

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

DƯƠNG THU TRANG

**ỨNG DỤNG MẠNG NƠON NHÂN TẠO DỰ BÁO SỐ HỌC SINH
TUYỂN VÀO TRUNG TÂM GDNN – GDTX QUẬN ĐÔNG ĐA**

LUẬN VĂN THẠC SĨ NGÀNH CÔNG NGHỆ THÔNG TIN

HÀ NỘI - 2017

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

DƯƠNG THU TRANG

**ỨNG DỤNG MẠNG NƠON NHÂN TẠO DỰ BÁO SỐ HỌC SINH
TUYỂN VÀO TRUNG TÂM GDNN – GDTX QUẬN ĐÔNG ĐA**

Ngành : Công nghệ thông tin

Chuyên ngành: Quản lý hệ thống thông tin

Mã số :

LUẬN VĂN THẠC SĨ NGÀNH CÔNG NGHỆ THÔNG TIN

Người hướng dẫn khoa học: PGS.TS. NGUYỄN QUANG HOAN

HÀ NỘI - 2017

LỜI CAM ĐOAN

Em xin cam đoan:

Những nội dung trong luận văn là do em thực hiện dưới sự hướng dẫn của thầy giáo PGS.TS. Nguyễn Quang Hoan.

Mọi tham khảo dùng trong luận văn đều được trích dẫn rõ ràng tác giả, tên công trình, thời gian, địa điểm công bố.

Mọi sao chép không hợp lệ, vi phạm quy chế đào tạo hay gian lận em xin chịu hoàn toàn trách nhiệm.

Hà Nội, ngày tháng năm 2017

Học viên

Dương Thu Trang

LỜI CẢM ƠN

Em xin bày tỏ lòng biết ơn sâu sắc đến PGS.TS. Nguyễn Quang Hoan, thầy đã hướng dẫn, chỉ bảo tận tình để em có thể hoàn thành luận văn này. Em cũng xin chân thành cảm ơn các thầy, cô giáo của khoa Công nghệ thông tin – Trường Đại học Công nghệ - Đại học Quốc gia Hà Nội đã truyền thụ kiến thức, hỗ trợ em trong suốt quá trình học tập vừa qua.

Em cũng xin chân thành cảm ơn bạn bè đồng nghiệp, gia đình và những người thân đã cùng chia sẻ, giúp đỡ, động viên, tạo mọi điều kiện thuận lợi để em hoàn thành nhiệm vụ học tập và cuốn luận văn này.

Hà Nội, ngày tháng năm 2017

Học viên

Dương Thu Trang

3
MỤC LỤC

LỜI CAM ĐOAN	1
LỜI CẢM ƠN	2
MỤC LỤC	3
DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT	6
DANH MỤC CÁC BẢNG BIỂU	7
DANH MỤC CÁC HÌNH VẼ	8
MỞ ĐẦU	9
1. Lý do chọn đề tài	9
2. Mục tiêu và nhiệm vụ của luận văn	9
3. Đối tượng và phạm vi nghiên cứu	9
4. Phương pháp nghiên cứu	9
5. Ý nghĩa khoa học và thực tiễn của luận văn	9
6. Bố cục luận văn	9
CHƯƠNG 1	11
TỔNG QUAN VỀ BÀI TOÁN DỰ BÁO	11
1.1. Lịch sử của quá trình dự báo	11
1.1.1. Khái niệm về dự báo	11
1.1.2. Mục đích của dự báo	11
1.1.3. Những thách thức trong phân tích dự báo	12
1.1.4. Các phương pháp dự báo	12
1.1.5. Quy trình thực hiện dự báo	13
1.2. Phương pháp dự báo sử dụng mạng nơron theo đề xuất của luận văn	16
1.3. Đánh giá mô hình dự báo	16
1.4. Kết luận chương 1	16
CHƯƠNG 2	17
TỔNG QUAN VỀ MẠNG NƠRON NHÂN TẠO	17
2.1. Mạng nơron nhân tạo là gì?	17
2.2. Sự tương đương nơron nhân tạo với nơron sinh học	18
2.3. Lịch sử phát triển của mạng nơron nhân tạo	18
2.4. Nơron sinh vật	18
2.5. Nơron nhân tạo	19
2.5.1. Cấu tạo nơron nhân tạo	19
2.5.2. Phân loại mạng nơron nhân tạo	20
2.6. Luật học	22

2.6.1. Học tham số (Parameter Learning).....	22
2.6.2. Học cấu trúc.....	23
2.7. Mạng truyền thẳng nhiều lớp và thuật toán lan truyền ngược.....	23
2.7.1. Kiến trúc mạng.....	23
2.7.2. Xác định cấu trúc mạng tối ưu.....	24
2.7.3. Cấu trúc của luật học lan truyền ngược.....	25
2.7.4. Luật học lan truyền ngược.....	26
2.7.5. Một số vấn đề của mạng nơron nhiều lớp.....	31
2.7.6. Ưu nhược điểm của mạng truyền thẳng.....	31
2.8. Kết luận chương 2.....	32
CHƯƠNG 3.....	33
ỨNG DỤNG MẠNG NƠON TRONG DỰ BÁO SỐ HỌC SINH.....	33
TUYỂN VÀO TRUNG TÂM GDNN-GDTX ĐỒNG ĐA.....	33
3.1. Giới thiệu về Trung tâm GDNN-GDTX Đồng Đa.....	34
3.2. Dữ liệu tuyển sinh của Trung tâm.....	36
3.3. Các phương án chọn cấu trúc dữ liệu.....	37
3.4. Phát biểu bài toán.....	38
3.5. Thiết kế mạng nơron.....	39
3.5.1. Số lớp nơron.....	39
3.5.2 Cấu trúc mạng.....	39
3.5.3. Hàm tương tác đầu ra.....	40
3.5.4. Giá trị trọng khởi đầu.....	40
3.6. Công cụ mô phỏng bài toán dự báo tuyển sinh.....	41
3.7. Chạy và thử nghiệm.....	41
3.8. So sánh các phương pháp dự báo.....	44
3.9 Kết luận chương 3.....	48
KẾT LUẬN.....	49
TÀI LIỆU THAM KHẢO.....	50
PHỤ LỤC A.....	52
GIỚI THIỆU CÔNG CỤ VÀ KẾT QUẢ DỰ BÁO.....	52
1. Giao diện chính của phần mềm.....	52
2. Giao diện huấn luyện.....	53
3. Giao diện Kiểm tra mạng.....	54
4. Chức năng Dự báo.....	54
5. Chức năng Giới thiệu vào Thoát.....	55

PHỤ LỤC B.....	56
MỘT SỐ MÃ NGUỒN CỦA CÔNG CỤ.....	56
1. Huấn luyện mạng.....	56
2. Chuẩn hóa dữ liệu.....	59
3. Kiểm tra mạng	60
4. Dự báo	61
PHỤ LỤC C.....	63
BÀI BÁO KHOA HỌC.....	63

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

STT	TÊN VIẾT TẮT	TÊN ĐẦY ĐỦ	DIỄN GIẢI
1	ANN	Artificial Neural Network	Mạng nơron nhân tạo
2	AR	Autoregressive	Tự hồi quy
3	BP	Back Propagation	Lan truyền ngược
4	HIDDENNODE	Hidden Node	Số nơron lớp ẩn
5	INPUTNODE	Input Node	Số nơron lớp vào
6	LEARNING RATE	Learning Rate	Tốc độ học
7	MA	Moving Average	Đường trung bình di động
8	MAE	Mean Absolute Error	Sai số tuyệt đối
9	MSE	Mean Square Error	Sai số trung bình
10	MFNN	Multi Feedforward Neural Network	Mạng truyền thẳng nhiều lớp
11	MLP	Multilayer Neural Network	Mạng nơron nhiều lớp
12	NƠRON	Neural	Tế bào thần kinh
13	OUTPUTNODE	Output Node	Số nơron lớp ra
14	GA	Genetic Algorithm	Giải thuật di truyền
15	EP	Evolutionary Programming	Lập trình tiến hóa

DANH MỤC CÁC BẢNG BIỂU

Bảng 2.1. Một số hàm phi tuyến thường được sử dụng trong các mô hình neuron	20
Bảng 3.1. Cơ sở vật chất của Trung tâm	34
Bảng 3.2. Dữ liệu tuyển sinh của Trung tâm.....	36
Bảng 3.3. Dữ liệu tuyển sinh từ năm 1988 đến 2017 của Trung tâm.....	38
Bảng 3.4. Bảng dữ liệu học	38
Bảng 3.5. Bảng trọng số	40
Bảng 3.6. Kết quả thử nghiệm dự báo tuyển sinh	43
Bảng 3.7. Kết quả huấn luyện dự báo tuyển sinh	44
Bảng 3.8. Kết quả dự báo bằng phương pháp trung bình dài hạn.....	45
Bảng 3.9. Kết quả dự báo bằng phương pháp trung bình động.....	45
Bảng 3.10. Kết quả dự báo bằng phương pháp hồi quy tương quan.....	46
Bảng 3.11. Kết quả dự báo bằng phương pháp san bằng mũ	47

DANH MỤC CÁC HÌNH VẼ

Hình 1.1. Quy trình thực hiện dự báo.....	14
Hình 1.2. Quy trình lựa chọn mô hình dự báo.....	15
Hình 2.1. Cấu trúc nơron sinh vật.....	19
Hình 2.2. Mô hình một nơron nhân tạo	19
Hình 2.3. Phân loại mạng nơron.....	21
Hình 2.4. Các dạng học	22
Hình 2.5. Mạng truyền thẳng nhiều lớp.....	24
Hình 2.6. Cấu trúc mạng lan truyền ngược.	26
Hình 2.7. Tối thiểu cục bộ (Local Minimum)	30
Hình 2.8. Tốc độ học của hằng số học η	31
Hình 3.1. Kết quả huấn luyện mạng	42
Hình 3.2. Kết quả kiểm tra mạng.....	43
Hình 3.3. Kết quả dự báo cho 3 năm 2018, 2019, 2020	44
Hình 3.4. Sơ đồ kết quả dự báo trung bình động.....	48
Hình 3.5. Kết quả dự báo san bằng mũ	50

1. Lý do chọn đề tài

Hiện nay, công tác dự báo được ứng dụng rộng rãi ở hầu hết các lĩnh vực như: Dự báo giá xăng dầu, dự báo chứng khoán, dự báo thời tiết... Công tác dự báo nhằm mục đích dự báo trước sự thay đổi của đối tượng được dự báo dựa trên cơ sở nghiên cứu các quy luật của đối tượng dự báo.

Những năm gần đây ngành giáo dục cũng bắt đầu chú trọng đến công tác dự báo nhằm đưa ra được những chính sách đúng đắn nhất. Có rất nhiều tác giả đã đi vào nghiên cứu dự báo các lĩnh vực khác nhau của giáo dục như: Dự báo nhu cầu giáo viên, dự báo tỉ lệ học sinh đỗ tốt nghiệp, dự báo tỉ lệ học sinh đỗ Đại học...

Chính vì thấy được lợi ích của việc dự báo trong giáo dục nên em chọn đề tài: “Ứng dụng mạng nơron nhân tạo dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa” để giúp đưa ra được những chính sách đúng đắn nhất trong thời gian tới cho Trung tâm của mình.

2. Mục tiêu và nhiệm vụ của luận văn

Luận văn tập trung khảo sát các mô hình mạng nơron, áp dụng phương pháp học máy mạng nơron và ứng dụng vào bài toán dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX Đống Đa. Luận văn phát triển công cụ thử nghiệm riêng và tiến hành thử nghiệm trên bộ số liệu thu thập được trong 30 năm.

3. Đối tượng và phạm vi nghiên cứu

Luận văn nghiên cứu lý thuyết mạng nơron nhân tạo, mạng nơron lan truyền thẳng và thuật toán lan truyền ngược, áp dụng vào bài toán dự báo nói chung và dự báo số học sinh tuyển vào Trung tâm GDNN – GDTX Đống Đa nói riêng.

4. Phương pháp nghiên cứu

Nghiên cứu lý thuyết để nắm rõ các mô hình mạng nơron, các phương pháp dự báo, lý thuyết về mạng nơron nhân tạo, mạng nơron lan truyền thẳng và thuật toán lan truyền ngược. Trên cơ sở đó lựa chọn mô hình phù hợp và thử nghiệm để hiệu chỉnh tối ưu mô hình đã chọn.

5. Ý nghĩa khoa học và thực tiễn của luận văn

Luận văn chỉ ra khả năng ứng dụng mạng nơron nhân tạo trong công tác dự báo nói chung và dự báo trong lĩnh vực giáo dục nói riêng là hoàn toàn khả thi và cho kết quả đáng khích lệ.

6. Bố cục luận văn

Nội dung của luận văn được tổ chức thành ba chương có nội dung như sau:

- Chương 1: Nghiên cứu về bài toán dự báo, các phương pháp dự báo và phương pháp dự báo mà luận văn áp dụng cho bài toán dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa.
- Chương 2: Nghiên cứu về mạng nơron nhân tạo, tập trung vào mạng nơron truyền thẳng nhiều lớp, chương này cung cấp chi tiết các vấn đề về quá trình

huấn luyện và thuật toán học của mạng, với trọng tâm là thuật toán lan truyền ngược sai số.

- Chương 3: Ứng dụng mạng nơron nhân tạo vào giải quyết bài toán dự báo số học sinh tuyển vào Trung tâm GDNN – GDTX quận Đống Đa. Chương này giới thiệu về số liệu thu thập được, các phương pháp đánh giá kết quả dự báo và tập trung vào thử nghiệm để dự báo số học sinh tuyển sinh vào Trung tâm GDNN – GDTX quận Đống Đa. Từ đó đánh giá khả năng ứng dụng phương pháp máy học vào dự báo chuỗi thời gian.
- Kết luận: Tổng kết các kết quả đã đạt được của luận văn và hướng nghiên cứu tiếp theo. Phần phụ lục giới thiệu phần mềm ứng dụng và kết quả thực nghiệm mà luận văn đã đạt được.

CHƯƠNG 1

TỔNG QUAN VỀ BÀI TOÁN DỰ BÁO

1.1. Lịch sử của quá trình dự báo

Nhu cầu dự báo về một sự việc sẽ diễn tiến thế nào trong tương lai đã có từ nhiều thế kỷ trước. Những dự báo đầu tiên là dự báo về các hiện tượng tự nhiên, các hiện tượng xã hội và hiện tượng về đời sống xã hội.

Ban đầu dự báo chỉ dựa vào những kinh nghiệm của người dự báo, dần dần dự báo đã được hỗ trợ bởi công nghệ giúp cho kết quả dự báo chính xác hơn.

Có rất nhiều các phương pháp dự báo đã được nghiên cứu và cho ra đời như: Phương pháp hồi quy bội, phương pháp Delphi, phương pháp Cross Impact Matrices, phương pháp ARIMA (kết hợp của AR – Autoregressive và MA – Moving Average)...

Có rất nhiều cách phân loại dự báo như:

Dựa vào thời gian dự báo phân làm ba loại là: Dự báo dài hạn, dự báo trung hạn và dự báo ngắn hạn.

Theo phương pháp dự báo lại được phân thành: Dự báo bằng phương pháp chuyên gia, dự báo theo phương trình hồi quy, dự báo dựa vào dãy số thời gian...

1.1.1. Khái niệm về dự báo

Dự báo là một khoa học và nghệ thuật tiên đoán những sự việc sẽ xảy ra trong tương lai.

Dự báo có tính khoa học vì nó dựa trên những số liệu thu thập được từ quá khứ và căn cứ vào kết quả phân tích các nhân tố ảnh hưởng đến kết quả dự báo.

Tính nghệ thuật của dự báo là dựa trên những kinh nghiệm thực tế và khả năng phán đoán của các chuyên gia để đưa ra được những dự đoán với độ chính xác cao nhất.

1.1.2. Mục đích của dự báo

Đưa ra được quyết định chính xác, nhất quán: Phân tích dự báo sẽ cung cấp thông tin chi tiết về đối tượng dự báo từ đó sẽ đưa ra được các hành động chiến lược. Phân tích dự báo được thực hiện liên tục và cho kết quả đáng tin cậy nhờ có sự hỗ trợ của kỹ thuật. Các quyết định sẽ được đưa ra một cách nhất quán, công bằng chứ không phải dựa trên tính chủ quan của con người.

Giải quyết công việc nhanh hơn: Dự báo sẽ trả lời các câu hỏi phức tạp và xử lý chúng với độ chính xác cao trong khoảng thời gian ngắn. Có những quyết định trước đây phải mất hàng giờ hoặc vài ngày thì nhờ có sự hỗ trợ của khoa học chỉ còn vài phút hoặc vài giây.

Giảm chi phí do giảm rủi ro: Với sự hiểu biết về đối tượng giúp các nhà lãnh đạo đánh giá được chính xác những rủi ro và giảm tổn thất.[18]

1.1.3. Những thách thức trong phân tích dự báo

Mục đích của dự báo là để giúp cải tiến về hiệu quả, hỗ trợ ra quyết định của các nhà lãnh đạo. Tuy nhiên, không phải lúc nào dự báo cũng chính xác, một số yếu tố ảnh hưởng đến độ chính xác của dự báo là: Trở ngại trong quản lý, dữ liệu, xây dựng mô hình và quá trình triển khai.[18]

Những trở ngại trong quản lý. Thông thường để triển khai mô hình dự báo đòi hỏi có sự chuyên đổi về các nguồn lực cho tổ chức nên cần có sự hỗ trợ từ các nhà lãnh đạo để chuyển các mô hình từ nghiên cứu sang vận hành.

Những trở ngại về dữ liệu. Các mô hình thường yêu cầu dữ liệu dưới dạng một bảng hoặc bảng có chứa hàng và cột (dữ liệu hai chiều). Nếu dữ liệu được lưu trữ trong các cơ sở dữ liệu thì cần phải kết nối các cơ sở dữ liệu để tạo ra một bảng.

Trở ngại trong việc xây dựng mô hình. Trở ngại lớn nhất là quá tải, tức là mô hình quá phức tạp và yêu cầu phải ghi nhớ dữ liệu huấn luyện. Hai trở ngại với mô hình là: Thứ nhất mô hình thực hiện kém với dữ liệu mới và việc giải thích mô hình không đáng tin cậy. Thứ hai, các nhà xây dựng mô hình quá tham vọng vào mô hình được xây dựng trên dữ liệu có sẵn trong khoảng thời gian nhất định. Cách tốt nhất để khắc phục là xây dựng một mô hình đơn giản sau đó có thể được cải tiến sau khi chạy thử nghiệm.

Trở ngại trong triển khai mô hình. Thông thường các mô hình không quá phức tạp về mặt tính toán. Tuy nhiên, các mô hình phải được kiểm tra bởi hệ thống hoạt động và đưa ra dự đoán phù hợp với hệ thống đó.[18]

1.1.4. Các phương pháp dự báo

Các phương pháp dự báo được chia thành 2 phương pháp là phương pháp định tính và phương pháp định lượng.

