Bài thực hành buổi 2

Hidden Markov Model với

Particle Swarm Optimization (PSO)

Sinh viên thực hiện:

Nguyễn Gia Bách \_Mã SV: 2251262579\_Lớp: 64TTNT1

Github: <https://github.com/b2004x/BTH2_NguyenGiaBach_2251262579_64TTNT1>

1. Lý thuyết ngắn gọn về Hidden Markov Model (HMM) và Particle Swarm Optimization

1.1 Giới thiệu về Hidden Markov Model (HMM)

HMM là phần mở rộng của mô hình Markov được ứng dụng vào nhiều lĩnh vực như Phân tích giọng nói((Jelinek et al., 1975; Rabiner, 1989), nhận diện chữ viết (Kosmala et al., 1999; Kundu and Bahl, 1988; Olivier et al., 1993) hay nhận diện hình ảnh (Samaria and Harter, 1994)

1.1.1 Cách hoạt động

HMM được định nghĩa bởi bộ tham số:

λ =(A,B,π)

N: Trạng thái ẩn là các biến cơ bản tạo ra dữ liệu quan sát được, nhưng chúng không thể quan sát trực tiếp được.

M: ký hiệu quan sát là những biến dữ liệu có thể được đo lường và quan sát

A = {aij} : Ma trận chuyển trạng thái,với:

aij​ = P(qt+1​=j ∣ qt​=i)

Xác suất chuyển trạng thái từ trạng thái ẩn i sang trạng thái ẩn j tại thời điểm t+1

B = {bj(k)} : xác suất phát xạ, với:

bj​(k) = P( ot​=vk​ ∣ qt​=j )

Xác suất quan sát ký hiệu vk tại thời điểm t, khi hệ thống ở trạng thái ẩn j

π = {πi​}: xác suất khởi đầu, với:

πi​ = P(q1 ​= i)

O = { o1,o2,…,oT} : Chuỗi quan sát

Q = {q1,q2,…,qT} : Chuỗi trạng thái ẩn

Để tính trạng thái ẩn dựa trên chuỗi quan sát trong mô hình Hidden Markov Model (HMM), ta cần giải bài toán:

Tìm chuỗi trạng thái ẩn Q = {q1,q2,…,qT} tối ưu, biết chuỗi quan sát O = { o1,o2,…,oT} và tham số mô hình λ = (A,B, π)

Các bước giải HMM bằng thuật toán Viterbi:

Bước 1: Khởi tạo (t = 1):

Bước 2: Đệ quy (t = 2 đến t=n):

Bước 3: Kết thúc (t = T)

Bước 4: Truy vết ngược(Backtracking)

1.2 Giới thiệu về Particle Swarm Optimization(PSO)

Particle Swarm Optimization là một thuật toán tối ưu heuristic mô phỏng theo hành vi bầy đàn của động vật như đàn chim hoặc đàn cá. Thuật toán lần đầu được đề cập đến do Kennedy and Eberhart (1995) như là một thuật toán tối ưu lấy cảm hứng từ xã hội học. Thuật toán này đã được ứng dụng thành công vào nhiều vấn đề khác nhau (Abido, 2002; Hu et al., 2003; van der Merwe and Engelbrecht, 2003; Omran et al., 2002; Paquet and Engelbrecht, 2003; Salman et al., 2003; Tasgetiren et al., 2004a; Tasgetiren et al., 2004b; Wachowiak et al., 2004).

