

**ĐẠI HỌC THÁI NGUYÊN
ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

NGUYỄN TUYẾT LAN

**MẠNG ANFIS VÀ ỨNG DỤNG CHO DỰ BÁO THỜI TIẾT KHU VỰC
MIỀN NÚI PHÍA BẮC**

LUẬN VĂN THẠC SĨ KHOA HỌC MÁY TÍNH

GVHD: TS. LÊ BÁ DŨNG

THÁI NGUYÊN 2015

LỜI CAM ĐOAN

Tên tôi là: Nguyễn Tuyết Lan

Lớp: Cao học K12A

Khóa học: 2013 - 2015

Chuyên ngành: Khoa học máy tính

Mã số chuyên ngành: 60 48 01

Cơ sở đào tạo: Trường Đại học Công nghệ thông tin và Truyền thông Thái Nguyên

Giáo viên hướng dẫn: PGS.TS Lê Bá Dũng

Cơ quan công tác: Viện công nghệ thông tin – Viện Hàn lâm Khoa học và Công nghệ Việt Nam.

Tôi xin cam đoan luận văn ***“Mạng ANFIS và ứng dụng cho dự báo thời tiết khu vực miền núi phía Bắc”*** này là công trình nghiên cứu của riêng tôi. Các số liệu sử dụng trong luận văn là trung thực, các kết quả nghiên cứu được trình bày trong luận văn chưa từng được công bố tại bất kỳ công trình nào khác.

Thái Nguyên, ngày 12 tháng 5 năm 2015

Học viên

Nguyễn Tuyết Lan

LỜI CẢM ƠN

Tôi xin bày tỏ lời cảm ơn chân thành tới tập thể các thầy cô giáo Viện công nghệ thông tin - Viện Hàn lâm Khoa học và Công nghệ Việt Nam, các thầy cô giáo Trường Đại học Công nghệ thông tin và truyền thông - Đại học Thái Nguyên đã tận tình giảng dạy cũng như tạo mọi điều kiện để tôi học tập và nghiên cứu trong 2 năm học cao học.

Tôi xin chân thành cảm ơn sâu sắc tới thầy giáo PGS.TS Lê Bá Dũng đã cho tôi nhiều sự chỉ bảo quý báu, đã tận tình hướng dẫn và tạo điều kiện cho tôi hoàn thành tốt luận văn tốt nghiệp này.

Quá trình thực hiện đề tài không tránh khỏi các thiếu sót, rất mong tiếp tục nhận được sự đóng góp ý kiến của các thầy, các cô giáo, các bạn đồng nghiệp đối với đề tài nghiên cứu của tôi để đề tài được hoàn thiện hơn.

Tôi xin trân trọng cảm ơn!

Thái Nguyên, ngày 12 tháng 5 năm 2015

MỤC LỤC

LỜI CAM ĐOAN	i
LỜI CẢM ƠN.....	iii
MỤC LỤC	iv
DANH MỤC CÁC BẢNG BIỂU.....	vi
DANH MỤC CÁC HÌNH VẼ	vii
DANH MỤC CÁC TỪ VIẾT TẮT	ix
MỞ ĐẦU	1
CHƯƠNG I. MẠNG NƠ-RON NHÂN TẠO VÀ HỆ MỜ	2
1.1. Cấu trúc và mô hình mạng nơron	2
1.1.1. Mô hình một nơron nhân tạo	2
1.1.2. Cấu trúc của mạng nơron nhân tạo	5
1.2. Cấu tạo và phương thức làm việc của nơron	5
1.3. Các luật học	8
1.4. Thuật toán lan truyền ngược	12
1.5. Hệ mờ và mạng nơron	13
1.5.1. Các khái niệm cơ bản của logic mờ	13
1.5.2 Suy luận mờ.....	19
1.5.3. Cấu trúc của hệ thống suy luận mờ	22
1.6. Kết luận.....	31
CHƯƠNG 2. MẠNG ANFIS VÀ KHẢ NĂNG ỨNG DỤNG	32
2.1. Hệ thống suy diễn mờ dựa trên mạng thích nghi	32
2.1.1. Các mô hình kết hợp giữa hệ mờ và mạng neural.....	32
2.1.2. Luật mờ if-then và hệ suy diễn mờ.....	32
2.1.3. Cấu trúc mạng ANFIS	33
2.2. Các thuật toán mạng ANFIS.....	39
2.2.1. Thuật toán học lan truyền ngược	39
2.2.2. Thuật toán học lai	45
2.3. Ứng dụng của mạng ANFIS	47

CHƯƠNG III. ỨNG DỤNG MẠNG ANFIS CHO BÀI TOÁN DỰ BÁO THỜI	
TIẾT KHU VỰC MIỀN NÚI PHÍA BẮC	47
3.1. Bài toán dự báo thời tiết	47
3.1.1. Một số khái quát cơ bản về khí hậu, môi trường tự nhiên	47
3.1.2. Sự cần thiết của việc dự báo thời tiết	48
3.2. Ứng dụng mạng ANFIS cho bài toán dự báo thời tiết khu vực miền núi phía Bắc	50
3.2.1. Thu thập dữ liệu.....	51
3.2.2. Huấn luyện mạng ANFIS cho dự báo thời tiết khu vực miền núi phía Bắc	53
3.3. Kết quả dự báo.....	58
KẾT LUẬN	62
TÀI LIỆU THAM KHẢO	63

DANH MỤC CÁC BẢNG BIỂU

Bảng 2.1: Hai pha trong thủ tục học lai cho hệ ANFIS.....	46
Bảng 3.1: Số liệu thu thập tại trạm khí tượng thủy văn trong 2 năm gần đây	52
Bảng 3.2. Kết quả số liệu khảo sát và giá trị dự báo	59

DANH MỤC CÁC HÌNH VẼ

Hình 1.1. Mô hình một noron nhân tạo	2
Hình 1.2. Đồ thị các dạng hàm truyền.....	4
Hình 1.3. Mạng truyền thẳng một lớp	7
Hình 1.4. Mô tả cấu trúc của mạng nơ-ron truyền thẳng nhiều lớp.	8
Hình 1.5. Mạng hồi quy một lớp có nối ngược	8
Hình 1.6. Mạng hồi quy nhiều lớp có nối ngược.....	8
Hình 1.7: Sơ đồ học tham số có giám sát.	10
Hình 1.8: Sơ đồ học tham số không có giám sát.	11
Hình 1.9: Sơ đồ học tăng cường.....	11
Hình 1.10: Một số dạng hàm thuộc cơ bản	15
Hình 1.11: Hàm phụ thuộc của tập mờ A.....	16
Hình 1.12: Các tập mờ điển hình dùng để định nghĩa biến ngôn ngữ tốc độ.....	17
Hình 1.13: Mô hình suy luận mờ với một luật-một tiên đề.....	20
Hình 1.14: Mô hình suy luận mờ một luật-nhiều tiên đề	21
Hình 1.15: Mô hình suy luận mờ hai luật hai tiên đề	22
Hình 1.16: Giải mờ bằng phương pháp cực đại	23
Hình 1.17: Phương pháp giải mờ điểm trọng tâm.....	24
Hình 1.18. Hệ thống suy luận mờ.....	25
Hình 1.19. Mô hình suy diễn mờ Mamdani	27
Hình 1.20. Mô hình suy luận mờ Tsukamoto.....	28
Hình 1.21: Hệ suy diễn mờ Takagi- Sugeno	29
Hình 1.22: Mô hình suy luận mờ Sugeno hai đầu vào một đầu ra	30
Hình 2.1. Lập luận mờ.....	33
Hình 2.2. Mạng thích nghi.....	34
Hình 2.3. Kiến trúc mạng ANFIS.....	37
Hình 2.4: Mạng 3 lớp lan truyền ngược	40
Hình 3.1. Dữ liệu vào ở bảng 3.1 biểu diễn dưới dạng đồ thị	55
Hình 3.2: Sơ đồ khối của mạng ANFIS.....	55

Hình 3.3: Dữ liệu học sau khi đã load vào chương trình.....**Error! Bookmark not defined.**

Hình 3.5: Mạng ANFIS được xây dựng cho huấn luyện dữ liệu để dự báo thời tiết khu vực miền núi phía Bắc	56
Hình 3.6. Hệ luật mờ được hình thành trong quá trình huấn luyện mạng ANFIS	57
Hình 3.6a Hệ luật mờ được sinh ra cho huấn luyện mạng	58
Hình 3.6b Mặt suy diễn của hệ ANFIS	58
Hình 3.7: Kết quả dự báo thời tiết	59

DANH MỤC CÁC TỪ VIẾT TẮT

ANN	Artificial Neural Network Mạng nơron nhân tạo
ANFIS	Adaptive Neuro Fuzzy Inference System Hệ suy luận mờ
PE	Processing Element Phần tử xử lý
TSK	Takagi – Sugeno – Kang Mô hình Takagi - Sugeno
BP	Back Propagation Lan truyền ngược
LSE	Least Square Error Ước lượng sai số bình phương tối thiểu
CPI	Chỉ số giá tiêu dùng

MỞ ĐẦU

Mạng nơron nhân tạo (Artificial Neural Network-ANN) nói chung, hệ suy diễn mờ - nơron thích nghi (ANFIS - Adaptive Neural Fuzzy Inference System) nói riêng đã và đang được nghiên cứu, ứng dụng mạnh mẽ và thành công ở nhiều lĩnh vực trong những năm gần đây [6,7]. Với các quá trình như: xấp xỉ phi tuyến, dự báo thị trường chứng khoán, dự báo mô phỏng các hệ thống điều khiển...được đưa ra, giải quyết có kết quả [1,2,3,4]. Các lớp bài toán của các lĩnh vực trên cũng có thể sử dụng và giải quyết theo các phương pháp truyền thống như phương pháp thống kê, quy hoạch tuyến tính [1],... Mạng nơron nhân tạo, mạng ANFIS được hình thành có nhiều khả năng vượt trội trong việc tuyến tính hóa, dự báo, phân tích, đánh giá dữ liệu và áp dụng thành công cho một số lĩnh vực khoa học, kỹ thuật, kinh tế...

Quá trình dự báo thời tiết ở Việt Nam nói chung, khu vực miền núi phía Bắc nói riêng có những đặc thù và có nhiều sự cố về thiên tai như: Rét đậm, rét hại, bão, tố, lốc, mưa đá, sạt lở đất, thời tiết khắc nghiệt, ảnh hưởng của sông ngòi, mưa nhiều dẫn đến lũ lụt, hạn hán... Để góp sức vào quá trình ứng dụng những thành tựu của công nghệ thông tin trong dự báo thủy văn [9,10], được sự gợi ý của thầy hướng dẫn và nhận thấy tính thiết thực của vấn đề em chọn đề tài: **“Mạng ANFIS và ứng dụng cho dự báo thời tiết khu vực miền núi phía Bắc”** làm khóa luận tốt nghiệp cho luận văn tốt nghiệp của mình, Luận văn bao gồm các nội dung sau:

Chương 1: Mạng Nơron nhân tạo và hệ mờ.

Chương 2: Mạng ANFIS và khả năng ứng dụng.

Chương 3: Ứng dụng mạng Anfis cho bài toán dự báo thời tiết khu vực miền núi phía Bắc.

CHƯƠNG I

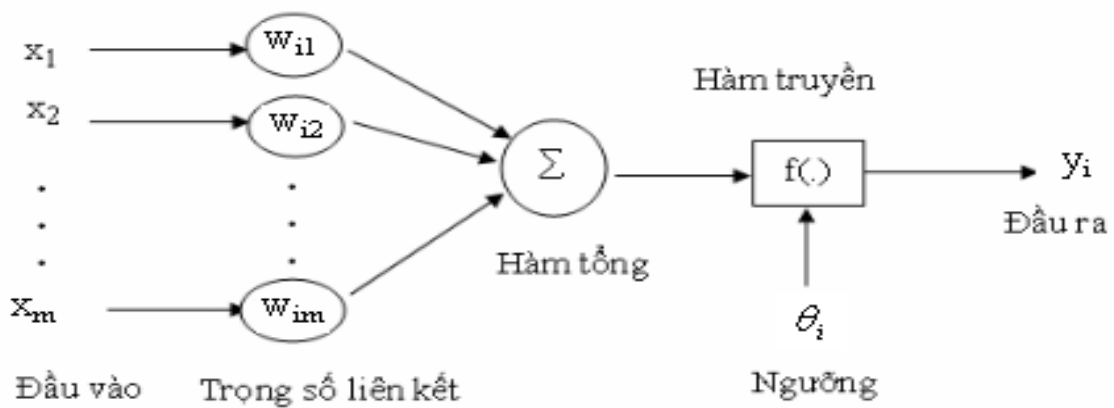
MẠNG NƠ-RON NHÂN TẠO VÀ HỆ MỜ

1.1. Cấu trúc và mô hình mạng nơron

1.1.1. Mô hình một nơron nhân tạo

Một nơron là một đơn vị xử lý thông tin và là thành phần cơ bản của một mạng nơron. Cấu trúc một nơron được mô tả trên hình dưới.

Mô hình nơron có m đầu vào x_1, x_2, \dots, x_m , và một đầu ra y_i như sau:



Hình 1.1. Mô hình một nơron nhân tạo

Giải thích các thành phần cơ bản:

- Tập các đầu vào: Là các tín hiệu vào (input signals) của nơron, các tín hiệu này thường được đưa vào dưới dạng một vector m chiều.
- Tập các liên kết (các trọng số): Mỗi liên kết được thể hiện bởi một trọng số (thường được gọi là trọng số liên kết – Synaptic weight). Trọng số liên kết giữa tín hiệu vào thứ j cho nơron i thường được ký hiệu là w_{ij} . Thông thường các trọng số này được khởi tạo ngẫu nhiên ở thời điểm khởi tạo mạng và được cập nhật liên tục trong quá trình học mạng.
- Bộ tổng (Hàm tổng): Thường dùng để tính tổng của tích các đầu vào với trọng số liên kết của nó.

- Ngưỡng: Ngưỡng này thường được đưa vào như một thành phần của hàm truyền.

- Hàm truyền: Hàm này dùng để giới hạn phạm vi đầu ra của mỗi nơron. Nó nhận đầu vào là kết quả của hàm tổng và ngưỡng đã cho. Thông thường, phạm vi đầu ra của mỗi nơron được giới hạn trong đoạn $[0,1]$ hoặc $[-1,1]$. Các hàm truyền rất đa dạng, có thể là các hàm tuyến tính hoặc phi tuyến. Việc lựa chọn hàm truyền tùy thuộc vào từng bài toán và kinh nghiệm của người thiết kế mạng.

- Đầu ra: Là tín hiệu đầu ra của một nơron, với mỗi nơron sẽ có tối đa một đầu ra.

Về mặt toán học, cấu trúc của một nơron i được mô tả bằng cặp biểu thức sau:

$$y_i = f(net_i - \theta_i) \text{ và } net_i = \sum_{j=1}^n w_{ij} x_j \quad (1.1)$$

trong đó: x_1, x_2, \dots, x_m là các tín hiệu đầu vào, còn $w_{i1}, w_{i2}, \dots, w_{im}$ là các trọng số kết nối của nơron thứ i , net_i là hàm tổng, f là hàm truyền, θ_i là một ngưỡng, y_i là tín hiệu đầu ra của nơron.

Như vậy, tương tự như nơron sinh học, nơron nhân tạo cũng nhận các tín hiệu đầu vào, xử lý (nhân các tín hiệu này với trọng số liên kết, tính tổng các tích thu được rồi gửi kết quả đến hàm truyền), và cho một tín hiệu đầu ra (là kết quả của hàm truyền).

- Hàm truyền có thể có các dạng sau:

- Hàm bước
$$y = \begin{cases} 1 & \text{khi } x \geq 0 \\ 0 & \text{khi } x < 0 \end{cases} \quad (1.2)$$

- Hàm giới hạn chặt (hay còn gọi là hàm bước)

$$y = \text{sgn}(x) = \begin{cases} 1 & \text{khi } x \geq 0 \\ -1 & \text{khi } x < 0 \end{cases} \quad (1.3)$$

- Hàm bậc thang

$$y = \text{sgn}(x) = \begin{cases} 1 & \text{khi } x > 1 \\ x & \text{khi } 0 \leq x \leq 1 \\ 0 & \text{khi } x < 0 \end{cases} \quad (1.4)$$

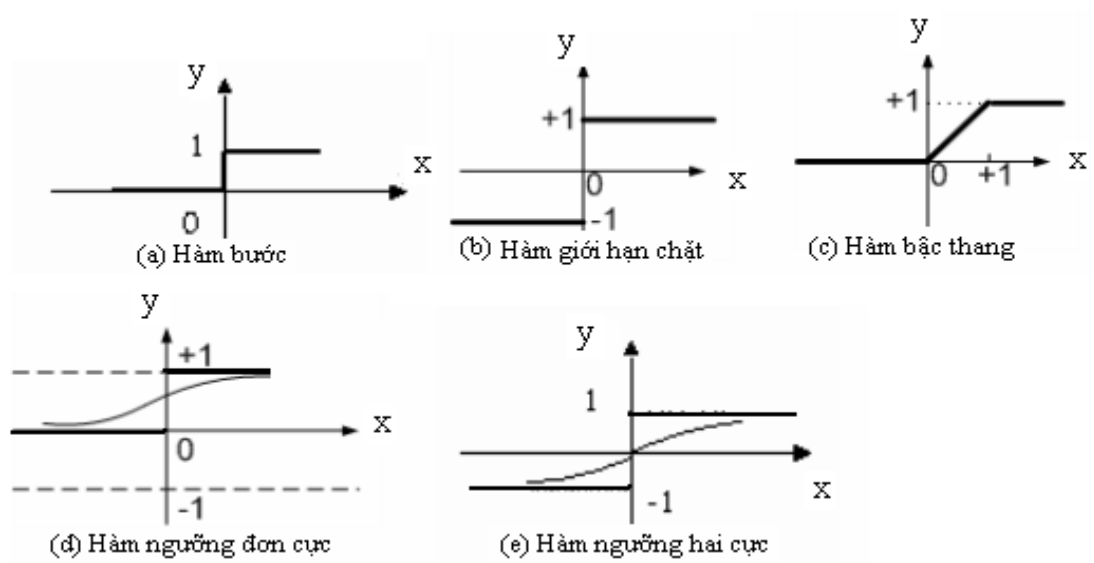
- Hàm ngưỡng đơn cực

$$y = \frac{1}{1 + e^{-\lambda x}} \quad \text{với } \lambda > 0 \quad (1.5)$$

- Hàm ngưỡng hai cực

$$y = \frac{2}{1 + e^{-\lambda x}} - 1 \quad \text{với } \lambda > 0 \quad (1.6)$$

- Đồ thị các dạng hàm truyền được biểu diễn như sau:



Hình 1.2. Đồ thị các dạng hàm truyền

1.1.2. Cấu trúc của mạng nơ-ron nhân tạo

Mạng nơ-ron nhân tạo (Artificial Neural Network) là một cấu trúc mạng được hình thành nên bởi số lượng các nơ-ron nhân tạo liên kết với nhau. Mỗi nơ-ron có các đặc tính đầu vào, đầu ra và thực hiện một số chức năng tính toán cục bộ.

Với việc giả lập các hệ thống sinh học, các cấu trúc tính toán, mạng nơ-ron có thể giải quyết được các lớp bài toán nhất định như: Bài toán xếp loại, bài toán lập lịch, bài toán tìm kiếm, bài toán nhận dạng mẫu... Các bài toán phức tạp cao, không xác định. Tuy nhiên, sự liên kết giữa một bài toán bất kỳ trong thực tế với một giải pháp mạng nơ-ron lại là một việc không dễ dàng.

Xét một cách tổng quát, mạng nơ-ron là một cấu trúc xử lý song song thông tin phân tán mang các đặc tính nổi bật sau:

- Là một mô hình tính toán dựa trên bản chất của nơ-ron.
- Bao gồm một số lượng rất lớn các nơ-ron liên kết với nhau.
- Mạng nơ-ron có khả năng học, khái quát hóa tập dữ liệu học thông qua việc gán và hiệu chỉnh các trọng số liên kết.
- Tổ chức theo kiểu tập hợp mang lại cho mạng nơ-ron khả năng tính toán rất lớn, trong đó không có nơ-ron nào mang thông tin riêng biệt.

1.2. Cấu tạo và phương thức làm việc của nơ-ron

• Liên kết mạng

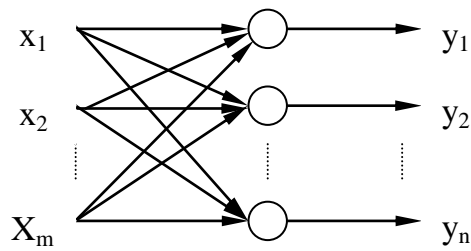
Sự liên kết trong mạng nơ-ron tùy thuộc vào nguyên lý tương tác giữa đầu ra của từng nơ-ron riêng biệt với các nơ-ron khác và tạo ra cấu trúc mạng. Khi liên kết các đầu vào/ra của nhiều nơ-ron với nhau, ta thu được một mạng nơ-ron trong mạng với nhau có thể theo nguyên tắc bất kỳ. Vì mạng nơ-ron là một hệ truyền đạt và xử lý tín hiệu. Trong phần này chúng ta đi tìm hiểu hai liên kết mạng đó là mạng truyền thẳng và mạng hồi quy.