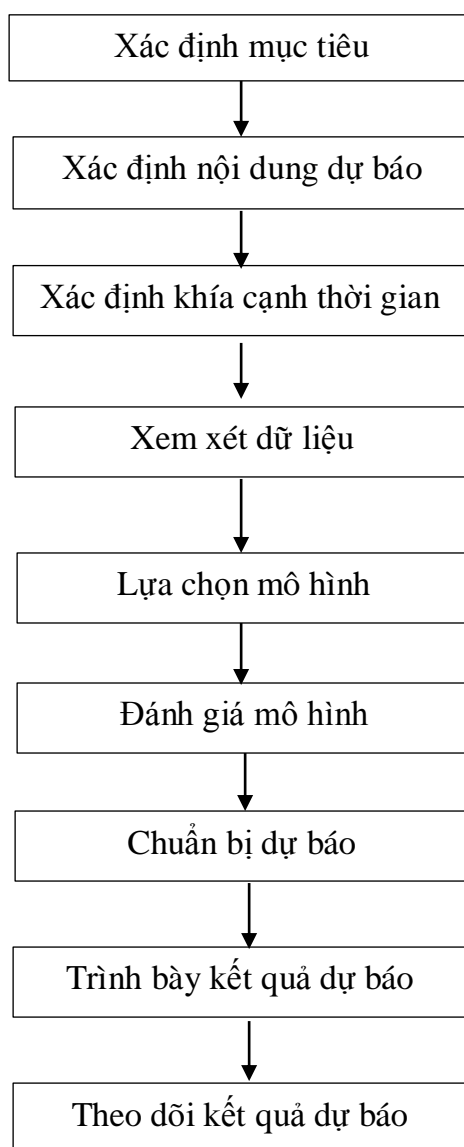
Phương pháp định tính: Hay còn gọi là phương pháp dự báo chuyên gia. Phương pháp này được sử dụng khi dữ liệu không có sẵn hoặc những dữ liệu có sẵn nhưng không đầy đủ để phân tích, đánh giá. Phương pháp định tính thường được sử dụng khi đối tượng dự báo bị tác động bởi những yếu tố không thể lượng hóa được. Nguyên tắc của phương pháp này là sử dụng ý kiến đánh giá của một hay nhiều chuyên gia trong lĩnh vực liên quan. Vì vậy, để đảm bảo tính chính xác của dự báo thì phải loại trừ ý kiến chủ quan của người dự báo.

Phương pháp định lượng: Sử dụng các dữ liệu từ quá khứ hoặc thu thập số liệu của các đối tượng ở hiện tại để dự báo. Với phương pháp định lượng sẽ cho kết quả dự báo nhanh và có thể đo được độ chính xác của dự báo. Tuy nhiên, phương pháp này chỉ áp dụng cho dự báo ngắn và trung hạn.

Để dự báo đạt được hiệu quả cao người ta thường kết hợp cả phương pháp định tính và phương pháp định lượng.

1.1.5. Quy trình thực hiện dự báo

Dự báo là một quá trình phức tạp nhưng về cơ bản quy trình có thể thực hiện theo các bước sau:



Hình 1.1. Quy trình thực hiện dự báo

Bước 1: Xác định mục tiêu

Xác định mục tiêu là xác định xem kết quả dự báo sẽ được sử dụng để làm gì. Mục tiêu chung của dự báo là lập kế hoạch và có những quyết định hành động hợp lý. Mục tiêu của dự báo tuyển sinh nhằm giúp các nhà lãnh đạo có những chiến lược phát triển một cách hợp lý nhất.

Bước 2: Xác định nội dung dự báo

Xác định chính xác dự báo cái gì. Ví dụ, mục tiêu chung là dự báo tuyển sinh nhưng một hệ thống dự báo gồm có: Số lượng học sinh, số lượng lớp... Luận văn xây dựng hệ thống thử nghiệm dự báo tuyển sinh với các thông số đầu vào là số liệu thu được trong 30 năm liên tiếp tại Trung tâm GDNN-GDTC Đổng Đa.

Bước 3: Xác định khía cạnh thời gian

Xác định độ dài của dự báo như: Dự báo dài hạn hay dự báo ngắn hạn và dự báo có tính cấp thiết như thế nào.

Bước 4: Xem xét dữ liệu

Xem xét dữ liệu là xem xét đến các yếu tố của dữ liệu như: Nguồn cung cấp, các bước phân loại, đánh giá, xử lý dữ liệu trước khi sử dụng. Trong luận văn này sử dụng nguồn dữ liệu thu thập được từ Trung tâm. Vì vậy, dữ liệu là chính xác và đáng tin cậy.

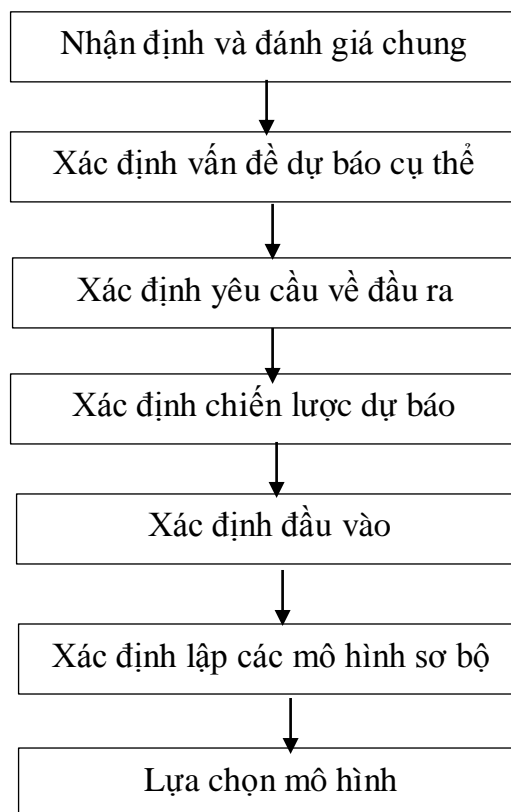
Bước 5: Lựa chọn mô hình

Để lựa chọn mô hình thích hợp cho dự báo cần xem xét các yếu tố như: Dữ liệu đầu vào, các yêu cầu về thời gian, yêu cầu về kết quả đầu ra, tài nguyên sẵn có...

Quy trình lựa chọn mô hình dự báo có thể được lựa chọn dựa trên một số chiến lược dự báo như sau:

1. Tiền định: Dựa trên mối quan hệ mật thiết giữa hiện tại và tương lai.
2. Triệu chứng: Dựa trên những dấu hiệu hiện tại để dự báo cho tương lai.
3. Hệ thống: Tức là cho rằng xu hướng phát triển trong tương lai sẽ tuân thủ theo một quy tắc nào đó, chẳng hạn các lý thuyết về giáo dục.

Luận văn sử dụng chiến lược dự báo tiền định để dự báo tuyển sinh.



Hình 1.2. Quy trình lựa chọn mô hình dự báo

Bước 6: Đánh giá mô hình

Với phương pháp dự báo định tính thì cần quan tâm đến việc đánh giá mô hình nhưng với phương pháp định lượng thì phải đánh giá mức độ phù hợp của mô hình và độ chính xác của dự báo. Nếu mô hình không phù hợp thì quay lại bước 5.

Để đánh giá độ phù hợp của mô hình luận văn sử dụng 2 tiêu chí MSE (Mean Square Error - công thức 1-1) và MAE (Mean Absolute Error - công thức 1-2).

Bước 7: Chuẩn bị dự báo

Chuẩn bị các số liệu để phục vụ cho việc dự báo.

Bước 8: Trình bày kết quả dự báo

Khi trình bày dự báo phải đảm bảo tính ngắn gọn, rõ ràng, chỉ ra được độ tin cậy của dự báo. Có rất nhiều cách để trình bày kết quả dự báo như: Bảng biểu, đồ thị hay hình ảnh minh họa và có thể trình bày ở dạng viết hoặc dạng nói.

Bước 9: Theo dõi kết quả dự báo

Sau khi dự báo phải xem xét kết quả dự báo tức xem xét độ lệch giữa giá trị dự báo và giá trị thực. Mục tiêu của việc theo dõi kết quả để tìm ra lý do tại sao lại có các sai số và xác định độ lớn của các sai số, qua đó bảo trì và nâng cấp hệ thống dự báo.

1.2. Phương pháp dự báo sử dụng mạng nơron theo đề xuất của luận văn

Mô hình mạng nơron là mô hình có khả năng “học” từ các dữ liệu quá khứ, có thể cập nhật các tham số. Vì vậy nếu lựa chọn được các tham số tối ưu thì nó sẽ là mô hình xấp xỉ rất tốt đường cong dịch chuyển của đối tượng cần dự báo. Kết quả dự báo cũng có độ chính xác cao. Đây là mô hình được lựa chọn cho bài toán dự báo tuyến tính của luận văn, cụ thể các vấn đề liên quan đến mạng nơron sẽ được trình bày trong chương 2.

1.3. Đánh giá mô hình dự báo

Để đánh giá độ chính xác của mô hình dự báo luận văn sử dụng hai đánh giá sau:

Sai số bình phương trung bình (*Mean Square Error*):

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{Q}_i - Q_i)^2 \quad (1-1)$$

Sai số tuyệt đối trung bình (*Mean Absolute Error*)

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{Q}_i - Q_i| \quad (1-2)$$

Trong đó:

\hat{Q}_i : Giá trị dự báo sinh ra bởi mô hình tại mẫu dữ liệu thứ i

Q_i : Giá trị quan sát tại mẫu dữ liệu thứ i

n : chiều dài chuỗi dự đoán

1.4. Kết luận chương 1

Dự báo là một nhu cầu thiết yếu đối với công tác quản lý vì nó mang tính định hướng cho tương lai, giúp đưa ra được những quyết định đúng đắn nhất. Trong các phương pháp dự báo hiện nay, mạng nơron tỏ ra có nhiều ưu điểm vì có mô hình tính toán linh hoạt, dễ thích nghi.

Dựa trên mô hình đã lựa chọn sau khi huấn luyện có thể đánh giá được hiệu quả của mô hình. Việc đánh giá chủ yếu dựa vào so sánh kết quả thực tế chứ chưa có phương pháp chuẩn để đánh giá.

TỔNG QUAN VỀ MẠNG NƠN NHÂN TẠO

Mạng nơon nhân tạo là một công cụ mạnh để giải quyết các bài toán có tính phi tuyến, phức tạp, đặc biệt trong các trường hợp mà yếu tố có mối quan hệ tác động qua lại không rõ rệt. Có rất nhiều loại mạng nơon khác nhau [3], [6], [12], [16] trong đó mạng nơon truyền thẳng nhiều lớp được sử dụng phổ biến nhất. Đã có nhiều nghiên cứu sử dụng mạng nơon truyền thẳng nhiều lớp trong bài toán dự báo [1], [2], [4], [5], [7], [8], [10], [13], [14], [15] và đã chứng tỏ đây là hướng tiếp cận rất hiệu quả. Trong chương này luận văn sẽ tìm hiểu những kiến thức về mạng nơon nhân tạo, mạng nơon truyền thẳng nhiều lớp và khả năng ứng dụng của chúng trong bài toán dự báo.

2.1. Mạng nơon nhân tạo là gì?

Một mạng nơon có thể được định nghĩa như một mô hình lý luận dựa trên bộ não của con người. Não người bao gồm một tập hợp các kết nối của các tế bào thần kinh. Bộ não con người có gần 10 tỷ tế bào thần kinh và 60 ngàn tỷ khớp thần kinh (Shepherd và Koch, 1990). Bằng cách sử dụng đồng thời nhiều tế bào thần kinh mà bộ não con người có thể thực hiện các chức năng nhanh hơn nhiều lần so với các máy tính nhanh nhất hiện nay.

Não người được coi như một hệ thống xử lý thông tin rất phức tạp, phi tuyến và song song. Não người có thể giải quyết rất nhiều vấn đề như: nghe, nói, nhìn, hồi ức thông tin, nhận dạng các mẫu... Não thực hiện được những nhiệm vụ như vậy nhờ các phần tử tính toán đó là nơon. Não sẽ phân bố việc xử lý cho hàng tỷ nơon có liên quan, điều khiển các mối liên hệ giữa các nơon. Nơon không ngừng nhận và truyền thông tin cho nhau. Trong quá trình xử lý các nơon tự liên kết với nhau tạo thành mạng. Mỗi mạng lại gồm hàng vạn các phần tử nơon khác nhau, mỗi nơon lại có khả năng liên kết với hàng vạn nơon khác.[3]

Ngoài ra, do tính mềm dẻo của sự kết nối giữa các tế bào thần kinh mà não người có khả năng học hỏi qua những kinh nghiệm từ quá khứ. Học là một đặc tính cơ bản và thiết yếu của mạng nơon sinh học.[17]

Dựa vào các đặc trưng cơ bản của nơon sinh học người ta đã xây dựng lên mạng nơon nhân tạo.

Mạng nơon nhân tạo (ANN) ngày nay gần giống với bộ não con người như: Có khả năng học, tức là sử dụng những kinh nghiệm để cải thiện hiệu suất. Khi thu thập được đủ một lượng mẫu thì ANN có thể khái quát hóa rất cao. Một ANN có thể nhận ra được một ký tự viết tay, có thể phát hiện bom ở sân bay...[17]

Chính vì vậy mà mạng nơon có thể giải quyết các vấn đề phức tạp và có quy mô lớn, khó hiện nay.

2.2. Sự tương đương nơon nhân tạo với nơon sinh học

Một mạng nơon nhân tạo bao gồm một số bộ xử lý đơn giản và có liên quan lẫn nhau, cũng được gọi là tế bào thần kinh, gần giống với bộ não của con người. Các tế bào thần kinh được kết nối bằng các trọng qua các tín hiệu từ một tế bào thần kinh khác. Mỗi tế bào thần kinh sẽ nhận được một số tín hiệu đầu vào thông qua các kết nối của nó. Các tín hiệu đầu ra được truyền đi thông qua các tế bào thần kinh kết nối (tương ứng với các sợi thần kinh sinh học).[17]

2.3. Lịch sử phát triển của mạng nơon nhân tạo

Năm 1890 William đã nghiên cứu về tâm lý học với sự liên kết các nơon thần kinh.

Năm 1940, McCulloch và Pitts đã chỉ ra rằng các nơon có thể được mô hình hóa như thiết bị ngưỡng (giới hạn) để thực hiện các phép tính logic. Trong thời gian này Wiener cũng nghiên cứu về mối liên hệ giữa nguyên lý phản hồi và chức năng của bộ não.

Vào những năm 1960 một số mô hình nơon được đưa ra như: Perception của Rosenbatt, Adaline của Widrow và ma trận học của Stinbuck. Trong đó mô hình Perception được chú ý vì tính đơn giản. Nhưng Minsky và Papert đã chứng minh rằng mô hình Perception không dùng được cho các hàm logic phức.

Đầu những năm 80 có sự đóng góp của Grossberg, Kohonen và Hoppfield. Đặc biệt là đóng góp lớn của Hoppfield khi đưa ra mạng rời rạc (1982) và mạng liên tục (1984). Từ đó làm cơ sở để Rumelhart và Hinton đề xuất thuật toán sai số truyền ngược để huấn luyện mạng nơon nhiều lớp nhằm giải những bài toán mà các mạng khác không làm được.

Từ những năm 1987 đến nay, hàng năm trên thế giới đều mở hội nghị toàn cầu chuyên ngành nơon IJCNN (International Joint Conference on Neural networks).

Ở Việt Nam mạng nơon được nghiên cứu từ những năm 1980 và được ứng dụng trong các lĩnh vực như: Tin học viễn thông, đo lường điều khiển... Một số chip nơon đã được sử dụng trong kỹ thuật lọc và một số ứng dụng khác. [3]

2.4. Nơon sinh vật

Mô hình nơon nhân tạo có nguồn gốc từ mô hình tế bào thần kinh (hay còn gọi là nơon) sinh vật.

Một tế bào nơon gồm bốn phần cơ bản là:

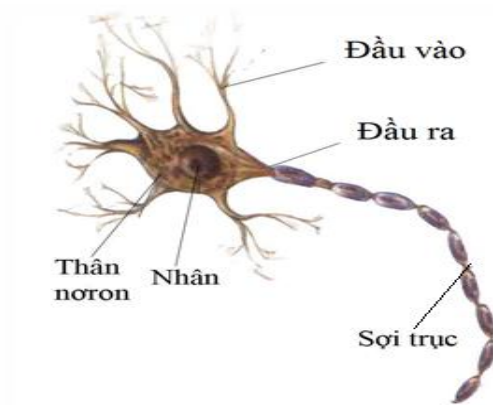
- **Các nhánh và rễ:** Là các bộ phận nhận thông tin. Các đầu nhập hoặc các đầu ra của các nơon khác bám vào rễ hoặc nhánh của một nơon. Khi các đầu vào từ ngoài này có sự chênh lệch về nồng độ K^+ , Na^+ hay Cl^- so với nồng độ bên trong của nó thì xảy ra hiện tượng thẩm thấu từ ngoài vào trong thông qua một cơ chế màng thẩm đặc biệt. Hiện tượng thẩm thấu tạo nên cơ chế truyền đạt thông tin. Mức độ thẩm thấu được đặc trưng bởi cơ chế màng tương trưng bằng một tỷ lệ. Tỷ lệ đó được gọi là tỷ trọng hoặc là trọng (weight).[3]

- **Thân Thần kinh (Soma):** Chứa các nhân và cơ quan tổng hợp protein. Các ion vào được tổng hợp và biến đổi. Khi nồng độ các ion đạt đến một giá trị nhất định,

xảy ra quá trình phát xung (hay kích thích). Xung đó được phát ở các đầu ra của nơron. Dây dẫn đầu ra xung được gọi là dây thần kinh (axon).[3]

- **Dây thần kinh (Axon):** Là một nhánh dài mang tín hiệu đầu ra. Đó là phương tiện truyền dẫn tín hiệu. Dây thần kinh được cấu tạo gồm các đốt và có thể dài từ micro mét đến vài mét tùy từng kết cấu cụ thể. Đầu ra này có thể truyền tín hiệu đến các nơron khác.[3]

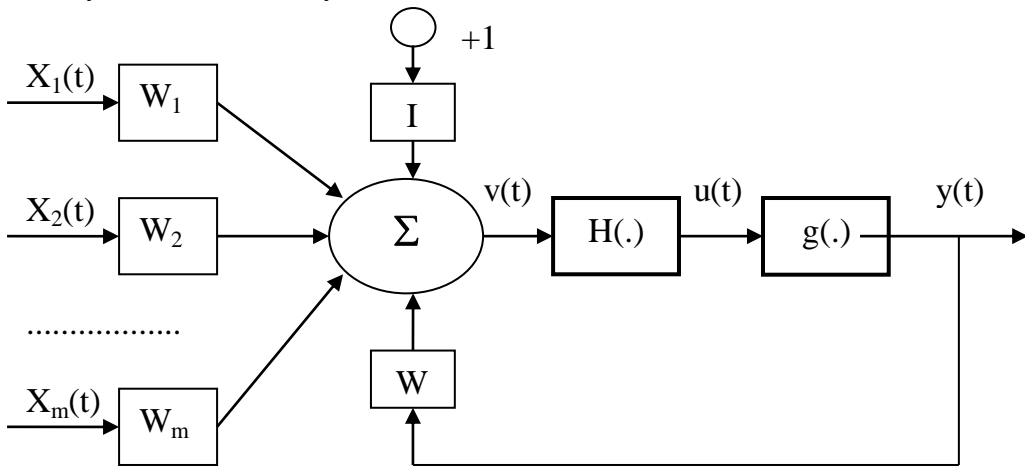
- **Khớp thần kinh (Synape):** Là bộ phận tiếp xúc của đầu ra nơron với rễ, nhánh của các nơron khác. Chúng có cấu trúc màng đặc biệt để tiếp nhận các tín hiệu khi có sự chênh lệch về nồng độ ion giữa bên trong và bên ngoài. Nếu nồng độ càng lớn thì việc truyền các ion càng nhiều và ngược lại.[3]



Hình 2.1. Cấu trúc nơron sinh vật

2.5. Nơron nhân tạo

2.5.1. Cấu tạo nơron nhân tạo



Hình 2.2. Mô hình một nơron nhân tạo

Một nơron nhân tạo được xây dựng từ ba thành phần chính: Bộ tổng các liên kết đầu vào, động học tuyến tính và phi tuyến không động học.