1.2.1 Cách hoạt động:

Xem xét một bầy đàn gồm n hạt và một hàm mục tiêu f, được xác định trên không gian tìm kiếm S, mục tiêu tìm được một hàm tối ưu toàn cục. Với mỗi hạt i, tại thời điểm t, ta xác định :

- là vị trí trong không gian tìm kiếm S

- vận tốc của hạt

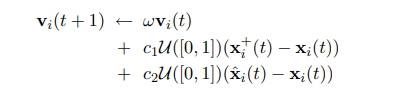
- vị trí tối nhất được tìm bởi hạt tính từ lúc bắt đầu

- vùng lân cận của hạt i tại thời điểm t. Nó tương ứng với tập hợp các hạt V gần nhất trong không gian tìm kiếm theo khoảng cách d, là Euclid trong hầu hết các trường hợp,

- the best personal position of particles from

U([0, 1]) là một giá trị ngẫu nhiên đồng đều được chọn ngẫu nhiên trong khoảng [0, 1] và ba tham số ω, c1 và c2 kiểm soát hệ thống hạt. ω là trọng số quán tính tỷ lệ với vận tốc bước thời gian trước đó, c1 và c2 là hai hệ số gia tốc tỷ lệ vớiảnh hưởng của giải pháp cá nhân tốt nhất của hạt (x + i (t)) và vị trí cá nhân tốt nhất trong vùng lân cận (Vi(t)). Tại mỗi lần lặp, chuyển động của hạt được tính như sau:

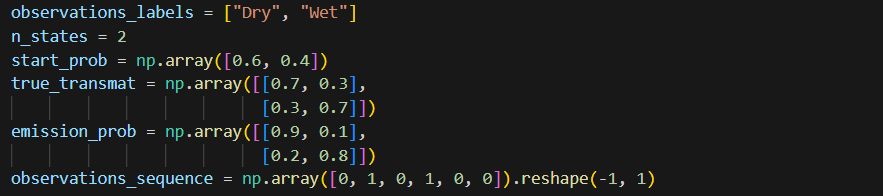
Vận tốc được update bằng công thức :



Nếu kích thước của vùng lân cận được đặt thành V < n (số lượng hạt) thì được gọi là lbest PSO (tốt nhất cục bộ). Nếu không (tức là V = n), thuật toán được gọi là gbest PSO (tốt nhất toàn cục)

2. Mô tả công việc đã làm

2.1 Tạo dữ liệu mô phỏng:



- state = [“Sunny”, “Rainy”] : Trạng thái ẩn của dữ liệu

- Observations\_labels = [“Dry”, “Wet”] : Dữ liệu quan sát

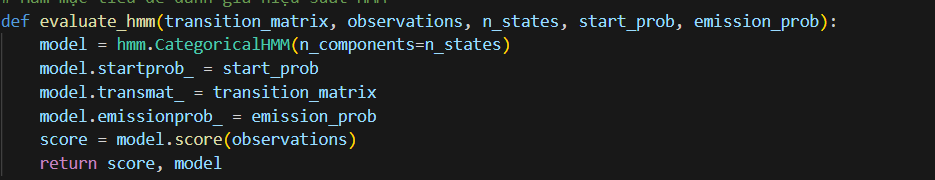
- n\_states = 2 : 2 trạng thái ẩn

- true\_transmat: Ma trận chuyển trạng thái

- emission\_Prob: Ma trận xác suất phát xạ

- observations\_sequence: Chuỗi quan sát

2. 2 Xây dựng hàm đánh giá HMM



- Nhận ma trận chuyển trạng thái, dữ liệu quan sát, và số trạng thái.-

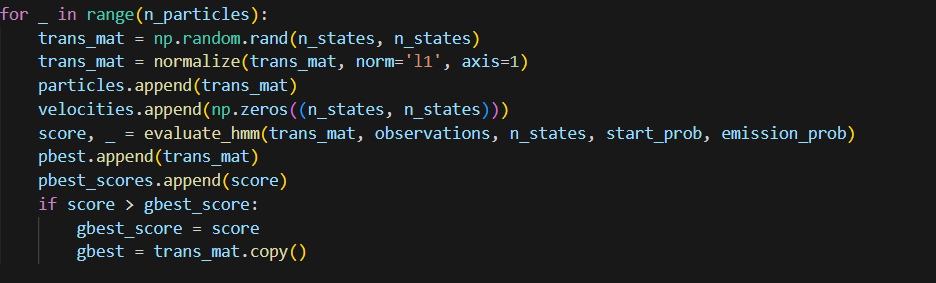
- Tạo mô hình HMM với ma trận chuyển trạng thái , xác suất bắt đầu, trung bình và phương sai