• Mạng truyền thẳng

Mạng truyền thẳng một lớp

Mô hình mạng nơ-ron truyền thẳng một lớp là mô hình liên kết cơ bản và đơn giản nhất. Các nơ-ron tổ chức lại với nhau tạo thành một lớp, đường truyền tín hiệu được truyền theo một hướng nhất định nào đó. Các đầu vào được nối với các

nơ-ron theo các trọng số khác nhau, sau quá trình xử lý cho ra một chuỗi các tín hiệu ra. Mạng nơ-ron là mô hình LTU thì nó được gọi là mạng Perception, còn mạng nơ-ron là mô hình LGU thì nó được gọi là mạng Adaline.



Hình 1.3. Mạng truyền thẳng một lớp

Với mỗi giá trị đầu vào $x = [x_1, x_2, \dots, x_n]^T$. Qua quá trình xử lý của mạng ta sẽ thu được một bộ tương ứng các giá trị đầu ra là $y = [y_1, y_2, \dots, y_n]^T$ được xác định như sau:

$$y_i = f_i \left(\sum_{j=1}^m w_{ij} x_j - \theta_i \right), \quad i = \overline{1, n}, \quad (1.7)$$

trong đó: m : số tín hiệu vào

n : số tín hiệu ra

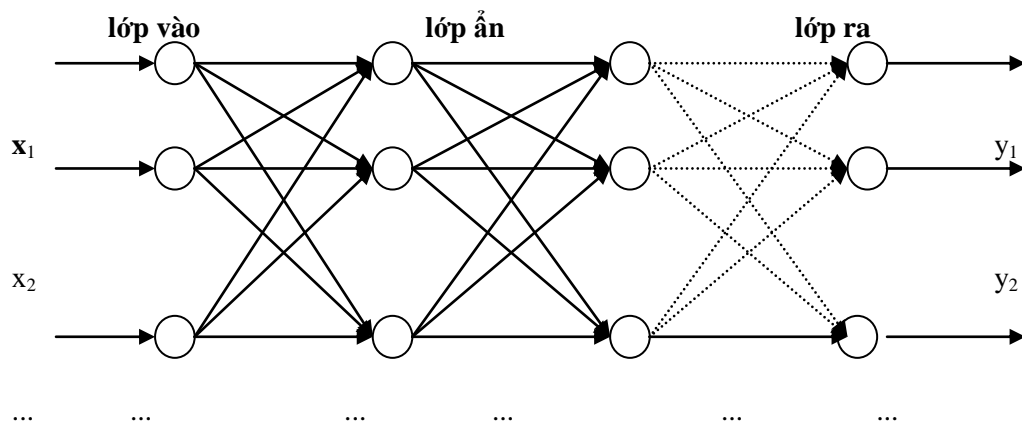
$W_i^T = [w_{i1}, w_{i2}, \dots, w_{in}]^T$ là véc tơ trọng số của nơ ron thứ i .

f_i : hàm kích hoạt của nơ ron thứ i

θ_i : là ngưỡng của nơ ron thứ i .

Mạng truyền thẳng nhiều lớp.

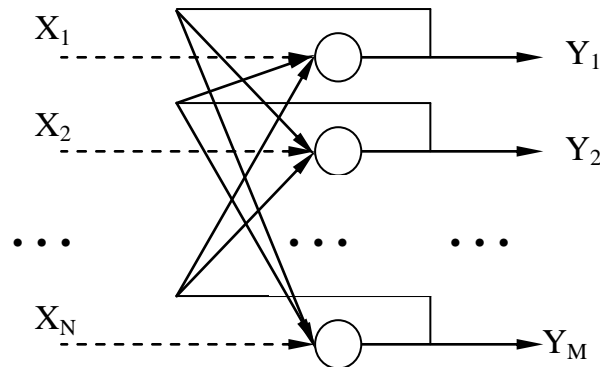
Với mạng nơ-ron truyền thẳng một lớp ở trên khi phân tích một bài toán phức tạp sẽ gặp rất nhiều khó khăn, để khắc phục vấn đề này người ta đưa ra mô hình mạng nơ-ron truyền thẳng nhiều lớp bằng việc kết hợp với một số lớp nơ-ron lại với nhau. Lớp nhận tín hiệu vào gọi là lớp vào, lớp đưa ra tín hiệu ra của mạng được gọi là lớp ra. Các lớp ở giữa lớp vào và lớp ra được gọi là các lớp ẩn.



Hình 1.4. Mô tả cấu trúc của mạng nơ-ron truyền thẳng nhiều lớp.

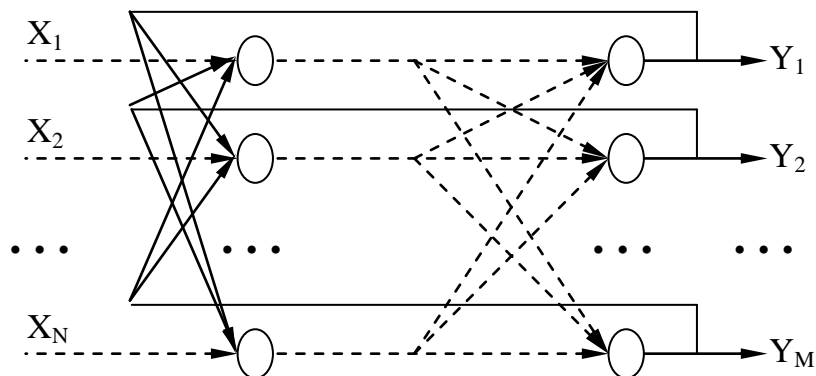
- **Mạng hồi quy**

Mạng hồi quy một lớp có nối ngược



Hình 1.5. Mạng hồi quy một lớp có nối ngược

Mạng hồi quy nhiều lớp có nối ngược



Hình 1.6. Mạng hồi quy nhiều lớp có nối

1.3. Các luật học

Mạng nơ-ron có một số ưu điểm so với máy tính truyền thống. Cấu trúc song song của mạng nơ-ron rất thích hợp cho những ứng dụng đòi hỏi tốc độ nhanh theo thời gian thực. Khả năng huấn luyện của mạng nơ-ron có thể khai thác để phát triển hệ học thích nghi. Mặt khác, với khả năng tổng quát hóa của mạng nơ-ron, nó có thể áp dụng để điều khiển nhiều tham số phức tạp đồng thời từ đó giải quyết dễ dàng một số bài toán NP - đầy đủ (NP - Complete).

Các luật học đóng vai trò quan trọng trong việc xác định một mạng nơ-ron nhân tạo. Một cách đơn giản về khái niệm học của mạng nơ-ron là cập nhật trọng số trên cơ sở các mẫu. Theo nghĩa rộng thì học có thể được chia thành hai loại: Học tham số và học cấu trúc

- **Học tham số:**

Các thủ tục học này nhằm tìm kiếm ma trận trọng số sao cho mạng có khả năng đưa ra dự báo sát với thực tế. Dạng chung của luật học tham số có thể được mô tả như sau:

$$\Delta W_{ij} = \eta r x_j, \quad i = \overline{1, N}, \quad j = \overline{1, M}, \quad (1.8)$$

trong đó:

ΔW_{ij} là sự thay đổi trọng số liên kết từ nơ-ron j đến nơ-ron i.

x_j là tín hiệu vào nơ-ron j.

η là tốc độ học, nằm trong khoảng (0,1).

r là hằng số học.

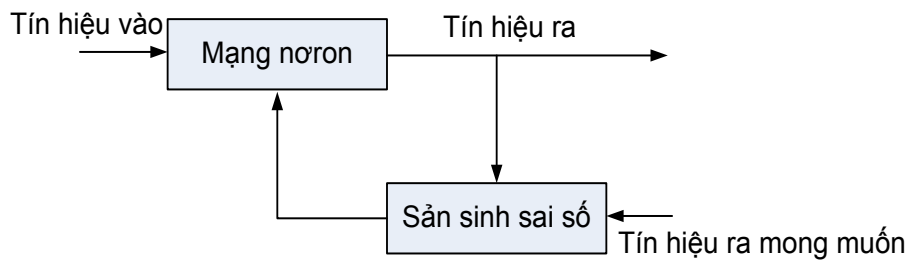
Vấn đề đặt ra ở đây là tín hiệu học r được sinh ra như thế nào để hiệu chỉnh trọng số của mạng.

Có thể chia thủ tục học tham số ra thành ba lớp nhỏ hơn: học có giám sát, học không có giám sát và học tăng cường.

+ **Học có giám sát:** Là quá trình học dựa vào sai số giữa đầu ra thực và đầu ra mong muốn để làm cơ sở cho việc hiệu chỉnh trọng số. Sai số này chính là hằng số học r. Luật học điển hình của nhóm này là luật học Della của Widrow (1962) nêu ra đầu tiên dùng xấp xỉ trọng số của Adaline dựa trên nguyên tắc gradient.

Trong nhóm luật học này cũng cần kể đến luật học Perceptron của Rosenblatt (1958). Về cơ bản luật học này thay đổi các giá trị trọng trong thời gian học, còn Perceptron thì thêm hoặc bỏ trọng tùy theo giá trị sai số là dương hay âm.

Một loạt các luật học khác cũng được dựa trên tư tưởng này. Luật oja là cải tiến và nâng cấp của luật Delta. Luật truyền ngược là luật mở rộng của luật Delta cho mạng nhiều lớp. Đối với mạng truyền thẳng thường sử dụng luật truyền ngược để chỉnh trọng số với tín hiệu chỉ đạo từ bên ngoài và người ta gọi mạng này là mạng lan truyền ngược.



Hình 1.7: Sơ đồ học tham số có giám sát.

+ **Học không có giám sát:** Luật học này sử dụng đầu ra của mạng làm cơ sở để hiệu chỉnh các trọng số liên kết. Hay trong luật này chính là tín hiệu ra của mạng. Điển hình là luật Hebb (1949) thường dùng cho các mạng tự liên kết, luật LVQ (Learning Vector Quantization) dùng cho mạng tự tổ chức một lớp thuộc lớp mạng ánh xạ đặc trưng của Kohonen.

Luật học Hebb là luật sinh học xuất phát từ tiên đề của Hebb cho rằng: Giữa hai nơ-ron có quan hệ và có thay đổi thế năng mạng thì giữa chúng có sự thay đổi trọng số liên kết. Nói cách khác, trọng số được điều chỉnh theo mối tương quan trước và sau, nghĩa là:

$$\Delta W_{ij} = \eta y_i x_j, i = \overline{1, N}, j = \overline{1, M}, \quad (1.9)$$

trong đó:

ΔW_{ij} : Là sự thay đổi trọng số liên kết từ nơ-ron j đến nơ-ron i.

x_j : là tín hiệu vào nơ-ron j.

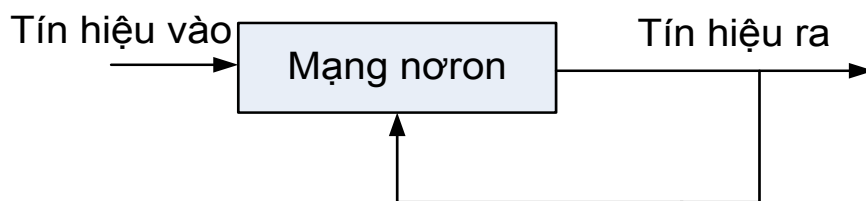
y_i là tín hiệu ra của nơ-ron i.

η là tốc độ học, nằm trong khoảng (0,1).

Luật Hebb giải thích việc chỉnh trọng số trong phạm vi cục bộ của mạng mà không cần tín hiệu chỉ đạo từ bên ngoài. Hopfield cũng cải tiến luật Hebb cho các mạng tự liên kết thành 16 dạng khác nhau theo kiểu luật Hebb, luật đối Hebb, luật Hopfield...

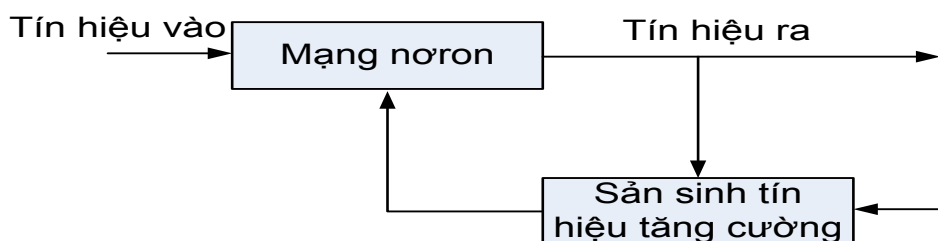
Như vậy, ứng với mỗi nhóm mạng thường áp dụng một luật học nhất định. Nếu tồn tại hàng chục loại mạng khác nhau thì các luật học dùng trong mạng nơ-ron có thể tăng lên rất nhiều lần.

Đối với mạng phản hồi thường sử dụng luật Hebb và các luật cải tiến của nó để chỉnh trọng số mà không cần tín hiệu chỉ đạo từ bên ngoài.



Hình 1.8: Sơ đồ học tham số không có giám sát.

+ **Học tăng cường**: Trong một số trường hợp, thông tin phản hồi chỉ là tín hiệu bao gồm hai trạng thái cho biết tín hiệu đầu ra của mạng là đúng hay sai. Quá trình học dựa trên các thông tin hướng dẫn như vậy được gọi là học có củng cố (học tăng cường) và tín hiệu mang thông tin phản hồi được gọi là tín hiệu củng cố cho quá trình học. Ta có thể thấy rằng quá trình học này là một dạng của quá trình học có tín hiệu chỉ đạo bởi vì mạng nhận được một số thông tin phản hồi từ bên ngoài.



Hình 1.9: Sơ đồ học tăng cường.

- **Học cấu trúc:**

Tìm kiếm các tham số của cấu trúc mạng để tìm ra một cấu trúc mạng hoạt động tốt nhất. Trong thực tế, việc học cấu trúc là tìm ra số lớp ẩn và tìm ra số nơ-ron trên mỗi lớp đó. Giải thuật di truyền thường được sử dụng trong các cấu trúc nhưng thường chạy rất lâu, thậm chí ngay cả đối với mạng có kích thước trung bình. Ngoài ra kỹ thuật gọt tĩa mạng hay mạng tăng dần cũng được áp dụng trong việc học cấu trúc của mạng có kích thước tương đối nhỏ.

1.4. Thuật toán lan truyền ngược

Ta sử dụng một số kí hiệu sau:

- j : nơron thứ j (hay nút thứ j)
- X_j : vector đầu vào của nút thứ j
- W_j : vector trọng số của nút thứ j
- x_{ji} : đầu vào của nút thứ j từ nút thứ i
- w_{ji} : trọng số trên x_{ji}
- b_j : ngưỡng tại nút thứ j
- o_j : đầu ra của nút thứ j
- t_j : đầu ra mong muốn của nút thứ j
- $\text{Downstream}(j)$: Tập tất cả các nút nhận đầu ra của nút thứ j làm một giá trị đầu vào.
- η : tốc độ học
- f : hàm truyền với $f(x) = 1 / (1 + e^{-x})$

Thuật toán lan truyền ngược được mô tả như sau:

Input:

- Mạng feed-forward với n_i đầu vào, n_h nút ẩn và n_o đầu ra.
- Hệ số học η
- Tập dữ liệu huấn luyện $D = \{ \text{là vector đầu vào, là vector đầu ra mong muốn} \}$.

Output: Các vector trọng số

Ký hiệu: Y .

Thuật toán:

Bước 1: Khởi tạo trọng số bởi các giá trị ngẫu nhiên nhỏ.

Bước 2: Lặp lại cho tới khi thỏa mãn điều kiện kết thúc.

Với mỗi mẫu, thực hiện các bước sau:

2.1 Tính đầu ra o_j cho mỗi nút j :

$$o_j = f(\mathbf{d} - \mathbf{b}_j) \text{ với } \mathbf{d} = \sum \mathbf{x}_{ji} \mathbf{w}_{ji}$$

2.2 Với mỗi nút k thuộc tầng ra, tính δ_k theo công thức:

$$\delta_k = (t_k - o_k)(1 - o_k)o_k$$

2.3 Với mỗi nút h thuộc tầng ẩn, tính δ_h theo công thức:

$$\delta_h = o_h(1 - o_h) \sum \delta_k \mathbf{w}_{kh} \text{ với } k \in \text{Downstream}(j)$$

2.4 Cập nhật: $\mathbf{w}_{ji} = \mathbf{w}_{ji} + \Delta \mathbf{w}_{ji}$

$$\text{Trong đó } \Delta \mathbf{w}_{ji} = \eta \delta_k \mathbf{x}_{ji}$$

1.5. Hệ mờ và mạng nơron

1.5.1. Các khái niệm cơ bản của logic mờ

a/ Định nghĩa tập mờ

Cơ sở của logic mờ là việc ánh xạ từ các biến x đầu vào thuộc tập A thành các biến y đầu ra thuộc tập B .

Nói cách khác, giá trị $x=a$ không được xác định rõ là có thuộc hay không thuộc tập B , và khái niệm mờ được đưa ra để làm nền tảng cho logic mờ và điều khiển mờ sau này.

Cơ chế cơ bản của logic mờ sau này có dạng là tập hợp các trạng thái *nếu...thì* hay còn được gọi là những quy luật.

Tập mờ được coi là phần mở rộng của tập kinh điển. Nếu X là một không gian nền (một tập nền) và những phần tử của nó được biểu thị bằng x , thì một tập mờ A trong X được xác định bởi một cặp các giá trị:

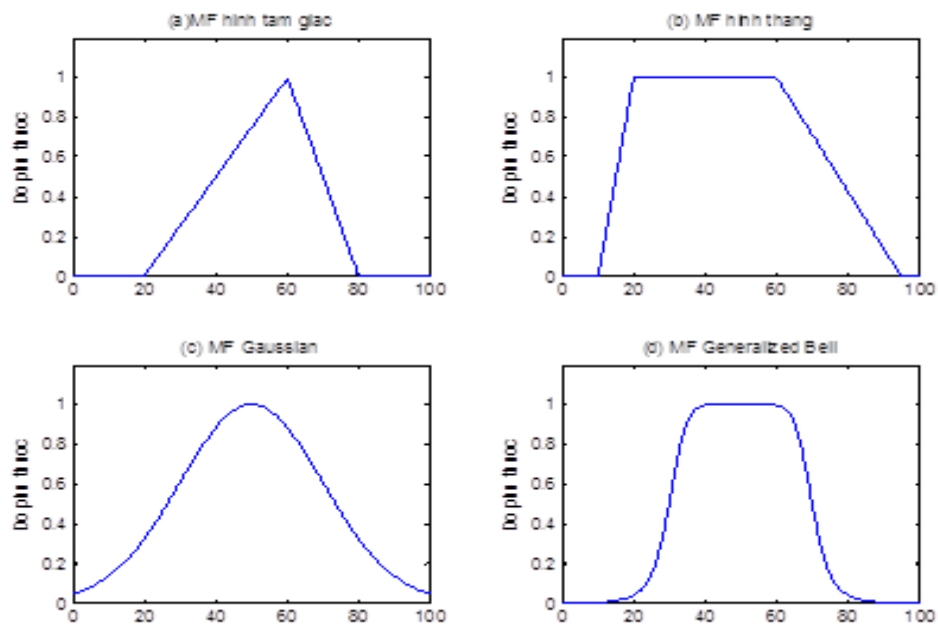
$$A = \{x, \mu_A(x) \mid x \in X\} \quad (1.10)$$

Với $0 \leq \mu_A(x) \leq 1$

Trong đó $\mu_A(x)$ được gọi là hàm liên thuộc của x trong A -viết tắt là MF (*Membership Function*). Nó không còn là hàm hai giá trị như đối với tập kinh điển nữa, mà là một hàm với một tập các giá trị hay còn gọi là một ánh xạ. Tức là, *hàm liên thuộc ánh xạ mỗi một phần tử của X tới một giá trị liên thuộc trong khoảng $[0,1]$.*

Như vậy, kiến trúc của một tập mờ phụ thuộc vào hai yếu tố: không gian nền và hàm liên thuộc phù hợp. Sự đặc biệt của hàm thuộc là nó mang tính chủ quan với ý nghĩa là với cùng định nghĩa một khái niệm nhưng với mỗi người khác nhau thì hàm thuộc có thể được xây dựng khác nhau.

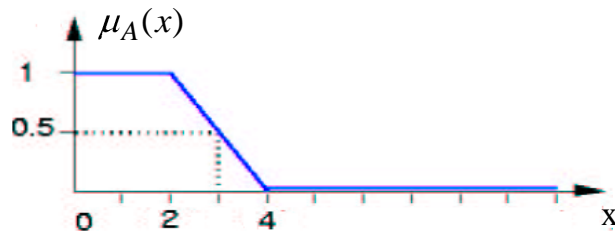
Các hàm liên thuộc được xây dựng từ những hàm cơ bản như: Kết nối hành vi, hàm bậc nhất, hình thang, hình tam giác, hàm phân bố Gaussian, đường cong xichma, đường cong đa thức bậc hai và bậc ba. Hình 1.10 dưới đây mô tả một vài dạng hàm thuộc cơ bản:



Hình 1.10: Một số dạng hàm thuộc cơ bản

Ví dụ 1.1:

Cho một tập mờ A của các số tự nhiên nhỏ hơn 4, với hàm phụ thuộc $\mu_A(x)$ có dạng như trong hình 1.10. Khi đó A sẽ chứa các phần tử như sau:



Hình 1.11: Hàm phụ thuộc của tập mờ A

$A = \{(1,1), (2,1), (3,0.5)\}$. Số tự nhiên 1 và 2 có độ phụ thuộc $\mu_A(1) = \mu_A(2) = 1$.

Số 3 có độ phụ thuộc nhỏ hơn $\mu_A(3) = 0.5$. Những số không được liệt kê đều có độ phụ thuộc bằng 0.

b/ Các phép toán đại số trên tập mờ

Các phép toán trên tập mờ được định nghĩa thông qua các hàm thuộc và được xây dựng tương tự như các phép toán trong lý thuyết tập mờ kinh điển, bao gồm tập con, phép giao, phép hợp và phép lấy phần bù.

