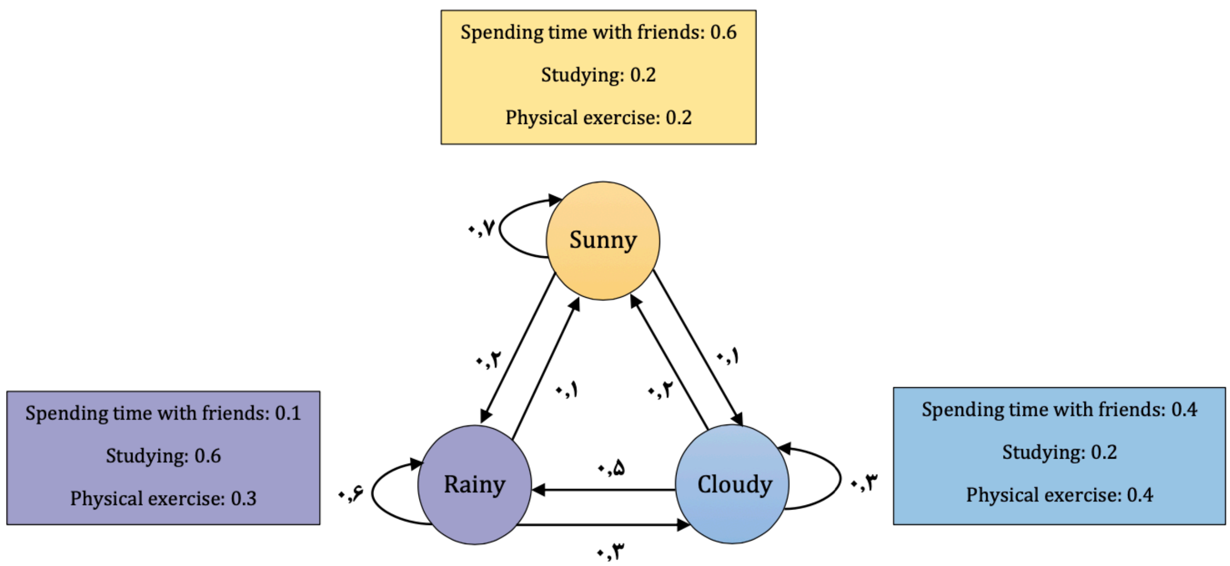
Implementation

هدف این قسمت، پیاده‌سازی و آزمایش عملی یکی از کاربردهای مدل پنهان مارکوف می‌باشد. به این منظور، دیاگرام زیر که تغییرات وضعیت هوا در شهر تهران را نمایش می‌دهد، در نظر بگیرید. فعالیت‌هایی که یک دانش‌آموز هر روز انجام می‌دهد، به وضعیت هوا وابسته است. برنامه‌ای بنویسید که با دریافت رشته‌ای از ورودی‌ها با توجه به دیاگرام حالت زیر و با استفاده از روش forward، احتمال رخداد این توالی از فعالیت‌ها را محاسبه کند. به عنوان مثال، نشان دهید احتمال این که این دانش‌آموز در سه روز متوالی درس بخواند، با دوستانش ملاقات کند و ورزش کند چقدر است؟ توجه کنید که به منظور ارزیابی، پیاده‌سازی شما باید این قابلیت را داشته باشد که با دریافت رشته‌ای از حالت‌ها (به‌جز رشته حالت‌های بالا)، احتمال پیشامد ورودی را در قالب عدد در خروجی نمایش دهد.

$$\pi_{\text{Sunny}} = \text{۰.۲} \quad \pi_{\text{Rainy}} = \text{۰.۴} \quad \pi_{\text{Cloudy}} = \text{۰.۴}$$



One of the applications of Hidden Markov chains is predicting the probability of occurrence of a sequence of observations based on the state diagram. In solving this question, we have been asked to use the Forward method to predict the probability of a student performing a sequence of activities based on weather conditions.