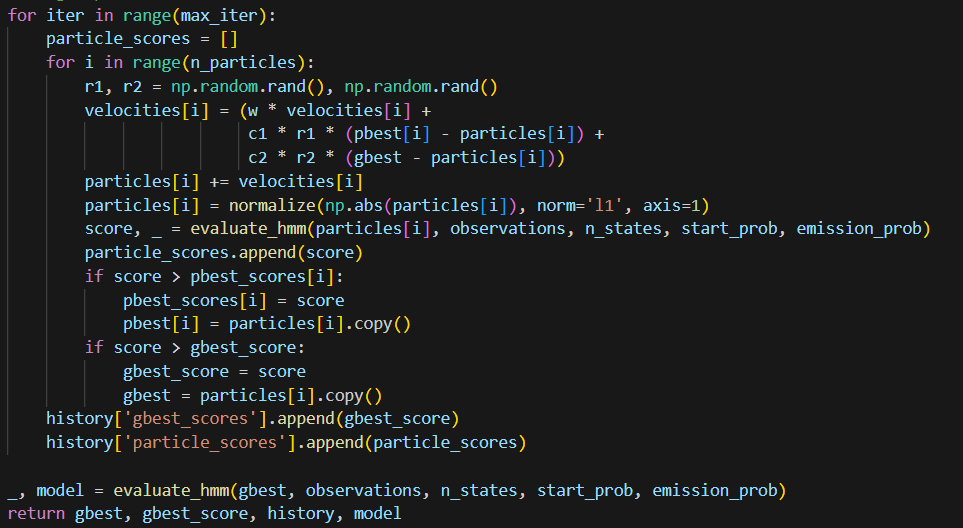
- Tính log-likelihood của dữ liệu bằng score

- Trả về điểm số và mô hình HMM

Mục đích : Đánh giá chất lượng ma trận chuyển trạng thái và dùng làm hàm mục tiêu PSO

2.3 Triển khai thuật toán PSO



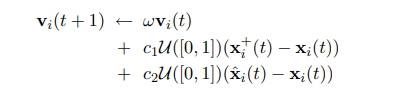


- Khởi tạo đàn hạt với 30 hạt, mỗi hạt là ma trận chuyển trạng thái 3x3 ngẫu nhiên, chuẩn hóa tổng xác xuất mỗi hàng bằng 1 (Normalize)

- Khởi tạo vận tốc bằng ma trận 0

- Đánh giá từng hạt để xác định pbest và gbest

- Tối ưu hóa (100 vòng lặp), mỗi vòng lặp cập nhập vận tốc theo công thức



với w = 0,5, c1 = c2 = 1.5, r1, r2 ngẫu nhiên

- Cập nhật vị trí: x = x + v, rồi chuẩn hóa ma trận.

- Đánh giá hạt mới, cập nhật pbest và gbest nếu điểm số cải thiện

2.4 Trực quan hóa kết quả

- Biểu đồ hội tụ:

Vẽ đường gbest\_scores qua các lần lặp để thể hiện sự cải thiện log-likelihood.

- So sánh ma trận chuyển trạng thái:

Vẽ heatmap của ma trận thực và ma trận tối ưu bằng imshow (matplotlib), với thanh màu thể hiện xác suất.

- Phân tán điểm số hạt:

Vẽ scatter plot cho điểm số của các hạt tại các lần lặp (0, 20, 40, 60, 80).

- Vẽ dữ liệu quan sát, tô màu các điểm theo trạng thái dự đoán.

Mục đích: Trực quan hóa quá trình tối ưu hóa, độ chính xác của ma trận, và khả năng phân loại trạng thái.