Giả sử A và B là hai tập mờ trên không gian nền U , có các hàm thuộc μ_A , μ_B . Khi đó:

- **Tập con:**

A là tập con của B , ký hiệu là $A \subseteq B$ khi và chỉ khi:

$$\mu_A(x) \leq \mu_B(x) \text{ với } \forall x \in U$$

- **Phép hợp:**

$$\mu_{A \cup B}(x) = \max \{ \mu_A(x), \mu_B(x) \} = \mu_A(x) \vee \mu_B(x)$$

Ở đây \vee là ký hiệu của phép toán max.

- **Phép giao:**

$$\mu_{A \cap B}(x) = \min \{ \mu_A(x), \mu_B(x) \} = \mu_A(x) \wedge \mu_B(x)$$

Ở đây \wedge là ký hiệu cho phép toán min.

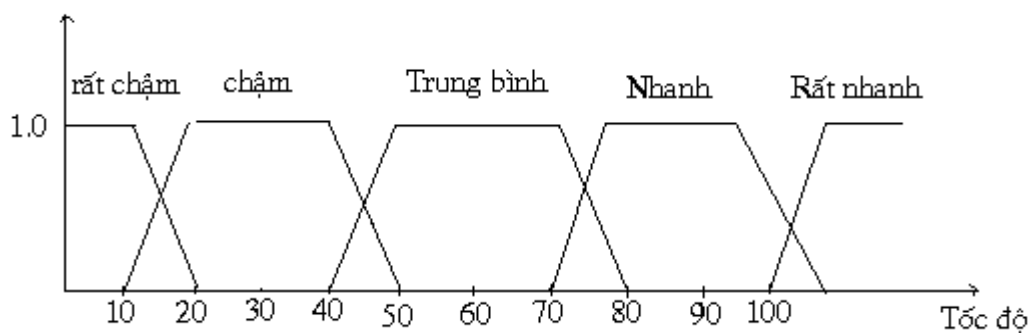
- **Phép lấy phần bù:**

Phép lấy phần bù A^C là tập mờ với hàm thuộc:

$$\mu_{A^C}(x) = 1 - \mu_A(x)$$

c/ Biến ngôn ngữ

Trong logic mờ, những biến ngôn ngữ đảm nhiệm những giá trị ngôn ngữ. Mỗi giá trị ngôn ngữ thực chất là một tập mờ xác định bởi một hàm thuộc và khoảng giá trị tương ứng. Logic mờ cho phép các tập này có thể xếp giao nhau. Chẳng hạn, ta có các giá trị ngôn ngữ như “*nhANH*”, “*trung bình*”, “*chậm*”, mỗi giá trị ngôn ngữ này có hàm thuộc dạng hình thang cân xác định trong khoảng tốc độ [20,100]. Khi đó, tốc độ 45 có thể trực thuộc cả tập mờ “*chậm*” lẫn tập mờ “*trung bình*”.



Hình 1.12: Các tập mờ điển hình dùng để định nghĩa biến ngôn ngữ tốc độ

Một dạng suy rộng khác trong cơ sở tri thức của nhiều hệ mờ thực tiễn có thể được phát biểu như sau:

Cho x_1, x_2, \dots, x_m là các biến vào của hệ thống, y là biến ra. Các tập A_{ij}, B_j , với $i=1, 2, \dots, m, j=1, 2, \dots, n$ là các tập mờ trong không gian nền tương ứng của các biến vào và biến ra. R_j là các suy diễn mờ dạng “ifthen...”

$$R_1 : \text{if } x_1 \text{ is } A_{1,1} \text{ and...and } x_m \text{ is } A_{m,1} \text{ then } y \text{ is } B_1$$

$$R_2 : \text{if } x_1 \text{ is } A_{1,2} \text{ and...and } x_m \text{ is } A_{m,2} \text{ then } y \text{ is } B_2$$

.....

$$R_n : \text{if } x_1 \text{ is } A_{1,n} \text{ and...and } x_m \text{ is } A_{m,n} \text{ then } y \text{ is } B_n$$

Bài toán:

Input	Nếu x_1 là e_1^* và...và x_m là e_m^*
Output	Giá trị y là u^*

ở đây e_1^*, \dots, e_m^* là các giá trị đầu vào hay sự kiện (có thể mờ hoặc rõ).

Chúng ta có thể nhận thấy, phần cốt lõi của nhiều hệ mờ cho bởi cơ sở tri thức dạng $R = \{\text{các luật } R_i\}$ và các cơ chế suy diễn cài đặt trong mô tơ suy diễn.

d/ Luật mờ IF-THEN

Luật mờ là biểu thức điều kiện có dạng “NẾU A THÌ B”, trong đó A và B là nhãn của các tập mờ được mô tả bằng cách xấp xỉ các thành viên. Nhờ vào dạng rút gọn, luật mờ thường được dùng để thiết lập những phương thức lập luận không chính xác, nhằm thể hiện tính đa dạng trong trí thức của con người. Ví dụ sau mô tả một sự kiện đơn giản là (đây là luật mờ loại Mamdani):

Nếu nhiệt độ cao, thì giá máy lạnh tăng. Trong đó *nhiệt độ* và *giá máy lạnh* là các biến ngôn ngữ, *cao*, và *tăng* là các giá trị ngôn ngữ hoặc các nhãn được mô tả bởi các hàm thành viên.

Một dạng khác của luật mờ do Takagi và Sugeno đề xuất, có các tập mờ chỉ xuất hiện trong phần giả thuyết của luật. Ví dụ (đây là luật mờ loại Sugeno):

*Nếu lưu lượng dòng chảy cao thì mực nước sông = k * lưu lượng dòng chảy.*
 Trong đó, *cao* là phần giả thuyết được mô tả bởi hàm thành viên xấp xỉ. Tuy nhiên, phần kết luận được định nghĩa bởi phương trình theo biến *lưu lượng dòng chảy*.

Cả hai loại dạng luật mờ *if then* trên đều được sử dụng rộng rãi trong mô hình hóa và điều khiển. Nhờ cách sử dụng những các biến ngôn ngữ và hàm thành viên mà một luật mờ có thể dễ dàng mô tả được sự suy luận chủ yếu của con người. Từ góc độ khác, do những định lượng ở phần giả thiết, mỗi luật mờ có thể được xem như là một sự mô tả cục bộ trong phạm vi giả định. Các luật mờ tạo nên phần cốt lõi của hệ thống suy luận mờ.

1.5.2 Suy luận mờ

Suy diễn mờ, còn được gọi là suy diễn xấp xỉ là một thuật toán suy luận nhằm thu được kết luận từ một tập các luật nếu-thì mờ được coi như chân lý. Luật cơ bản của phép suy luận truyền thống với hai giá trị logic là modus ponens, từ luật này ta có thể suy luận ra mệnh đề B từ mệnh đề A và phép kéo theo $R = A \rightarrow B$. Có thể minh họa luật modus ponens :

Giả thiết 1 (sự kiện) : x là A

Giả thiết 2 (luật) : Nếu x là A thì y là B

Suy diễn (kết luận) : y là B

Tuy nhiên, trong suy diễn của con người, luật modus ponens được sử dụng theo cách thức xấp xỉ điều này được minh họa như sau:

Giả thiết 1 (sự kiện) : x là A'

Giả thiết 2 (luật) : Nếu x là A thì y là B

Suy diễn (kết luận) : y là B'

Trong đó A' gần với A và B' gần với B. Khi A, B, A', B' là các tập mờ trên các không gian nền tương ứng thì thuật toán trên được gọi là suy luận xấp xỉ hay suy diễn mờ. Sử dụng luật hợp thành mờ đã nêu ở trên ta có thể thành lập thuật toán suy diễn mờ như sau:

Gọi A , A' và B là các tập mờ trên không gian X , X và Y . Giả thiết phép kéo theo mờ $A \rightarrow B$ được diễn giải như một quan hệ mờ trên không gian $X \times Y$ thì tập mờ B suy ra từ “ x là A ” và luật mờ “nếu x là A thì y là B ” được xác định bởi:

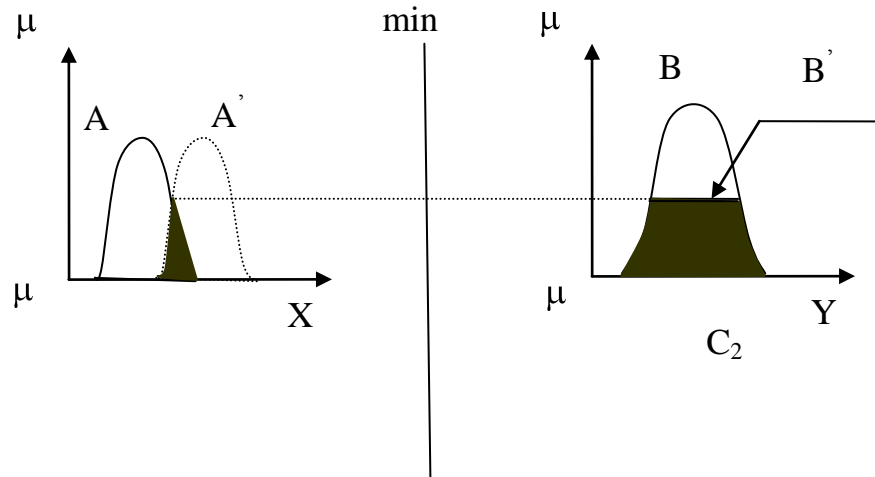
$$\mu_{B'}(y) = \max_x \min[\mu_A(x), \mu_R(x, y)] = \vee_x [\mu_A(x) \wedge \mu_R(x, y)] \quad (1.11)$$

hay tương đương

$$B' = A' \circ R = A' \circ (A \rightarrow B) \quad (1.12)$$

Các trường hợp có thể có của phép suy diễn mờ bao gồm:

a/ Suy diễn với một luật với một tiền đề:



Hình 1.13: Mô hình suy luận mờ với một luật-một tiền đề

Đây là trường hợp đơn giản nhất như đã được minh họa ở trên. Biến đổi ta có:

$$\mu_{B'}(y) = [\vee_x (\mu_A(x) \wedge \mu_{A'}(x))] \wedge \mu_B(y) = w \wedge \mu_B(y) \quad (1.13)$$

$$w = \max(\mu_A(x) \wedge \mu_{A'}(x))$$

b/ Suy luận một luật với nhiều tiền đề

Một luật nếu ... thì mờ với hai tiền đề thường được viết dưới dạng: “nếu x là A và y là B thì z là C ” và được minh họa như sau:

Giả thiết 1 (sự kiện) : x là A' và y là B'

Giả thiết 2 (luật) : Nếu x là A và y là B thì z là C

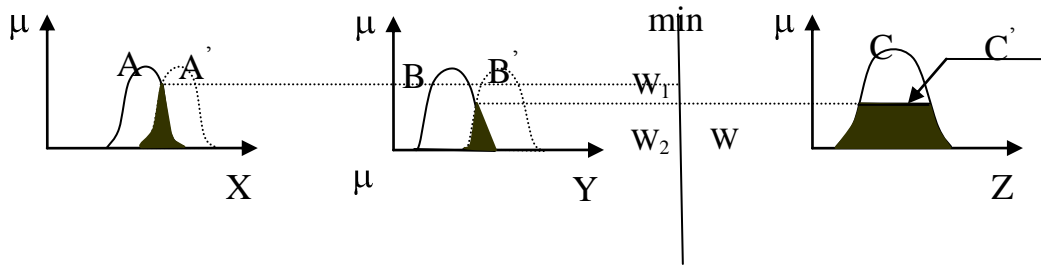
Suy diễn (kết luận) : z là C'

Luật mờ trong giả thiết 2 có thể đưa về dạng: $AxB \rightarrow C$ từ đó ta tính được:

$$C' = (A'xB') \circ R = A' \circ (AxB \rightarrow C) = [A' \circ (A \rightarrow C)] \cap [B' \circ (B \rightarrow C)] \quad (1.14)$$

$$\begin{aligned} \mu_{C'}(z) &= \bigvee_{x,y} [\mu_{A'}(x) \wedge \mu_{B'}(y)] \wedge [\mu_A(x) \wedge \mu_B(y) \wedge \mu_C(z)] \\ &= \bigvee_x [\mu_{A'}(x) \wedge \mu_A(x)] \bigwedge_y [\mu_{B'}(y) \wedge \mu_B(y)] \bigvee_z \mu_C(z) = (w_1 \wedge w_2) \vee \mu_C(z) \end{aligned} \quad (1.15)$$

Với $w_1 = \bigvee_x [\mu_{A'}(x) \wedge \mu_A(x)]$ và $w_2 = \bigvee_y [\mu_{B'}(y) \wedge \mu_B(y)]$



Hình 1.14: Mô hình suy luận mờ một luật-nhiều tiền đề

Ta lại có:

$$\begin{aligned} \mu_{C'}(z) &= \bigvee_{x,y} [\mu_{A'}(x) \wedge \mu_{B'}(y)] \wedge [\mu_A(x) \wedge \mu_B(y) \wedge \mu_C(z)] \\ &= \mu_{A' \circ (A \rightarrow C)}(y) \wedge \mu_{B' \circ (B \rightarrow C)}(y) \end{aligned} \quad (1.16)$$

Từ công thức trên ta thấy rằng, kết quả C' có thể được xem như giao của hai biểu thức $C_1' = A' \circ (A \rightarrow C)$ và $C_2' = B' \circ (B \rightarrow C)$. Mỗi một biểu thức liên quan đến phép suy diễn mờ trong trường hợp một luật mờ và một tiền đề đã xét ở trên.

c/ Nhiều luật mờ với nhiều tiền đề

Sự diễn dịch của luật mờ phức hợp được tiến hành như hợp của các quan hệ mờ tương ứng với luật mờ. Do vậy ta có:

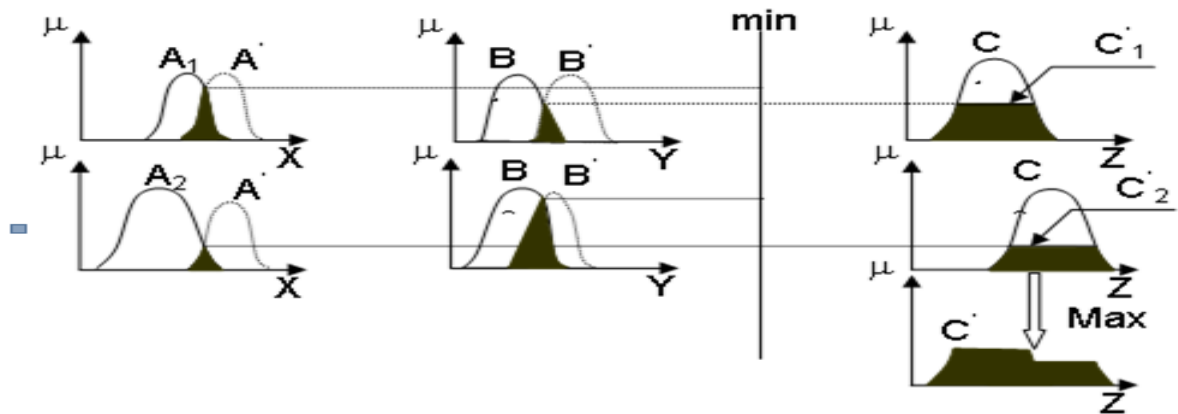
Giả thiết 1 (sự kiện): x là A' và y là B'

Giả thiết 2 (luật) : Nếu x là A_1 và y là B_1 thì z là C

Giả thiết 3 (luật) : Nếu x là A_2 và y là B_2 thì z là C_2

Suy diễn (kết luận) : z là C'

Chúng ta có thể sử dụng suy diễn mờ được trình bày trong hình 1.15 dưới đây như một thuật toán suy luận để tìm được kết quả của tập mờ đầu ra C' .



Hình 1.15: Mô hình suy luận mờ hai luật hai tiên đề

Để xác định thuật toán suy luận mờ, gọi $R_1 = A_1 \times B_1 \rightarrow C_1$ và $R_2 = A_2 \times B_2 \rightarrow C_2$.

Do phép hợp thành max-min \circ có tính chất phân phối với phép \cup nên:

$$C' = (A' \times B') \circ (R_1 \cup R_2) = [A' \times B'] \circ R_1 \cup [A' \times B'] \circ R_2 = C'_1 \cup C'_2 \quad (1.17)$$

Với C'_1 và C'_2 là tập mờ được suy diễn từ luật 1 và 2. Hình vẽ 3.7 cho thấy về mặt hình học thuật toán suy diễn mờ với luật phức hợp và nhiều tiên đề.

1.5.3. Cấu trúc của hệ thống suy luận mờ

a. Giải mờ

Sau khi thực hiện xong việc tính giá trị luật hợp thành, chúng ta thu được kết quả là một tập mờ cùng nền với tín hiệu ra [1]. Tuy nhiên, kết quả đó chưa phải là một giá trị thích hợp để điều khiển. Chẳng hạn, bài toán điều khiển tốc độ xe, tuy đã xác định được kết quả của luật điều khiển là tập mờ có hàm thuộc $\mu_R(y)$ cho tốc độ

50km, ta vẫn không biết phải chỉnh tay ga như thế nào, nói cách khác ta vẫn chưa biết phải điều chỉnh tay ga một góc mở là bao nhiêu?

Công việc xác định một góc mở tay ga cụ thể, hay nói một cách tổng quát, việc xác định một giá trị rõ y_0 từ tập mờ $\mu_R(y)$ của nó, được gọi là giải mờ. Giá trị rõ y_0 xác định được có thể được xem như “*phần tử đại diện xứng đáng*” cho tập mờ.

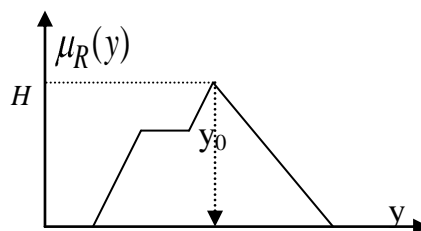
Căn cứ theo những quan niệm khác nhau về *phần tử đại diện xứng đáng* mà ta có các phương pháp giải mờ khác nhau. Trong điều khiển, người ta thường hay sử dụng hai phương pháp chính:

- Phương pháp điểm cực đại
- Phương pháp điểm trọng tâm.

* Phương pháp điểm cực đại

Tư tưởng chính của phương pháp giải mờ điểm cực đại là tìm trong tập mờ có hàm thuộc $\mu_R(y)$ một phần tử rõ y_0 với độ phụ thuộc lớn nhất. Tức là:

$$y_0 = \arg \max_y \mu_R(y)$$



Hình 1.16: Giải mờ bằng phương pháp cực đại

Tuy nhiên, do việc tìm y_0 như trên có thể dẫn tới vô số nghiệm, nên ta phải đưa thêm những yêu cầu cho phép chọn trong số các nghiệm đó một giá trị y_0 cụ thể chấp nhận được. Như vậy, việc giải mờ bằng phương pháp cực đại gồm hai bước:

- Xác định miền chứa giá trị rõ y_0 . Giá trị rõ y_0 là giá trị mà tại đó hàm thuộc đạt giá trị cực đại, tức là miền:

$$G = \{y \in Y \mid \mu_R(y) = H\}$$

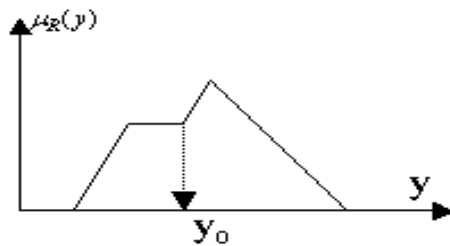
- Xác định y_0 có thể chấp nhận được từ G

* Phương pháp điểm trọng tâm

Phương pháp điểm trọng tâm sẽ cho ra kết quả y_0 là hoành độ của điểm trọng tâm miền được bao bởi trục hoành và đường cong $\mu_R(y)$

$$y_0 = \frac{\int_S y \mu_R(y) dy}{\int_S \mu_R(y) dy}$$

Với $S = \text{supp } \mu_R(y) = \{y \mid \mu_R(y) \neq 0\}$ là miền xác định của tập mờ R.