To solve this question, first we need to calculate the probability of occurrence of a sequence of observations in an HMM using the Forward algorithm. The Forward algorithm is a dynamic programming algorithm that uses the transition probability matrix (A), the observation probability matrix in each state (B), and the initial probability distribution vector to calculate the probability of occurrence of a sequence of observations in an HMM. The steps of this algorithm are as follows:

•Creating the Forward Table:
First, we need to create a Forward table with the size of observations over time and the possible number of states in the HMM. In this table, the element i, j indicates the probability of occurrence of a sequence of observations up to time i and to state j.

•Filling the first row of the Forward table:
In the next step, we need to fill the first row of the Forward table using the initial probability distribution vector and the observation probability matrix in each state. For each state j, the first row element of the Forward table with column number corresponding to state j is equal to the multiplication of the initial probability of this state and the probability of observing the first observation in that state.

•Filling the next columns of the Forward table:
Then, for each time i, we use the previous column of the Forward table and the transition probability matrix to calculate the elements of the new column. For each state j, the Forward table column element with a row number corresponding to state j is equal to the sum of the multiplication of the observation probability in state j and the elements of the previous Forward table column multiplied by the probability of transition from the previous state to state j.

•Calculating the probability of occurrence of a sequence of observations:
Finally, we need to calculate the sum of the elements of the last column of the Forward table to calculate the probability of occurrence of a sequence of observations in an HMM.

$$P(Y^1, Y^2, \dots, Y^t) = \sum_{i=0}^{n-1} \alpha_{t-1}(X_i)$$

Explanation of implementation functions:

1- Importing the required libraries:

```
import numpy as np
```

To calculate the transition matrices and predict the probabilities of Markov chain observations, we use the Numpy library.

2- Defining the diagram and initial states:

```
# States of the HMM
states = ("Sunny", "Cloudy", "Rainy")

# Observations or events
obs = ("Study", "Hangout", "Sports")

# Start probability of each state
start_prob = {"Sunny": 0.2, "Cloudy": 0.4, "Rainy": 0.4}

# Transition probabilities between states
trans_prob = {
    "Sunny": [0.7, 0.2, 0.1],
    "Cloudy": [0.1, 0.3, 0.3],
    "Rainy": [0.2, 0.5, 0.6]
}

# Emission probabilities of each state for each observation
emit_prob = {
    "Sunny": {"Study": 0.2, "Hangout": 0.6, "Sports": 0.2},
    "Cloudy": {"Study": 0.2, "Hangout": 0.4, "Sports": 0.4},
    "Rainy": {"Study": 0.6, "Hangout": 0.1, "Sports": 0.3}
}

# Sequence of observations(Observations may change)
input_obs = ("Study", "Hangout", "Sports")
```

First, a hidden Markov model with 3 states "Sunny", "Cloudy" and "Rainy" is defined for weather changes in Tehran. Also, 3 events or activities named "Study", "Hangout" and "Sports" are considered as observations. It is assumed that a

student's activities depend on weather conditions, and the probability of performing each activity for each state is specified.
Inside this piece of code, all the probabilities in the state diagram inside the question are specified and defined.
The default sequence of observations is studying, hanging out with friends and sports activities as mentioned in the question. The implementation is designed so that by changing the input variable or input_obs, you can predict the probability of occurrence of any sequence using the Forward method and see the predicted result at the output.

3 - Forward algorithm function:

```
def hmm_forward(observations, hidden_states, initial_state_probs, transition_probs, emission_probs):
    alpha = np.zeros((len(observations), len(hidden_states)))
    for i, state in enumerate(hidden_states):
        alpha[0, i] = initial_state_probs[state] * emission_probs[state][observations[0]]

    for t in range(1, len(observations)):
        for i, state in enumerate(hidden_states):
            alpha[t, i] = emission_probs[state][observations[t]] * np.sum(alpha[t-1] * transition_probs[state])

    sequence_prob = np.sum(alpha[-1])

    return sequence_prob, alpha
```

The hmm_forward function is implemented with inputs observations, hidden_states, initial_state_probs, transition_probs and emission_probs respectively. The inputs of this function include the sequence of observations, hidden states, the probability of starting each state, the probability of transition from one state to another, and the probability of performing each observation in each state. And is called when needed to predict the probability of occurrence of a sequence of observations. In the hidden Markov model, the Forward algorithm is one of the methods used to calculate the probability of occurrence of a sequence of observations. This algorithm seeks to calculate the probability that a sequence of observations of length t ends in state j, assuming that this probability has been calculated independently for a sequence of observations of length t-1. In this function, first an array with the size of the number of observations in the input and the number of states in the hidden Markov model is created. Then, using the probability of starting each state and the probability of performing the first observation in each state, the first value of the alpha array is filled.