Bộ tổng liên kết: Bộ tổng hợp các liên kết đầu vào của một phân tử nơron có thể mô tả như sau:

$$v(t) = W \cdot y(t) + \sum_{k=1}^m W_k x_k(t) + I \tag{2-1}$$

Trong đó:

- $v(t)$: Tổng tất cả đầu vào mô tả toàn bộ thể năng tác động ở thân nơron;
- $x_k(t)$: Các đầu vào ngoài, mô tả tín hiệu vào từ các đầu nhạy thần kinh hoặc từ các nơron khác đưa vào;
- W_k : Trọng liên kết vào ngoài, là hệ số mô tả mức độ liên kết giữa các đầu vào ngoài tới nơron hiện tại, m là số đầu vào; $k = 1, \dots, m$;
- $y(t)$: Đầu ra nơron mô tả tín hiệu đưa ra;
- I : Hằng số, còn gọi là ngưỡng, xác định mức kích thích hay ức chế.

Phần động học tuyến tính: Đầu vào của phần động học là $v(t)$. Đầu ra của nó là $u(t)$ gọi là đầu ra tương tự. Hàm truyền tương ứng của phần động học tuyến tính có thể mô tả dưới dạng biến đổi Laplace như sau:

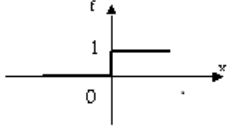
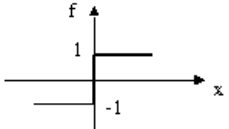
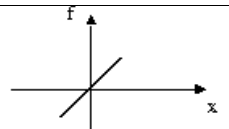
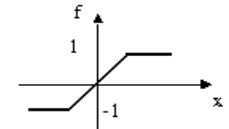
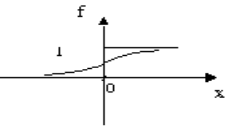
$$U(s) = H(s)V(s) \quad (2-2)$$

Phần phi tuyến: Các đầu ra của các nơron sinh vật là các xung, có giới hạn chặn. Khi mô phỏng để đảm bảo hệ ổn định đầu ra thường gán hàm chặn ở lối ra cho các tín hiệu. Tức là ở mỗi đầu ra của nơron phải đặt một hàm chặn, thường ở dạng phi tuyến với hàm $g(\cdot)$. Như vậy, đầu ra y có đặc trưng của một hàm :

$$y = g(u(t)) \quad (2-3)$$

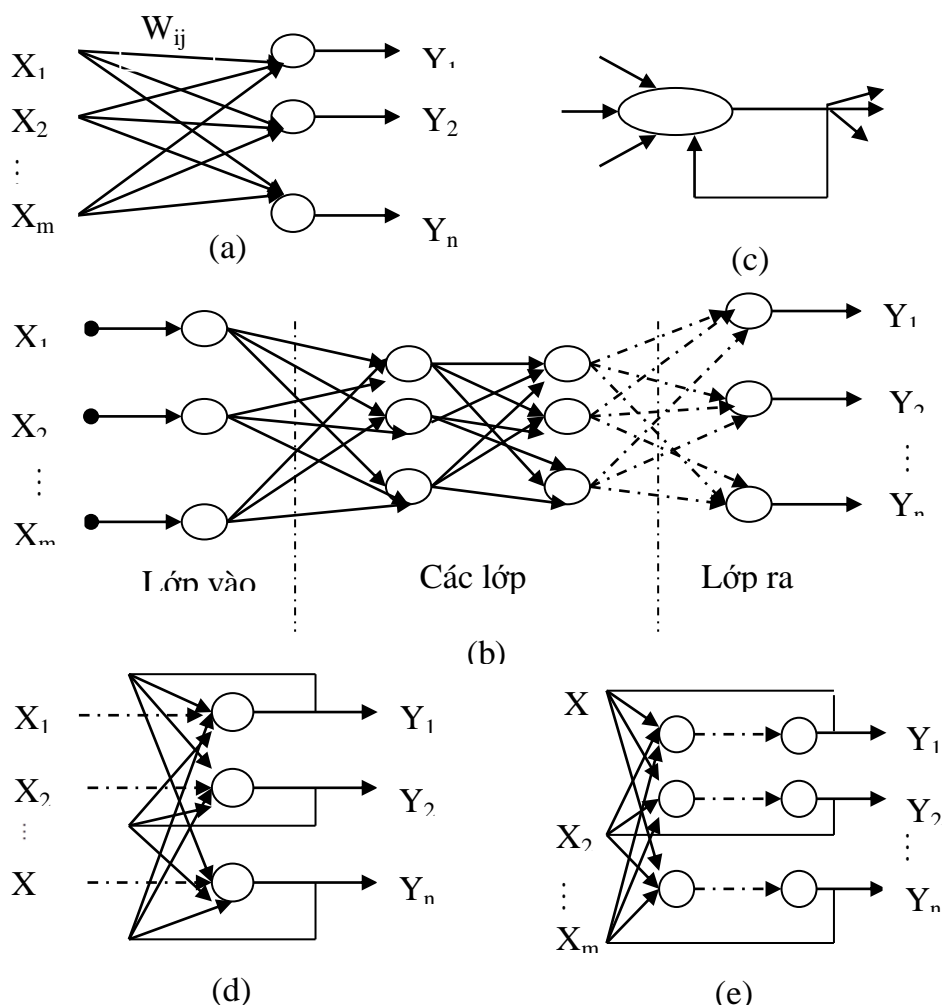
Một số hàm phi tuyến thường được sử dụng trong các mô hình nơron được mô tả trong bảng 2.1.

Bảng 2.1. Một số hàm phi tuyến thường được sử dụng trong các mô hình nơron

Tên hàm	Công thức	Đặc tính
Bước nhảy đơn vị	$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$	
Hàm dấu (sgn)	$f(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases}$	
Hàm tuyến tính	$f(x) = x$	
Hàm tuyến tính bão hòa đối xứng	$f(x) = \begin{cases} 1 & x > 1 \\ x & -1 \leq x \leq 1 \\ -1 & x < -1 \end{cases}$	
Hàm Sigmoid đơn cực	$f(x) = \frac{1}{1 + e^{-2x}}$	

2.5.2. Phân loại mạng nơron nhân tạo

Có rất nhiều cách phân loại mạng nơron nhân tạo. Dựa vào các đặc trưng mạng nơron nhân tạo được phân loại như sau:[3]



Hình 2.3. Phân loại mạng nơron

(a) Mạng truyền thẳng một lớp, (b) Mạng truyền thẳng nhiều lớp.

(c) Nơron tự phản hồi, (d) Mạng phản hồi một lớp, (e) Mạng phản hồi nhiều lớp

2.5.2.1. Phân loại mạng theo số lớp trong mạng

- **Mạng một lớp:** Là tập hợp các phần tử nơron có đầu vào và đầu ra trên cùng một phần tử. Nếu mạng nối đầu ra của các phần tử này với đầu vào của phần tử kia gọi là mạng tự liên kết (*Autoassociative*).[3]

- **Mạng nhiều lớp:** Gồm một lớp đầu vào và một lớp đầu ra riêng biệt. Các lớp nằm giữa lớp đầu vào và lớp đầu ra gọi là lớp ẩn (*Hidden Layer*).[3]

2.5.2.2. Phân loại theo đường truyền tín hiệu

- **Mạng truyền thẳng:** Là mạng hai hay nhiều lớp mà quá trình truyền tín hiệu từ đầu ra lớp này đến đầu vào lớp kia theo một hướng.[3]

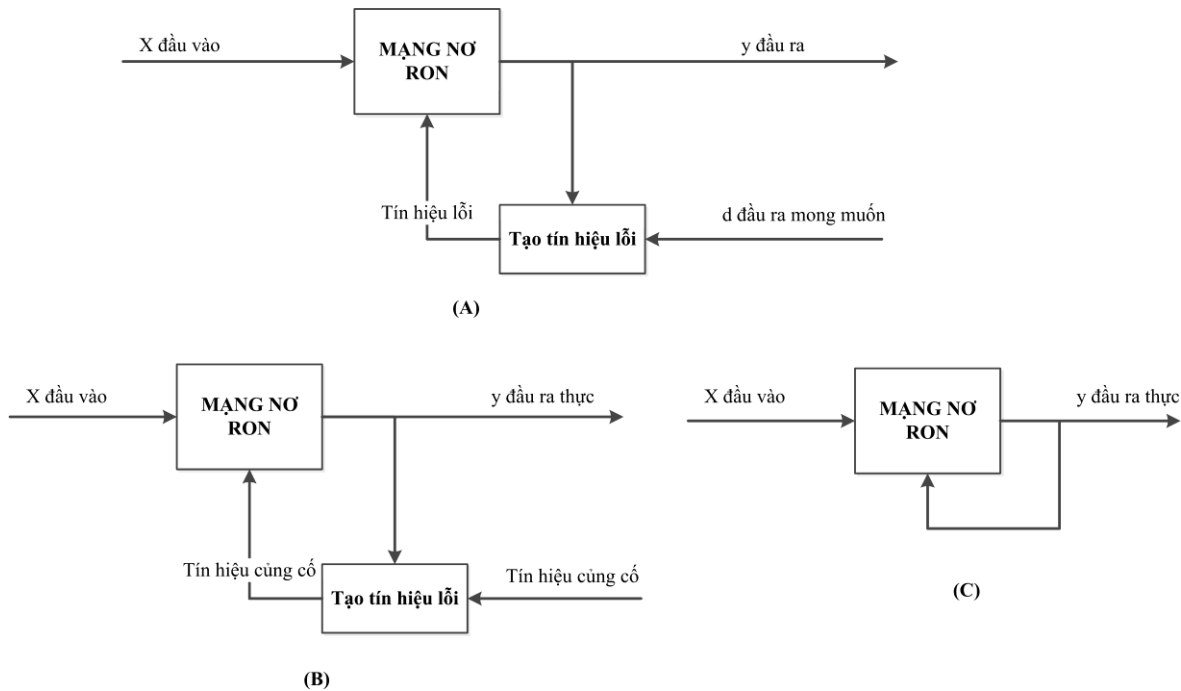
- **Mạng phản hồi:** Là mạng mà trong đó một hoặc nhiều đầu ra của các phần tử lớp sau truyền ngược tới đầu vào của lớp trước.[3]

- **Mạng tự tổ chức:** Là mạng có khả năng sử dụng những kinh nghiệm của quá khứ để thích ứng với những biến đổi của môi trường (không dự báo trước). Loại mạng này thuộc nhóm hệ học, thích nghi không cần có tín hiệu chỉ đạo từ bên ngoài.[3]

2.6. Luật học

2.6.1. Học tham số (Parameter Learning)

Học tham số tức là thay đổi, cập nhật các trọng liên kết. Hầu hết các luật học tồn tại thuộc kiểu học tham số. Luật học tham số thường được chia thành 3 dạng chính: Học giám sát, học không giám sát và học củng cố. [3]



Hình 2.4. Các dạng học

(A): Học giám sát; (B): Học củng cố; (C): Học không giám sát

- **Học giám sát (Supervised Learning):** Đầu vào và đầu ra của mạng được cho sẵn.

Mạng được cung cấp một tập các mẫu $\{(x(1), d(1)); (x(2), d(2)); \dots; (x(k), d(k))\}$ là các cặp đầu vào – đầu ra mong muốn. Khi một đầu vào $x(k)$ được đưa vào mạng, đầu ra mong muốn $d(k)$ cũng được đưa vào mạng. Như hình 2.4 (A), khi có sự sai khác giữa giá trị đầu ra thực sự $y(k)$ và đầu ra mong muốn $d(k)$ thì trọng sẽ được thay đổi sao cho giá trị sai khác là nhỏ nhất. [3]

- **Học củng cố (Reinforcement Learning):** Mạng chỉ được biết là giá trị đầu ra thực sự “quá cao” hay chỉ “chính xác 50%”, hoặc chỉ có được thông tin phản hồi là đầu ra đúng hay sai. Như hình 2.4 (B), luật học củng cố là một dạng của luật học giám sát vì mạng vẫn có được vài thông tin phản hồi từ môi trường. Tuy nhiên, thông tin phản hồi chỉ mang tính đánh giá chứ không mang tính chất dạy. [3]

- **Học không giám sát (Unsupervised Learning):** Không có bất kì một thông tin phản hồi nào từ môi trường. Mạng sẽ chỉ nhận được một số dữ liệu đầu vào và phải tự tìm ra các mẫu, đặc tính, quy tắc, sự tương quan trong dữ liệu đầu vào và tập hợp lại để tạo đầu ra. [3]

Các luật học có thể được đánh giá thành dạng chung với lượng điều chỉnh trọng như sau:

$$\Delta w_{ij} = \alpha r x_j(t) \quad (2-4)$$

Trong đó:

- i : Nơon thứ i ;
- j : Là đầu vào thứ j ;
- α : Là hằng số học (dương) xác định tốc độ học và được xác định bằng thực nghiệm;
- R : Tín hiệu học. Tín hiệu học tổng quát là một hàm của w , x và d tức là $r=f(w, x, d)$.

Đối với các trọng biến đổi liên tục, có thể sử dụng dạng sau:

$$\frac{dw_i(t)}{dt} = \alpha r x(t) \quad (2-5)$$

2.6.2. Học cấu trúc

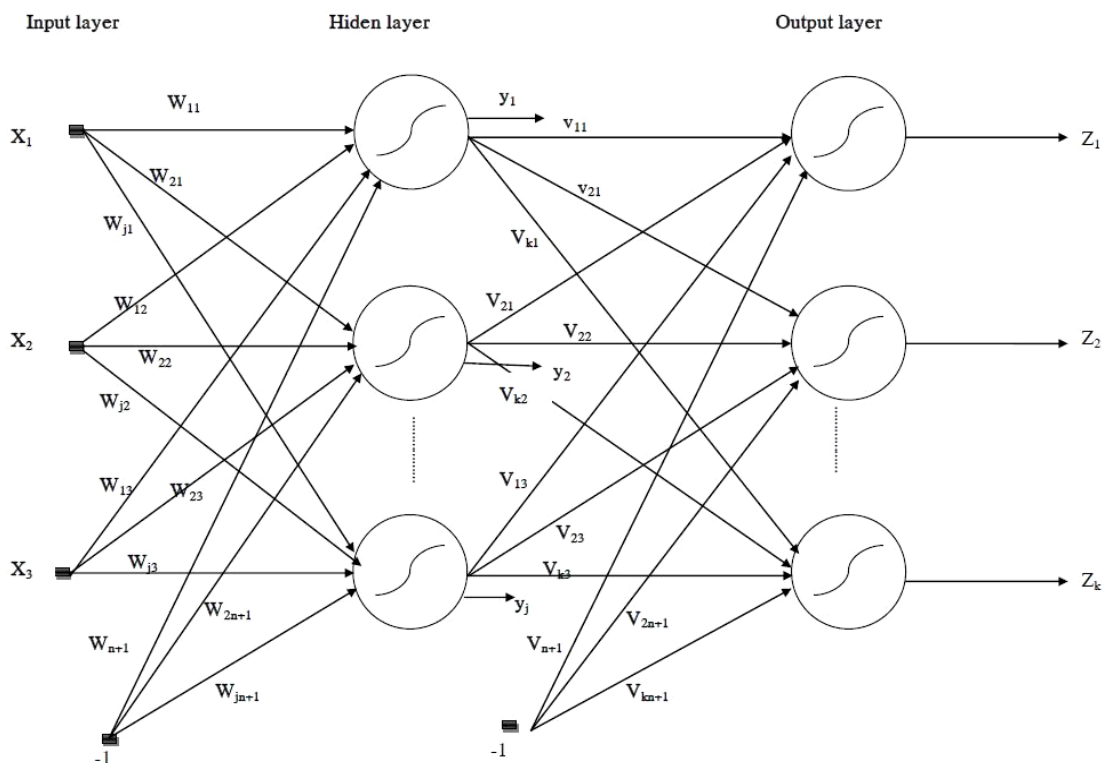
Học cấu trúc tức là thay đổi cấu trúc mạng bằng cách thay đổi số nơon, kiểu liên kết. Với học cấu trúc có thể sử dụng các kỹ thuật liên quan đến thuật toán Gen (GAs: *Genetic Algorithm*) và lập trình tiến hóa (EP: *Evolutionary Programming*).[3]

2.7. Mạng truyền thẳng nhiều lớp và thuật toán lan truyền ngược

2.7.1. Kiến trúc mạng

Một mạng truyền thẳng nhiều lớp bao gồm một lớp vào (*Input Layer*), một lớp ra (*Output Layer*) và một hoặc nhiều lớp ẩn (*Hidden Layers*) nằm giữa lớp vào và lớp ra. Các lớp đầu vào nhận tín hiệu vào và tái phân phối cho các nơon trong lớp ẩn. Các nơon đầu vào không thực hiện bất kỳ một tính toán nào. Các nơon lớp ẩn sẽ phát hiện các tính năng và trọng của các nơon đại diện cho các tính năng ẩn của lớp đầu vào. Những tính năng này sẽ được sử dụng bởi các lớp ra để xác định mô hình đầu ra.[17]

Luồng thông tin trong mạng nơon truyền thẳng sẽ đi từ trái qua phải, các giá trị đầu vào x được truyền tới các nơon lớp ẩn thông qua trọng số kết nối sau đó đưa tới lớp ra. Trọng số kết nối từ phần tử vào thứ i tới nơon ẩn thứ j được ký hiệu là w_{ij} , trọng số kết nối từ nơon ẩn thứ j tới các nơon ra thứ k được ký hiệu là v_{jk} . [17]



Hình 2.5. Mạng truyền thẳng nhiều lớp

Với lớp nơron ẩn thứ j :

$$a_j = \sum_{i=1}^n w_{ij} x_i + \theta_j, y_j = f(a_j) \quad (2-6)$$

Với nơron ra thứ k :

$$a_k = \sum_{j=1}^k v_{kj} y_j + \theta_k, z_k = f(a_k) \quad (2-7)$$

Hàm kích hoạt:

$$f(x) = \frac{1}{1+e^{-x}} \quad (2-8)$$

Hàm f đơn điệu tăng, khả vi và cho giá trị thuộc $[0; 1]$.