3. Quá trình huấn luyện

3.1 Input và Output

Quá trình huấn luyện sử dụng PSO để tối ưu cho ma trận chuyện trạng thái của HMM, nhằm có thể đạt được log-likelihood cao nhất cho chuỗi quan sát [Dry, Wet, Dry, Wet, Dry, Dry].

Input :



- Chuỗi quan sát [0,1,0,1,0,0] với (0 = Dry, 1 = Wet)



- 2 trạng thái ẩn : Sunny và Rainy



- Xác suất ban đầu [0.6, 0.4]



- Xác suất phát sinh [[0.9, 0.1], [0.2, 0.8]]

Output:

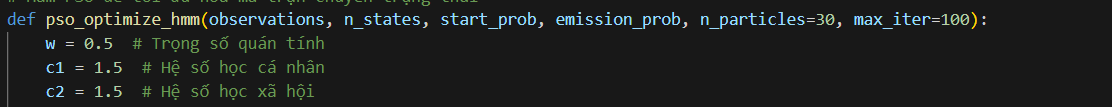
- Ma trận chuyển trạng thái tối ưu.

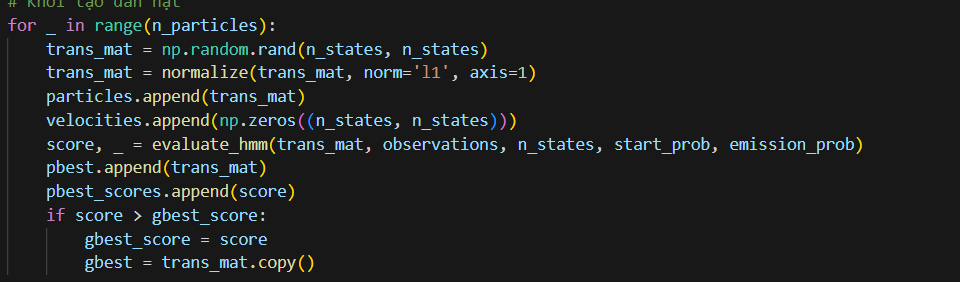
- Mô hình HMM với ma trận tối ưu, dùng để dự đoán trạng thái ẩn.

- Lịch sử tối ưu hóa và biểu đồ trực quan.