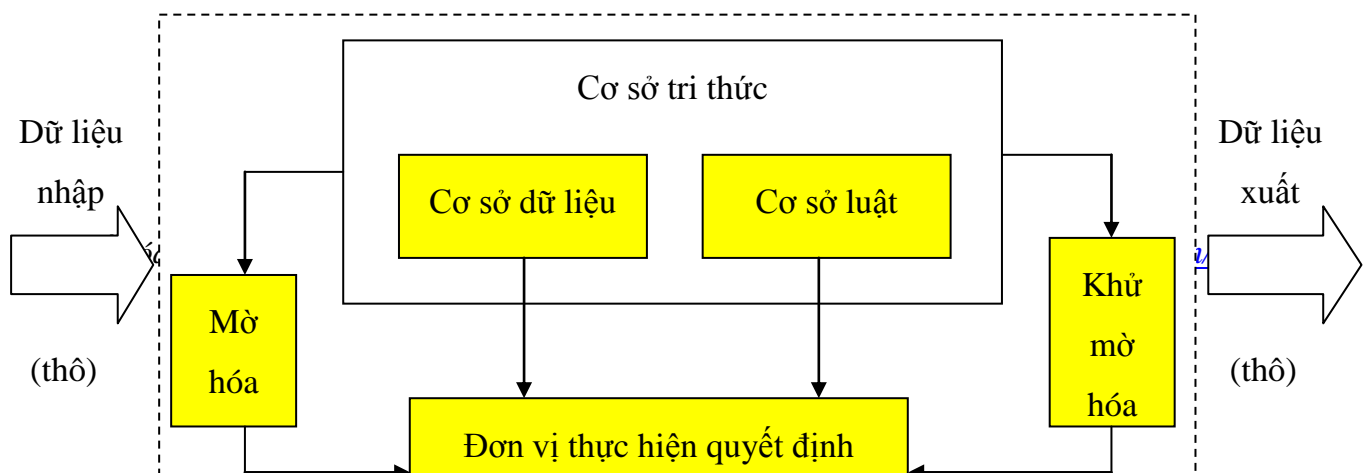


Hình 1.17: Phương pháp giải mờ điểm trọng tâm

Đây là phương pháp ưa được sử dụng nhất. Nó cho phép ta xác định giá trị y_0 với sự tham gia của tất cả các tập mờ đầu ra của luật điều khiển một cách bình đẳng và chính xác.

b. Cấu trúc của hệ thống suy luận mờ

Một hệ thống suy luận mờ gồm 5 khối cơ bản sau (Hình 1.18):



Hình 1.18. Hệ thống suy luận mờ

- **Một cơ sở luật** chứa các luật mờ nếu – thì
- **Một cơ sở dữ liệu** định nghĩa các hàm thành viên của các tập mờ được sử dụng trong các luật mờ
- **Một đơn vị thực hiện quyết định** thực hiện phép toán suy luận trên các luật
- **Một suy luận mờ** chuyển đổi dữ liệu thô thành các mức độ kết nối với biến ngôn ngữ
- **Một suy luận khử mờ** chuyển đổi các kết quả mờ của suy luận thành dữ liệu thô

Thông thường cơ sở luật và cơ sở dữ liệu được liên kết với nhau tạo nên cơ sở tri thức. Các bước của lập luận mờ (những tác vụ suy luận dựa trên các luật mờ) được thực hiện bởi hệ thống suy luận mờ gồm:

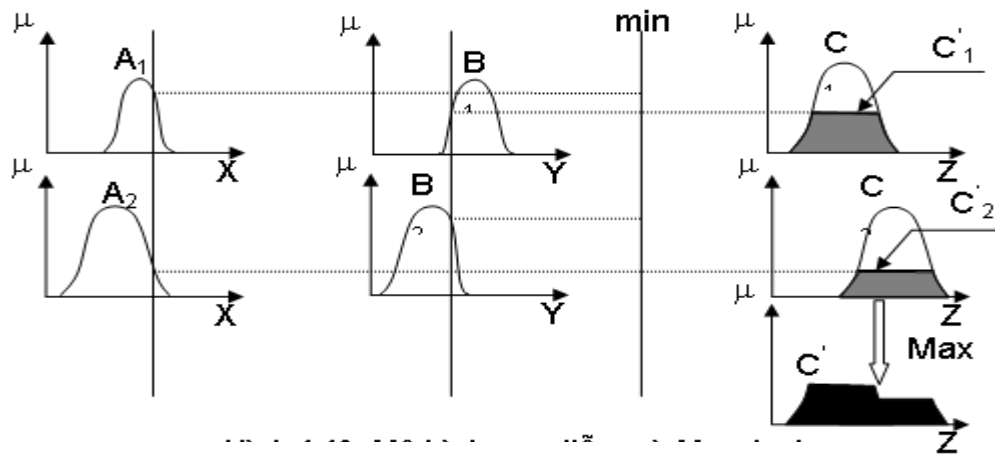
1. So sánh giá trị đầu vào với hàm thành viên của phần giả thiết để có được giá trị thành viên (hay những độ đo tương tự) của những tập ngữ nghĩa (bước này được gọi là bước mờ hóa)
2. Kết nối (qua toán tử chuẩn T-norm, thường là nhân hay lấy tối thiểu) giá trị tạo thành viên của phần giả thiết để có được trọng số vào của luật.
3. Tạo kết luận đang tính toán (có thể là mờ hay rõ) cho mỗi luật dựa vào *trọng số* của luật.
4. Tổng hợp các giá trị kết luận này để tạo kết quả rõ (bước này được gọi là giải mờ).

Có một số suy diễn mờ được sử dụng trong lĩnh vực này, dựa vào loại của lập luận mờ và loại sử dụng luật mờ if-then mà các hệ suy diễn mờ được chia ra làm ba loại như sau:

***Hệ suy diễn mờ Mamdani**

Hệ suy diễn mờ Mamdani (còn được gọi là mô hình ngôn ngữ (linguistic model(1975))) được đề xuất với mục đích ban đầu là điều khiển tổ hợp động cơ hơi nước và nồi hơi thông qua một tập luật dạng ngôn ngữ thu được từ những thao tác viên, người có kinh nghiệm. Đây là hệ suy diễn mờ điển hình nhất, với bộ luật bao gồm các luật mà phần giả thiết và phần kết luận đều là các tập mờ.

Hình 1.19 minh họa hệ suy diễn mờ Mamdani hai luật điển hình với một đầu ra z , chỉ u tác động của hai đầu vào rõ x, y với phép hợp thành min-max.

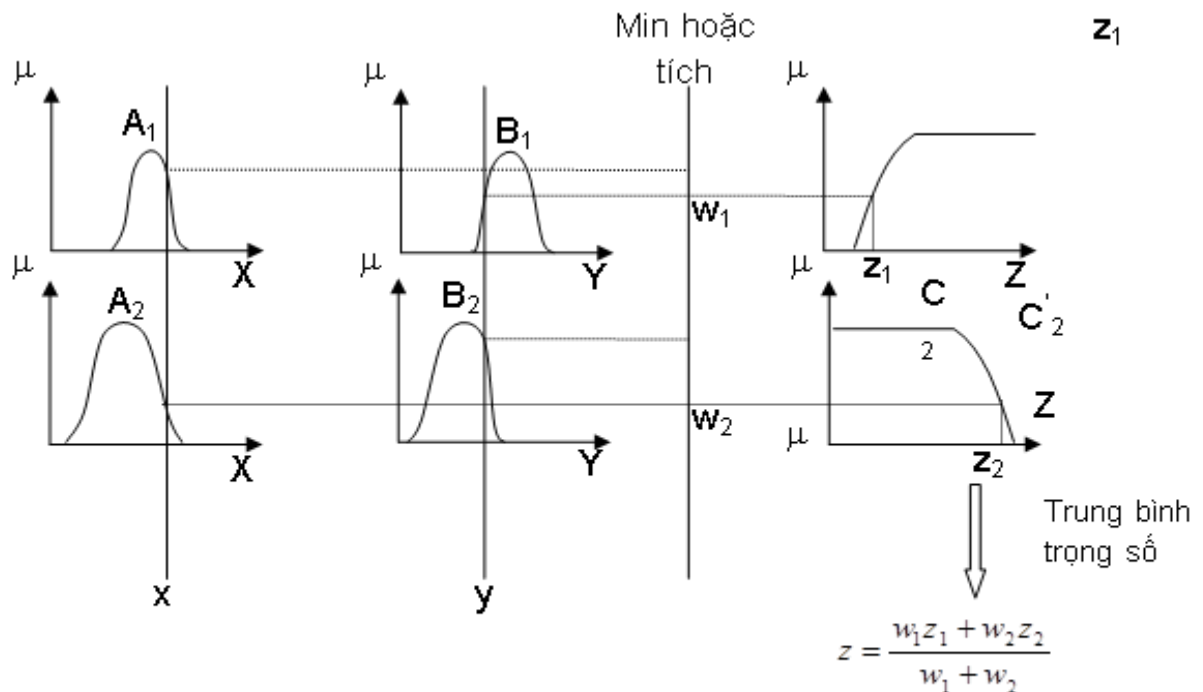


Hình 1.19. Mô hình suy diễn mờ Mamdani

Kết quả của hệ suy diễn này lại không được cụ thể, bởi vì đầu vào và đầu ra của hệ đều là mờ.

* Hệ suy diễn mờ Tsukamoto

Trong hệ suy diễn mờ Tsukamoto, phần kết luận của mỗi luật mờ *if - then* được biểu diễn bằng một tập mờ với hàm thuộc đơn điệu. Giá trị đầu ra trong hệ suy diễn mờ này là trung bình của mỗi đầu ra đã được gán trọng số của mỗi luật được xác định bởi một giá trị rõ được suy ra từ cường độ luật và hàm thuộc đầu ra.



Hình 1.20. Mô hình suy luận mờ Tsukamoto

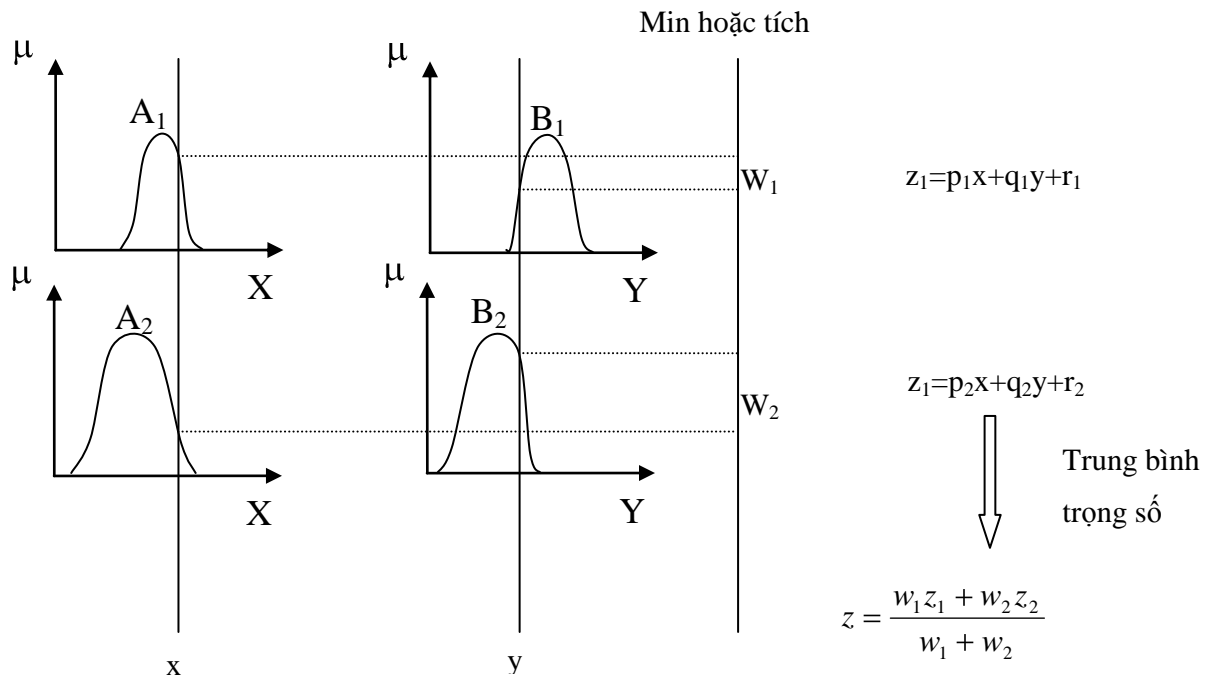
Hình 1.20 minh hoạ toàn bộ thủ tục suy diễn của hệ suy diễn mờ Tsukamoto với 2 đầu vào hai luật. Đối với hệ suy diễn mờ Tsukamoto, thì mỗi một luật đều tạo ra một giá trị ở đầu ra nên chỉ cần kết hợp đầu ra của các luật đó theo phương pháp trung bình có trọng số là ra được kết quả đầu ra. Như vậy, đối với hệ suy diễn này ta không mất nhiều thời gian và chi phí cho quá trình giải mờ như trong hệ suy diễn mờ Mamdani. Tuy nhiên, chúng ta cũng sẽ gặp phải một số khó khăn trong việc xác định hàm thuộc đầu ra của mỗi luật.

* Hệ suy diễn mờ Takagi- Sugeno

Mô hình Takagi-Sugeno (còn được gọi là mô hình TSK) được công bố bởi Takagi, Sugeno, Kang với mong muốn phát triển một cách tiếp cận có hệ thống nhằm thiết lập các luật mờ từ tập dữ liệu vào ra. Luật mờ cơ bản trong mô hình Takagi-Sugeno có dạng:

$$\text{Nếu } x \text{ là } A \text{ và } y \text{ là } B \text{ thì } z = f(x,y)$$

Với A và B là các tập mờ tiền đề trong khi $z=f(x,y)$ là một hàm số rõ trong kết luận. Thông thường $f(x,y)$ là đa thức của các biến đầu vào x và y hoặc bất cứ hàm số nào miễn là miêu tả thích hợp được đầu ra của hệ thống trong miền mờ xác định bởi các tiền đề của luật. Khi $f(x,y)$ là đa thức bậc nhất ta có mô hình Takagi-Sugeno bậc nhất, khi f là hằng số chúng ta có mô hình Takagi-Sugeno bậc không và đây có thể xem như trường hợp đặc biệt của mô hình Mamdani.



Số hóa

Hình 1.21: Hệ suy diễn mờ Takagi-Sugeno

Hình 1.21 cho thấy thuật toán suy diễn của mô hình Takagi-Sugeno bậc nhất. Do mỗi luật có một đầu ra rõ nên đầu ra của hệ thống đạt được nhờ phép tính trung bình có trọng số.

mô hình Takagi-Sugeno đơn giản có thể tạo ra các hành vi phức tạp. Có thể lấy một ví dụ minh họa mô hình Takagi-Sugeno hai đầu vào với 4 luật mờ:

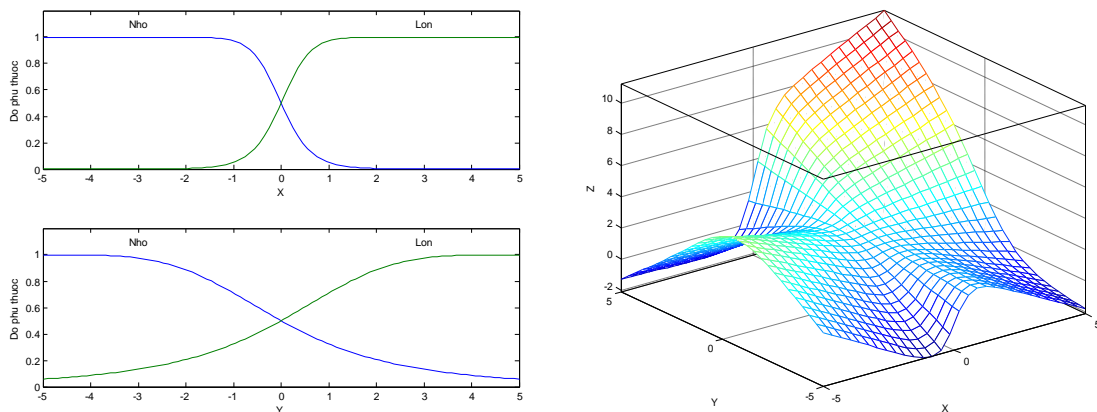
Nếu X nhỏ và Y nhỏ thì $z = -x + y + 1$

Nếu X nhỏ và Y lớn thì $z = -y + 3$

Nếu X lớn và Y nhỏ thì $z = -x + 3$

Nếu X lớn và Y lớn thì $z = x + y + 2$

Trên hình 1.22 ta có thể thấy được hàm thuộc của đầu vào X , y và bề mặt không gian đầu vào-đầu ra. Bề mặt này rất phức tạp nhưng hiển nhiên sẽ gồm bốn mặt phẳng mỗi mặt đại diện cho một phương trình đầu ra của luật mờ.



Hình 1.22: Mô hình suy luận mờ Sugeno hai đầu vào một đầu ra

Không giống như mô hình Mamdani, mô hình Takagi-Sugeno không tuân theo luật hợp thành của suy luận một cách chặt chẽ như trong các cơ chế suy luận mờ. Điều này đặt ra những khó khăn nhất định khi đầu vào của mô hình Takagi-Sugeno là mờ. Tuy nhiên, không tốn nhiều thời gian và việc thực hiện giải mờ hoá nên mô

hình Takagi-Sugeno cho tới nay vẫn là một ứng cử viên thông dụng nhất trong các mô hình học mẫu mờ.

1.3. Kết luận

Mạng nơron là một lĩnh vực đã, đang được nghiên cứu thành công và ứng dụng rộng rãi trong thực tế, tìm hiểu về hệ suy diễn mờ cũng như những ứng dụng của nó trong thực tế cho một số ứng dụng khoa học và kỹ thuật trong những năm gần đây, xấp xỉ phi tuyến, dự báo thị trường chứng khoán, dự báo mô phỏng các hệ thống điều khiển...được đưa ra, giải quyết có kết quả. Tất cả những điều đó làm cơ sở cho việc xây dựng hệ suy diễn mờ trên cơ sở mạng thích nghi.

Cũng trong chương này chúng ta đã được đề cập đến mạng thích nghi: cấu trúc cơ bản cũng như thuật toán học của mạng thích nghi. Đây là một mạng được sử dụng nhiều trong thực tế với khả năng tự thích nghi nhằm điều chỉnh các tham số trong mạng để được một đầu ra như mong đợi.

Hệ thống nơron mờ nói chung và hệ thống suy diễn nơron thích nghi mờ nói riêng đã được ứng dụng rộng rãi trong nhiều lĩnh vực. Trong lĩnh vực tài chính ngân hàng, dự báo các chỉ số kinh tế, dự báo chỉ số tài chính sử dụng mạng nơron đã được đề cập trong nhiều công trình nghiên cứu.

CHƯƠNG 2

MẠNG ANFIS VÀ KHẢ NĂNG ỨNG DỤNG

2.1. Hệ thống suy diễn mờ dựa trên mạng thích nghi

2.1.1. Các mô hình kết hợp giữa hệ mờ và mạng neural

- Neural Fuzzy system: Sử dụng mạng neural như là công cụ trong mô hình mờ.
- Fuzzy Neural Network: Mờ hóa mô hình mạng neural.
- Fuzzy Neural Hybrid System: Kết hợp kỹ thuật mờ và mạng neural vào trong các hệ lai.

Trong cách tiếp cận thứ nhất hệ thống Neural Fuzzy cung cấp cho hệ thống mờ các phương pháp điều chỉnh tự động của mạng Neural không những làm thay đổi các chức năng của chúng (Mờ hóa, giải mờ, suy diễn mờ...). Trong hệ thống Neural Fuzzy được sử dụng xử lý số thông qua các hàm thành viên để sử dụng các luật mờ.

Trong cách tiếp cận thứ hai, mạng Neural Fuzzy giữ lại các đặc tính và kiến trúc cơ bản của mạng Neural mờ hóa một vài phần tử của hệ thống. Trong hệ thống Neural Fuzzy một neural rõ có thể trở thành mờ và sự đáp ứng của neural đối với tín hiệu kích hoạt của tầng kế tiếp là quan hệ mờ.

Trong cách tiếp cận thứ ba kết hợp hệ thống mờ và mạng Neural Fuzzy vào trong một hệ thống. Trong đề tài này hệ suy diễn mờ dựa trên mạng thích nghi (ANFIS).

2.1.2. Luật mờ *if-then* và hệ suy diễn mờ

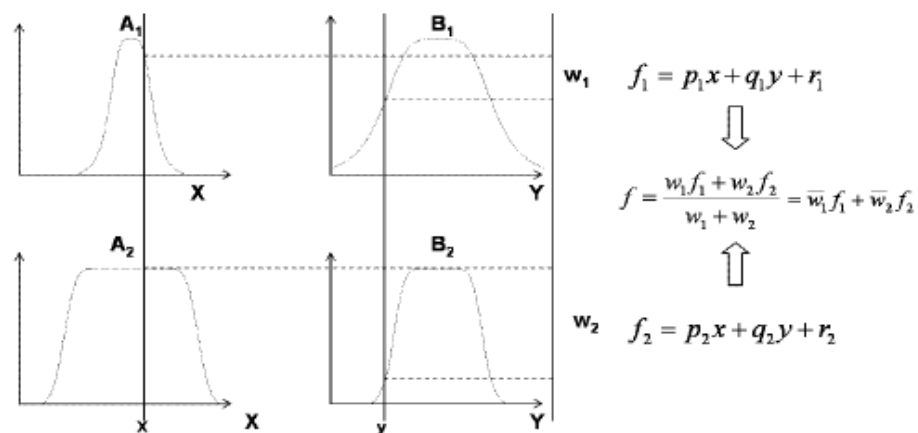
Chúng ta đã được tìm hiểu về hệ suy diễn mờ sử dụng các luật mờ *if-then*. Đây là một hệ phù hợp cho việc biểu diễn tri thức con người và xử lý suy luận mà không cần sử dụng những phân tích chính xác. Tuy nhiên, nó cũng có một số hạn chế nhất định như:

- Không có một phương pháp chuẩn cho việc biến đổi các tri thức con người hoặc các kinh nghiệm vào các luật và dữ liệu của một hệ suy diễn mờ.

- Cần có một phương pháp đem lại hiệu quả cho việc điều chỉnh các hàm thuộc (MF's) để tối thiểu hoá lỗi đầu ra.

Một mô hình kiến trúc khác có thể khắc phục được những hạn chế đó một cách tốt nhất đó là mô hình hệ suy diễn mờ dựa trên mạng thích nghi (ANFIS). Mô hình này sẽ cho phép tạo lên một tập các luật mờ *if then* với việc xấp xỉ các hàm thuộc để sinh ra các cặp dữ liệu vào ra theo điều kiện. Phần tiếp theo trong chương này, chúng ta sẽ đi tìm hiểu kiến trúc hệ suy diễn mờ trên cơ sở mạng thích nghi.