Với một mẫu đầu vào được cho trước, mạng sẽ tạo ra giá trị đầu ra thực tế z_k , giá trị này được so sánh với giá trị mẫu mong muốn ở đầu ra d_k . Các trọng số của mạng sau đó được hiệu chỉnh để giảm lỗi và đưa ra các mẫu tiếp theo. Sau đó trọng số sẽ tiếp tục được hiệu chỉnh tới khi tổng lỗi qua tất cả các mẫu học được giảm tới mức cho phép. Thuật toán học này được hiểu như thuật toán lan truyền ngược.

2.7.2. Xác định cấu trúc mạng tối ưu

Dưới đây luận văn trình bày một số vấn đề cần quan tâm khi thiết kế một mạng.

2.7.2.1. Số lớp ẩn

Với mạng có một lớp ẩn có thể biểu diễn cho bất kỳ một tín hiệu đầu vào nào đó của lớp vào. Với mạng có hai lớp ẩn có thể thể hiện các hàm với các dạng điệu bất kỳ.[17]

Phần lớn các thuật toán huấn luyện mạng cho các nơron truyền thẳng đều dựa trên phương pháp Gradient. Các lớp thêm vào sẽ tăng thêm việc lan truyền các lỗi làm cho vector Gradient không ổn định. [17]

Với thuật toán tối ưu dựa trên Gradient chỉ có thể tìm ra tối thiểu cục bộ và rất khó để tìm ra tối thiểu toàn cục vì xác suất bị tắc tại tối thiểu cục bộ là khá lớn.[17]

2.7.2.2. Số nơron trong lớp ẩn

Nếu số nơron trong lớp ẩn quá ít có thể dẫn đến việc không thể nhận dạng được đầy đủ các tín hiệu trong một tập dữ liệu phức tạp, hay thiếu ăn khớp (*Underfitting*). Nhưng nếu có quá nhiều lại làm tăng thời gian huấn luyện mạng và có thể dẫn đến tình trạng thừa ăn khớp (*Overfitting*). Tức là mạng có quá nhiều thông tin, hoặc lượng thông tin trong tập dữ liệu mẫu (*Training Set*) không đủ các dữ liệu đặc trưng để huấn luyện mạng. Số lượng các đơn vị trong lớp ẩn phụ thuộc vào rất nhiều yếu tố như số đầu vào, đầu ra của mạng, số trường hợp trong tập mẫu, độ nhiễu của dữ liệu đích, độ phức tạp của hàm lỗi, kiến trúc mạng và thuật toán huấn luyện mạng.

Thông thường để xác định được số đơn vị tối ưu trong lớp ẩn cần phải thông qua huấn luyện mạng với một bộ số các đơn vị trong lớp ẩn và dự báo lỗi tổng quát hóa của từng lựa chọn. Cách tốt nhất để tìm ra số đơn vị tối ưu là sử dụng phương pháp thử sai (*Trial – and – Error*).

2.7.3. Cấu trúc của luật học lan truyền ngược

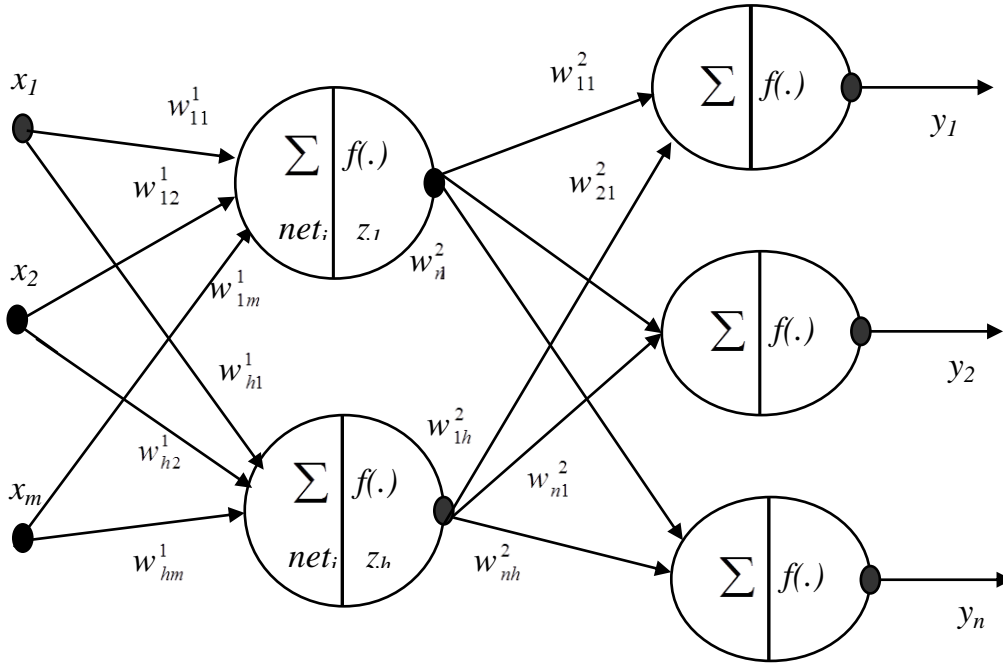
Thuật toán lan truyền ngược được quyết định bởi sự kết nối giữa các nơron (các kiến trúc mạng), hàm kích hoạt và các thuật toán học (hay luật học) để điều chỉnh trọng số.[17]

Thuật toán lan truyền ngược thường được sử dụng cho mạng đa lớp. Các lớp trong mạng được kết nối đầy đủ với nhau, tức là nơron của một lớp sẽ kết nối với tất cả các nơron khác trong lớp trước.[17]

Luật học lan truyền ngược thực hiện 2 pha.

- **Pha 1:** Mẫu đầu vào $x^{(k)}$ được truyền tiến (từ lớp vào tới lớp ra), kết quả của luồng dữ liệu thẳng (forward) là đầu ra thực $y^{(k)}$.
- **Pha 2:** Tín hiệu lỗi được tính trên cơ sở sai khác giữa $d^{(k)}$ và $y^{(k)}$ được lan truyền ngược (từ lớp ra quay trở lại các lớp trước đó) để hiệu chỉnh trọng số.[17]

Xét một mạng ba lớp: Lớp vào có m nơron, lớp ẩn có h nơron và lớp ra có n nơron (Hình 2.6).



Hình 2.6. Cấu trúc mạng lan truyền ngược.

- **Lớp ẩn:** Với mẫu đầu vào x , tại nơron thứ q của lớp ẩn:

$$net = \sum_{j=1}^m w_{jq}^1 x_j \quad j=1, 2, \dots, m; q=1, 2, \dots, h; \quad (2-9)$$

- Với đầu ra:

$$z_q = f(net_q) = f\left(\sum_{j=1}^m w_{jq}^1 x_j\right) \quad (2-10)$$

- Trong đó, $f(\cdot)$ là hàm kích hoạt đầu ra.

- **Lớp ra:** Giả thiết dùng 1 hàm kích hoạt đầu ra $f(\cdot)$. Tín hiệu tổng đầu vào trên nơron thứ i :

$$net_i = \sum_{q=1}^h w_{iq}^2 z_q = \sum_{q=1}^h w_{iq}^2 f\left(\sum_{j=1}^m w_{jq}^1 x_j\right) \quad (2-11)$$

Đầu ra:

$$y_i = f(net_i) = f\left(\sum_{q=1}^h w_{iq}^2 z_q\right) = f\left(\sum_{q=1}^h w_{iq}^2 f\left(\sum_{j=1}^m w_{jq}^1 x_j\right)\right) \quad (2-12)$$

2.7.4. Luật học lan truyền ngược

2.7.4.1. Thuật toán

Giải thuật lan truyền ngược tìm kiếm một vector các trọng số (*weights vector*) giúp cực tiểu hóa lỗi tổng thể của hệ thống đối lập với tập học.

Giải thuật lan truyền ngược bao gồm hai giai đoạn:

Giai đoạn lan truyền tiến tín hiệu (*Signal Forward*). Các tín hiệu đầu vào (vector các giá trị đầu vào) được lan truyền tiến từ tầng đầu vào đến tầng đầu ra (đi qua các tầng ẩn).

Giai đoạn lan truyền ngược lỗi (*Error Backward*)

- Căn cứ vào giá trị đầu ra mong muốn của vector đầu vào, hệ thống sẽ tính toán giá trị lỗi.
- Giá trị lỗi được lan truyền ngược từ tầng đầu ra đến tầng đầu vào.
- Các trọng sẽ được cập nhật đến khi lỗi có thể chấp nhận được.[17]

Ta sử dụng các ký hiệu sau:

- j : Nơron thứ J (hay nút thứ J);
- x_j : Vector đầu vào của nơron thứ J ;
- w_j : Vector trọng số của nơron thứ J ;
- x_{ij} : Trọng số trên x_{ij} ;
- b_j : Ngưỡng tại nút thứ J ;
- $u_j = x_j * w_j - b_j$: Tổng trọng số trên các đầu vào của nút thứ J ;
- o_j : Đầu vào của nút thứ J (với $o_j = f(u_j)$);
- t_j : Đầu ra mong muốn của nút thứ j ;
- $D(j)$: Tập các nút nhận đầu ra của nút thứ J làm một giá trị đầu vào;
- *Outputs*: Tập các nút trong tầng ra;
- η : Tốc độ học;
- f : Hàm truyền, $f(x) = \frac{1}{1+e^{-x}}$

Ta tìm công thức tính w_{ij} . Sự thay đổi trong mỗi lần cập nhật với:

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}}$$

Trong đó E là hàm lỗi.

Do trọng số cập nhật sau mỗi mẫu huấn luyện, để đơn giản có thể coi tập huấn luyện chỉ gồm một mẫu. Khi đó hàm lỗi được tính như sau :

$$E = \frac{1}{2} \sum_{k \in \text{Outputs}} (t_k - f(u_k))^2 \quad (2-13)$$

Theo quy tắc chuỗi ta có:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial u_j} \frac{\partial u_j}{\partial w_{ji}} \quad (2-14)$$

Mặt khác :

$$u_j = \sum_i w_{ji} \cdot x_{ji} - b_j \quad (2-15)$$

Do đó :

$$\frac{\partial u_j}{\partial w_{ji}} = x_{ji} \quad (2-16)$$

Thay vào (2-16) ta được :

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial u_j} x_{ji} \quad (2-17)$$

Xét hai trường hợp :

Trường hợp 1: j là nơron tầng ra ($j \in \text{Outputs}$)

$$\text{Đặt : } \delta_j = -\frac{\partial E}{\partial u_j}$$

Ta có :

$$\Delta w_{ji} = -\frac{\partial E}{\partial w_{ji}} = \eta \delta_j x_{ji} \quad (2-18)$$

Tất cả các nút đầu ra $k \neq j$ là độc lập với w_{ji} , ta có thể coi :

$$E = \frac{1}{2}(t_j - o_j)^2 \quad (2-19)$$

Khi đó :

$$\begin{aligned} \delta_j &= \frac{\partial E}{\partial u_j} = -\frac{\partial}{\partial u_j} \frac{1}{2}(t_j - o_j)^2 \\ &= (t_j - o_j) \frac{\partial o_j}{\partial u_j} \\ &= (t_j - o_j) \frac{\partial}{\partial u_j} f(u_j) \\ &= (t_j - o_j)(1 - f(u_j))f(u_j) \\ &= (t_j - o_j)(1 - o_j)o_j \end{aligned}$$

Trường hợp 2: j là neuron tầng ẩn

Ta có nhận xét sau :

1. Với mỗi nút $k \in D(j)$, u_k là một hàm u_j .
2. Việc tạo ra các lỗi của tất cả các nút $l \neq j$ trong cùng một tầng với J là độc lập với w_{ji} .
3. Cũng theo quy tắc chuỗi :

$$\begin{aligned} \frac{\partial E}{\partial w_{ji}} &= \sum_{k \in D(j)} \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial o_j} \frac{\partial o_j}{\partial u_j} \frac{\partial u_j}{\partial w_{ji}} \\ &= \sum_{k \in D(j)} \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial o_j} \frac{\partial o_j}{\partial u_j} x_{ji} \end{aligned}$$

Nếu đặt:

$$\delta_j = \sum_{k \in D(j)} \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial o_j} \frac{\partial o_j}{\partial u_j}$$

Ta có:

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}} = \eta \delta_j x_{ji} \quad (2-20)$$

Từ đó:

$$\begin{aligned} \delta_j &= -\sum_{k \in D(j)} \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial o_j} \frac{\partial o_j}{\partial u_j} = -\sum_{k \in D(j)} (-\delta_k) w_{kj} o_j (1 - o_j) \\ &= o_j (1 - o_j) \sum_{k \in D(j)} \delta_k w_{kj} \end{aligned}$$

Như vậy từ (2-18) và (2-20) ta thấy rằng trong mọi trường hợp đều có:

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}} = \eta \delta_j x_{ji} \quad (2-21)$$

Trong đó δ_j được tính như sau:

Với j là nút xuất thì :

$$\delta_j = (t_j - o_j)(1 - o_j)o_j \quad (2-22)$$

Với j là nút ẩn thì:

$$\delta_j = o_j(1 - o_j) \sum_{k \in D(j)} \delta_k w_{kj} \quad (2-23)$$

Thuật toán lan truyền ngược được mô tả lại như sau :

Input:

- Mạng feed-forward với n_i đầu vào, n_k nút ẩn vào neuron đầu ra.
- Hệ số học η .
- Tập dữ liệu huấn luyện $X = \left\{ \left(\begin{matrix} \mathbf{x} \\ \mathbf{t} \end{matrix} \right) \mid \mathbf{x} = (x_1, x_2, x_n) \right\}$ là vector đầu vào
 $\mathbf{t} = (t_1, t_2, t_m)$ là vector đầu ra mong muốn

Output: Các vector trọng số \mathbf{w}_j

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 $\left(\begin{matrix} \mathbf{x} \\ \mathbf{t} \end{matrix} \right) \in X$ thực hiện các bước sau:

1. Tính đầu ra o_j cho mỗi nút j
2. Với mỗi nút k thuộc tầng ra, tính δ_k theo công thức:

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

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

$$\delta_h = o_h(1 - o_h) \sum_{k \in D(h)} \delta_k w_{kh}$$

4. Cập nhật: $w_{ji} = w_{ji} + \Delta w_{ji}$

Trong đó $\Delta w_{ji} = \eta \delta_j x_{ji}$

2.7.4.2. Các yếu tố ảnh hưởng đến quá trình học theo phương pháp lan truyền ngược sai số

Khởi tạo các trọng số

Xét trường hợp sử dụng luật học Gradient, ta xem giá trị khởi tạo sẽ ảnh hưởng thế nào tới quá trình học (gồm tốc độ học, tính tối ưu...)

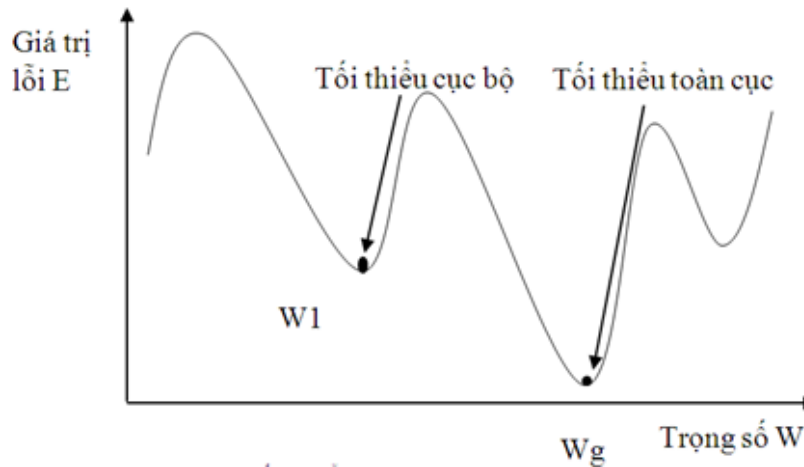
Trong không gian tương tự 1 chiều:

Với tổng sai số:

$$E = \frac{1}{2} \sum_{i=1}^n (e_i^2) = \frac{1}{2} \sum_{i=1}^n (d_i - y_i)^2 \quad (2-24)$$

Phương pháp hạ Gradient:

$$\Delta W_{ij} = -\alpha \frac{\partial E}{\partial W_{ij}} = -\alpha \frac{1}{2} \frac{\partial}{\partial W_{ij}} \sum (d_i - y_i)^2 = \alpha \sum_{i=1}^n (d_i - y_i) \frac{\partial \sum_{j=1}^m W_{ij} x_j}{\partial W_{ij}} \quad (2-25)$$



Hình 2.7. Tối thiểu cục bộ (Local Minimum)

Giả sử mô hình thực hiện giảm dần độ dốc (Gradient Descent) và đạt giá trị lỗi tại điểm W_1 mà không phải là W_g . Điểm W_1 được gọi là điểm tối thiểu cục bộ (Local Minimum). Điểm W_g được gọi là điểm tối thiểu toàn cục. Tuy nhiên mô hình chỉ có thể đạt đến điểm tối thiểu toàn cục là điểm W_g khi và chỉ khi mô hình thoát khỏi điểm W_1 .

