Assignment 1

Vedanshu Goyal
B19CSE096

Compile : **g++ -std=c++17**

a)

I have implemented the Baum-Welch algorithm to solve this problem. The Baum-Welch algorithm, also known as the Forward-Backward algorithm, is a dynamic programming algorithm used in the estimation of parameters for hidden Markov models (HMMs).

Firstly I InitalizeModel or _a and _b matrix where
_a[i][j] -> Probability for state i to state j
_b[i][j] -> Probability of observation j in state i
I Initialize this Matrix with random probabilities and conditions given in the questions.

Then I implement Alpha and Beta Matrix, with forward and backward Algorithms.
Then I use the Baum-Welch algorithm, to update the probabilities;

b)
For Part b), We can directly use the Alpha Matrix and then accumulate the result of the last column.

c)

For Part c), I use the dynamic programming algorithm with backtracking to generate the answer, like VITERBI ALGORITHM

$$v_t(j) = \max_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t)$$

function VITERBI(*observations* of len *T*,*state-graph* of len *N*) **returns** *best-path*, *path-prob*

create a path probability matrix *viterbi[N,T]*
for each state *s* **from** 1 **to** *N* **do** ; initialization step
viterbi[*s*,1] $\leftarrow \pi_s * b_s(o_1)$
backpointer[s,1] $\leftarrow 0$
for each time step *t* **from** 2 **to** *T* **do** ; recursion step
for each state *s* **from** 1 **to** *N* **do**
viterbi[s,t] $\leftarrow \max_{s'=1}^{N}$ *viterbi*$[s', t-1] * a_{s',s} * b_s(o_t)$
backpointer[s,t] $\leftarrow \operatorname{argmax}_{s'=1}^{N}$ *viterbi*$[s', t-1] * a_{s',s} * b_s(o_t)$
bestpathprob $\leftarrow \max_{s=1}^{N}$ *viterbi*$[s,T]$; termination step
bestpathpointer $\leftarrow \operatorname{argmax}_{s=1}^{N}$ *viterbi*$[s,T]$; termination step
bestpath $\leftarrow$ the path starting at state *bestpathpointer*, that follows backpointer[] to states back in time
return *bestpath*, *bestpathprob*