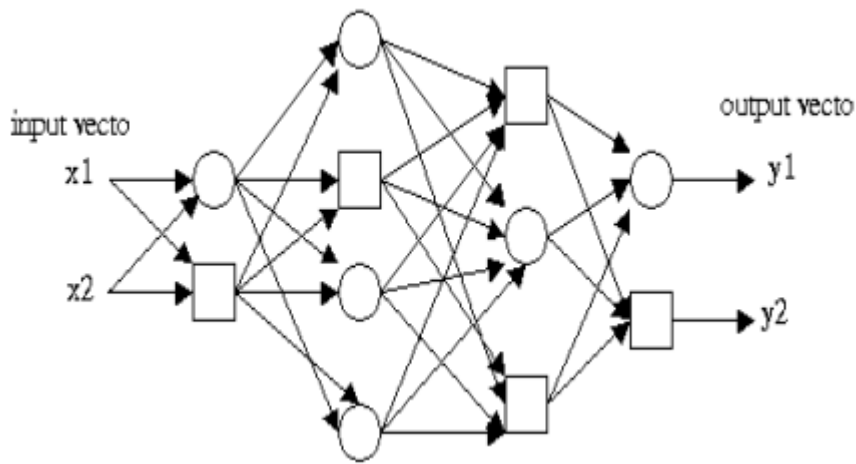
Trong phần trước đó, chúng ta đã được tìm hiểu về kiến trúc của mạng thích nghi. Trong phần này chúng ta sẽ đưa ra cấu trúc của mạng thích nghi tương đương với hệ suy diễn mờ, trong đó chúng ta sẽ đi sâu vào mô tả cách phân chia các tập tham số để áp dụng luật học lai và cách thực hiện các luật mờ *if then*.



Hình 2.1. Lập luận mờ

2.1.3. Cấu trúc mạng ANFIS

Mạng thích nghi là một mạng lan truyền thẳng gồm có nhiều lớp, mỗi lớp có nhiều nút khác nhau, mỗi nút thực hiện một hàm đặc biệt (còn được gọi là nút hàm - node function) trên dữ liệu đến nút đó như một tập các tham số thuộc về lớp đó. Để phản ánh khả năng thích nghi khác nhau, người ta sử dụng hai loại nút, đó là nút hình tròn và nút hình vuông. Trong đó, nút hình tròn (nút cố định) không có tham số, còn nút hình vuông (nút thích nghi) có các tham số. Tập các tham số của mạng thích nghi là hợp của các tham số của mỗi nút.



Hình 2.2. Mạng thích nghi

Bình thường, các nút hàm có thể biến đổi từ nút này đến nút khác và việc chọn mỗi một nút hàm phụ thuộc vào tất cả các hàm vào ra mà mạng thích nghi yêu cầu thực hiện. Hình 2.2 mô tả cấu trúc của mạng thích nghi. Chú ý rằng, các liên kết trong mạng thích nghi chỉ cho biết tín hiệu trực tiếp giữa các nút, không có trọng số được kết hợp với liên kết. Như vậy, đối với mạng thích nghi để có thể đưa ra được một đầu ra mong đợi thì những tham số được cập nhật theo tập dữ liệu huấn luyện, thủ tục học trên cơ sở gradient được mô tả dưới đây:

Giả sử rằng chúng ta có một mạng thích nghi gồm L lớp và lớp thứ k có $\#k$ nút. Ta có thể ký hiệu vị trí của nút thứ i trong lớp k là (k, i) và hàm của nút (nút ra) là O_i^k . Vì nút ra phụ thuộc vào dữ liệu đến nút và tập các tham số, nên ta có:

$$O_i^k = O_i^k (O_i^{k-1}, \dots, O_{\#(k-1)}^{k-1}, a, b, c \dots) \quad (2.1)$$

Ở đây a, b, c là các tham số thuộc về nút này.

Giả sử rằng có tập dữ liệu huấn luyện P đầu vào, chúng ta có thể định nghĩa lỗi cho đầu vào thứ p ($1 \leq p \leq P$) của dữ liệu huấn luyện là tổng sai số bình phương.

$$J_p = \sum_{i=1}^{\#(L)} (T_{i,p} - O_{i,p}^L)^2 \quad (2.2)$$

Trong đó, $T_{i,p}$ là thành phần thứ i của vectơ đầu ra đích thứ p , $O_{i,p}^L$ là thành phần thứ i của vectơ đầu ra thực tế có được từ vectơ đầu vào thứ p . Do vậy, nếu ta có P đầu vào thì tổng lỗi là:

$$J = \sum_{p=1}^P J_p \quad (2.3)$$

Để phát triển thủ tục học sử dụng phương pháp giảm gradient trong J trên không gian tham số, trước tiên chúng ta đi tính tỷ lệ lỗi $\frac{\partial J_p}{\partial O}$ cho dữ liệu huấn luyện thứ p và cho mỗi nút ra O. Tỷ lệ lỗi cho nút ra tại (L, i) có thể được tính theo phương trình sau:

$$\frac{\partial J_p}{\partial O_{i,p}^L} = -2(T_{i,p} - O_{i,p}^L) \quad (2.4)$$

Tại nút (k,i) bên trong, tỷ lệ lỗi có được tính bởi công thức sau:

$$\frac{\partial J_p}{\partial O_{i,p}^k} = \sum_{m=1}^{\#(k+1)} \frac{\partial J_p}{\partial O_{m,p}^{k+1}} \frac{\partial O_{m,p}^{k+1}}{\partial O_{i,p}^k} \quad (2.5)$$

Với $1 \leq k \leq L-1$. Hay tỷ lệ lỗi của mỗi nút trong có thể được diễn tả như một sự kết hợp tuyến tính tỷ lệ các lỗi trong lớp tiếp theo. Do đó, với tất cả các $1 \leq k \leq L$ và $1 \leq i \leq \#(k)$, chúng ta có: $\frac{\partial J_p}{\partial O_{i,p}^k}$ từ công thức (2.4) và (2.5)

Bây giờ, nếu α là tham số của mạng thích nghi, khi đó ta có:

$$\frac{\partial J_p}{\partial \alpha} = \sum_{O^* \in S} \frac{\partial J_p}{\partial O^*} \frac{\partial O^*}{\partial \alpha} \quad (2.6)$$

Trong đó, S là tập các nút có đầu ra phụ thuộc α . Khi đó đạo hàm của J theo α là:

$$\frac{\partial J}{\partial \alpha} = \sum_{p=1}^P \frac{\partial J_p}{\partial \alpha} \quad (2.7)$$

Và như vậy, công thức cập nhật cho tham số di truyền α là:

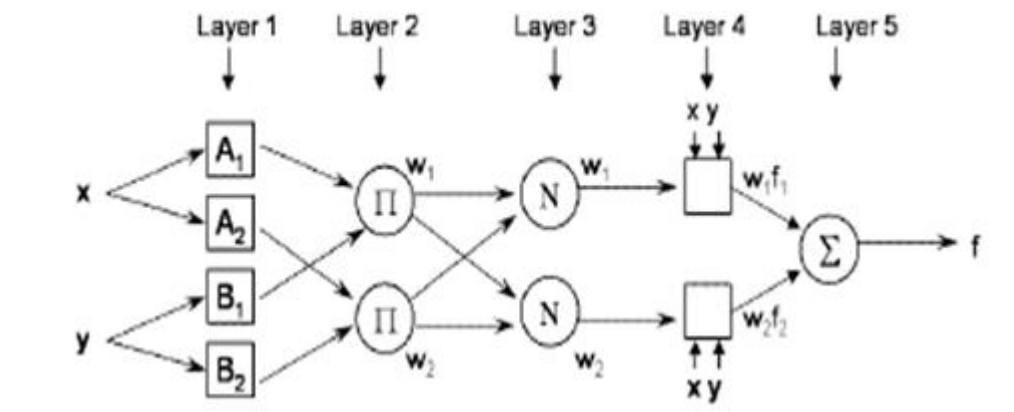
$$\Delta \alpha = -\gamma \frac{\partial J}{\partial \alpha} \quad (2.8)$$

Ở đây γ là tỷ lệ học và có giá trị là:

$$\gamma = \frac{k}{\sqrt{\sum_{\alpha} \left(\frac{\partial J}{\partial \alpha}\right)^2}} \quad (2.9)$$

Với k là kích thước bước, chiều dài của mỗi lần chuyển đổi gradient trong không gian tham số. Chúng ta cũng có thể biến đổi giá trị của k từ sự biến đổi tốc độ hội tụ.

Như vậy, chúng ta đã đưa ra được công thức cập nhật cho tham số α dựa vào công thức (2.8). Tuy nhiên, phương pháp gradient trên để xác định các tham số trong mạng thích nghi nhìn chung chậm và hay dẫn đến cực tiểu nội bộ.



Hình 2.3. Kiến trúc mạng ANFIS

Để đơn giản, ta giả sử rằng hệ suy diễn mờ có hai đầu vào x và y và một đầu ra. Giả sử rằng có hai luật cơ sở mờ *if - then* của Takagi và Sugeno như sau:

Rule 1 : If x is A_1 and y is B_1 , then $f_1 = p_1x + q_1y + r_1$

Rule 2 : If x is A_2 and y is B_2 , then $f_2 = p_2x + q_2y + r_2$

Các nút trong cùng một lớp có chức năng như nhau như mô tả dưới đây:

* **Lớp 1:** Tầng này gồm $n*m$ nút thích nghi (nút hình vuông) với hàm của nút là

$$O_{i,j}^1 = \mu_{A_{i,j}}(X_i), \quad (2.10)$$

Trong đó $X_i (0 \leq i \leq n-1)$ là đầu vào thứ i , $A_{i,j} (0 \leq i \leq n-1, 0 \leq j \leq m-1)$ là nhãn ngôn ngữ thứ j của đầu vào i , (ví dụ như nhỏ, trung bình, lớn...). $\mu_{A_{ij}}$ là hàm thuộc của $A_{i,j}$ và nó chỉ rõ mức độ X_i thuộc về $A_{i,j}$. Thông thường ta chọn hàm thuộc là hàm Chuông (Bell) hoặc hàm Gaussian với giá trị trả về nằm trong đoạn $[0,1]$.

$$\mu_{gbell}(x) = \frac{1}{1 + \left| \frac{x - c_k}{a_k} \right|^{2b_k}}, \quad \mu_{gaussian}(x) = \exp\left[-\left(\frac{x - c_k}{s_k}\right)^2\right] \quad (2.11)$$

Với $\{a_k, b_k, c_k\}$ hay $\{a_k, c_k\}$ ($0 \leq k \leq n*m-1$) là tập tham số phi tuyến của nút k . Khi những giá trị của tham số thay đổi, hình dạng của hàm thành viên của nhãn ngôn A_{ij} sẽ thay đổi theo.

* **Lớp 2:** Mỗi nút trong lớp này có hình tròn và có nhãn là Π . Nó chính là tích của các tín hiệu đến và mỗi giá trị đầu ra của nút biểu diễn cường độ của một luật (toán tử T - norm thực hiện phép AND tổng quát có thể được sử dụng ở đây)

$$O_i^2 = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \quad i = 1, 2 \quad (2.12)$$

* **Lớp 3:** Mỗi nút trong lớp này có hình tròn có nhãn là N . Nút thứ i được tính là tỉ lệ của cường độ luật của nút thứ i so với tổng tất cả các cường độ luật:

$$O_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2 \quad (2.13)$$

Để thuận tiện, giá trị đầu ra của nút này sẽ được gọi là giá trị đầu vào luật được chuẩn hoá (*normalized firing strengths*)

* **Lớp 4:** Mỗi lớp i trong lớp này là một nút hình vuông có chức năng

$$O_1^4(x) = \bar{w}_i f_i = \bar{w}_i(p_x + q_i y + r_i)$$

(2.14)

Trong đó, \overline{w}_i là đầu ra của lớp thứ ba. $\{p_i, q_i, r_i\}$ là tập các tham số, các tham số trong lớp này sẽ được xem như là tham số kết luận.

* **Lớp 5:** Là một lớp có một nút hình tròn, trong lớp này có ký hiệu là \sum là đầu ra bằng tổng của tất cả các tín hiệu đầu vào.

$$O_1^5(x) = \sum_i \overline{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (2.15)$$

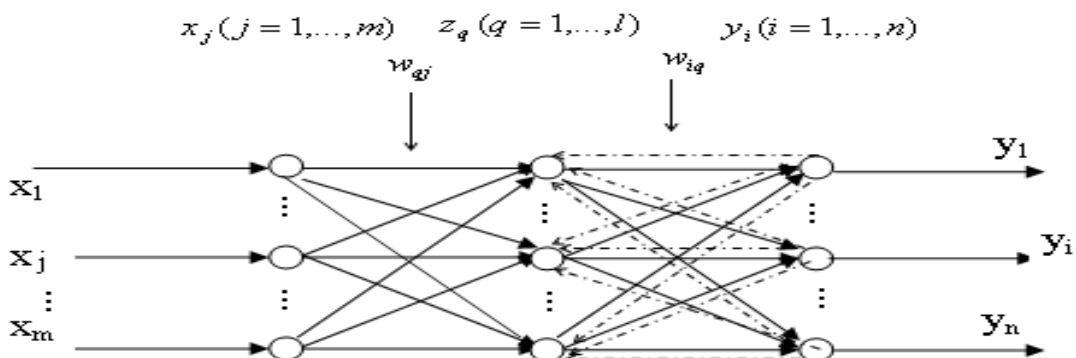
2.2. Các thuật toán mạng ANFIS

2.2.1. Thuật toán học lan truyền ngược

Thuật toán lan truyền ngược (BP) được giới thiệu lần đầu vào năm 1970 bởi Werbos. Tập hợp các tham số được cập nhật suốt trong quá trình huấn luyện dữ liệu bởi phương pháp độ dốc gradient. Hầu hết mạng nơ-ron cơ bản hiện nay được huấn luyện bởi thuật toán BP.

Lan truyền ngược là một phương pháp cho phép xác định tập trọng số tốt nhất của mạng để giải một bài toán đã cho. Việc áp dụng phương pháp lan truyền ngược là một quá trình lặp đi lặp lại nhiều lần hai tiến trình chính: Lan truyền tiến để thực hiện ánh xạ và lan truyền ngược sai số để cập nhật các trọng số. Các trọng số của mạng là các hệ số của mô hình. Phương pháp giảm gradient được dùng để cập nhật những hệ số này sao cho giảm thiểu được sai số của mô hình.

Xét lớp mạng 3 lớp như hình (2.4)



Hình 2.4: Mạng 3 lớp lan truyền ngược

Thuật toán: Đầu tiên ta cho lan truyền thẳng suốt trong mạng, qua các phần tử noron và được tiếp tục với các hàm kích hoạt của phần tử noron. Các mạng được nghiên cứu cùng với thuật toán học lan truyền ngược được gọi là mạng lan truyền ngược.

Huấn luyện các cặp vào/ra.

$$\{(x^{(k)}, d^{(k)})\}, \quad k = 1, 2, \dots, p$$

Thuật toán cung cấp một thủ tục cho việc thay đổi các vector trọng số trong mạng, đầu ra của mạng được lan truyền ngược trở lại lớp đầu vào cho đúng các mẫu. Cơ sở cho việc cập nhật các trọng số là phương pháp độ dốc Gradient.

Với cặp vào ra $(x^{(k)}, d^{(k)})$, thuật toán lan truyền ngược thực hiện các bước như sau:

Đầu tiên, mẫu $x^{(k)}$ được lan truyền từ lớp đầu vào qua các lớp ẩn đi đến lớp đầu ra có kết quả là $y^{(k)}$. Sau đó, sai số giữa $y^{(k)}$ và $d^{(k)}$ được lan truyền ngược trở lại từ lớp đầu ra tới lớp đầu vào để cập nhật trọng số. Hình (2.4) diễn giải thuật toán lan truyền ngược. Kết quả có thể mở rộng sang mạng noron nhiều lớp.

Trên hình (2.4) có m phần tử noron đầu vào, l phần tử noron ở lớp ẩn, và n phần tử noron ở lớp đầu ra. Đường nét liền diễn tả lan truyền thẳng của các tín hiệu, đường nét đứt diễn tả lan truyền ngược của các sai số. Đầu tiên huấn luyện vào cặp vào/ra ký hiệu (x, d) để cho đơn giản ta bỏ chỉ số k . Khi một mẫu đầu vào x được đưa vào thì các phần tử trong mạng sẽ được tính như sau:

Đầu vào phần tử q của lớp ẩn sẽ được tính theo phương trình:

$$net_q = \sum_{j=1}^m v_{qj} x_j \quad (2.16)$$

$$\text{Phương trình đầu ra của } q \text{ sẽ là: } z_q = a(net_q) = a\left(\sum_{j=1}^m v_{qj} x_j\right) \quad (2.17)$$

Đầu vào phần tử thứ i của lớp đầu ra sẽ là:

$$net_i = \sum_{q=1}^l w_{iq} z_q = \sum_{q=1}^l w_{iq} a\left(\sum_{j=1}^m v_{qj} x_j\right) \quad (2.18)$$

Phương trình đầu ra của phần tử nơron thứ i sẽ là:

$$y_i = a(net_i) = a\left(\sum_{q=1}^l w_{iq} z_q\right) = a\left(\sum_{q=1}^l w_{iq} \cdot a\left(\sum_{j=1}^m v_{qj} x_j\right)\right) \quad (2.19)$$

Các chỉ số trên được tính toán cho sự lan truyền tiến của các tín hiệu đầu vào xuyên suốt qua các lớp mạng nơron. Trước khi ta đề cập đến các tín hiệu sai số của sự lan truyền ngược, ta sẽ định nghĩa một hàm mục tiêu như sau:

$$E(w) = \frac{1}{2} \sum_{i=1}^n (d_i - y_i)^2 = \frac{1}{2} \sum_{i=1}^n \left[d_i - a\left(\sum_{q=1}^l w_{iq} z_q\right) \right]^2 \quad (2.20)$$

Sau đó, theo phương pháp độ dốc Gradient, các trọng số nối giữa lớp ẩn và lớp đầu ra được cập nhật bởi Δw_{iq} , và nó được tính theo công thức sau:

$$\Delta w_{iq} = -\eta \frac{\partial E}{\partial w_{iq}} \quad (2.21)$$

Sử dụng các công thức (1.19-1.21) và thay đổi luật với $\frac{\partial E}{\partial w_{iq}}$, ta có:

$$\Delta w_{iq} = -\eta \left[\frac{\partial E}{\partial y_i} \right] \left[\frac{\partial y_i}{\partial net_i} \right] \left[\frac{\partial net_i}{\partial w_{iq}} \right] = \eta [d_i - y_i] [a'(net_i)] [z_q] = \eta \delta_{oi} z_q \quad (2.22)$$

Trong đó, δ_{oi} là tín hiệu sai số, chỉ số dưới thứ hai là điểm thứ i trong lớp đầu ra. Sai số tín hiệu được định nghĩa bởi:

$$\delta_{oi} = -\frac{\partial E}{\partial net_i} = -\left[\frac{\partial E}{\partial y_i} \right] \left[\frac{\partial y_i}{\partial net_i} \right] = [d_i - y_i] a'(net_i) \quad (2.23)$$

Trong đó, net_i là đầu vào của phần tử nơron thứ i trong lớp đầu ra và $a'(net_i) = \frac{\partial a(net_i)}{\partial net_i}$.

Bây giờ ta phải tính đầu ra z_q của lớp ẩn:

Với trọng số nối giữa đầu vào và các lớp ẩn, ta sử dụng thay đổi luật cùng phương pháp độ dốc Gradient, ta cập nhật trọng số để kết nối giữa phần tử thứ j của lớp đầu vào với phần tử thứ q của lớp ẩn. Khi đó:

$$\Delta v_{qj} = -\eta \left[\frac{\partial E}{\partial v_{qj}} \right] = -\eta \left[\frac{\partial E}{\partial net_q} \right] \left[\frac{\partial net_q}{\partial v_{qj}} \right] = -\eta \left[\frac{\partial E}{\partial z_q} \right] \left[\frac{\partial z_q}{\partial net_q} \right] \left[\frac{\partial net_q}{\partial v_{qj}} \right] \quad (2.24)$$

Từ công thức (2.20), thì mỗi sai số $[d_i - y_i]$, với $i=1,2,...,n$ là một hàm của z_q .

Đánh giá thay đổi luật ta có:

$$\Delta v_{qj} = -\eta \sum_{i=1}^n [d_i - y_i] a'(net_i) \cdot w_{iq} a'(net_q) \cdot x_j \quad (2.25)$$

Sử dụng công thức (2.23), ta có thể viết lại công thức (2.33) như sau:

$$\Delta v_{qj} = -\eta \sum_{i=1}^n [d_{oi} \cdot w_{iq}] a'(net_q) \cdot x_j = \eta \partial_{hq} x_j \quad (2.26)$$

Ở đây, ∂_{hq} là sai số tín hiệu của phần tử thứ q của lớp ẩn và được định nghĩa như dưới đây:

$$\partial_{hq} = -\frac{\partial E}{\partial net_q} = -\left[\frac{\partial E}{\partial z_q} \right] \left[\frac{\partial z_q}{\partial net_q} \right] = a'(net_q) \sum_{i=1}^n \delta_{oi} w_{iq} \quad (2.27)$$

Trong đó, net_q là đầu vào phần tử thứ q của lớp ẩn.

Tín hiệu sai số của một phần tử trong lớp ẩn khác so với tín hiệu sai số của một phần tử trong lớp đầu ra, như đã được chỉ ra trong công thức (2.27) và (2.23). Do có sự khác nhau này, nên các thủ tục cập nhật các trọng số trên được gọi là luật học delta tổng quát. Chúng ta xem xét công thức (2.27), sai số tín hiệu ∂_{hq} của phần tử lớp ẩn q có thể được xác định trong các mẫu của các tín hiệu sai số δ_{oi} của các phần tử ở lớp ra thứ i (y_i) cung ứng. Các hệ số là các trọng số được sử dụng cho lan truyền thẳng, nhưng ở đây chúng truyền các tín hiệu sai số (δ_{oi}) ngược trở lại, đó chính là các đường nét đứt trong hình (2.4). Điều này đã chứng tỏ được đặc điểm quan trọng của thuật toán lan truyền ngược - luật cập nhật cục bộ, đây chính là tính

toán trọng số thay đổi dựa vào sự kết nối và chúng ta chỉ cần giá trị ở hai đầu của kết nối này.