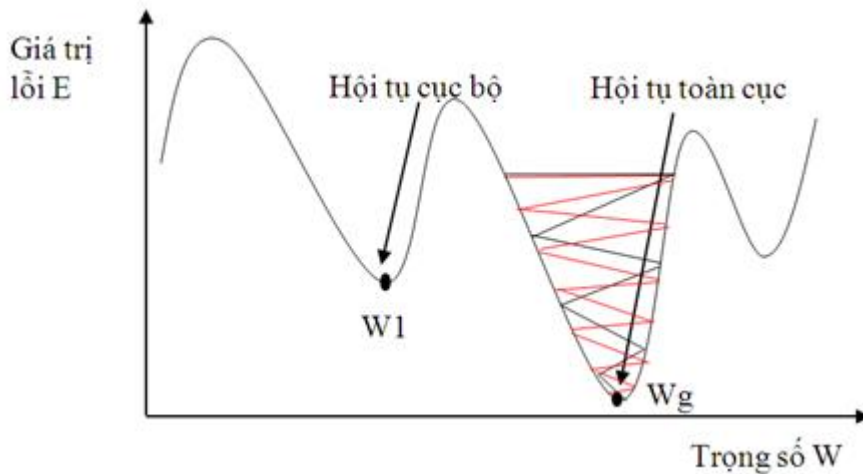
Giải thuật BP có nhược điểm lớn là mô hình mạng có thể rơi vào điểm tối thiểu cục bộ mà không phải là tối thiểu toàn cục. Để khắc phục nhược điểm này ta đưa thêm vào một khái niệm là Momentum. Momentum là hệ số góp phần giúp cho quá trình chỉnh sửa trọng số giảm khả năng rơi vào các điểm tối thiểu cục bộ, đồng thời làm giảm thời gian huấn luyện. Giá trị trọng số ở lần huấn luyện thứ $t+1$ được tính dựa trên giá trị của các trọng số ở các lần lặp trước đó.

$$\Delta W_{ij}(t) = -\eta \frac{\partial E}{\partial W_{ij}}(t) + \mu * \Delta W_{ij}(t-1) \quad (2-26)$$

Hệ số quán tính μ có tác dụng điều chỉnh mức độ ảnh hưởng của giá trị $\Delta W_{ij}(t-1)$ ở bước lặp trước lên giá trị $\Delta W_{ij}(t)$. Giúp cho giải thuật không bị dừng ở hội tụ cục bộ.

Hằng số học η

Giả sử ta cho $\eta=0.45$. Mỗi lần thay đổi 1 lượng ΔW_{ij} tỷ lệ với 0.45 và 1 lượng ΔW_{ij} tỷ lệ với 0.2 ta có thể mô tả trên hình 2.8 như sau :



Hình 2.8. Tốc độ học của hằng số học η
Đường màu đỏ $\eta=0.2$
Đường màu đen $\eta=0.45$

Như vậy ta nhận thấy với hằng số học càng lớn thì tốc độ học càng nhanh nhưng độ chính xác lại thấp. Ngược lại hằng số học càng nhỏ thì tốc độ học càng lâu nhưng độ chính xác cao.

Hàm giá

Hàm giá hay còn gọi là hàm mục tiêu hoặc hàm sai số. Nếu E_0 càng lớn thì sai số càng cao và ngược lại. Hàm giá đạt cực tiểu khi hai đối số d_i và y_i bằng nhau. Thông thường hàm giá được chọn có dạng:

$$E = \frac{1}{p} \sum_{i=1}^n (d_i - y_i)^p \quad \text{với } 1 \leq p \leq \infty \quad (2-27)$$

2.7.5. Một số vấn đề của mạng nơron nhiều lớp

Mạng nơron nhiều lớp có một số vấn đề cần được quan tâm như sau:

1. Thời gian huấn luyện lâu và không phải luôn hội tụ.
2. Không biết trước được sự ràng buộc giữa lỗi huấn luyện và cấu trúc mạng.
3. Không biết trước hiệu suất huấn luyện (thời gian huấn luyện, độ lỗi).
4. Khó xác định độ lỗi của mạng từ tập huấn luyện.
5. Khó xác định độ lớn của tập huấn luyện.

2.7.6. Ưu nhược điểm của mạng truyền thẳng

Ưu điểm:

- Hỗ trợ tính toán song song ở mức rất cao.
- Có khả năng chịu nhiễu, nhờ các tính toán song song.
- Có thể được thiết kế để tự thích nghi (Các trọng số, cấu trúc mạng).

Nhược điểm:

- Không có quy tắc tổng quát để xác định cấu trúc mạng và các tham số học tối ưu cho một lớp, một bài toán xác định.
- Không có phương pháp tổng quát để đánh giá hoạt động bên trong của ANN (vì vậy hệ thống ANN được xem như là một "hộp đen").
- Rất khó (không thể) để giải thích cho người dùng.

- Rất khó để dự đoán hiệu năng của hệ thống trong tương lai (khả năng khái quát hóa của hệ thống học)

2.8. Kết luận chương 2

Chương này nêu ra các khái niệm cơ bản nhất về mạng nơron sinh vật, làm tham chiếu cho các khái niệm về mạng nơron nhân tạo mô tả sau đó. Chương này đưa ra mô hình của mạng nơron và mạng liên kết các nơron cũng như các phương pháp huấn luyện mạng. Mạng nhiều lớp lan truyền thẳng cũng được mô tả kỹ để làm tiền đề cho việc thiết kế phần mềm dự báo của luận văn.

CHƯƠNG 3

ỨNG DỤNG MẠNG NƠN TRONG DỰ BÁO SỐ HỌC SINH TUYỂN VÀO TRUNG TÂM GDNN-GDTX ĐÔNG ĐA

Dự báo giáo dục có thể chia làm nhiều loại như: Dự báo tỉ lệ nghỉ học của sinh viên, dự báo tỉ lệ đỗ tốt nghiệp, tỉ lệ đỗ đại học, số học sinh tuyển vào của những năm tiếp theo...

Luận văn “Dự báo kết quả đậu đại học” của ThS. Thái Trung Hải cũng sử dụng mạng nơon truyền thẳng với thuật toán lan truyền ngược. Trong luận văn sử dụng mạng 3 lớp (1 lớp vào, 1 lớp ẩn, 1 lớp ra) với số nơon đầu vào là 13, lớp ẩn là 13 và đầu ra là 3. Đầu vào của luận văn là 13 yếu tố ảnh hưởng đến dự báo đó là kết quả học tập của 13 môn học. 3 đầu ra là điểm thi đại học, điểm thi tốt nghiệp và nhóm trường đậu đại học.[1]

Trong đồ án tốt nghiệp của tác giả Nguyễn Văn Trịnh về “Ứng dụng mạng nơon trong dự báo tỷ lệ nghỉ học” lại xây dựng mô hình mạng nơon khác với luận văn trên. Đồ án xây dựng ba mô hình là: Mô hình dự báo tình trạng nghỉ học của học viên với đầu vào của mạng là hồ sơ của những học viên đã tốt nghiệp và những học viên nghỉ học và đầu ra là trạng thái học viên đó là: Nghỉ học và tốt nghiệp. Mô hình thứ hai là: Dự báo tình trạng học tập của học viên qua 1 kỳ xác định (ký hiệu là kỳ $x=2, 3, 4$) có đầu vào là hồ sơ của những học viên đã học qua kỳ x và những học viên đã nghỉ học trước kỳ x và đầu ra là trạng thái học viên là nghỉ học và học vượt qua kỳ x . Mô hình thứ 3 là: Dự báo tiềm năng nghỉ học của học viên trong quá trình học có đầu vào là kỳ x , hồ sơ học viên, quy mô lớp, điểm tích lũy trung bình, số lần đăng nhập diễn đàn, tỉ lệ nghỉ học kỳ trước và đầu ra là trạng thái học viên với màu xanh là học thêm được ít nhất 2 kỳ nữa, màu vàng là kỳ tới sẽ nghỉ học và màu đỏ là kỳ này nghỉ học. Đồ án luyện mạng sử dụng các quy tắc học là Delta, momen và delta-bar-delta.[7]

Bài toán dự báo có nhiệm vụ phân tích và sử dụng số liệu đã có trong quá khứ để dự đoán giá trị tương lai. Hiện nay, có rất nhiều phương pháp dự báo đã được sử dụng trên mô hình vật lý và mô hình toán học. Trong nhiều trường hợp, kết quả nghiên cứu dự báo theo các mô hình trên đã đạt được những thành công nhất định [1], [2], [4], [5], [7], [8]. Tuy nhiên, phương pháp nào cũng có những hạn chế, do vậy bài toán dự báo vẫn là nội dung được nghiên cứu hiện nay. Trên thế giới, các phương pháp khai phá dữ liệu (đặc biệt là các phương pháp học máy mạng nơon) cũng được áp dụng nhiều vào dự báo [10], [11], [13], [14], [15].

Luận văn giải quyết bài toán dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa nhằm phục vụ công tác quản lý, hỗ trợ lên kế hoạch tuyển sinh. Chương này sẽ trình bày một số nội dung liên quan đến bài toán dự báo và một số thử nghiệm sử dụng những nghiên cứu trong chương 2 để dự báo số học sinh tuyển vào Trung tâm.

3.1. Giới thiệu về Trung tâm GDNN-GDTX Đống Đa

Tháng 10 năm 1978, Sở GD&ĐT Hà Nội đã quyết định cho các quận được mở trường Bổ túc văn hóa thanh thiếu niên cấp III - Quận Đống Đa khi đó mở 3 trường là: Trường số 3, số 4 và số 5 – cùng với hệ thống trường vừa học – vừa làm và trường Bổ túc văn hóa Dân Chính đã có từ trước tạo thành mạng lưới trường Bổ túc văn hóa. Mọi hoạt động Dạy và Học cùng các hoạt động ngoại khóa đều được thực hiện như các trường trung học phổ thông khác. Riêng về chương trình, học sinh chỉ học một số môn khoa học cơ bản như: Văn, Toán, Lý, Hóa, Sinh, Chính trị và học nghề.

Những ngày đầu các trường Bổ túc văn hóa số 3, số 4 và số 5 mới chỉ có 5-15 cán bộ, giáo viên biên chế từ các trường cấp III chuyển về và một số giáo viên thỉnh giảng, cơ sở vật chất vô cùng nghèo nàn: Chỉ có một phòng làm việc khoảng gần 20m², một số tủ đựng đồ thô sơ và vài bàn giáo viên, địa điểm phải học nhờ các trường vào buổi tối. Mặc dù vậy, số học sinh lại rất đông, mỗi trường tuyển sinh được từ 400 đến 600 em.

Năm học 1998 – 1999, trường Bổ túc văn hóa số 4 được đổi tên thành Trung tâm GDTX Đống Đa. Chức năng, nhiệm vụ của Trung tâm GDTX là: Xóa mù chữ; giáo dục tiếp tục sau biết chữ; phổ cập giáo dục cấp THCS; GDTX cấp THPT cho học sinh, cán bộ và nhân dân; công tác dạy nghề (điện dân dụng, nấu ăn, cắt may...), dạy vi tính, ngoại ngữ, tham gia tư vấn hoạt động cho các Trung tâm học tập cộng đồng phường góp phần nâng cao dân trí, nâng cao chất lượng nguồn nhân lực.

Tháng 11 năm 2009, trường Bổ túc văn hóa số 3 và Trung tâm GDTX Đống Đa hợp nhất thành Trung tâm GDTX Đống Đa theo quyết định số 5998/QĐ-UBND của UBND thành phố Hà nội.

Về cơ sở vật chất Trung tâm có:

Bảng 3.1. Cơ sở vật chất của Trung tâm

STT	Nội dung	Đơn vị tính	Số lượng
1	Tổng diện tích	m ²	4.487,3
2	Số phòng học	phòng	19
3	Số phòng thí nghiệm, đa năng	phòng	03
4	Phòng làm việc	phòng	11
5	Phòng Tin học	phòng	03

Đội ngũ cán bộ, giáo viên, nhân viên gồm 24 đồng chí với 3 tổ: Xã hội, Tự nhiên, Hành chính – Tổng hợp; 100% đạt chuẩn, 7 đồng chí trên chuẩn (thạc sỹ).

Tỉ lệ học sinh đỗ tốt nghiệp của Trung tâm luôn cao hơn tỉ lệ đỗ tốt nghiệp trung bình của thành phố Hà Nội.

Trung tâm GDTX Đống Đa là Trung tâm duy nhất trên toàn Quốc đã đưa giáo dục giá trị sống và kỹ năng sống vào giảng dạy chính thức.

Tháng 1 năm 2017 Trung tâm GDTX Đống Đa sát nhập với Trung tâm Dạy nghề Đống Đa, Trung tâm giáo dục Kỹ thuật tổng hợp số 3 thành Trung tâm GDNN-GDTX quận Đống Đa theo Quyết định số 5399/QĐ-UBND ngày 28 tháng 9 năm 2016. Trung tâm có 5 cơ sở dạy học và 50 cán bộ giáo viên, nhân viên.

3.2. Dữ liệu tuyển sinh của Trung tâm

Dữ liệu tuyển sinh của Trung tâm từ năm 1988 đến năm 2017 được thống kê trong bảng 3.2 như sau:

Bảng 3.2. Dữ liệu tuyển sinh của Trung tâm

TT	Năm	Số HS tuyển sinh	Các đặc trưng tuyển chọn	
			Số HS đỗ tốt nghiệp	Số hs đỗ CD, ĐH
1	1988	396	394	15
2	1989	263	262	18
3	1990	341	339	26
4	1991	205	204	31
5	1992	197	194	18
6	1993	329	325	37
7	1994	412	400	31
8	1995	139	137	13
9	1996	348	344	37
10	1997	95	93	10
11	1998	345	340	45
12	1999	356	351	38
13	2000	307	305	26
14	2001	248	243	51
15	2002	147	139	36
16	2003	310	307	58
17	2004	421	415	67
18	2005	342	335	38
19	2006	142	139	36
20	2007	354	349	73
21	2008	277	275	55
22	2009	465	460	83
23	2010	187	184	40
24	2011	387	385	63
25	2012	254	250	71
26	2013	187	185	54
27	2014	150	143	61
28	2015	108	101	57
29	2016	230	211	49
30	2017	250	243	47

Các đặc trưng có ảnh hưởng đến số lượng học sinh tuyển vào Trung tâm như:

- Số học sinh thi đỗ tốt nghiệp hàng năm của Trung tâm.
- Số học sinh thi đỗ cao đẳng, đại học của Trung tâm.
- Các yếu tố khác: Ảnh hưởng của quy chế thi cử, điều kiện kinh tế xã hội.

3.3. Các phương án chọn cấu trúc dữ liệu

Để lựa chọn dữ liệu đầu vào cho bài toán tuyển sinh có rất nhiều phương án như:

Phương án 1: Dựa vào các yếu tố đặc trưng ảnh hưởng đến đầu vào như: Số học sinh thi đỗ tốt nghiệp hàng năm; số học sinh thi đỗ cao đẳng, đại học.

Số lượng học sinh thi đỗ tốt nghiệp và cao đẳng, đại học thể hiện danh tiếng hay chất lượng đào tạo của một cơ sở giáo dục. Danh tiếng của cơ sở đó càng lớn thì số lượng học sinh đăng ký vào học sẽ càng cao.

Tuy nhiên với đặc thù của Trung tâm thì phương án này không được chọn vì số đặc trưng quá ít có quan hệ không rõ với mục đích nhập học của học sinh.

Ngoài ra, số các đặc trưng ảnh hưởng đến số liệu tuyển sinh quá ít.

Phương án 2: Chọn 1 đặc trưng đó là số học sinh tuyển vào trong 30 năm của Trung tâm.

Số lượng học sinh tuyển vào Trung tâm trong 30 năm có nhiều biến động, có ảnh hưởng đến số liệu trong những năm tiếp theo. Ngoài ra, do số liệu trải dài trong 30 năm nên số đặc trưng là tương đối đủ để dự báo.

Sử dụng 1 đặc trưng làm cấu trúc dữ liệu cho bài toán nên phương án giải bài toán sẽ là dự báo dạng hồi quy. Tức là sử dụng dữ liệu của những năm trước để dự báo cho những năm tiếp theo.

Ta có bảng dữ liệu sau:

Bảng 3.3. Dữ liệu tuyển sinh từ năm 1988 đến 2017 của Trung tâm

TT	Năm	Số HS tuyển sinh	TT	Năm	Số HS tuyển sinh
1	1988	396	16	2003	310
2	1989	263	17	2004	421
3	1990	341	18	2005	342
4	1991	205	19	2006	142
5	1992	197	20	2007	354
6	1993	329	21	2008	277
7	1994	412	22	2009	465
8	1995	139	23	2010	187
9	1996	348	24	2011	387
10	1997	95	25	2012	254
11	1998	345	26	2013	187
12	1999	356	27	2014	150
13	2000	307	28	2015	108
14	2001	248	29	2016	230
15	2002	147	30	2017	250

3.4. Phát biểu bài toán

Từ bảng dữ liệu 3.3 luận văn đã xây dựng bài toán như sau:

- Lấy 10 năm đầu từ 1988 đến 1997 để dự báo cho năm 1998.
- Sau đó lại lấy tiến lên 1 năm từ 1989 đến 1998 để dự báo cho năm 1999.
- Cứ tiếp tục như vậy cho dự báo đến năm 2017.

Bài toán sẽ gồm 2 pha như sau:

Pha 1: Pha học

Dữ liệu dùng để học sẽ lấy từ năm 1988 đến năm 2011 như vậy ta sẽ có 14 mẫu mỗi mẫu gồm 10 đầu vào là số học sinh của 10 năm. Mẫu 1 là số học sinh từ năm 1988 đến năm 1997 và đầu ra là số học sinh năm 1998. Mẫu 2 lấy tiến lên 1 năm là số học sinh từ năm 1989 đến năm 1998 và đầu ra là số học sinh năm 1999. Cứ như vậy ta sẽ có đến mẫu dữ liệu 14 là số học sinh từ năm 2001 đến năm 2011. Dữ liệu từ năm 2012 đến năm 2017 sẽ được sử dụng để làm dữ liệu kiểm tra độ chính xác của dự báo. Ta có bảng dữ liệu học như sau :

Bảng 3.4. Bảng dữ liệu học

Đầu vào	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	Đầu ra
k=1	396	263	341	205	197	329	412	139	348	95	345
k=2	263	341	205	197	329	412	139	348	95	345	356
k=3	341	205	197	329	412	139	348	95	345	356	307
k=4	205	197	329	412	139	348	95	345	356	307	248
k=5	197	329	412	139	348	95	345	356	307	248	147
k=6	329	412	139	348	95	345	356	307	248	147	310

k=7	412	139	348	95	345	356	307	248	147	310	421
k=8	139	348	95	345	356	307	248	147	310	421	342
k=9	348	95	345	356	307	248	147	310	421	342	142
k=10	95	345	356	307	248	147	310	421	342	142	354
k=11	345	356	307	248	147	310	421	342	142	354	277
k=12	356	307	248	147	310	421	342	142	354	277	465
k=13	307	248	147	310	421	342	142	354	277	465	187
k=14	248	147	310	421	342	142	354	277	465	187	387

Ta có thể mô tả bài toán như sau:

$$\begin{aligned} y_1 &= w_{11}x_1 + w_{12}x_2 + \dots + w_{110}x_{10} = x_{11} (1998) = d_1 \\ y_2 &= w_{21}x_1 + w_{22}x_2 + \dots + w_{210}x_{10} = x_{12} (1999) = d_2 \end{aligned} \quad (3-1)$$

....

$$y_{14} = w_{141}x_1 + w_{142}x_2 + \dots + w_{1410}x_{10} = x_{14} (2011) = d_{14}$$

Hay

$$Y = WX = d \quad (3-2)$$

$$Y = [y_1, y_2, \dots, y_{14}]^T; x = [x_1, x_2, \dots, x_{10}]^T; W = [w_{ij}]$$

Đây là pha học của mạng nơron. Tức là cần xác định w_{ij} .

