates the likelihood of observing a given sequence (sequence) from a Hidden Markov Model (HMM) with specified states (states), initial probabilities (initial_probabilities), transition probabilities (transition_matrix), and emission probabilities (emission_matrix). It initializes a matrix to hold the forward probabilities and then populates it using a dynamic programming approach, iterating over the sequence and computing probabilities based on previous states' contributions and current emission probabilities. The final probability of the sequence is the sum of the last set of state probabilities.

The `Q2` function serves as a wrapper that sets up the inputs and calls `hmm_forward_probability` to get the sequence probability.

3)

js (text, theta, sigma, alignment, alphabet)

Path: M1 D2 D3 M4 M5 I5 M6 M7 M8

M stands for matches, D stands for deletion, I stand for Insertions, and the numbers beside it represents the number of residues which need to be matched or need to be inserted or deleted

And these are the operations performed after multiple sequence alignment of the given sequence in the test case using a profile HMM

The above terms mean the following

Multiple Sequence Alignment (MSA): MSA is the process of aligning and comparing multiple sequences of biological macromolecules (such as proteins or nucleic acids) to identify conserved regions and similarities among them. The goal of MSA is to create a representation of these sequences where corresponding positions are lined up, allowing for the identification of conserved residues and functional motifs.

Profile Hidden Markov Model (Profile HMM): A Profile HMM is a probabilistic model that represents the consensus or profile of a multiple sequence alignment. It captures the statistical properties of the aligned sequences, including the probabilities of specific amino acids or nucleotides occurring at each position in the alignment. A Profile HMM is a versatile tool for detecting homologous sequences, identifying functional domains, and annotating sequences with structural and functional information.