Sự đạo hàm ở trên có thể dễ dàng mở rộng cho mạng có nhiều hơn một lớp ẩn, bằng cách sử dụng chuỗi luật liên tiếp. Trong trường hợp chung, với số lớp tùy ý, thì sự lan truyền ngược được cập nhật luật ở dạng sau:

$$\Delta w_{ij} = \eta \delta_i x_j = \eta \delta_{\text{output}-i} x_{\text{input}-j} \quad (2.28)$$

Ở đây, (output-i) và (input-j) quy vào hai đầu của sự kết nối từ phần tử thứ j tới phần tử thứ i, x_j là đầu vào cuối cùng kích hoạt từ một phần tử lớp ẩn, hoặc từ một đầu vào bên ngoài. Ngoài ra, δ_i là tín hiệu học được định nghĩa bởi công thức (2.23) với đầu ra hoặc lớp cuối cùng của các trọng số kết nối, và được định nghĩa bởi công thức (2.27) cho tất cả các lớp khác. Khi hàm sigmoid lưỡng cực được sử dụng làm hàm kích hoạt, đồng thời sử dụng (2.23) và (2.27) ta có hàm y được xác định như sau:

$$y = a(\text{net}) = \frac{1}{1 + e^{-\text{net}}} - 1$$

$$\text{Khi đó ta có: } a'(\text{net}) = \frac{\partial a(\text{net})}{\partial \text{net}} = \frac{1}{2} [1 - a^2(\text{net})] = \frac{1}{2} (1 - y^2)$$

$$\delta_{oi} = \frac{1}{2} (1 - y^2) [y_i - y_i] \quad (2.29)$$

$$\delta_{hq} = \frac{1}{2} (1 - z^2) \sum_{i=1}^n \delta_{oi} w_{iq} \quad (2.30)$$

Thuật toán lan truyền ngược

Xem xét một mạng với Q lớp lan truyền ngược, $q=1,2,\dots,Q$; với ${}^q\text{net}_i$ và qy_i lần lượt là đầu vào và đầu ra của khối trong lớp thứ q. Mạng có m nơron đầu vào, l nơron ở lớp ẩn, và n nơron đầu ra. Với ${}^qw_{ij}$ là trọng số nối từ ${}^{q-1}w_j$ đến qy_i .

Đầu vào: Các cặp huấn luyện $\{x^{(k)}, d^{(k)} \mid k=1,2,\dots,p\}$, ở đó giá trị đầu vào của phần tử cuối cùng bằng -1, tức là $x_{m+1}^{(k)} = -1$.

Bước 0 (Đặt giá trị ban đầu)

- Lựa chọn bước tính (Hằng số học) $0 < \eta < 1$ và E_{\max} (sai số lớn nhất cho phép).
- Chọn các trọng số ban đầu nối từ phần tử thứ j của lớp $(q - 1)$ đến phần tử thứ i của lớp q là ${}^q w_{ij}$ có giá trị nhỏ và ngẫu nhiên.
- Cho sai số $E = 0$ và $k = 1$.

Bước 1 (Vòng lặp huấn luyện)

Áp dụng mẫu vào thứ k , cho lớp đầu vào $q=1$. Khi đó ta có:

$${}^q y_i = {}^1 y_i = x_i^{(k)} \text{ cho tất cả các } i = 1, 2, 3, \dots, m. \quad (2.31)$$

Bước 2 (Lan truyền thẳng)

Lan truyền tín hiệu thẳng xuyên suốt mạng cho mỗi i và q cho tới khi các đầu ra của lớp đầu ra ${}^Q y_i$ được thực hiện.

$${}^q y_i = a({}^q net_i) = a\left(\sum_j {}^q w_{ij} {}^{q-1} y_j\right) \quad (2.32)$$

Bước 3 (Đo lường sai số đầu ra)

Tính toán giá trị sai lệch và tín hiệu sai lệch ${}^Q \delta_i$ cho lớp đầu ra như sau:

$$E = \frac{1}{2} \sum_{i=1}^n (d_i^{(k)} - {}^Q y_i)^2 + E; \quad (2.33)$$

$${}^Q \delta_i = (d_i^{(k)} - {}^Q y_i) \cdot a'({}^Q net_i) \quad (2.34)$$

Trong đó: ${}^Q \delta_i$ là tín hiệu sai lệch của nơron thứ i cho lớp ra Q ; $a'({}^Q net_i)$ là đạo hàm của hàm truyền $a(.)$ theo tổng trọng số của phần tử i của lớp đầu ra là ${}^Q net_i$.

$$a'({}^Q net_i) = \frac{da}{d({}^Q net_i)}$$

Bước 4 (lan truyền ngược sai số)

Các sai số lan truyền ngược với mục đích để cập nhật các trọng số và tính toán các tín hiệu sai lệch ${}^{q-1}\delta_i$ cho các lớp xử lý:

$$\Delta^q w_{ij} = \eta \cdot {}^q\delta_i \cdot {}^{q-1}y_j; \quad {}^q w_{ij}^{new} = {}^q w_{ij}^{old} + \Delta^q w_{ij} \quad (2.35)$$

$${}^{q-1}\delta_i = a'({}^{q-1}net_i) \sum_j {}^q w_{ji} {}^q\delta_j; \quad \text{với } q=Q, Q-1, \dots, 2 \quad (2.36)$$

Trong đó:

- $\Delta^q w_{ij}$ là sai lệch tại thời điểm tính của giá trị trọng số liên kết cập nhật mới và cũ, liên kết từ phần tử thứ j của lớp q-1 đến phần tử i của lớp q.
- ${}^q w_{ij}^{new}$ là giá trị trọng số liên kết cập nhật mới từ phần tử thứ j của lớp (q-1) đến phần tử i của lớp q.
- ${}^q w_{ij}^{old}$ là giá trị trọng số liên kết cũ từ phần tử thứ j của lớp (q-1) đến phần tử i của lớp q.
- ${}^{q-1}y_j$ là tín hiệu ra của phần tử j của lớp (q-1).

Bước 5 (Sau mỗi vòng lặp)

Kiểm tra xem đã lặp hết các giá trị mẫu huấn luyện chưa, nếu chưa quay vòng hết (tức là $k < p$) tăng $k=k+1$, và nhảy tới bước 1, ngược lại (tức $k=p$) thì chuyển sang bước 6.

Bước 6 (Kiểm tra tổng sai số)

Kiểm tra sự khác nhau giữa tổng sai số và sai số cho phép:

- Nếu tổng sai số nhỏ hơn sai số cho phép (tức là $E < E_{max}$) thì kết thúc quá trình huấn luyện, và ghi lại các giá trị trọng số cuối cùng.
- Trái lại, thì lại gán $E=0$, $k=1$ và bắt đầu một quá trình huấn luyện mới bằng cách nhảy tới bước 1.

2.2.2. Thuật toán học lai

Để có thể đem lại hiệu quả cao nhất trong quá trình tính toán, hệ ANFIS sử dụng thuật toán học lai. Từ kiến trúc mạng ANFIS đã cho trong hình (2.2), chúng ta thấy rằng nếu cho giá trị các tham số giả thiết, đầu ra có thể được diễn tả bởi một hàm kết hợp tuyến tính các tham số kết luận. Nói một cách cụ thể, đầu ra f trong hình (2.2) có thể được cụ thể hoá như sau:

$$f = \overline{w_1}f_1 + \overline{w_2}f_2 = (\overline{w_1})r_1 + (\overline{w_1}x)p_1 + (\overline{w_1}y)q_1 + (\overline{w_2})r_2 + (\overline{w_2}x)p_2 + (\overline{w_2}y)q_2 \quad (2.37)$$

f là tuyến tính trong các tham số kết quả ($p_1, q_1, r_1, p_2, q_2, r_2$). Kết quả chúng ta có:

S = tập của tổng các tham số

S_1 = tập của các tham số giả thiết

S_2 = tập của các tham số kết luận

Cụ thể hơn, trong pha đi tới của thuật toán học lai, hàm tín hiệu đi qua bốn lớp (đi đến lớp thứ 4) và tham số kết luận được xác định bởi phương pháp ước lượng bình phương tối thiểu (LSE). Còn trong pha đi lui, tỉ lệ lỗi được lan truyền ngược và các tham số giả thiết được cập nhật bởi phương pháp giảm gradient, các hoạt động trong mỗi pha hoạt động được tổng quát hoá qua bảng sau:

	Pha đi tới	Pha đi lui
Tham số giả thiết	Cố định	Giảm gradient
Tham số kết luận	LSE	Cố định
Tín hiệu	Nút ra	Tỉ lệ lỗi

Bảng 2.1: Hai pha trong thủ tục học lai cho hệ ANFIS

Theo như bảng trên, chúng ta nhận thấy, các công thức cập nhật các tham số giả thiết và kết luận được tách ra trong luật học lai. Do đó, việc tăng tốc độ học là có thể bằng cách sử dụng một cách khác của phương pháp giảm gradient trong phần giả thiết như giảm gradient lan truyền nhanh, tối ưu hoá phi tuyến...

Trên thực tế, với mỗi phương pháp đều có những ưu và nhược điểm riêng của nó. Phương pháp ước lượng bình phương tối thiểu và phương pháp giảm gradient cũng vậy. Do đó, người ta đã đưa ra được một số cách để cập nhật tham số (tùy từng trường hợp cụ thể áp dụng vì mỗi một phương pháp đều có một độ phức tạp tính toán nhất định nào đó)

- Chỉ giảm gradient: Tất cả các tham số được cập nhật bởi phương pháp giảm gradient.
- Giảm gradient và chỉ một lần dùng LSE, trong đó LSE được sử dụng chỉ một lần tại thời điểm bắt đầu đặt các giá trị khởi tạo của các tham số kết quả và sau đó sử dụng giảm gradient để cập nhật tất cả các tham số.
- Giảm gradient và LSE: Đây là luật học lai.

- Chỉ LSE: Hệ ANFIS là tuyến tính và các tham số giả thiết và sự mở rộng thuật toán lọc Kalman được dùng để cập nhật tất cả các tham số

2.3. Ứng dụng của mạng ANFIS

Mạng Anfis được sử dụng thành công trong nhiều lĩnh vực khác nhau, dự đoán cổ phiếu trên thị trường chứng khoán, dự báo chỉ số kinh tế tài chính, dự báo mô phỏng các hệ thống điều khiển, bài toán nhận dạng, các ứng dụng ngân hàng, quân sự, khoa học như dự báo thời tiết, chuẩn đoán, theo dõi.

CHƯƠNG III

ỨNG DỤNG MẠNG ANFIS CHO BÀI TOÁN DỰ BÁO THỜI TIẾT KHU VỰC MIỀN NÚI PHÍA BẮC

3.1. Bài toán dự báo thời tiết

3.1.1. Một số khái quát cơ bản về khí hậu, môi trường tự nhiên

- Khí hậu bao gồm các yếu tố nhiệt độ, độ ẩm, lượng mưa, áp suất khí quyển, các hiện tượng xảy ra trong khí quyển và nhiều yếu tố khí tượng khác trong khoảng thời gian dài ở một vùng, miền xác định. Điều này trái ngược với khái niệm thời tiết về mặt thời gian, do thời tiết chỉ đề cập đến các diễn biến hiện tại hoặc tương lai gần. Khí hậu của một khu vực ảnh hưởng bởi tọa độ địa lí, địa hình, độ cao, độ ổn định của băng tuyết bao phủ cũng như các dòng nước lưu ở các đại dương lân cận. Khí hậu phân ra các kiểu khác nhau dựa trên các thông số chính xác về nhiệt độ và lượng mưa. Sơ đồ phân loại khí hậu được sử dụng phổ biến nhất hiện nay do ông Wladimir Koeppen phát triển. Hệ thống Thornthwaite ^[1] sử dụng từ năm 1948 kết hợp thêm sự thoát-bốc hơi nước với nhiệt độ và các thông tin về lượng mưa được dùng trong việc nghiên cứu về các loài động vật và tiềm năng tác động của sự biến đổi khí hậu. Bergeron và Hệ thống Phân loại không gian khái quát (Spatial Synoptic Classification – SSC) tập trung vào nguồn gốc của các khối không khí xác định cho khí hậu từng khu vực nào đó.

- Môi trường tự nhiên bao gồm các nhân tố thiên nhiên như vật lý, hoá học, sinh học, tồn tại ngoài ý muốn của con người, nhưng cũng ít nhiều chịu tác động của con người. Đó là ánh sáng mặt trời, núi sông, biển cả, không khí, động, thực vật, đất, nước... Môi trường tự nhiên cho ta không khí để thở, đất để xây dựng nhà

cửa, trồng cây, chăn nuôi, cung cấp cho con người các loại tài nguyên khoáng sản cần cho sản xuất, tiêu thụ và là nơi chứa đựng, đồng hoá các chất thải, cung cấp cho ta cảnh đẹp để giải trí, làm cho cuộc sống con người thêm phong phú.

Ngoài ra, người ta còn phân biệt khái niệm môi trường nhân tạo, bao gồm tất cả các nhân tố do con người tạo nên, làm thành những tiện nghi trong cuộc sống, như ô tô, máy bay, nhà ở, công sở, các khu vực đô thị, công viên nhân tạo...

Môi trường bao gồm tất cả các nhân tố bao quanh sinh vật, có tác động trực tiếp hoặc gián tiếp đến sự tồn tại, phát triển, sinh sản... và các hoạt động khác của sinh vật.

Các loại môi trường:

-Môi trường trên cạn: mặt đất và bầu khí quyển.

-Môi trường nước: nước ngọt, lợ, mặn.

-Môi trường đất: các lớp đất nơi có sinh vật sinh sống.

-Môi trường sinh vật: động vật, thực vật, con người...: Nơi sống ký sinh, cộng sinh của sinh vật khác.

3.1.2. Sự cần thiết của việc dự báo thời tiết

Dự báo thời tiết là một ngành ứng dụng của khoa học và công nghệ để tiên đoán trạng thái và vị trí của bầu khí quyển trong tương lai gần. Loài người đã nỗ lực dự báo thời tiết một cách không chính thức từ nhiều thiên niên kỷ trước, và việc dự báo thời tiết một cách chính thức bắt đầu từ thế kỷ mười chín. Công tác dự báo thời tiết được thực hiện bằng cách thu thập số liệu về trạng thái hiện tại của bầu khí quyển và áp dụng những hiểu biết khoa học về các quá trình của khí quyển để tiên đoán sự tiến triển của khí quyển.

Hệ thống dự báo thời tiết là một trong những hệ thống phương trình phức tạp nhất mà máy tính có thể giải quyết. Một số lượng lớn các dữ liệu đến từ các vệ tinh, trạm mặt đất và cảm biến nằm xung quanh hành tinh của chúng tôi gửi thông tin hàng ngày phải được sử dụng để đoán trước tình hình thời tiết trong giờ tiếp theo và

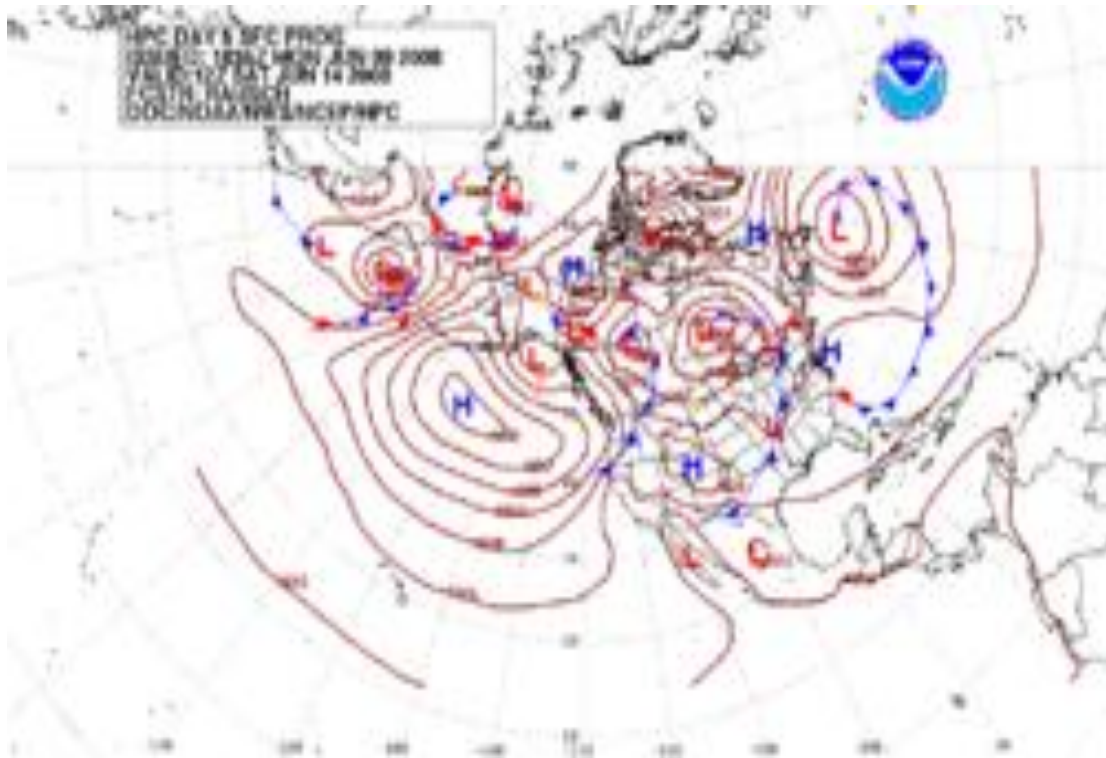
những ngày trên toàn thế giới. Dự báo thời tiết cung cấp cho dự báo cho 24, 48 và 72 giờ tiếp theo cho các khu vực rộng.

Nỗ lực dự báo của con người chủ yếu dựa trên cơ sở về sự thay đổi của áp suất khí quyển, điều kiện hiện tại của thời tiết, và điều kiện bầu trời, các mô hình dự báo được sử dụng để dự báo trong tương lai. Những dữ liệu đầu vào của con người vẫn đòi hỏi phải thực hiện việc lựa chọn mô hình dự báo tốt nhất có thể để làm căn cứ cho việc dự báo, bao gồm kỹ năng nhận định các phần tham gia, teleconnection (liên hệ từ xa), kiến thức về hoạt động của mô hình và kiến thức về khuynh hướng của mô hình. Do bản chất hỗn loạn của khí quyển nên cần phải có những siêu máy tính để giải các phương trình mô tả bầu khí quyển. Những sai số trong việc đo đạc các số liệu đầu vào và sự hiểu biết chưa hoàn thiện về các hoạt động của khí quyển đã làm cho công tác dự báo trở lên ít chính xác trên nhiều địa điểm trong cùng một khoảng thời gian và khi thời gian dự báo tăng lên. Việc sử dụng kết hợp và liên ứng các mô giúp giảm thiểu sai số và chọn ra được kết quả khả quan nhất.

Dự báo thời tiết cung cấp các thông tin quan trọng về thời tiết trong tương lai. Có những kỹ thuật khác nhau tham gia vào dự báo thời tiết, từ quan sát tương đối đơn giản của bầu trời với các mô hình toán học trên máy vi tính rất phức tạp.

Dữ liệu được ghi bởi một trạm thời tiết tại các trung tâm khí tượng nổi bật của Göztepe, İstanbul đã được sử dụng để phân tích và dự báo áp dụng ANFIS và ARIMA, họ được đánh giá và so sánh ...

Cảnh báo thời tiết là dự báo quan trọng bởi vì nó cung cấp thông tin nhằm bảo vệ cuộc sống con người cũng như tài sản và các hoạt động ngoài trời. Dự báo về nhiệt độ và lượng mưa là quan trọng trong nông nghiệp, giao thông,...



3.2. Ứng dụng mạng ANFIS cho bài toán dự báo thời tiết khu vực miền núi phía Bắc

Dự báo thời tiết cung cấp các thông tin quan trọng về thời tiết trong tương lai. Có những kỹ thuật khác nhau tham gia vào dự báo thời tiết, từ quan sát tương đối đơn giản của bầu trời với các mô hình toán học trên máy vi tính rất phức tạp.

Dữ liệu được ghi bởi một trạm thời tiết tại các trung tâm khí tượng nổi bật của Göztepe, Istanbul đã được sử dụng để phân tích và dự báo áp dụng ANFIS và ARIMA, họ được đánh giá và so sánh ..

Một số phương pháp dự báo thời tiết hiện nay:

Phương pháp numerical weather prediction: Dự báo cho nhiều quốc gia trong khoảng thời gian dài từ vài giờ đến vài tháng.

Phương pháp persistence: Dựa trên phương pháp NWP, dự báo trong tương lai gần từ vài phút đến vài giờ.

Phương pháp Statistical and neural networks: Dự báo trong thời gian ngắn vài phút đến một giờ, kết quả kém chính xác khi thời gian dự báo xong.

3.2.1. Thu thập dữ liệu

Để dự báo được độ ẩm cho các tỉnh tại khu vực miền núi phía Bắc, cần thu thập đầy đủ dữ liệu về nhiệt độ cao nhất, nhiệt độ thấp nhất, áp suất, lượng mưa, độ ẩm trong thời gian gần nhất. Trên thực tế, dữ liệu sử dụng cho luận văn này được thu thập ở một vài trạm thủy văn của một vài tỉnh khu vực miền núi phía Bắc và đây là số liệu trong gần 2 năm, gồm 639 bản ghi dữ liệu, mỗi bản ghi đại diện cho một ngày trong năm.