Pha 2 : Pha chạy

Từ kết quả của pha 1 ta tìm được $W = [w_{ij}]$.

Cho tập dữ liệu năm [1988 ... 2017].

Cho tập dữ liệu số học sinh [396...250].

Bài toán xác định y là số học sinh năm 2018.

Nếu muốn dự báo số học sinh tuyển vào năm 2019 ta sẽ có 2 bước như sau :

Bước 1 : Lấy dữ liệu năm 2018 cho học lại.

Bước 2 : Lấy dữ liệu 10 năm từ 2010 đến 2018 để tìm y của năm 2019.

Cứ như vậy chúng ta có thể sử dụng dữ liệu của 10 năm trước để dự báo cho năm sau.

3.5. Thiết kế mạng nơron

Trong bài toán, ta xác định cặp đầu vào $(x^{(k)}, d^{(k)})$ trong đó $k=1 \dots 14$ là giá trị dữ liệu số lượng học sinh tuyển vào Trung tâm trong các năm 1988...2011.

3.5.1. Số lớp nơron

Luận văn lựa chọn mô hình mạng nơron với 3 lớp (1 lớp vào, 1 lớp ẩn, 1 lớp ra) như trong cấu trúc bài toán ở trên.

3.5.2 Cấu trúc mạng

Số nơron lớp vào

Trong luận văn lựa chọn 10 nơron lớp vào cho từng tập mẫu ứng vào giá trị đầu vào của mạng là số học sinh tuyển vào Trung tâm trong 10 năm liên tiếp.

Đầu ra của tập mẫu trước sẽ là 1 đầu vào cho tập mẫu sau.

Số nơon lớp ẩn

Để lựa chọn được chính xác số nơon cho lớp ẩn là rất khó vì vậy số nơon lớp ẩn sẽ được lựa chọn qua thực nghiệm của bài toán.

Qua thực nghiệm bài toán lựa chọn số nơon lớp ẩn là 10 nơon.

Số nơon lớp ra

Bài toán đặt ra là dự báo số lượng học sinh tuyển vào Trung tâm trong năm tiếp theo. Do vậy sẽ có 1 đầu ra là: Số lượng học sinh tuyển vào Trung tâm trong năm tới.

Hằng số học

Hằng số học η được lựa chọn bằng phương pháp thực nghiệm.

3.5.3. Hàm tương tác đầu ra

Hàm kích hoạt được sử dụng trong lớp vào và lớp ẩn là hàm Log-sigmoid, hàm này rất thuận lợi khi sử dụng cho các mạng huấn luyện bằng thuật toán lan truyền ngược BP, bởi nó rất dễ lấy đạo hàm.

$$y = f(Net_i) = \frac{1}{1 + e^{-Net_i}}$$

Hàm log-sigmoid này phù hợp với các bài toán có đầu ra mong muốn rơi vào khoảng [0,1].
















Hàm kích hoạt được sử dụng trong lớp ra là hàm tổng SUM vì giá trị đầu ra là giá trị thực tế đã chuẩn hóa theo đơn vị đầu ra.

3.5.4. Giá trị trọng khởi đầu

Giá trị trọng khởi đầu là giá trị khởi tạo ngẫu nhiên trong quá trình học của mạng.

Bảng thực nghiệm giá trị trọng khởi đầu của bài toán là:

Bảng 3.5. Bảng trọng số

	[0, 0]	0.52398867324180376
	[0, 1]	0.67137902680383021
	[0, 2]	0.46584385049801497
	[0, 3]	0.59960697200131
	[0, 4]	0.45414992955240885
	[0, 5]	0.43398760698455741
	[0, 6]	0.51030320092584158
	[0, 7]	0.93913115930703062
	[0, 8]	0.82730298807253266
	[0, 9]	0.56257637243837877
	[1, 0]	0.45133728787830907
	[1, 1]	0.97052624028666235
	[1, 2]	0.39243029728179346
	[1, 3]	0.63617314381346723
	[1, 4]	0.71603614870274257

Sau khi chạy chương trình huấn luyện mạng để dự báo số học mạng đã tạo ra được bộ trọng số. Bộ trọng số sẽ được thay đổi qua các vòng huấn luyện để sai số giữa giá trị dự báo và giá trị thực tế là nhỏ nhất.

3.6. Công cụ mô phỏng bài toán dự báo tuyển sinh

Để thử nghiệm mô hình dự báo ở chương 2 và thay đổi các thông số khác nhau đối với mạng nơron nhân tạo, luận văn đã xây dựng một công cụ mô phỏng mạng nơron truyền thẳng sử dụng thuật toán lan truyền ngược sai số. Cách sử dụng và giao diện của công cụ được trình bày trong phần phụ lục của luận văn.

Phần mềm được xây dựng bằng ngôn ngữ VB.NET trong bộ Microsoft Visual Studio.NET 2012 chạy trên nền Net FrameWork 4.0. Cấu hình yêu cầu: CPU tương đương Pentium 3, RAM 512MB, dung lượng ổ cứng còn trống trên 10MB.

3.7. Chạy và thử nghiệm

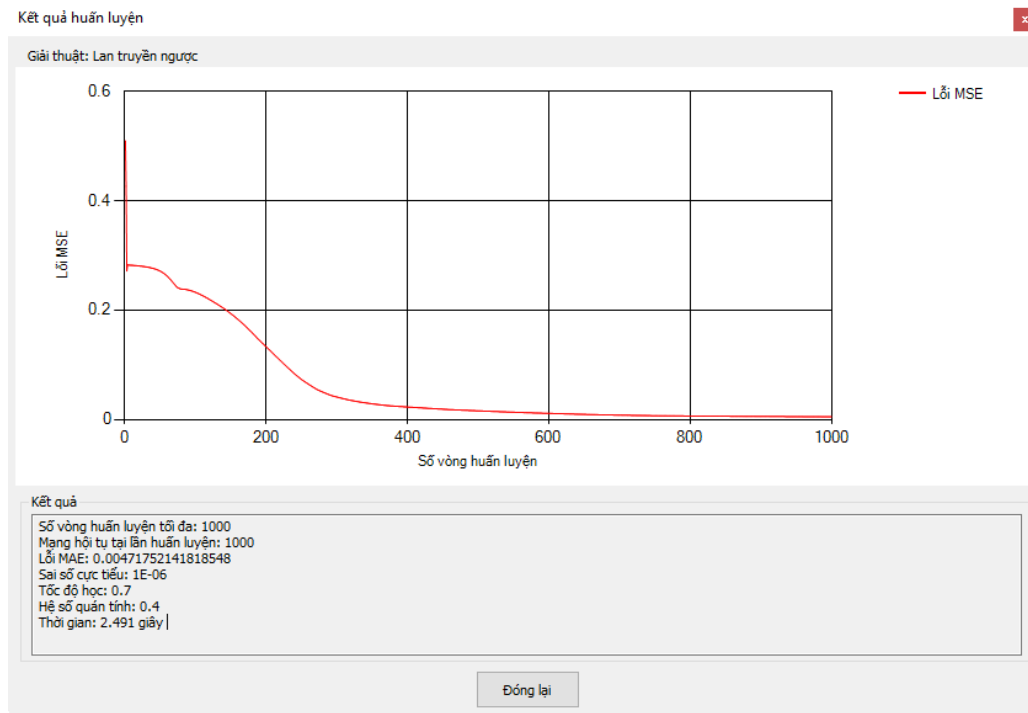
Chương trình này đã ứng dụng mạng nơron nhân tạo sử dụng giải thuật lan truyền ngược sai số vào việc dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa.

Qua các thử nghiệm cho kết quả dự báo như sau:

Bảng 3.6. Kết quả thử nghiệm dự báo tuyển sinh

Stt	Số nơron lớp ẩn	Tổng thời gian huấn luyện	Tổng số vòng huấn luyện	Lỗi MAE
1	6	2.752 giây	1000	0.0040497845057569
2	7	2.375 giây	1000	0.00485736880308239
3	8	2.218 giây	1000	0.0055487184283217
4	9	2.52 giây	1000	0.00761645708861354
5	10	2.686 giây	1000	0.00254915503276571
6	11	2.753 giây	1000	0.00466860652219338
7	12	4.084 giây	1000	0.00528905213344223
8	13	2.778 giây	1000	0.0045620026164574
9	14	2.738 giây	1000	0.00586705619117314
10	15	3.28 giây	1000	0.00529373821264322
11	16	2.606 giây	1000	0.00509736590845608
12	17	2.556 giây	1000	0.00497131752813026

Ban đầu luận văn lựa chọn số nơron trong lớp ẩn là 6 thì thời gian huấn luyện mạng là 2,752 giây và lỗi là 0,00404978 với số vòng huấn luyện là 1000 vòng. Sau đó số nơron của lớp ẩn được tăng dần lên để kiểm tra thời gian huấn luyện của mạng và lỗi MAE. Sau khi lựa chọn một số lớp ẩn thì ta nhận thấy với số nơron lớp ẩn là 10 thì lỗi MAE là nhỏ nhất nên luận văn chọn số nơron lớp là 10 nơron là phù hợp nhất với bài toán.



Hình 3.1. Kết quả huấn luyện mạng

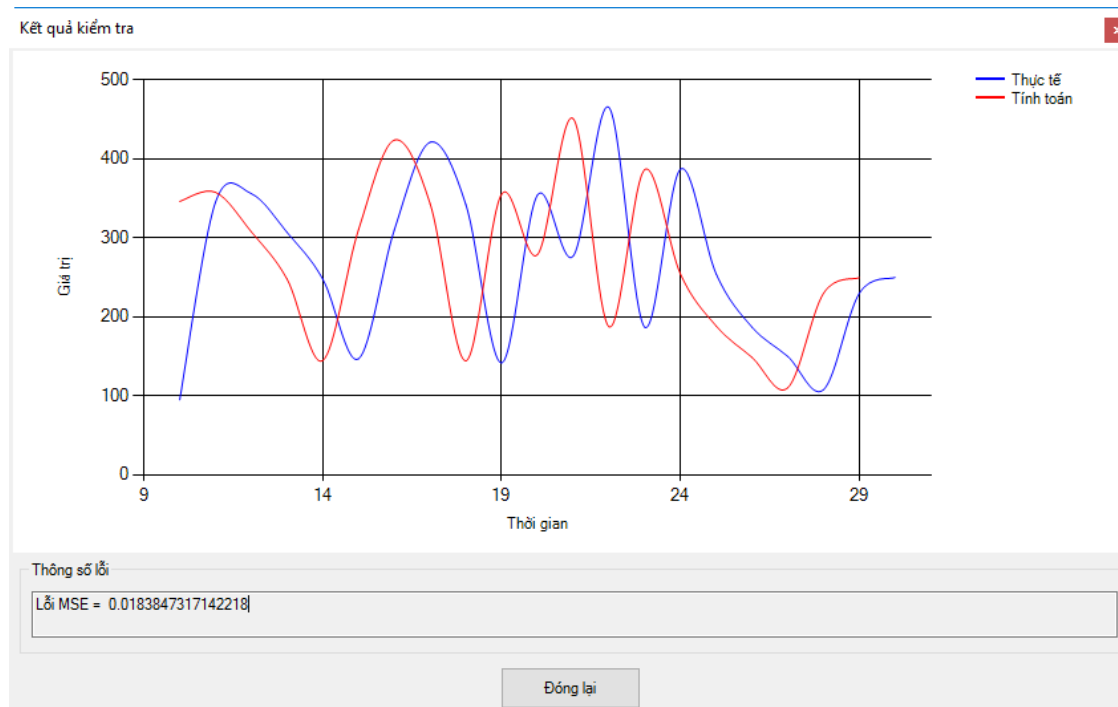
Trong hình 3.1 là biểu đồ hiển thị kết quả huấn luyện mạng sau 1000 vòng với số lỗi MSE và MAE tìm được trong quá trình huấn luyện. Các thông số của mạng được khởi tạo là: Mạng hội tụ tại vòng huấn luyện thứ 1000; tốc độ học là 0,7; hệ số quán tính là 0,4 và thời gian huấn luyện của mạng.

Bảng 3.7. Kết quả huấn luyện dự báo tuyến sinh

Mẫu	Giá trị thực tế	Giá trị dự báo	Tỷ lệ %	Lỗi MSE
0	345	346.04	99.7	0.0104
1	356	356.9	99.75	0.0045
2	307	307.49	99.84	0.0016
3	248	248.48	99.81	0.0012
4	147	144.2	98.1	0.0056
5	310	310.08	99.97	0.0001
6	421	424.55	99.16	0.0051
7	342	343.03	99.7	0.0013
8	142	142.17	99.88	0.0002
9	354	354.77	99.78	0.0008
10	277	277.84	99.7	0.0008
11	465	451.42	97.08	0.0113
12	187	187.15	99.92	0.0001
13	387	386.46	99.86	0.0004
14	254	253.81	99.93	0.0001
15	187	187.99	99.47	0.0006
16	150	149.24	99.5	0.0004

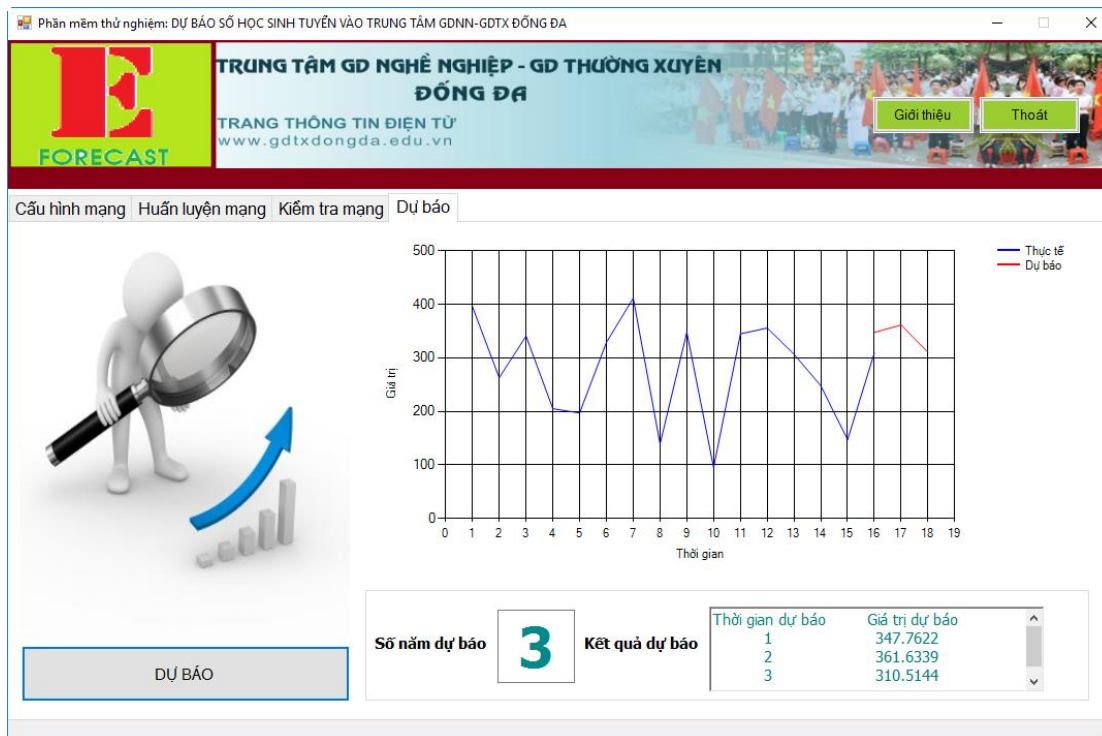
17	108	112.13	96.18	0.0023
18	230	229.69	99.86	0.0002
19	250	249.81	99.92	0.0001

Như vậy kết quả dự báo sử dụng công cụ là mạng nơron truyền thẳng với thuật toán lan truyền ngược sai số cho kết quả dự báo tương đối chính xác, sai số là chấp nhận được.



Hình 3.2. Kết quả kiểm tra mạng

Hình 3.2 mô tả kết quả kiểm tra mạng với bộ dữ liệu kiểm tra từ năm 2011 đến 2017. Đường màu xanh mô tả giá trị thực tế thu thập được và đường màu đỏ là đường thể hiện giá trị dự báo, ngoài ra giao diện còn thể hiện lỗi MSE của dự báo. Nhìn vào biểu đồ kiểm tra kết quả của mạng ta nhận thấy kết quả dự báo của mạng là tương đối chính xác, tức là sai số của dự báo là nhỏ.



Hình 3.3. Kết quả dự báo cho 3 năm 2018, 2019, 2020

Trong bảng là kết quả dự báo của ba năm liên tiếp là 2018, 2019 và 2020. Số dự báo là đường màu đỏ. Kết quả dự báo của năm 2019 sẽ có 1 đầu vào là kết quả dự báo của năm 2018 và năm 2020 sẽ lấy kết quả dự báo của năm 2019 làm 1 đầu vào.

3.8. So sánh các phương pháp dự báo.

Hiện nay có rất nhiều các phương pháp khác nhau được sử dụng để dự báo như : Phương pháp dự báo trung bình dài hạn, phương pháp dự báo trung bình động, phương pháp hồi quy tuyến tính, phương pháp san bằng số mũ...

Để đánh giá độ chính xác và tính ưu việt của mạng nơron trong dự báo luận văn có sử dụng các phương pháp này để so sánh kết quả dự báo.

