Let's break down the \*\*Hidden Markov Model (HMM)\*\* with a simple example.

### Scenario: Weather Prediction

Imagine you're trying to predict the weather (hidden states) based on whether your friend carries an umbrella (observable data). You don't directly know the weather each day, but you can see whether your friend carries an umbrella.

#### Hidden States (Weather):

- \*\*Sunny\*\* (S)

- \*\*Rainy\*\* (R)

You don’t directly observe the weather, but you assume it is in one of these states each day.

#### Observations:

- \*\*Umbrella\*\* (U)

- \*\*No Umbrella\*\* (N)

Your friend carries an umbrella on rainy days, but sometimes also on sunny days (just to be cautious). You only observe whether they carry an umbrella, not the actual weather.

### Components of HMM in this Example:

1. \*\*State Transition Probabilities (A)\*\*:

- These probabilities tell us the likelihood of the weather changing from one state to another.

- Example:

- P(Sunny → Sunny) = 0.7, P(Sunny → Rainy) = 0.3

- P(Rainy → Rainy) = 0.6, P(Rainy → Sunny) = 0.4

- If it’s sunny today, there’s a 70% chance it’ll be sunny tomorrow and a 30% chance it’ll rain.

2. \*\*Emission Probabilities (B)\*\*:

- These probabilities tell us how likely the observation (umbrella or no umbrella) is given the hidden state (weather).

- Example:

- P(Umbrella | Sunny) = 0.1, P(No Umbrella | Sunny) = 0.9

- P(Umbrella | Rainy) = 0.8, P(No Umbrella | Rainy) = 0.2

- If it’s rainy, your friend is much more likely (80%) to carry an umbrella.

3. \*\*Initial State Distribution (π)\*\*:

- This is the probability of the weather on the first day.

- Example:

- P(Sunny on Day 1) = 0.6, P(Rainy on Day 1) = 0.4

### Problem 1: \*\*Evaluation\*\*

You observe your friend carrying an umbrella for three consecutive days: \*\*U, U, N\*\*. What's the probability that this sequence was generated by the HMM?

- Use the \*\*forward algorithm\*\* to calculate the likelihood of this sequence given the HMM. It computes the probabilities for each possible hidden state path that could generate the observed sequence (umbrella, umbrella, no umbrella).

### Problem 2: \*\*Decoding\*\*

Now you want to know the most likely sequence of weather (hidden states) given the observation of your friend carrying or not carrying an umbrella: \*\*U, U, N\*\*.

- Use the \*\*Viterbi algorithm\*\* to find the most likely hidden state sequence (weather):

- Day 1: Rainy (likely because your friend carried an umbrella).

- Day 2: Rainy (again, because your friend carried an umbrella).

- Day 3: Sunny (because your friend didn't carry an umbrella).

### Problem 3: \*\*Learning\*\*

If you don’t know the transition and emission probabilities, you can use the \*\*Baum-Welch algorithm\*\* to estimate these from a large set of observed sequences (whether your friend carried an umbrella or not) and learn the best parameters for predicting the hidden states.

---

### Summary of the Example:

- \*\*Hidden States (Weather)\*\*: Sunny, Rainy

- \*\*Observations (Umbrella)\*\*: Umbrella, No Umbrella

- \*\*Goal\*\*: Use the observations (your friend’s umbrella) to infer the hidden states (weather).

HMM helps us model this kind of problem, where the actual underlying state is hidden but we can make educated guesses based on observed data.

A \*\*Hidden Markov Model (HMM)\*\* is a statistical model used to represent systems that are governed by hidden (unobservable) states. It’s widely used in machine learning, especially for sequential data analysis like speech recognition, time series prediction, and natural language processing.

### Key Concepts of HMM:

1. \*\*States\*\*:

- The system is assumed to be in one of a finite number of hidden states at any given time.

- Transitions between states follow a set of probabilities, known as the \*\*state transition probabilities\*\*.

2. \*\*Observations\*\*:

- Each hidden state produces an observable output, and the likelihood of observing a particular output depends on the hidden state.

- The observations are related to states through the \*\*emission probabilities\*\*.

3. \*\*Markov Property\*\*:

- The probability of transitioning to the next state depends only on the current state, not on the sequence of states that preceded it.

4. \*\*Key Probabilities\*\*:

- \*\*Transition Probability (A)\*\*: The probability of transitioning from one state to another.

- \*\*Emission Probability (B)\*\*: The probability of an observation being generated by a hidden state.

- \*\*Initial State Distribution (π)\*\*: The probability distribution of the initial state.

### Three Main Problems in HMM:

1. \*\*Evaluation Problem\*\*:

- Given a model and a sequence of observations, determine the likelihood that the sequence was generated by the model (solved using the \*\*forward-backward algorithm\*\*).

2. \*\*Decoding Problem\*\*:

- Given a model and a sequence of observations, find the most likely sequence of hidden states (solved using the \*\*Viterbi algorithm\*\*).

3. \*\*Learning Problem\*\*:

- Given a sequence of observations, learn the most likely model parameters (transition and emission probabilities). This is typically solved using the \*\*Baum-Welch algorithm\*\*, a form of the Expectation-Maximization (EM) algorithm.

### Applications of HMM:

- \*\*Speech Recognition\*\*: Mapping sequences of audio signals to sequences of words.

- \*\*Natural Language Processing (NLP)\*\*: Part-of-speech tagging and named entity recognition.

- \*\*Bioinformatics\*\*: Gene prediction and protein structure analysis.

- \*\*Finance\*\*: Modeling stock prices and market trends.

HMMs are particularly useful in scenarios where you have observable data but the underlying process generating the data is hidden or partially observable.