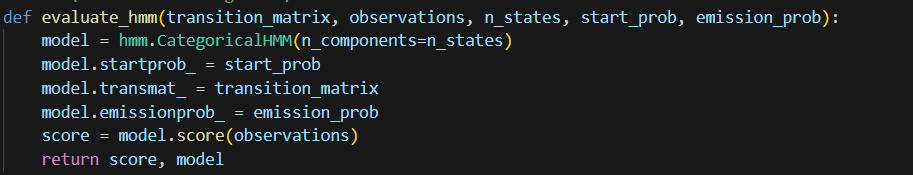
3.2 Quá trình trainning

Bước 1 : Khởi tạo



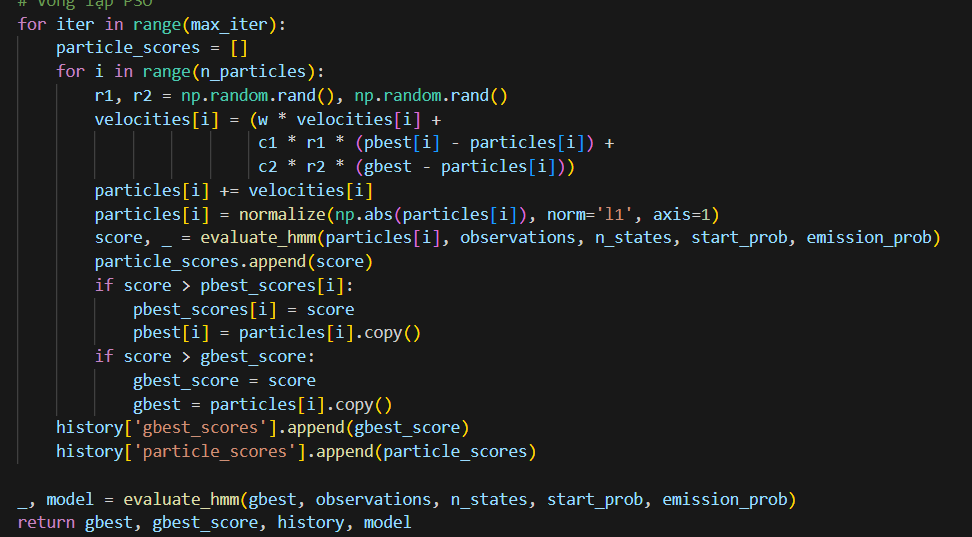


Tạo một đàn gồm 30 hạt (n\_particles = 30), mỗi hạt là một ma trận chuyển trạng thái 2x2 ngẫu nhiên (cho 2 trạng thái Sunny, Rainy). Sau đó chuẩn hóa ma trận dùng (normalize) để đảm bảo xác suất mỗi hàng bằng 1. Khởi tạo vận tốc mỗi hạt bằng ma trận 0

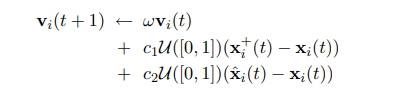


Tạo mô hình HMM với dữ liệu vào là ma trận chuyển trạng thái, xác suất ban đầu và xác xuất phát sinh. Tính log-likelihood cho chuỗi quan sát với score, lưu điểm tốt nhất cục bộ (pbest) và toàn cục(gbest)

Bước 2: Tối ưu hóa bằng POS



Lặp với max\_inter = 100 để cập nhận đàn hạt. Mỗi lần lặp cập nhập vận tốc hạt với công thức



trọng số tuyến tính(w) = 0.5, hệ số cục bộ c1 = 1.5, hệ số toàn cục c2 = 1.5, r1,r2 thuộc [0,1]

Cập nhập vị trí mới của hạt x = x + v, chuẩn hóa ma trận vị trí đảm bảo tổng xác suất bằng 1

Tạo mô hình HMM với ma trận chuyển trạng thái mới, tính log-likelihood cho chuỗi quan sát, cập nhập pbest nếu pbest tốt hơn pbest trước đó, tương tự với gbest. Lưu lại gbest\_score(tốt nhất toàn cục) và particle\_score(điểm số tất cả các hạt) cho mỗi lần lặp

Đạt được ma trận chuyển trạng thái tối ưu bằng cách điều chỉnh các hạt sau mỗi lần lặp

Bước 3: Truyền tham số vào Model



Sau 100 vòng lặp chọn ma trận chuyển trạng thái tốt nhất, truyền vào model HMM với ma trận chuyển trạng thái tối ưu và các biến khởi tạo ban đầu.