Phương pháp dự báo trung bình dài hạn : là phương pháp mà số dự báo bằng trung bình cộng của các quan sát thực tế trước đó. Ta có công thức :

$$F_{t+1} = \frac{\sum_{i=0}^{n-1} D_{t-1}}{n} \quad (3-3)$$

Trong đó :

F_{t+1} : Số dự báo ở kỳ thứ t+1

D_t : Số quan sát ở kỳ thứ t

n : Tổng quan sát

Qua tính toán bằng phương pháp cho giá trị sai số như sau :

Bảng 3.8. Kết quả dự báo bằng phương pháp trung bình dài hạn

STT	Giá trị thực tế	Giá trị dự báo	Tỷ lệ %
1	345	273	78.99
2	356	279	78.40
3	307	286	93.00
4	248	287	86.36
5	147	284	51.70
6	310	275	88.77
7	421	277	65.88
8	342	286	83.57
9	142	289	49.14
10	354	281	79.44
11	277	285	97.24
12	465	284	61.18
13	187	293	63.89
14	387	288	74.44
15	254	292	86.92
16	187	291	64.33
17	150	287	52.32
18	108	282	38.35
19	230	275	83.51
20	250	274	91.29

Phương pháp trung bình động : Số dự báo ở kỳ thứ t+1 bằng trung bình cộng của n kỳ trước đó. Cứ mỗi kỳ dự báo sẽ bỏ đi số liệu xa nhất trong quá khứ và thêm vào số liệu mới nhất. Ta có công thức :

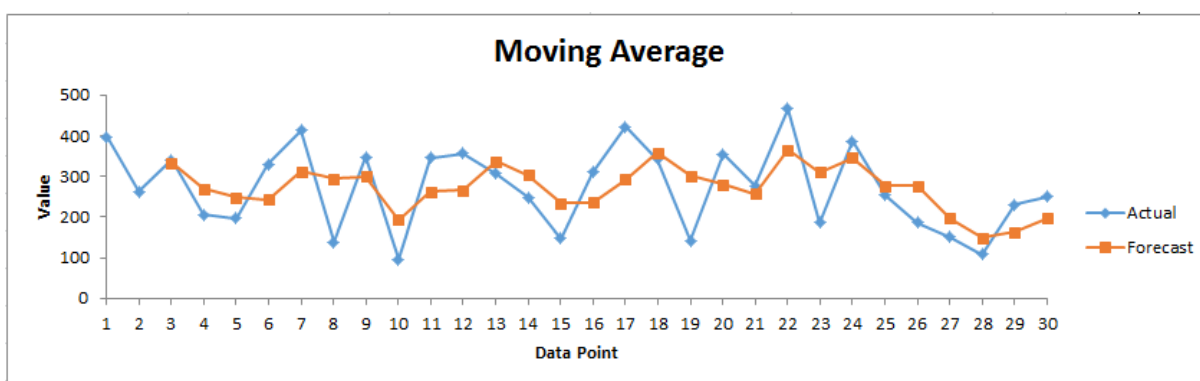
$$F_{t+1} = \frac{D_t + D_{t-1} + \dots + D_{t-n}}{n+1} \quad (3-4)$$

Bảng 3.9. Kết quả dự báo bằng phương pháp trung bình động

STT	Giá trị thực tế	Giá trị dự báo	Tỷ lệ %
1	345	194	56.23
2	356	263	73.78
3	307	265	86.43
4	248	336	73.81
5	147	304	48.41
6	310	234	75.48
7	421	235	55.82
8	342	293	85.58

9	142	358	39.70
10	354	302	85.22
11	277	279	99.16
12	465	258	55.41
13	187	365	51.19
14	387	310	80.02
15	254	346	73.34
16	187	276	67.75
17	150	276	54.35
18	108	197	54.82
19	230	148	64.49
20	250	163	65.07

Phương pháp này có sử dụng công cụ dự báo Moving Average và cho ra sơ đồ dự báo như sau :



Hình 3.4. Sơ đồ kết quả dự báo trung bình động

Phương pháp hồi quy tương quan : Phương pháp này sử dụng mối phụ thuộc của một biến (biến phụ thuộc) với một hay nhiều biến khác (biến độc lập) để ước lượng và dự báo giá trị trung bình của biến phụ thuộc với các giá trị đã biết của biến độc lập.

Ta có kết quả dự báo như sau :

Bảng 3.10. Kết quả dự báo bằng phương pháp hồi quy tương quan

STT	Giá trị thực tế	Giá trị dự báo	Tỷ lệ %
1	345	345	99.91
2	356	356	99.92
3	307	310	99.13
4	248	248	99.86
5	147	144	97.69
6	310	312	99.46
7	421	420	99.70
8	342	340	99.33
9	142	144	98.88

10	354	354	99.92
11	277	280	99.05
12	465	465	99.95
13	187	189	99.14
14	387	390	99.30
15	254	255	99.74
16	187	190	98.61
17	150	148	98.40
18	108	106	97.76
19	230	216	93.76
20	250	248	99.06

Phương pháp dự báo san bằng mũ đơn giản : Phương pháp này sẽ dự báo giá trị mới bằng giá trị cũ cộng với khoảng chênh lệch giữa nhu cầu dự báo thực và nhu cầu dự báo của kỳ trước, có điều chỉnh hệ số.

Ta có công thức sau :

$$F_t = F_{t-1} + \alpha (D_{t-1} - F_{t-1}) \text{ với } 0 < \alpha < 1 \quad (3-5)$$

Trong đó :

F_t : Nhu cầu dự báo kỳ t

F_{t-1} : Nhu cầu dự báo kỳ t-1

D_{t-1} : Nhu cầu thực kỳ t-1

α : Hệ số san bằng mũ

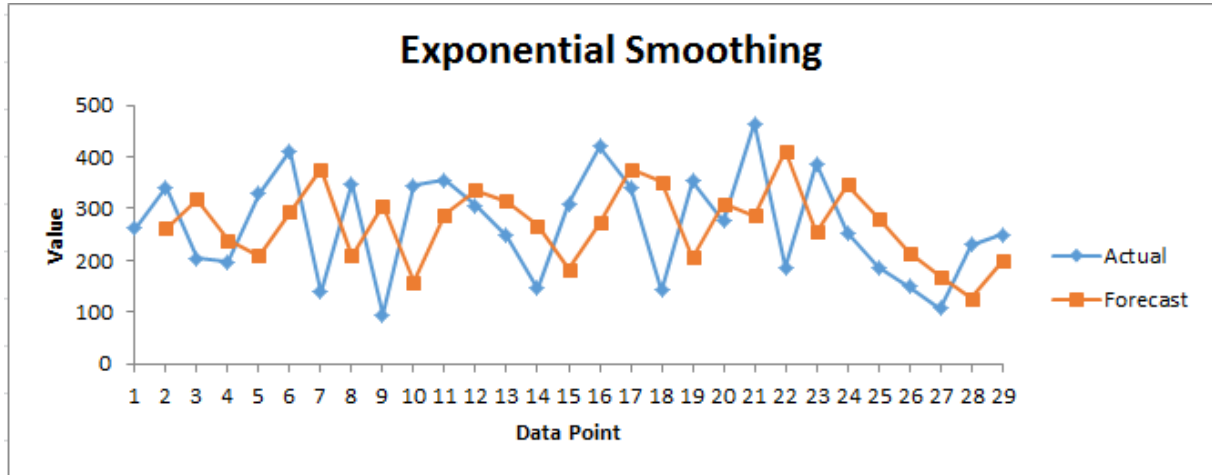
Ta có bảng kết quả dự báo như sau :

Bảng 3.11. Kết quả dự báo bằng phương pháp san bằng mũ

STT	Giá trị thực tế	Giá trị dự báo	Tỷ lệ %
1	345	289	83.78
2	356	336	94.36
3	307	316	97.25
4	248	268	92.43
5	147	183	80.16
6	310	272	87.75
7	421	376	89.38
8	342	352	97.08
9	142	205	69.24
10	354	309	87.38
11	277	287	96.62
12	465	412	88.50

13	187	254	73.52
14	387	347	89.72
15	254	282	90.08
16	187	215	86.78
17	150	170	88.42
18	108	126	85.38
19	230	199	86.50
20	250	165	65.87

Sơ đồ biểu diễn độ sai số của dự báo :



Hình 3.5. Kết quả dự báo san bằng mũ

Qua các kết quả của các phương pháp dự báo thì kết quả dự báo bằng công cụ mạng nơron cho kết quả chính xác và đáng tin cậy nhất. Vì vậy, luận văn lựa chọn dự báo tuyến sinh bằng mạng nơron truyền thẳng với thuật toán lan truyền ngược sai số là hoàn toàn có cơ sở.

3.9 Kết luận chương 3

Qua các kết quả thực nghiệm thu được trong quá trình thử nghiệm trên công cụ dự báo trong luận văn đã chỉ ra được những điều sau:

- Mạng nơron nhân tạo có tính ứng dụng rất đa dạng, khá hiệu quả trong các bài toán dự báo.
- Mức độ chính xác của mô hình phụ thuộc vào rất nhiều thông số, tuy nhiên lại chưa có một phương pháp nào để xác định được chính xác định tính cũng như định lượng của các thông số. Ta phải thông qua phương pháp thực nghiệm để xác định giá trị thông số tối ưu.
- Số liệu của mỗi đơn vị tuyển sinh có đặc trưng riêng và thông số tối ưu thay đổi theo mức độ đặc trưng của số liệu.

KẾT LUẬN

Luận văn nghiên cứu mạng nơron nhân tạo, mô hình mạng nơron truyền thẳng được huấn luyện với giải thuật lan truyền ngược cho bài toán dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa.

Các đóng góp của luận văn:

1. Luận văn đã nghiên cứu tổng quan về mạng nơron nhân tạo, đi sâu vào nghiên cứu mạng nơron lan truyền thẳng huấn luyện bằng thuật toán lan truyền ngược sai số nhằm đạt tới một kết quả tốt nhất cho bài toán tối ưu trọng số mạng nơron nhân tạo.
2. Luận văn cũng đã xây dựng được phần mềm dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa trong năm tiếp theo.

Hướng phát triển tiếp theo

Những kết quả thực nghiệm khả quan dựa trên nghiên cứu về ứng dụng mạng nơron nhân tạo với thuật toán học là lan truyền ngược sai số trong bài toán dự báo số học sinh tuyển vào Trung tâm GDNN-GDTX quận Đống Đa trong năm tới cho thấy đây là một mô hình hiệu quả.

Vì vậy, hướng phát triển tiếp theo của luận văn là cải tiến phương pháp dự báo để có kết quả chính xác nhất.

Ngoài ra, có thể sử dụng kết hợp với logic mờ và giải thuật di truyền (GA) để cho kết quả dự báo chính xác hơn.

Tiếng Việt

1. Thái Trung Hải (2014), *Sử dụng mạng nơron trong việc dự đoán kết quả đậu đại học*, Luận văn Thạc sỹ, Đại học Lạc Hồng.
2. Nguyễn Quang Hoan, Hoàng Thị Lan Phương (2006), “Dự báo giá chứng khoán sử dụng công nghệ mạng nơron”, *Kỷ yếu hội nghị khoa học lần 3 về nghiên cứu, phát triển và ứng dụng công nghệ thông tin và truyền thông Hà Nội ICT.rda'06 (20-21/5/2006) tr 157 -164*
3. Nguyễn Quang Hoan (2005), *Giáo trình mạng nơron nhân tạo*, Học viện công nghệ bưu chính viễn thông.
4. Hoàng Phúc Lâm, Nguyễn Hương Điền, Công Thanh, Hoàng Thanh Vân (2007), “Sử dụng mạng nơron đa lớp truyền thẳng và mạng truy hồi dự báo tổng lượng bức xạ ngày cho một số trạm ở đồng bằng phía Bắc Việt Nam”, *Tạp chí Khí tượng Thủy văn số 10 (559)*.
5. Trần Đức Minh (2002), *Ứng dụng mạng nơron truyền thẳng trong dự báo dữ liệu*, Luận văn thạc sỹ, Viện Công nghệ Thông tin.
6. Nguyễn Đình Thúc (2000), *Mạng nơron nhân tạo – phương pháp và ứng dụng*, Nhà xuất bản Giáo dục.
7. Nguyễn Văn Trịnh (2012), *Ứng dụng mạng nơron trong dự báo tỷ lệ nghỉ học*, Luận văn thạc sỹ, Viện Toán Ứng dụng và Tin học – Đại học Bách khoa Hà Nội.
8. Phạm Thị Hoàng Nhung (2007), *Nghiên cứu, ứng dụng các phương pháp học máy tiên tiến trong công tác dự báo, vận hành hồ Hòa Bình*, Luận văn thạc sỹ, trường Đại học Công nghệ (ĐHQGHN).

Tiếng Anh

9. John C. Chambers, Satinder K. Mullick and Donald D. Smith (1971), *How to Choose the Right Forecasting Technique*, <https://hbr.org/1971/07/how-to-choose-the-right-forecasting-technique>.
10. Petar Halachev (2012), “Prediction of e-Learning Efficiency by Neural Networks”, *Cybernetics And Information Technologies – volume 12 – Nơron 12*, Bulgarian Academy of Sciences.
11. Hopfield, J.J (1982), “Neural Networks and Physical Systems with Emergent Collective Computational Abilities”, *Proceeding of Natural Academic Sciences, USA, vol. 79, pp. 2.554 – 2.558*.
12. Robert J. Schalkoff (1997), *Artificial Neural Networks*, The McGraw – Hill Companies, Inc 1997.
13. David Silverman, Jonh A. Dracup (2000), “Artificial Neural Networks and Long-Range Precipitation Prediction in California”, *Journal of Applied Meteorology, vol 39 (Jan 2000), pp. 57-66*.

14. Dipti Srinivasan, A.C. Liew, Jonh S., P. Chen (1991), “Short Term Forecasting Using Neural Networks Approach”, *IEEE 91TH0374-9/91/0000-0012*, pp 12-16, 1991.
15. Morioka Y., Sakurai K., Yokoyama A. Sekine Y. (1993), “Next Day Peak Load Forecasting Using a Multilayer Neural Network with an Additional Learning”, *IEEE, 0-7803-1217-1/93*, 1993.
16. Michael Negnevitsky (2005), *Artificial Intelligence*, Addison Wesley.
17. Fico Corporation (2009), *Understanding Predictive Analytics*.
18. Dean Abbott (2014), *Applied Predictive Analytics: Principles and Techniques for the Professional Data Analyst*, Wiley.

PHỤ LỤC A

GIỚI THIỆU CÔNG CỤ VÀ KẾT QUẢ DỰ BÁO

1. Giao diện chính của phần mềm

Chức năng Cấu hình mạng:

- Nút lệnh Nhập từ tệp: Cho phép người dùng nạp dữ liệu từ tệp dữ liệu.
- Nút lệnh Tạo thông số mạng: Để tạo mới một mạng nơron lan truyền ngược. Thông số mạng được tạo bao gồm: Số nơron lớp vào, số nơron lớp ẩn, số nơron lớp ra, tốc độ học, hệ số quán tính, số lần học tối đa, sai số cực tiểu.
- Nút lệnh Sửa: Để sửa các thông số của mạng.

2. Giao diện huấn luyện

Phần mềm thử nghiệm: DỰ BÁO SỐ HỌC SINH TUYỂN VÀO TRUNG TÂM GDNN-GDTX ĐỒNG ĐÀ

TRUNG TÂM GD NGHỀ NGHIỆP - GD THƯỜNG XUYÊN ĐỒNG ĐÀ
TRANG THÔNG TIN ĐIỆN TỬ
www.gdtxdongda.edu.vn

Giới thiệu Thoát

Cấu hình mạng Huấn luyện mạng Kiểm tra mạng Dự báo

Tập dữ liệu huấn luyện

STT	Input 1	Input 2	Input 3	Input 4	Input 5	Input 6	Input 7	Input 8	Input 9	Input 10	Output
<input type="checkbox"/> 0	396	263	341	205	197	329	412	139	348	95	345
<input checked="" type="checkbox"/> 1	263	341	205	197	329	412	139	348	95	345	356
<input checked="" type="checkbox"/> 2	341	205	197	329	412	139	348	95	345	356	307
<input checked="" type="checkbox"/> 3	205	197	329	412	139	348	95	345	356	307	248
<input checked="" type="checkbox"/> 4	197	329	412	139	348	95	345	356	307	248	147
<input checked="" type="checkbox"/> 5	329	412	139	348	95	345	356	307	248	147	310
<input checked="" type="checkbox"/> 6	412	139	348	95	345	356	307	248	147	310	421
<input checked="" type="checkbox"/> 7	139	348	95	345	356	307	248	147	310	421	342

Dữ liệu đã chuẩn hóa

STT	Input 1	Input 2	Input 3	Input 4	In5	In6	In7	In8	In9	In10	Output
<input checked="" type="checkbox"/> 0	0.9148	0.515	0.7495	0.3407	0.3166	0.7134	0.9629	0.1423	0.7706	0.01	0.7615
<input checked="" type="checkbox"/> 1	0.515	0.7495	0.3407	0.3166	0.7134	0.9629	0.1423	0.7706	0.01	0.7615	0.7946
<input checked="" type="checkbox"/> 2	0.7495	0.3407	0.3166	0.7134	0.9629	0.1423	0.7706	0.01	0.7615	0.7946	0.6473
<input checked="" type="checkbox"/> 3	0.3407	0.3166	0.7134	0.9629	0.1423	0.7706	0.01	0.7615	0.7946	0.6473	0.4699
<input checked="" type="checkbox"/> 4	0.3166	0.7134	0.9629	0.1423	0.7706	0.01	0.7615	0.7946	0.6473	0.4699	0.1663
<input checked="" type="checkbox"/> 5	0.7134	0.9629	0.1423	0.7706	0.01	0.7615	0.7946	0.6473	0.4699	0.1663	0.6563
<input checked="" type="checkbox"/> 6	0.9629	0.1423	0.7706	0.01	0.7615	0.7946	0.6473	0.4699	0.1663	0.6563	0.99
<input checked="" type="checkbox"/> 7	0.1423	0.7706	0.01	0.7615	0.7946	0.6473	0.4699	0.1663	0.6563	0.99	0.7525

Huấn luyện Nhập dữ liệu Chuẩn hóa Xóa dữ liệu

Liên kết Liên kết No-ron

Trước khi huấn luyện phải Nhập dữ liệu cho mạng.

- Nút lệnh Huấn luyện: Dùng để huấn luyện mạng, màn hình hiển thị thông số huấn luyện mạng gồm Tập dữ liệu huấn luyện và dữ liệu đã chuẩn hóa.
- Nút lệnh Chuẩn hóa: Dùng để chuẩn hóa dữ liệu huấn luyện.
- Nút lệnh Xóa dữ liệu: Để xóa các dữ liệu.