Bảng dưới đây liệt kê 30 bản dữ liệu từ dòng thứ 100 đến dòng thứ 129.

Nhiệt độ cao nhất (độ C)	Nhiệt độ thấp nhất (độ C)	Áp suất (mm atm)	Lượng mưa (mm)	Độ ẩm (%)
30,4	18,4	1009,9	0	93
29,8	16,8	1009,3	0	88
28,5	17,5	1009,2	0,2	93
30	18	1011,5	0	86
32,4	20,4	1011,3	0	87
29,7	19,7	1010,5	0	85
30,2	17,2	1008,6	0	86
31,4	19,4	1008,5	0	89
30,6	19,6	1009,1	0	87
30,7	17,7	1007,7	0	88
32,6	20,6	1008,3	0	82
33,3	19,3	1007,1	0	77
31,2	17,2	1010,9	0,2	90
29	16	1007,6	1,6	92
33	21	1005,7	0,1	91
31,2	19,2	1007,2	0,1	89
32,3	20,3	1007,9	0,1	87
27,3	17,3	1014,2	11,6	92
26,9	14,9	1015,3	0	88
25,7	13,7	1012,7	10,5	91
26	15	1011,3	12,8	84
27,7	16,7	1011	0	82
30,2	17,2	1013,1	0	77
31,4	17,4	1011,5	0	84
29,4	17,4	1009	0,4	86
25,7	12,7	1016	26,1	86
25,4	12,4	1015,9	0	82
26,3	15,3	1011,7	0	86
20,6	16,8	1006,8	0	80
21,1	20,6	1001,4	0	79

Bảng 3.1: Số liệu thu thập tại trạm khí tượng thủy văn trong 2 năm gần đây

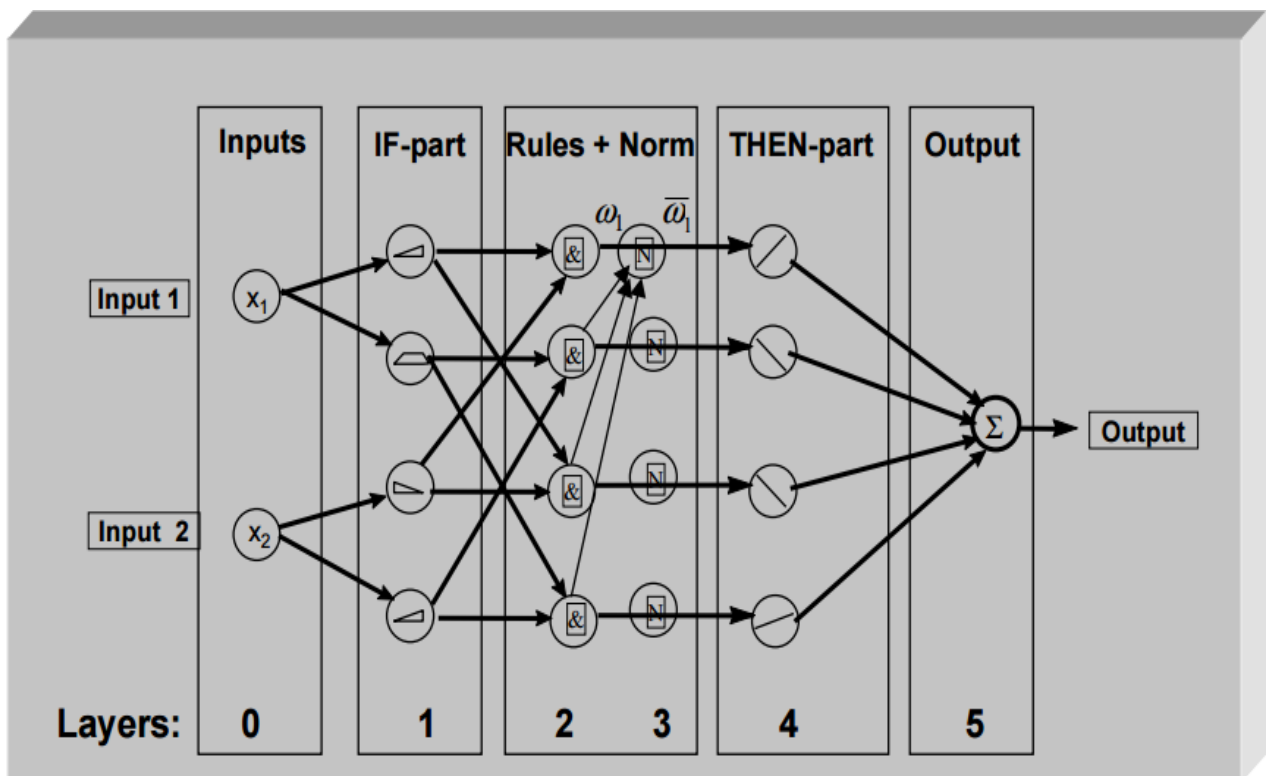
3.2.2. Xây dựng hệ luật

Mô hình mờ trong mạng Anfis là mô hình mờ Sugeno. Tập các giá trị đầu vào được phân bổ hình chuông, mỗi biến ngôn ngữ *Nhiệt độ cao nhất*, *Nhiệt độ thấp nhất*, *Lượng mưa*, *Áp suất* có 3 tập giá trị mờ là *Cao*, *Trung bình*, *Thấp*.

Do vậy, tổng cộng có $3 \times 3 \times 3 \times 3 = 81$ luật được sinh ra, các luật có dạng như sau:

R1: If *Nhiệt độ cao nhất* is *Cao* and *Nhiệt độ thấp nhất* is *Thấp* and *Lượng mưa* is *Cao* and *Áp suất* is *Thấp* then *Độ ẩm* = $f(\text{Nhiệt độ cao nhất}, \text{Nhiệt độ thấp nhất}, \text{Lượng mưa}, \text{Áp suất})$

Hình dưới đây thể hiện cách hệ luật được áp dụng trong mạng Anfis



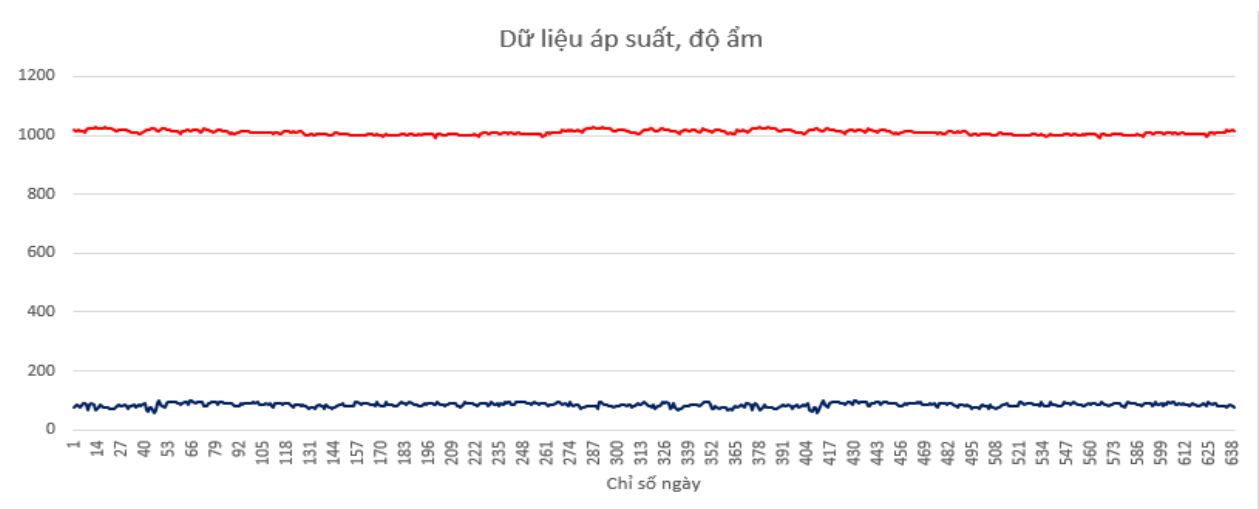
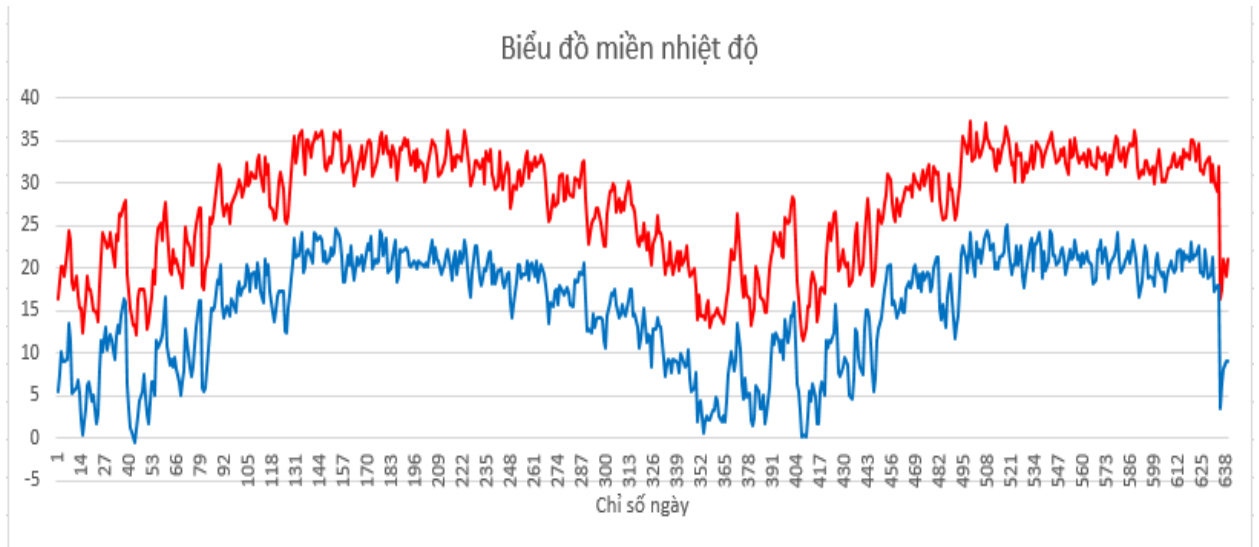
3.2.2. Huấn luyện mạng ANFIS cho dự báo thời tiết khu vực miền núi phía Bắc

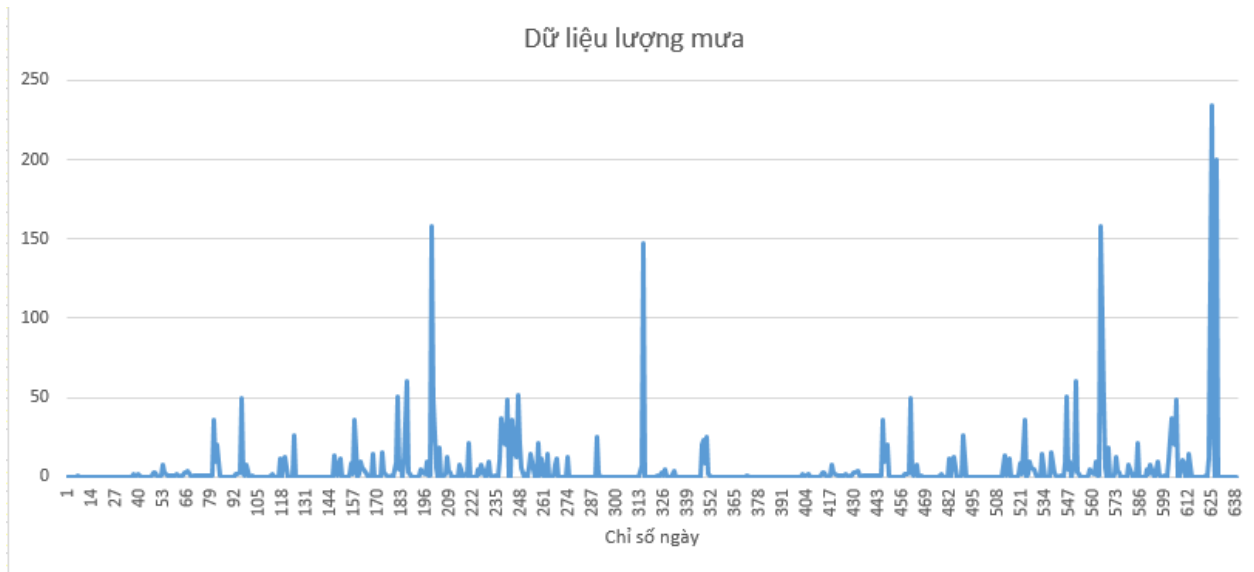
Trong quá trình huấn luyện mạng ANFIS các thông số cần quan tâm đó là:

- Số hàm thuộc cho mỗi một đầu vào (số giá trị ngôn ngữ)
- Đầu ra của mạng ANFIS
- Số bước huấn luyện (epoch number)
- Mục tiêu của sai số huấn luyện
- Độ lớn của bước nhảy

- Tốc độ của các bước nhảy là tăng hay giảm

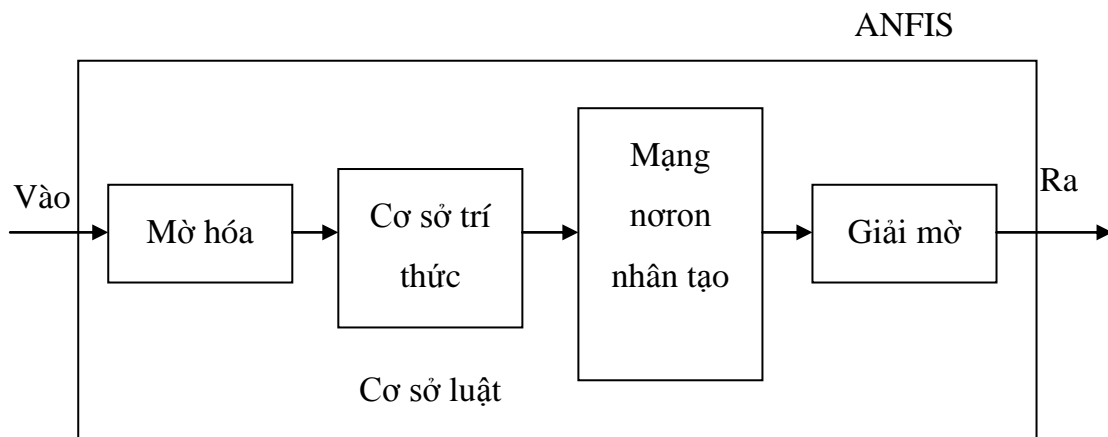
Một số thông số khác có thể sử dụng các dữ liệu kiểm tra sai số của quá trình huấn luyện. Các dữ liệu kiểm tra được sử dụng để kiểm soát lỗi (sự khác biệt giữa các dữ liệu thu thập và dự báo)





Hình 3.1. Dữ liệu vào ở bảng 3.1 biểu diễn dưới dạng đồ thị

Sơ đồ khối cho ANFIS có thể thấy trên hình 3.2



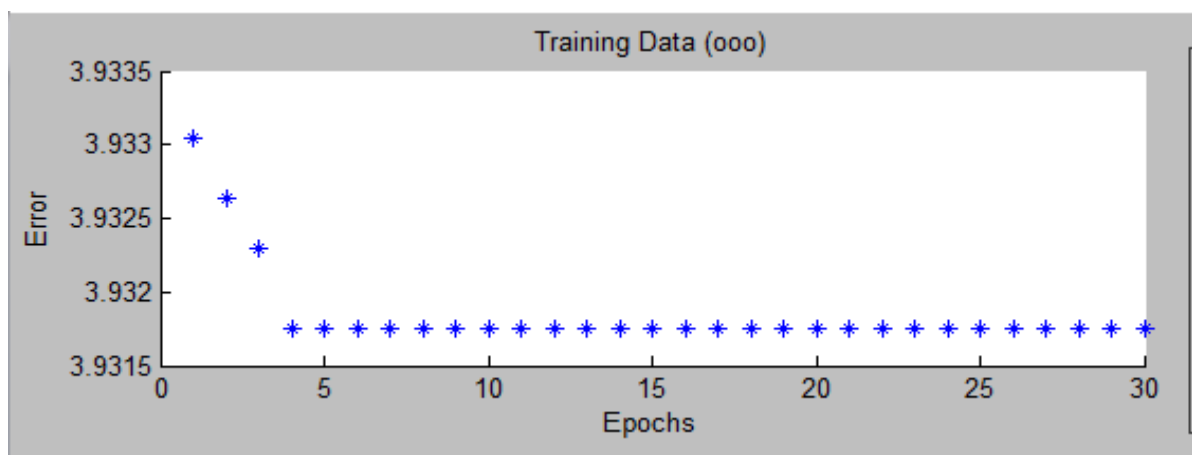
Hình 3.2: Sơ đồ khối của mạng ANFIS

Chương trình mô phỏng được viết trên Matlab, có sử dụng một số thư viện có sẵn trên Matlab để rút ngắn thời gian nghiên cứu. Toàn bộ dữ liệu thu thập sẽ được chuẩn hóa và làm dữ liệu đầu vào cho chương trình.

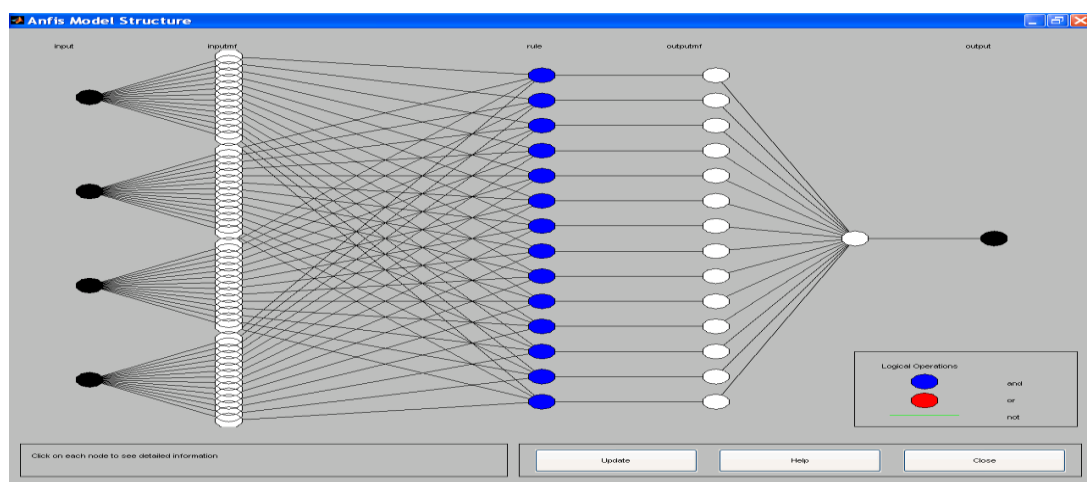
Tiến hành cấu hình cho mạng ANFIS, tổng số giá trị đầu vào là 4, mỗi giá trị đầu vào có 3 hàm thuộc, sử dụng thuật toán học lai là **gbell**, kiểu hàm thuộc là **linear**.


```
%Số hàm liên thuộc
numMFs = 4;
%Kiểu thuật toán học lai
mfType = 'gbellmf';
%Kiểu đầu ra hàm thuộc
default_output_type = 'linear';
```

Quá trình huấn luyện được chạy qua 30 bước lặp đã đạt đến độ ổn định về sai số, sai số trung bình là: **3.9318**.

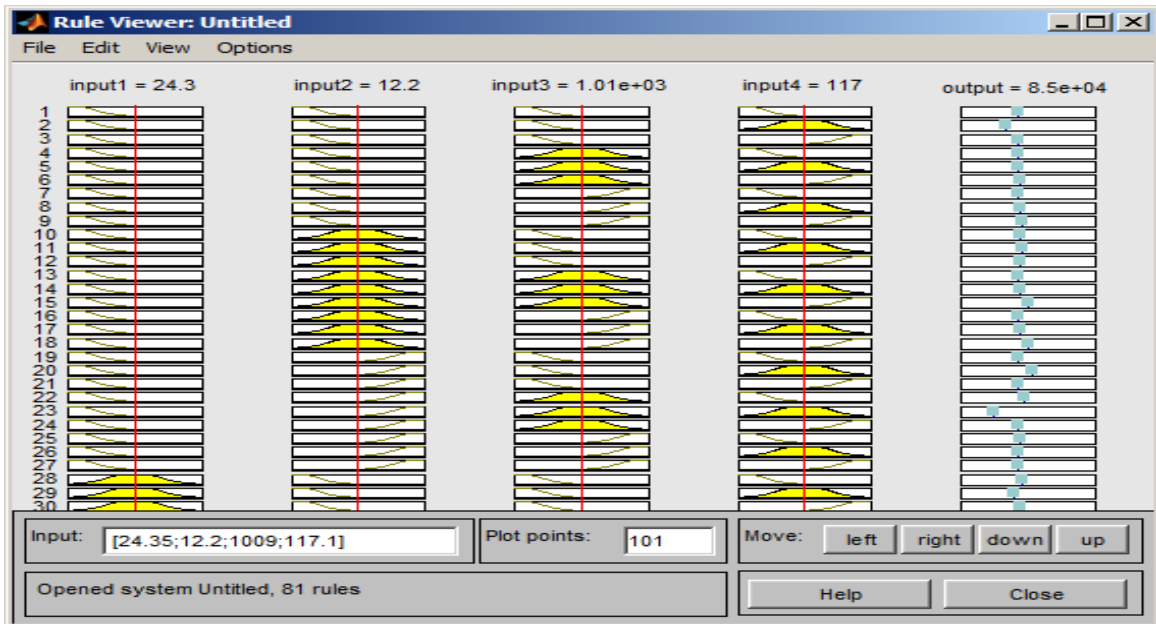


Hình 3.4: Kiểm tra sai số của quá trình huấn luyện mạng.