Bước 4: Sử dụng Model đã trainning để dự đoán trạng thái ẩn

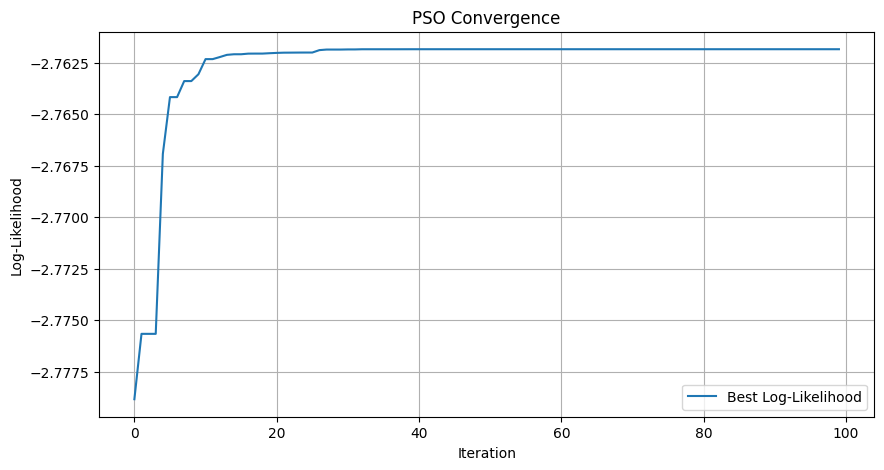


Sử dụng optimized\_model.predict(observations\_sequence) để dự đoán trạng thái ẩn cho chuỗi [0, 1, 0, 1, 0, 0]. Chuyển đổi các trạng thái (0, 1) thành nhãn Sunny, Rainy.

4. Mô tả kết quả đạt được

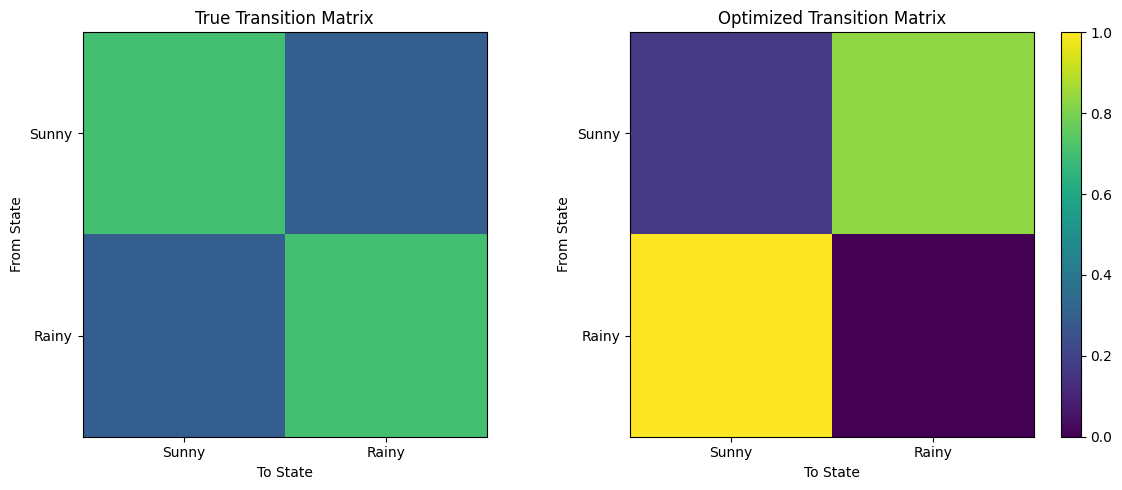
- Ma trận tối ưu sau khi sử dụng thuật toán POS

-Log-likelihood: Giá trị cao hơn so với các ma trận ngẫu nhiên ban đầu, cho thấy mô hình phù hợp hơn với dữ liệu.

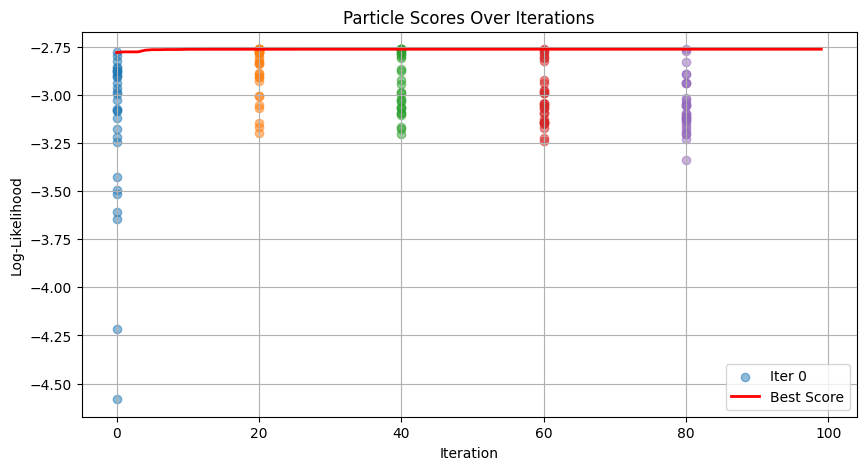


Log-likelihood tăng lên trong những lần lặp đầu tiên từ lần 0 đến trước lần 20. Sau đó ổn định không giao động ở những lần lặp còn lại cho thấy POS đã tìm được giải pháp trong những lần lặp đầu sau đó hội tụ.