3. Giao diện Kiểm tra mạng

Phần mềm thử nghiệm: DỰ BÁO SỐ HỌC SINH TUYỂN VÀO TRUNG TÂM GDNN-GDTC ĐỒNG ĐÀ

TRUNG TÂM GD NGHỀ NGHIỆP - GD THƯỜNG XUYÊN ĐỒNG ĐÀ
TRANG THÔNG TIN ĐIỆN TỬ
www.gdtxdongda.edu.vn

Giới thiệu Thoát

Cấu hình mạng Huấn luyện mạng Kiểm tra mạng Dự báo

Tập dữ liệu kiểm tra

STT	Input 1	Input 2	Input 3	Input 4	Input 5
<input checked="" type="checkbox"/> 0	396	263	341	205	197
<input checked="" type="checkbox"/> 1	263	341	205	197	329
<input checked="" type="checkbox"/> 2	341	205	197	329	412
<input checked="" type="checkbox"/> 3	205	197	329	412	139
<input checked="" type="checkbox"/> 4	197	329	412	139	348
<input checked="" type="checkbox"/> 5	329	412	139	348	95
<input checked="" type="checkbox"/> 6	412	139	348	95	345
<input checked="" type="checkbox"/> 7	139	348	95	345	356

Kết quả kiểm tra mạng Ghi vào tệp

STT	Thực tế	Dự báo	Tỷ lệ %	MAE
<input checked="" type="checkbox"/> 0	345	346.02	99.71	0.0102
<input checked="" type="checkbox"/> 1	356	357.49	99.58	0.0074
<input checked="" type="checkbox"/> 2	307	308.74	99.43	0.0058
<input checked="" type="checkbox"/> 3	248	248.12	99.95	0.0003
<input checked="" type="checkbox"/> 4	147	142.8	97.14	0.0084
<input checked="" type="checkbox"/> 5	310	309.04	99.69	0.0016
<input checked="" type="checkbox"/> 6	421	423.5	99.41	0.0036
<input checked="" type="checkbox"/> 7	342	341.51	99.86	0.0006
<input checked="" type="checkbox"/> 8	142	143.3	99.09	0.0014
<input checked="" type="checkbox"/> 9	354	353.88	99.97	0.0001
<input checked="" type="checkbox"/> 10	277	276.19	99.71	0.0007
<input checked="" type="checkbox"/> 11	465	450.41	96.86	0.0122
<input checked="" type="checkbox"/> 12	187	185.27	99.07	0.0013
<input checked="" type="checkbox"/> 13	387	384.56	99.37	0.0017
<input checked="" type="checkbox"/> 14	254	252.26	99.31	0.0012
<input checked="" type="checkbox"/> 15	187	187.53	99.71	0.0003
<input checked="" type="checkbox"/> 16	150	149.63	99.75	0.0002
<input checked="" type="checkbox"/> 17	108	106.21	98.35	0.001
<input checked="" type="checkbox"/> 18	230	230.24	99.9	0.0001
<input checked="" type="checkbox"/> 19	250	249.87	99.95	0.0001

Dữ liệu đã chuẩn hóa:

STT	Input 1	Input 2	Input 3	Input 4	Input 5
<input checked="" type="checkbox"/> 0	0.8072	0.455	0.6616	0.3014	0.2802
<input checked="" type="checkbox"/> 1	0.455	0.6616	0.3014	0.2802	0.6298
<input checked="" type="checkbox"/> 2	0.6616	0.3014	0.2802	0.6298	0.8496
<input checked="" type="checkbox"/> 3	0.3014	0.2802	0.6298	0.8496	0.1265
<input checked="" type="checkbox"/> 4	0.2802	0.6298	0.8496	0.1265	0.6801
<input checked="" type="checkbox"/> 5	0.6298	0.8496	0.1265	0.6801	0.01
<input checked="" type="checkbox"/> 6	0.8496	0.1265	0.6801	0.01	0.6722
<input checked="" type="checkbox"/> 7	0.1265	0.6801	0.01	0.6722	0.3014

Kiểm tra mạng Nhập dữ liệu Chuẩn hóa Xóa dữ liệu

Trước khi kiểm tra mạng phải Nhập dữ liệu cho mạng.

Nút lệnh Kiểm tra mạng: Để kiểm tra độ chính xác kết quả huấn luyện của mạng.

Giao diện hiển thị tập dữ liệu được kiểm tra và dữ liệu đã chuẩn hóa của mạng.

4. Chức năng Dự báo

Phần mềm thử nghiệm: DỰ BÁO SỐ HỌC SINH TUYỂN VÀO TRUNG TÂM GDNN-GDTC ĐỒNG ĐÀ

TRUNG TÂM GD NGHỀ NGHIỆP - GD THƯỜNG XUYÊN ĐỒNG ĐÀ
TRANG THÔNG TIN ĐIỆN TỬ
www.gdtxdongda.edu.vn

Giới thiệu Thoát

Cấu hình mạng Huấn luyện mạng Kiểm tra mạng Dự báo

DỰ BÁO

Số năm dự báo **3** Kết quả dự báo

Thời gian dự báo	Giá trị dự báo
1	346.7786
2	360.7604
3	309.5254

Thời gian Giá trị

— Thực tế
— Dự báo

Trước khi nhấn nút lệnh Dự báo phải chọn số năm định dự báo.

5. Chức năng Giới thiệu vào Thoát

Giới thiệu

ĐẠI HỌC CÔNG NGHỆ - ĐHQG HN	PHẦN MỀM THỬ NGHIỆM DỰ BÁO
	Luận văn tốt nghiệp Thạc sỹ CNTT

ĐỀ TÀI

**ỨNG DỤNG MẠNG NƠON TRONG DỰ BÁO
SỐ HỌC SINH TUYỂN VÀO TRUNG TÂM GDNN-GD TX
ĐỒNG ĐA**

Tác giả:	Dương Thu Trang
Giáo viên hướng dẫn:	PGS. TS. Nguyễn Quang Hoan

Tháng 9 - 2017

Chức năng Giới thiệu: Hiện các thông tin tên đề tài, tác giả, giáo viên hướng dẫn luận văn.

Chức năng Thoát: Dùng để thoát chương trình

PHỤ LỤC B

MỘT SỐ MÃ NGUỒN CỦA CÔNG CỤ

1. Huấn luyện mạng

```

public void Bp_Run(List<double> sampleSeries, double learnRate, double momentum, int
theEpoches = 10000, double residual = 0.00001)
{
    InitForTrain();

    int i, j, k, n;
    int epoch = 0;
    double MAE = Double.MaxValue;
    double LastError = Double.MaxValue;
    List<double> MAError = new List<double>();

    double[,] deltaInputHidden = new double[s_Network.m_iNumInputNodes + 1,
s_Network.m_iNumHiddenNodes];
    double[,] deltaHiddenOutput = new double[s_Network.m_iNumHiddenNodes + 1,
s_Network.m_iNumOutputNodes];
    double[,] lagDeltaInputHidden = new double[s_Network.m_iNumInputNodes + 1,
s_Network.m_iNumHiddenNodes];
    double[,] lagDeltaHiddenOutput = new double[s_Network.m_iNumHiddenNodes +
1, s_Network.m_iNumOutputNodes];

    for (j = 0; j < s_Network.m_iNumHiddenNodes; j++) // initialize weight-
step of Input Hidden connection
    {
        for (i = 0; i <= s_Network.m_iNumInputNodes; i++)
        {
            deltaInputHidden[i, j] = 0.0;
            lagDeltaInputHidden[i, j] = 0.0;
        }
    }
    for (k = 0; k < s_Network.m_iNumOutputNodes; k++) // initialize weight-
step of Hidden Output connection
    {
        for (j = 0; j <= s_Network.m_iNumHiddenNodes; j++)
        {
            deltaHiddenOutput[j, k] = 0.0;
            lagDeltaHiddenOutput[j, k] = 0.0;
        }
    }

    TrainingProgress frmTrainingProgress = new TrainingProgress();
    frmTrainingProgress.SetInit(sampleSeries.Count(), theEpoches , 1);
    System.Windows.Forms.Application.DoEvents();
    frmTrainingProgress.Show();
    bool isUserStop = false;
    RunTime BeginRunTime = new RunTime();
    BeginRunTime.GetCurrTime();

    while (epoch < theEpoches)
    {
        MAE = 0.0;

```

```

for (n = s_Network.m_iNumInputNodes; n < sampleSeries.Count; n++)
{
    // forward
    double[] lstTemp = new double[s_Network.m_iNumInputNodes];
    for (i = s_Network.m_iNumInputNodes; i > 0; i--)
    {
        lstTemp[s_Network.m_iNumInputNodes - i] = sampleSeries[n - i];
    }
    s_Network.CalculateOutput(lstTemp);
    for (k = 0; k < s_Network.m_iNumOutputNodes; k++)
    {
        MAE += Math.Abs(sampleSeries.ElementAt(n + k) -
s_Network.m_arOutputNodes[k].GetOutput());
    }

    // backward
    /*calculate weight-step for weights connecting from hidden nodes
to output nodes*/
    for (k = 0; k < s_Network.m_iNumOutputNodes; k++)
    {
        for (j = 0; j <= s_Network.m_iNumHiddenNodes; j++)
        {
            double parDerv = -s_Network.m_arOutputNodes[k].GetOutput()
* (1 - s_Network.m_arOutputNodes[k].GetOutput()) *
s_Network.m_arHiddenNodes[j].GetOutput() * (sampleSeries.ElementAt(n) -
s_Network.m_arOutputNodes[k].GetOutput());
            deltaHiddenOutput[j, k] = -learnRate * parDerv + momentum
* lagDeltaHiddenOutput[j, k];
            lagDeltaHiddenOutput[j, k] = deltaHiddenOutput[j, k];
        }
    }
    /*calculate weight-step for weights connecting from input nodes to
hidden nodes*/
    for (j = 0; j < s_Network.m_iNumHiddenNodes; j++)
    {
        double temp = 0.0;
        for (k = 0; k < s_Network.m_iNumOutputNodes; k++)
        {
            temp += -(sampleSeries.ElementAt(n) -
s_Network.m_arOutputNodes[k].GetOutput()) * s_Network.m_arOutputNodes[k].GetOutput() *
(1 - s_Network.m_arOutputNodes[k].GetOutput()) * s_Network.m_arHiddenOutputConn[j, k];
        }
        for (i = 0; i <= s_Network.m_iNumInputNodes; i++)
        {
            double parDerv = s_Network.m_arHiddenNodes[j].GetOutput()
* (1 - s_Network.m_arHiddenNodes[j].GetOutput()) *
s_Network.m_arInputNodes[i].GetInput() * temp;
            deltaInputHidden[i, j] = -learnRate * parDerv + momentum *
lagDeltaInputHidden[i, j];
            lagDeltaInputHidden[i, j] = deltaInputHidden[i, j];
        }
    }
    /*updating weight from Input to Hidden*/
    for (j = 0; j < s_Network.m_iNumHiddenNodes; j++)

```

```

        {
            for (i = 0; i <= s_Network.m_iNumInputNodes; i++)
            {
                s_Network.m_arInputHiddenConn[i, j] += deltaInputHidden[i,
j]];
            }
        }
        /*updating weight from Hidden to Output*/
        for (k = 0; k < s_Network.m_iNumOutputNodes; k++)
        {
            for (j = 0; j <= s_Network.m_iNumHiddenNodes; j++)
            {
                s_Network.m_arHiddenOutputConn[j, k] +=
deltaHiddenOutput[j, k];
            }
        }

        } // end outer for
        MAE = MAE / (sampleSeries.Count - s_Network.m_iNumInputNodes); //
caculate mean square error
        if (Math.Abs(MAE - LastError) < residual) // if the Error is not
improved significantly, halt training process and rollback
        {
            RollBack();
            break;
        }
        else
        { //else backup the current configuration and continue training
            LastError = MAE;
            BackUp();
            MAError.Add(MAE);
            epoch++;
        }
        frmTraningProgress.SetStep(epoch);
        System.Windows.Forms.Application.DoEvents();
        if (frmTraningProgress.isStop == true)
        {
            isUserStop = true;
            break;
        }
        //Thread.Sleep(10);
    }
    frmTraningProgress.OnClose();
    RunTime EndRunTime = new RunTime();
    EndRunTime.GetCurrTime();
    RunTime RunTimeUse = new RunTime();
    RunTimeUse = BeginRunTime.Subtraction(EndRunTime);
    /* output training result */

    TrainingResult result = new TrainingResult();
    result.trainResult.AppendText("Số vòng huấn luyện tối đa: " + theEpoches +
"\n");
    result.trainResult.AppendText("Mạng hội tụ tại lần huấn luyện: " + epoch +

```

```

"\n");
    result.trainResult.AppendText("Lỗi MAE: " + MAE + "\n");
    result.trainResult.AppendText("Sai số cực tiểu: " + residual + "\n");
    result.trainResult.AppendText("Tốc độ học: " + learnRate + "\n");
    result.trainResult.AppendText("Hệ số quán tính: " + momentum + "\n");
    result.trainResult.ReadOnly = true;
    //result.chart1.Series["MAE"].Color = System.Drawing.Color.Red;
    result.chart1.Series["Lỗi MSE"].Color = System.Drawing.Color.Red;
    for (int t = 0; t < MAError.Count; t++)
    {
        //result.chart1.Series["MAE"].Points.AddXY(t + 1,
MAError.ElementAt(t));
        result.chart1.Series["Lỗi MSE"].Points.AddXY(t + 1,
MAError.ElementAt(t));
    }
    result.trainResult.AppendText("Thời gian: " +
RunTimeUse.ConvertTimeString() + "\n");
    result.label1.Text = "Giải thuật: Lan truyền ngược";
    result.ShowDialog();
}

```

2. Chuẩn hóa dữ liệu

```

private void btChuanHoa_Click(object sender, EventArgs e)
{
    double max = ListInput.Max();
    double min = ListInput.Min();
    int count = ListInput.Count;
    int k = 0;
    int j = 0;
    ListViewItem ltmp = default(ListViewItem);
    ListChuanHoa.Clear();

    for (int i = 0; i < ListInput.Count; i++)
    {
        double a = ListInput.ElementAt(i);
        double b = (a - min) / (max - min) * (0.99 - 0.01) + 0.01;
        ListChuanHoa.Add(Math.Round(b,4));
    }

    //Load to ListView
    for(int i=0;i< ListChuanHoa.Count; i++)
    {
        ltmp = ListTrainingData1.Items.Add(k.ToString());
        for (j = 0; j < numInput + numOutput; j++)
        {
            ltmp.SubItems.Add(ListChuanHoa.ElementAt(j + i).ToString ());
        }
        ltmp.Checked = true;
        if (i + j >= ListChuanHoa.Count)
            break;
        k = k + 1;
    }
}

```

}

3. Kiểm tra mạng

```

public void Test(List<double> testDataSeries, double Max, double Min, int numInput)
{
    List<double> testDataOutput = new List<double>();
    List<double> testDataSeriesProcessed = new List<double>();
    List<double> testResultSeries = new List<double>();
    double MSE_ERROR = 0;

    double max = testDataSeries.Max();
    double min = testDataSeries.Min();
    int count = testDataSeries.Count;
    for (int i = 0; i < count; i++)
    {
        double a = testDataSeries.ElementAt(i);
        double b = (a - min) / (max - min) * (0.99 - 0.01) + 0.01;
        testDataSeriesProcessed.Add(b);
    }

    for (int i = 0; i < numInput; i++)
    {
        testResultSeries.Add(testDataSeries[i]);
    }

    for (int i = numInput; i < testDataSeriesProcessed.Count; i++)
    {
        double[] input = new double[numInput];
        for (int j = numInput; j > 0; j--)
        {
            input[numInput - j] = testDataSeriesProcessed[i - j];
        }

        s_Network.CalculateOutput(input);
        double temp = s_Network.m_arOutputNodes[0].GetOutput();
        temp = (temp - 0.01) / (0.99 - 0.01) * (max - min) + min;
        testResultSeries.Add(temp);
        MSE_ERROR += Math.Abs(temp - testDataSeries.ElementAt(i));
    }

    //MSE = MSE_ERROR / sampleTest.Count();
    TestForm formTest = new TestForm();
    formTest.textBox1.AppendText("Lỗi MSE = " + MSE_ERROR/
testResultSeries.Count() + "\n");
    formTest.textBox1.ReadOnly = true;
    for (int t = 0; t < testDataSeries.Count; t++)
    {
        if (testDataSeries.ElementAt(t) == testResultSeries.ElementAt(t))
            formTest.chart1.Series["Thực tế"].Points.AddXY(t + 1,
testDataSeries.ElementAt(t)+5);
        else
            formTest.chart1.Series["Thực tế"].Points.AddXY(t + 1,

```

```

testDataSeries.ElementAt(t));
        formTest.chart1.Series["Tính toán"].Points.AddXY(t + 1,
testResultSeries.ElementAt(t));
    }
    formTest.ShowDialog();
}

```

4. Dự báo

```

private void btForeCast_Click(object sender, EventArgs e)
{
    picForecast.Visible = false;
    int nHead = Convert.ToInt16(txtYearForecast.Text);
    ResultTextBox.Text = "";
    List<double> forecastResultSeries = new List<double>();
    List<double> testResultSeries = new List<double>();
    training.Forecast(ListInput, ListInput.Count, nHead, out
forecastResultSeries);
    DrawResult(ListInput, forecastResultSeries);
}

public void Forecast(List<double> testDataForeCast, int numInput, int nHead, out
List<double> forecastResultSeries)
{
    List<double> forecastDataSeriesProcessed = new List<double>();
    forecastResultSeries = new List<double>();

    double max = testDataForeCast.Max();
    double min = testDataForeCast.Min();
    int count = testDataForeCast.Count;

    for (int i = 0; i < count; i++)
    {
        double a = testDataForeCast.ElementAt(i);
        double b = (a - min) / (max - min) * (0.99 - 0.01) + 0.01;
        forecastDataSeriesProcessed.Add(b);
    }

    for (int i = 0; i < nHead; i++)
    {
        double[] input = new double[numInput];
        for (int j = numInput; j > 0; j--)
        {
            input[numInput - j] =
forecastDataSeriesProcessed[forecastDataSeriesProcessed.Count - j];
        }
        s_Network.CalculateOutput(input);
        double temp = s_Network.m_arOutputNodes[0].GetOutput();
        forecastDataSeriesProcessed.Add(temp);
        temp = (temp - 0.01) / (0.99 - 0.01) * (max - min) + min;
        forecastResultSeries.Add(temp);
    }
}
}

```

```

public void Forecast(List<double> testDataForeCast, int numInput, int nHead, out
List<double> forecastResultSeries)
{
    List<double> forecastDataSeriesProcessed = new List<double>();
    forecastResultSeries = new List<double>();

    double max = testDataForeCast.Max();
    double min = testDataForeCast.Min();
    int count = testDataForeCast.Count;

    for (int i = 0; i < count; i++)
    {
        double a = testDataForeCast.ElementAt(i);
        double b = (a - min) / (max - min) * (0.99 - 0.01) + 0.01;
        forecastDataSeriesProcessed.Add(b);
    }

    for (int i = 0; i < nHead; i++)
    {
        double[] input = new double[numInput];
        for (int j = numInput; j > 0; j--)
        {
            input[numInput - j] =
forecastDataSeriesProcessed[forecastDataSeriesProcessed.Count - j];
        }
        s_Network.CalculateOutput(input);
        double temp = s_Network.m_arOutputNodes[0].GetOutput();
        forecastDataSeriesProcessed.Add(temp);
        temp = (temp - 0.01) / (0.99 - 0.01) * (max - min) + min;
        forecastResultSeries.Add(temp);
    }
}

```


PHỤ LỤC C
BÀI BÁO KHOA HỌC