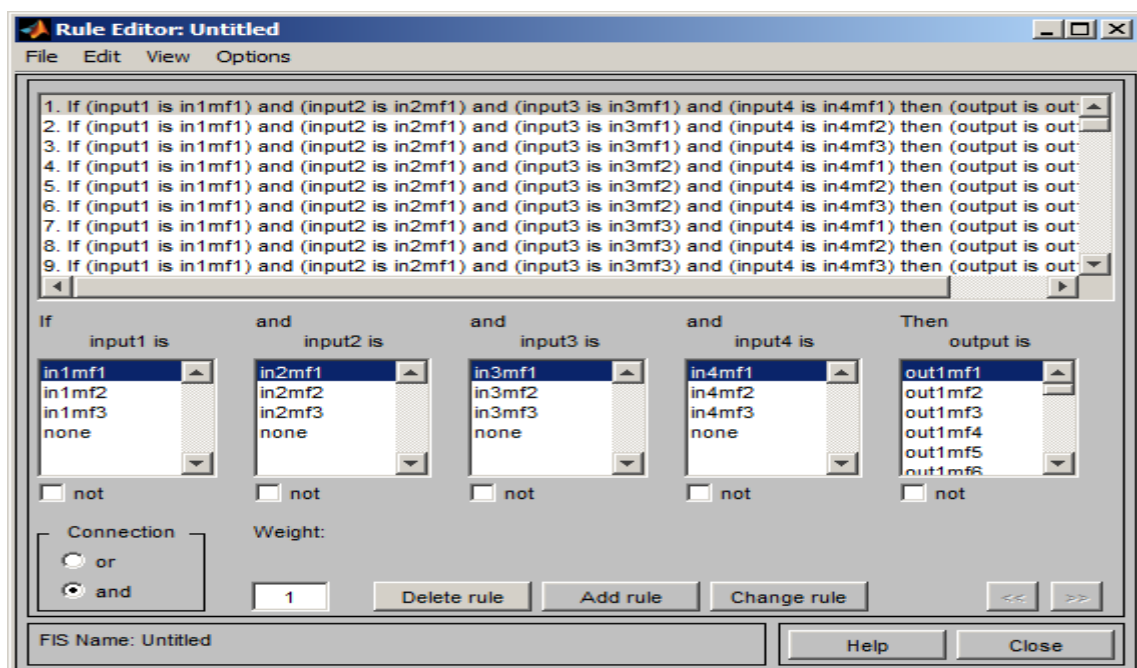
Hình 3.5: Mạng ANFIS được xây dựng cho huấn luyện dữ liệu để dự báo thời tiết khu vực miền núi phía Bắc

Các luật sinh ra từ tập dữ liệu đầu vào. Có tất cả 81 luật.



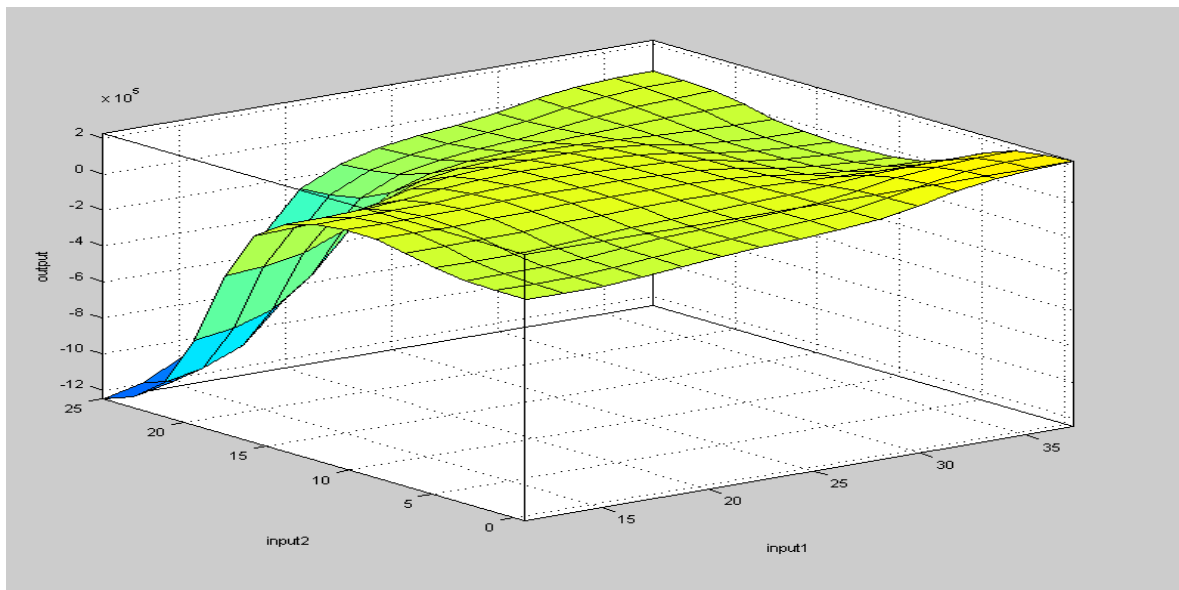
Hình 3.6. Hệ luật mờ được hình thành trong quá trình huấn luyện mạng ANFIS

Trên hình 3.6 là hệ luật mờ được hình thành cho huấn luyện mạng ANFIS hình 3.5. Hệ luật với 4 đầu vào và 1 đầu ra được mờ hóa từ bảng dữ liệu 3.1 hay trên đồ thị hình 3.1. Các giá trị ngữ nghĩa được mờ hóa là NB, ZO, PB. Hệ luật với 81 luật được sử dụng trong quá trình huấn luyện mạng. Các giá trị ngữ nghĩa sử dụng hàm thuộc Gbellmenbeship, Hình 3.6, 3.6a. Ở đây gọi là hệ luật mờ hay như trên Hình 3.2 là hệ cơ sở luật.



Hình 3.6a Hệ luật mờ được sinh ra cho huấn luyện mạng

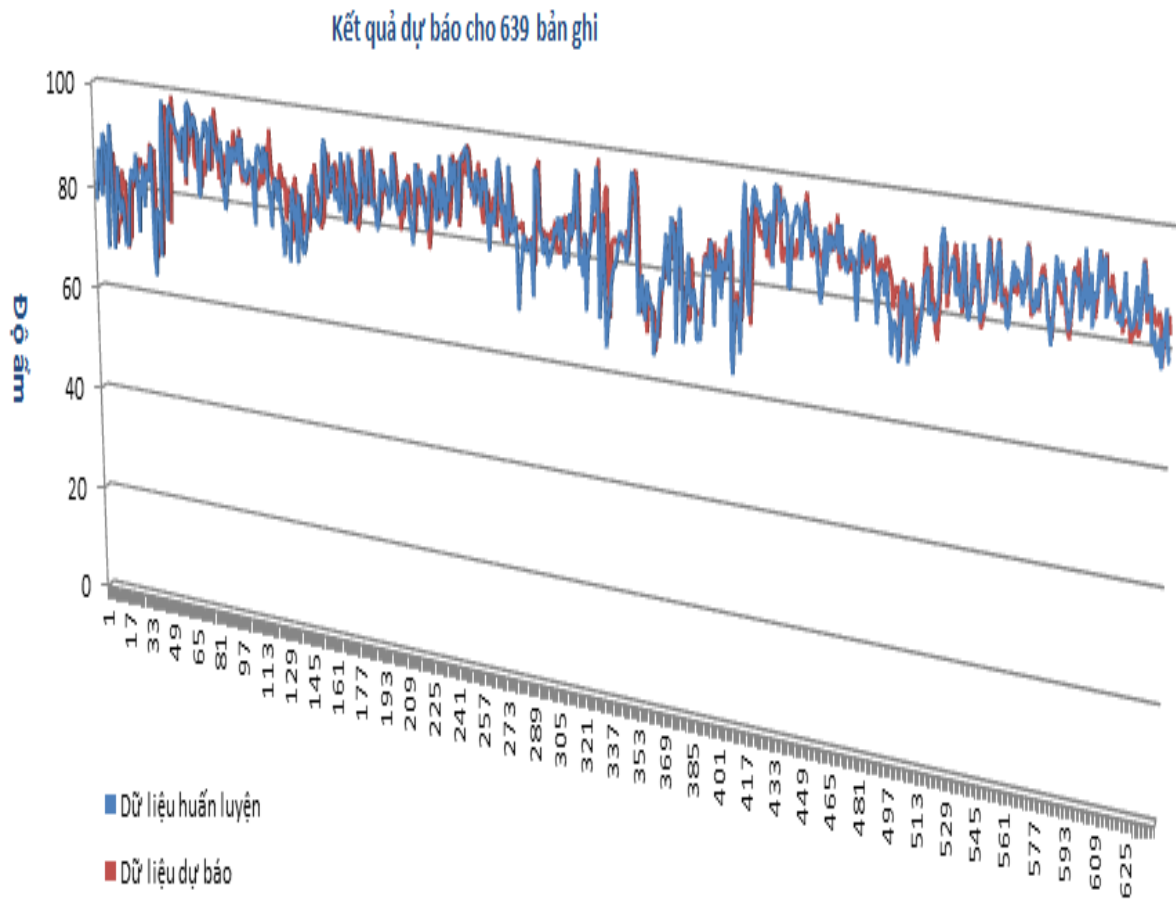
Trong quá trình huấn luyện mạng ANFIS chỉ ra sai số của quá trình huấn luyện, điều đó cho thấy việc xấp xỉ hàm dự báo đảm bảo yêu cầu và mô hình ANFIS thể hiện được các hành vi về dữ liệu huấn luyện cũng như dữ liệu kiểm tra. Trên hình 3.4 biểu diễn kiểm tra sai số với 30 bước lặp cho huấn luyện mạng. Hình 3.4 biểu diễn sai số trong quá trình huấn luyện



Hình 3.6b Mặt suy diễn của hệ ANFIS

3.3. Kết quả dự báo

Với việc sử dụng hàm thuộc dạng gbell cho hệ thống mờ của mô hình ANFIS dự báo độ ẩm của vùng núi phía Bắc. Quá trình dự báo được thể hiện trên hình 3.7 giá trị thể hiện màu xanh là các giá trị dữ liệu thu thập, các giá trị màu đỏ là các giá trị dự báo. Các kết quả dự báo có sai lệch tương đối nhỏ so với dữ liệu khảo sát, điều đó cho thấy mạng ANFIS có khả năng dự báo tương đối chính xác cho các lĩnh vực dự báo thời tiết.



Hình 3.7: Kết quả dự báo độ ẩm

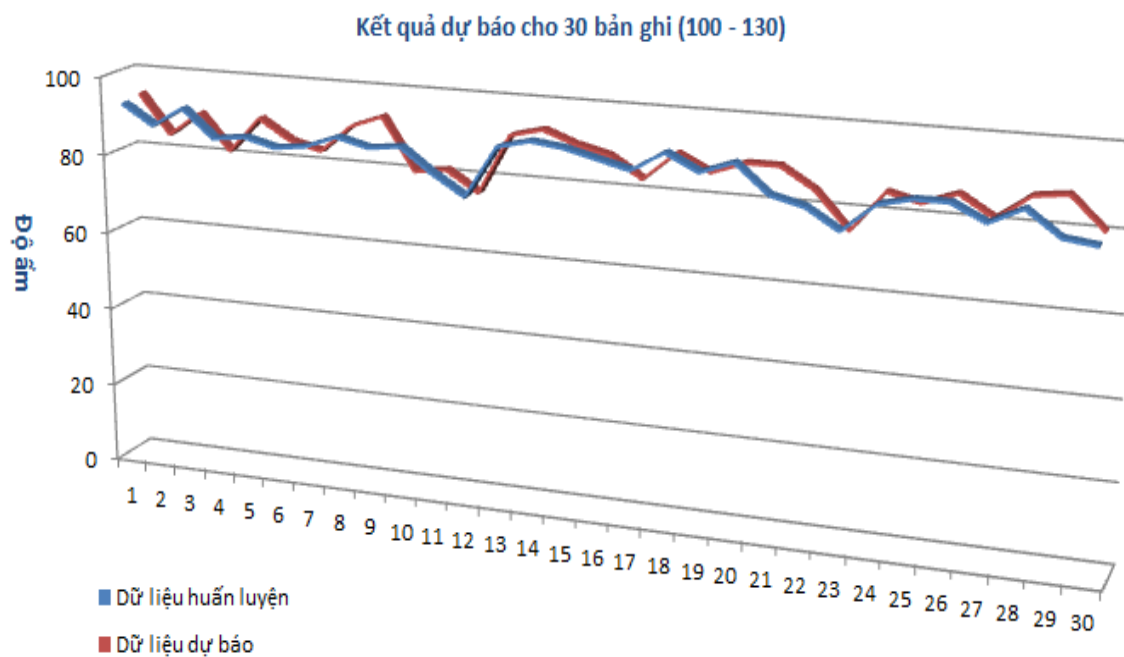
Như vậy với 4 đầu vào khảo sát (nếu có cơ hội có thể tăng số đầu vào lên thì việc đánh giá dự báo sẽ thực tế hơn).

Để dễ quan sát, tôi tách riêng 30 bản ghi, từ bản ghi thứ 100 đến 130 được thể hiện trong bảng 3.2 và dữ liệu mô phỏng dạng biểu đồ được thể hiện ở hình 3.8

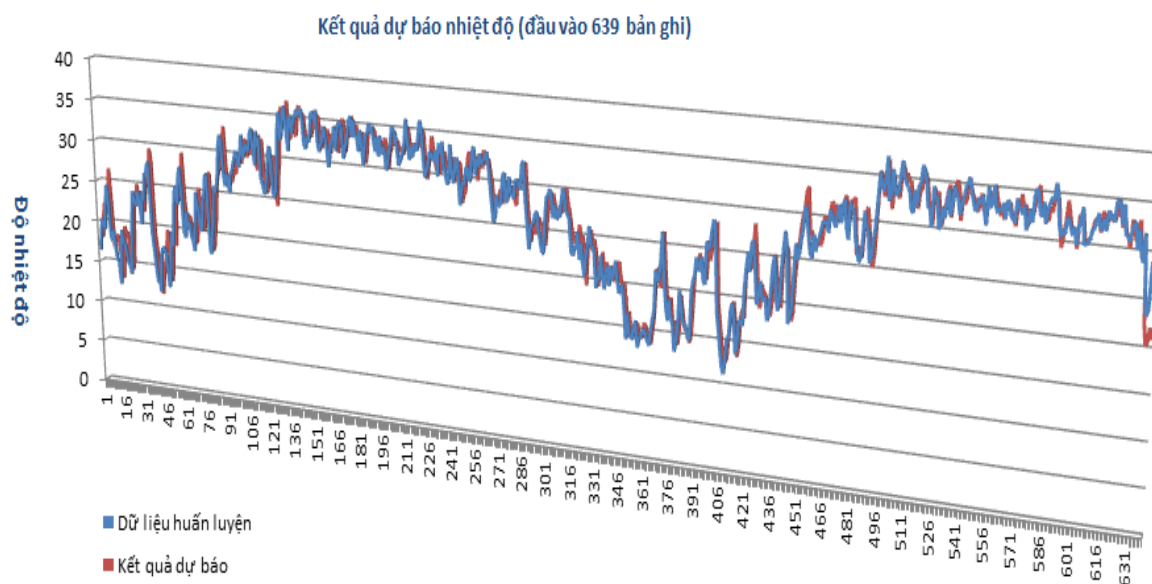
Bảng 3.2. Kết quả số liệu khảo sát và giá trị dự báo

Độ ẩm khảo sát	Độ ẩm dự báo
93.0	94.0
88.0	84.0
93.0	90.0
86.0	81.0
87.0	90.0

85.0	85.0
86.0	83.0
89.0	90.0
87.0	93.0
88.0	80.0
82.0	81.0
77.0	76.0
90.0	91.0
92.0	93.0
91.0	90.0
89.0	88.0
87.0	83.0
92.0	90.0
88.0	86.0
91.0	89.0
84.0	89.0
82.0	84.0
77.0	75.0
84.0	85.0
86.0	83.0
86.0	86.0
82.0	81.0
86.0	87.0
80.0	88.0
79.0	80.0



Hình 3.8: Kết quả dự báo độ ẩm



Hình 3.9: Kết quả dự báo nhiệt độ

KẾT LUẬN

Trong luận văn này đã đề cập đến một vấn đề nghiên cứu rộng rãi, đó là hệ suy diễn mờ trên cơ sở mạng thích nghi (ANFIS), đây là một hệ đã và đang được ứng dụng rất nhiều trong thực tế, đặc biệt là trong các lĩnh vực như dự đoán, phân lớp,.. Dự đoán trước tình hình thời tiết các tỉnh khu vực miền núi phía Bắc, tuy nhiên để tìm ra quyết định chính xác luôn là nhiệm vụ khó khăn để có thể tiếp cận đến kỹ thuật mới luận văn đã thực hiện được:

- + Nghiên cứu lý thuyết mạng Noron bao gồm các loại mạng, kiến trúc của chúng, các thuật toán luyện mạng, phân tích được các khả năng hoạt động của chúng. Nghiên cứu lý thuyết logic mờ, các hệ mờ thường gặp nhằm làm sáng tỏ quá trình suy luận của một hệ mờ trong thực tế để áp dụng xây dựng một hệ mờ cho dự báo thời tiết.

- + Nêu rõ cấu trúc mạng ANFIS, các thuật toán huấn luyện và những ứng dụng rộng lớn trong thực tế đặc biệt là trong bài toán dự báo thời tiết.

- + Khái quát các kiến thức cơ bản về dự báo, về bài toán dự báo thời tiết khu vực miền núi phía Bắc.

- + Xây dựng mô hình mạng ANFIS cho bài toán dự báo thời tiết khu vực miền núi phía Bắc.

- + Thu thập dữ liệu, tiền xử lý dữ liệu phục vụ việc huấn luyện và kiểm tra.

- + Cài đặt mô hình mạng ANFIS trên môi trường matlab.

Chương trình thực nghiệm chạy cho kết quả dự đoán khá chính xác. Để có kết quả dự báo chính xác hơn hướng nghiên cứu tiếp theo là:

- + Thời gian tới sẽ tiếp tục nghiên cứu xây dựng hệ ANFIS có nhiều đầu ra.

- + Phân tích dữ liệu loại bỏ những mẫu dữ liệu cực đoan để có được tập dữ liệu tốt phục vụ huấn luyện.

- + Trong tương lai để có thể hoàn thiện tốt hơn quá trình dự báo đề tài sẽ cập nhật dữ liệu thời tiết của nhiều trạm khí tượng thủy văn hơn và luận văn sẽ có thể làm tài liệu giúp ích tốt hơn cho các cán bộ làm công tác dự báo khí tượng thủy văn.

Do thời gian và năng lực còn hạn chế, kính mong sự chỉ bảo và đóng góp ý kiến của các Thầy, Cô và các Bạn để luận văn đạt kết quả tốt hơn.

TÀI LIỆU THAM KHẢO

*** Tiếng Việt**

[1] Lê Bá Dũng, *Bài giảng mạng Noron và ứng dụng* lớp Cao học Thái Nguyên, 2013.

[2] Nguyễn Như Hiền, Lại Khắc Lãi, *Hệ mờ và Noron trong kỹ thuật điều khiển*, NXB khoa học tự nhiên và công nghệ, 2007.

[3] Bùi Công Cường, Nguyễn Doãn Phước, *Hệ thống mạng Noron mờ và ứng dụng*, NXB Khoa học và Công nghệ, 2006.

[4] Phan Xuân Minh, Nguyễn Doãn Phước, *Lý thuyết điều khiển mờ*, NXB Khoa học và kỹ thuật, 2002.

*** Tiếng Anh**

[5] P. Bonissone, K. Goebel, “When will it break? A Hybrid Soft Computing Model to Predict Time-to-break Margins in paper machine’s”, in Proc. SPIE 7th Annual Meeting, Int. Symp. Optical Science and Technology, 4787, 53-64, 2002

[6] J.M. Liu, R. Chen, L.M. Liu, J.L. Harris, “ A semiparametric time series approach in modeling hourly electricity loads”, Journal of Forecasting, vol. 25, pp. 537–559, 2006

[7] L.J. Soares, M.C. Medeiros , “Modeling and forecasting short-term electricity load: a comparison of methods with an application to Brazilian data”, International Journal of Forecasting, vol. 24, pp. 630–644, 2008

[8] O.E. Dragomir, R. Gouriveau, F. Dragomir, E. Minca, “Review of the prognosis problem in academics and industrial area of interest”, in Proc. European Control Conference 2009- ECC’09, 23-26 August Budapest, Hungary, 2009

[9] L. Rutkowski, *Flexible neuro-fuzzy systems structure, learning and performance*, Kluwer Aca-demic Pub., 2004.

[10] J.S.R.Jang, C.I. Sun and E.Mizutani, *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentice-Hall, NJ, 1997.

[11] C.T.Lin and C.S.G. Lee, *Neural Fuzzy Systems*, Prentice Hall, London, 1996.

[12] Weather Forecasting Using ANFIS and ARIMA MODELS. A Case Study for Istanbul

[13] Prediction of River Water Quality by Adaptive Neuro Fuzzy Inference System

*** Trang web**

[10] tailieu.vn

[11] <http://www.ebook.edu.vn/>

[12] <http://www.amazon.com>