Sự cải thiện từ -2.775 đến -2.7625 (khoảng 0.0125) cho thấy thuật toán đã điều chỉnh ma trận chuyển trạng thái để mô hình hóa tốt hơn chuỗi quan sát [0, 1, 0, 1, 0, 0].



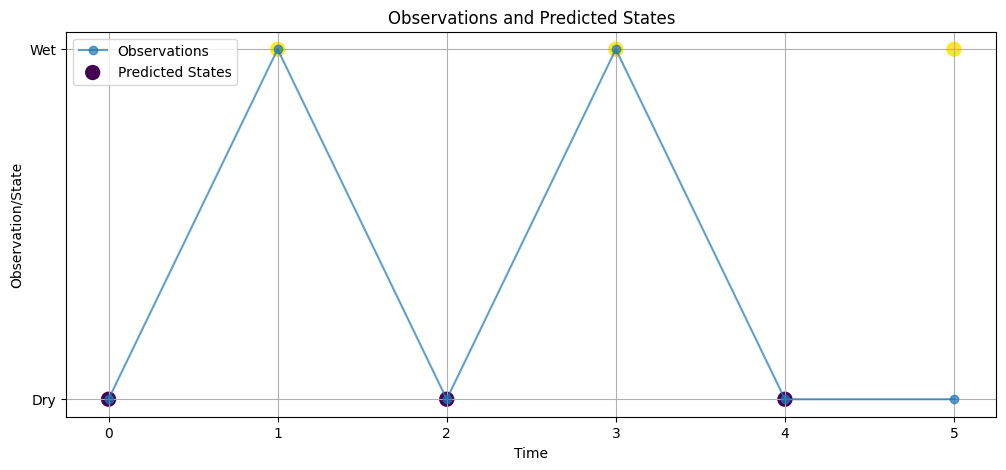
Biểu đồ 2 ma trận chuyển trạng thái thực tế và ma trận tối ưu có sự khác biệt rõ ràng



Biểu đồ thể hiện sự phân tán của điểm số log-likelihood của các hạt (particles) trong quá trình tối ưu hóa bằng Particle Swarm Optimization (PSO)

Biểu đồ thể hiện quá trình PSO đưa các hạt từ trạng thái phân tán ban đầu (log-likelihood thấp, -4.5) đến trạng thái hội tụ (log-likelihood cao hơn, -2.75 đến -2.8).

Đường Best Score thể hiện log-likelihood dần tối ưu và ổn định, phù hợp với kết quả ma trận chuyển trạng thái đã tối ưu hóa.



Biểu đồ thể hiện kết quả dựa đoán trạng thái ẩn của mô hình HMM sau khi tối ưu hóa bằng Particle Swarm Optimization (PSO). Kết quả của Chuỗi quan sát [0,1,0,1,0,0] là [Dry, Wet, Dry, Wet, Dry, Dry]

5. Tài liệu tham khảo

[1] S. Aupetit, N. Monmarché, and M. Slimane, "Hidden Markov Models training by a Particle Swarm Optimization algorithm," Université François-Rabelais de Tours, Laboratoire d’Informatique, Polytech’Tours, France. Available: [sebastien.aupetit@univ-tours.fr](mailto:sebastien.aupetit@univ-tours.fr)

[2] https://www.geeksforgeeks.org/hidden-markov-model-in-machine-learning/