$$\alpha_t(X_i) = \sum_{j=0}^{n-1} \alpha_{t-1}(X_j)\, P(X_i|X_j) P(Y^t|X_i)$$

Here, alpha [t] [j] indicates the probability that a sequence of observations of length t has ended and ended in state j. Then, in the next step, using the Forward relationship and the transition probability and the probability of observation in each state at time t, the value of each cell in the alpha array is calculated recursively. Finally, by summing the values in the last row of the alpha array, the probability of occurrence of a sequence of observations is calculated and returned as the output of the function along with the alpha matrix containing the values calculated at each step of the Forward algorithm. At the end, using the print statement, the probability of occurrence of a sequence of observations and the alpha matrix containing the values calculated at each stage of the Forward algorithm are printed.

4 - Calculating the probability of occurrence of a sequence of observations:

```
# Calculate the probability of the input sequence using the forward algorithm
prob, alpha = hmm_forward(input_obs, states, start_prob, trans_prob, emit_prob)


# Print the result
print("The probability of the input sequence {} is {:.5f}".format(input_obs, prob))


# Print the alpha matrix with state and observation axes
print("The alpha matrix is:")
print("{:<10}".format("Obs."), end="")
for s in states:
    print("{:<10}".format(s), end="")
print("")
for t in range(len(input_obs)):
    print("{:<10}".format(t+1), end="")
    for i in range(len(states)):
        print("{:<10.5f}".format(alpha[t,i]), end="")
    print("")
```

This part of the code is used to use the hmm_forward function and display the results.

In the first line, the hmm_forward function is called with the given inputs and two values prob and alpha are returned as the output of the function. Prob indicates the probability of occurrence of the input sequence of observations calculated

using the Forward algorithm and alpha is an array containing alpha [t] [j] values for each time t and each state j calculated using the Forward algorithm.
The prob value, which is the probability of occurrence of a sequence of observations, is printed in the output.
Also, the alpha array containing the alpha [t] [j] values for each time t and each state j is displayed to the user as a matrix. This matrix contains alpha values calculated using the Forward algorithm for the input sequence of observations. The values of this matrix indicate the probability of occurrence of a sequence of observations of length t given state j at time t. The rows of the matrix also indicate time t and its columns indicate the different states of the hidden Markov model.
You can see the results obtained at each stage in Review of Results.

Review and analysis of the obtained results

In this section, we will review and analyze the results obtained from implementing the models. We first provided an overview of the concepts and solved the descriptive questions related to the hidden Markov model, and then explained the implementation section.

به عنوان مثال، نشان دهید احتمال این که این دانش‌آموز در سه روز متوالی درس بخواند، با دوستانش ملاقات کند و ورزش کند چقدر است؟ توجه کنید که به منظور ارزیابی، پیاده‌سازی شما باید این قابلیت را داشته باشد که با دریافت رشته‌ای از حالت‌ها (به‌جز رشته حالت‌های بالا)، احتمال پیشامد ورودی را در قالب عدد در خروجی نمایش دهد.

As you can see in the implementation section, functions related to the hidden Markov model for predicting a student's activities were implemented. The state diagram and initial probabilities were entered into the implementation section, and by default, the probability of the sequence of activities of studying, meeting friends, and doing sports was predicted. You can see the output of this section in the image provided.

```
The probability of the input sequence ('Study', 'Hangout', 'Sports') is 0.02834
```

Additionally, the alpha matrix is also displayed to provide more information about the intermediate states.

```
The alpha matrix is:
Obs.      Sunny      Cloudy     Rainy
1         0.04000    0.08000    0.24000
2         0.04080    0.04000    0.01920
3         0.00770    0.00874    0.01190
```

Furthermore, as mentioned in the implementation section, it is possible to change the input sequence by modifying the input_obs variable. By changing this variable and rerunning the implemented code, you can observe the prediction result for that observation sequence.