# CSE 494/559: Algorithms in Computational Biology - Assignment 5

Fall 2023

**Instructor:** Heewook Lee **Due Date:** **11/17/2023 11:59pm**

Hidden Markov Models

**Note:** that any program you implement should run within **5 mins (wall time)** to produce an output to be satisfactory.

## Hidden Markov Models

**Note: Assume initial probabilities are equally likely. Remember to compute probabilities in log space to prevent any underflow errors resulting in probabilities of zero.**

1. [**Viterbi Algorithm - 60pt**] Implement the Viterbi Algorithm for finding the most likely sequence of hidden states $\pi$, where $Pr(x, \pi)$ is maximized.

   **Input:** An observed sequence of symbols $x$, followed by alphabet $\Sigma$, followed by a list of states *states*, followed by transition matrix $T$, followed by emission matrix $E$ of an HMM $(\Sigma, states, T, E)$.

   **Output:** The sequence of hidden states that maximizes $Pr(x, \pi)$.

   **Sample input file content:**
   ```
   xyxzzxyxyy
   --------
   x y z
   --------
   A B
   --------
       A      B
   A  0.641  0.359
   B  0.729  0.271
   --------
       x      y      z
   A  0.117  0.691  0.192
   B  0.097  0.42   0.483
   ```

   **Sample output:**
   ```
   AAABBAAAAA
   ```

2. [**Parameter Estimation via Viterbi Learning - 60pt**] Implement Viterbi Learning algorithm to estimate the unknown parameters of HMM to maximize $Pr(x, \pi)$ over all possible parameter sets.

   **Input:** A number of iterations $i$, followed by an observed sequence of symbols $x$, followed by alphabet $\Sigma$, followed by a list of states *states*, followed by initial transition matrix $T$, followed by emissions matrix $E$.

   **Output:** Transition matrix followed by emission matrix that maximizes $Pr(x, \Pi)$ over all $\Pi$ and parameters (transition and emission matrices).

**Sample input file content:**

```
100
--------
xxxzyzzxxzxyzxzxyxxzyzyzyyyyzzxxxzzxzyzzzxyxzzzxyzzxxxxzzzxyyxzzzzzyzzzxxzzxxxyx
yzzyxzxxxyxzyxxyzyxz
--------
x y z
--------
A B
--------
```

| | A | B |
|---|---|---|
| A | 0.582 | 0.418 |
| B | 0.272 | 0.728 |

```
--------
```

| | x | y | z |
|---|---|---|---|
| A | 0.129 | 0.35 | 0.52 |
| B | 0.422 | 0.151 | 0.426 |

**Sample output:**

| | A | B |
|---|---|---|
| A | 0.875 | 0.125 |
| B | 0.011 | 0.989 |

```
--------
```

| | x | y | z |
|---|---|---|---|
| A | 0.0 | 0.75 | 0.25 |
| B | 0.402 | 0.174 | 0.424 |

3. [**Soft Decoding Problem - 60pt**] Instead of finding the most likely hidden path, we want to find the conditional probability $Pr(\Pi_i = k \mid x)$ that the HMM was in hidden state $k$ when emitting $i$th symbol (at step $i$).

   **Input:** An observed sequence of symbols $x = x_1 \ldots x_n$, followed by alphabet $\Sigma$, followed by a list of states *states*, followed by transition matrix $T$, followed by emission matrix $E$ of an HMM $(\Sigma, states, T, E)$.

   **Output:** All conditional probabilities $Pr(\Pi_i = k \mid x)$ for each state k and each step $i$ (1 to $n$)

   **Sample input file content:**

```
yzzzyxzxxx
--------
x y z
--------
BBABABABAB
--------
A B C
```

**Sample output:**

| | A | B | C |
|---|---|---|---|
| A | 0.0 | 1.0 | 0.0 |
| B | 0.8 | 0.2 | 0.0 |
| C | 0.333 | 0.333 | 0.333 |

```
--------
```

| | x | y | z |
|---|---|---|---|
| A | 0.25 | 0.25 | 0.5 |
| B | 0.5 | 0.167 | 0.333 |
| C | 0.333 | 0.333 | 0.333 |

4. **[Baum-Welch Learning for HMM - 60pt] (OPTIONAL for CSE494)** Implement Baum-Welch algorithm to learn the unknown parameters of HMM. You will need soft decoding of both nodes and edges in Viterbi graph (node soft decoding done in the previous problem).

   **Input:** A number of iterations $i$, followed by an observed sequence of symbols $x$, followed by alphabet $\Sigma$, followed by a list of states *states*, followed by initial transition matrix $T$, followed by emissions matrix $E$.

   **Output:** Transition matrix followed by emission matrix that maximize $Pr(x, \Pi)$ over all possible $\Pi$ and parameters (transition and emission matricies).

   **Sample input file content:**

```
10
--------
xzyyzyzyxy
--------
x y z
--------
A B
--------
     A      B
A  0.019  0.981
B  0.668  0.332
--------
     x      y      z
A  0.175  0.003  0.821
B  0.196  0.512  0.293
```

   **Sample output:**

```
     A      B
A  0.000  1.0
B  0.786  0.214
--------
     x      y      z
A  0.242  0.0    0.758
B  0.172  0.828  0.0
```