**TASK 11**:Write a program to construct a Hideenmarkov model for a medical data. Use this model to demonstrate the diagnosis of medial data.

**Tools: Google co-lab, Python, Scikitlearn, Anaconda navigator**

**Algorithm (HMM):**

1. Define the hidden states as diseases like Flu and Cold.
2. Define the observable symptoms like Fever, Cough, and Normal.
3. Set initial probabilities for starting with each disease.
4. Set transition probabilities between diseases.
5. Set emission probabilities of symptoms given each disease.
6. Input observed symptoms and use HMM to predict the most likely disease sequence.

**CODE:**

# Step 1: Install hmmlearn (if not already installed)

!pip install hmmlearn --quiet

# Step 2: Import libraries

import numpy as np

from hmmlearn import hmm

# Step 3: Define hidden states (diseases)

states = ["Flu", "Cold"]

n\_states = len(states)

# Step 4: Define observable symptoms

observations = ["Fever", "Cough", "Normal"]

n\_observations = len(observations)

# Step 5: Create the HMM model

model = hmm.MultinomialHMM(n\_components=n\_states, n\_iter=100, tol=0.01)

# Step 6: Set initial probabilities (probability of starting with each disease)

model.startprob\_ = np.array([0.6, 0.4]) # Flu more likely initially

# Step 7: Set transition probabilities (disease progression)

model.transmat\_ = np.array([

[0.7, 0.3], # Flu to Flu/Cold

[0.4, 0.6] # Cold to Flu/Cold

])

# Step 8: Set emission probabilities (symptoms given disease)

model.emissionprob\_ = np.array([

[0.6, 0.3, 0.1], # Flu: Fever, Cough, Normal

[0.2, 0.5, 0.3] # Cold: Fever, Cough, Normal

])

# Step 9: Map observations to integers

obs\_map = {"Fever": 0, "Cough": 1, "Normal": 2}

# Example observed symptoms sequence

obs\_sequence = ["Fever", "Cough", "Normal", "Fever", "Cough"]

obs\_seq\_int = np.array([[obs\_map[sym]] for sym in obs\_sequence])

# Step 10: Predict the hidden states (diseases)

logprob, hidden\_states = model.decode(obs\_seq\_int, algorithm="viterbi")

# Step 11: Print results

print("Observed Symptoms: ", obs\_sequence)

print("Predicted Disease States: ", [states[s] for s in hidden\_states])

**OUTPUT:**

Observed Symptoms: ['Fever', 'Cough', 'Normal', 'Fever', 'Cough']

Predicted Disease States: ['Flu', 'Flu', 'Cold', 'Flu', 'Flu']

**RESULT:** Thus the program was executed successfully and output was verified.