Week10 - HMM

DS3010: Introduction to Machine Learning Lab

Timing: 02:00 PM to 04:30 PM Max Marks: 5

Instructions

1. Submit one .ipynb file containing all answers. The name should be [student_name] _[rollno]_[lab].ipynb
2. Write the questions in separate text blocks before the answers.
3. Outputs for all sub-questions should be given and the code should be executable.
4. Write justifications for your choices where needed.
5. Ensure that all plots include clear labels and legends for better interpretation.
6. Use of generative AI tools (such as ChatGPT, Gemini, etc.) is strictly prohibited. Any submission found to contain AI-generated or plagiarized content will receive a score of zero, and disciplinary action.

Your goal is to **model daily weather patterns** using a **Hidden Markov Model (HMM)**, where the *true weather state* (Sunny, Cloudy, Rainy) is **hidden**, and the *sensor observation* (Dry, Damp, Wet) is **visible**. Use the `hmmlearn` library to complete the following tasks:

1. Model Definition (0.5 Marks)

1. Define and initialize the HMM parameters as follows:
   - Hidden states = ["Sunny", "Cloudy", "Rainy"]
   - Observations = ["Dry", "Damp", "Wet"]

   Transition matrix (A):
   $$A = \begin{bmatrix} 0.75 & 0.20 & 0.05 \\ 0.25 & 0.50 & 0.25 \\ 0.05 & 0.25 & 0.70 \end{bmatrix}$$
   Emission matrix (B):
   $$B = \begin{bmatrix} 0.70 & 0.25 & 0.05 \\ 0.30 & 0.50 & 0.20 \\ 0.05 & 0.25 & 0.70 \end{bmatrix}$$
   Initial probabilities (π) = [0.6, 0.3, 0.1]

2. HMM Construction (0.5 Marks)

1. Construct the HMM using the `MultinomialHMM` class from `hmmlearn`.
2. Use `n_components = 3`, `random_state = 42`, and `n_trials = 1` .
3. Initialize the model with the transition, emission, and start probability matrices.
4. Print and verify that all parameters were set correctly.

3. Sequence Probability (1 Marks)

Given the observation sequence:

$$O = [Dry, Damp, Wet, Wet, Damp]$$

1. Convert the observation names to numeric form and apply **one-hot encoding**.
2. Compute the **log-likelihood** of this sequence under the model using the **Forward algorithm** (`model.score()`).
3. Interpret what the resulting log-likelihood value indicates about how well the model explains the sequence.

4. Viterbi Decoding (1 Marks)

1. Use the **Viterbi algorithm** (`model.decode()`) to find the most probable sequence of hidden states for the same observation sequence.
2. Print the mapping as:

```
Observation → Most likely hidden state
Dry → Sunny
Damp → Cloudy
Wet → Rainy
...
```

3. Explain how the Forward and Viterbi algorithms differ in their purpose.

5. Simulation (1 Marks)

1. Simulate a 12-day weather sequence using `model.sample()`.
2. Display both the generated *observations* and *hidden states*.
3. Plot a bar chart of the simulated observations.
4. Comment on whether the simulation seems realistic based on transition probabilities.

6. Stable Weather Model (1 Marks)

Create a second "stable" weather model where states change less frequently using:

$$A_{stable} = \begin{bmatrix} 0.90 & 0.08 & 0.02 \\ 0.10 & 0.80 & 0.10 \\ 0.02 & 0.08 & 0.90 \end{bmatrix}$$

1. Compute and compare the log-likelihood of the given observation sequence under both models.
2. Discuss which model fits the data better and why.