**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BỘ MÔN KHOA HỌC MÁY TÍNH**

- - - - 🙞🙞🟔🙜🙜 - - - -

**NHÓM THỰC HIỆN**

**0712133 – LÊ MINH DUY**

**0712228 – TRẦN TRUNG KIÊN**

**0712263 – VẠN DUY THANH LONG**

**071 – BÀNH TRÍ THÀNH**

**DỰ ĐOÁN GIÁ VÀ XU HƯỚNG CHỨNG KHOÁN BẰNG CÁC PHƯƠNG PHÁP MÁY HỌC**

**ĐỒ ÁN MÔN HỌC MÁY HỌC**

**NĂM 2010**

# MỤC LỤC

MỤC LỤC 2

Chương 1 NỀN TẢNG LÝ THUYẾT 4

1.1 Lý thuyết về mạng nơron 4

1.1.1 Nơron 4

1.1.1.1 Cấu trúc của một nơron 4

1.1.1.2 Các phép toán trên một nơron 5

1.1.2 Mạng nơron 9

1.1.2.1 Định nghĩa 9

1.1.2.2 Một số cấu trúc mạng nơron 9

1.1.2.3 Phương pháp học 11

1.1.3 Mạng truyền thẳng ba lớp 14

1.1.3.1 Cấu trúc mạng 14

1.1.3.2 Lan truyền tiến 14

1.1.3.3 Thuật giải lan truyền ngược 16

Chương 2 HƯỚNG TIẾP CẬN 17

2.1 Tối thiểu hóa độ lỗi: 17

2.2 Các chiến lược dự đoán: 18

2.2.1 Cách tổ chức dữ liệu đưa vào mô hình: 18

2.2.2 Tiếp cận theo hướng dùng hàm TSCFD 19

2.2.3 Một số độ đo lỗi đánh giá mô hình: 20

Chương 3 CÀI ĐẶT 21

3.1 Sơ đồ tổng quan các chức năng trong mô hình: 21

3.2 Training: 21

3.2.1 Cài đặt Mạng: 21

3.2.1.1 Tổ chức dữ liệu training: 21

3.2.1.2 Lựa chọn hàm chi phí: 22

3.2.1.3 Các tham số: 22

3.2.2 Tiền xử lý dữ liệu: 22

3.2.3 Quá trình training: 23

Chương 4 THỰC NGHIỆM 24

4.1 Dữ liệu 24

4.2 Kịch bản 24

Chương 5 KẾT LUẬN 25

5.1 Nhận xét về mô hình 25

5.1.1 Ưu điểm 25

5.1.2 Khuyết điểm 25

5.2 Hướng phát triển 25

PHỤ LỤC. HƯỚNG DẪN SỬ DỤNG CHƯƠNG TRÌNH 25

TÀI LIỆU THAM KHẢO 25

# NỀN TẢNG LÝ THUYẾT

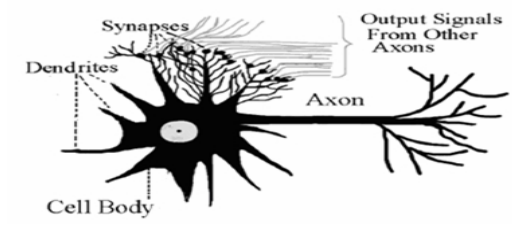
## Lý thuyết về mạng nơron

### Nơron

#### Cấu trúc của một nơron

##### Nơron sinh học

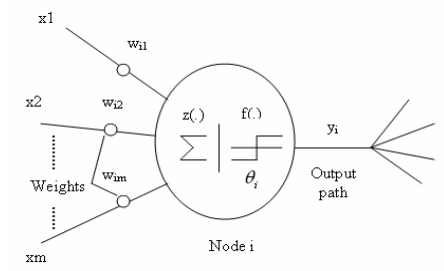
Nơron là tế bào thần kinh nằm trong bộ não con người, có chức năng xử lý thông nhận được từ các tín hiệu đầu vào. Nơron sinh học gồm các bộ phận sau: sysnapse, dendrite, cell body và axon



Dendrite có nhiệm vụ mang các tín hiệu đầu vào tới cell body để bộ phận này thực hiện việc xử lý, axon có nhiệm vụ đưa các tín hiệu sau khi xử lý ra ngoài .Các sysnapse là những khớp thần kinh có vai trò sắp xếp sự mạnh yếu của các tín hiệu đầu vào

##### Nơron nhân tạo

Các nhà tin học đã xây dựng nên một cấu trúc tương tự như nơron sinh học, gọi là nơron nhân tạo (hay gọi tắt là nơron) .Một nơron trong tin học cũng có cấu trúc tương tự như nơron sinh học .Sơ đồ hình dưới cho thấy cấu trúc của một nơron gồm 3 thành phần cơ bản



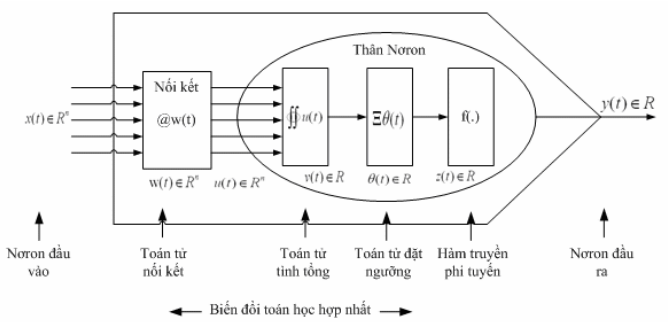
Tập hợp các kết nối mà mỗi một trong số chúng được đặc trưng bởi trọng số w. Một tín hiệu đầu vào xi trước khi được đưa vào nơron sẽ được nhân với trọng số w. Chức năng của trọng số w cũng tương đương với sysnapse trong nơron nhân tạo nhằm điều chỉnh mức độ ảnh hưởng của các tín hiệu đầu vào

Một bộ tính tổng các tín hiệu đầu vào sau khi đã được nhân với trọng số tương ứng của chúng

Kết quả từ phép tổng trọng sẽ được cho qua hàm truyền phi tuyến để thu nhận kết quả đầu ra của nơron. Chức năng của hàm truyền này dùng để giới hạn đầu ra của nơron trong một khoảng cho phép phù hợp với yêu cầu của bài toán

#### Các phép toán trên một nơron

Mô hình tổng quát của một nơron



x(t) € Rn : vector đầu vào của nơron

w(t) : vector trọng số

v(t) : kết quả của phép kết nối

θ(t) : phép đặt ngưỡng

y(t) : đầu ra của nơron

##### Phép hợp nhất

Phép hợp nhất gồm các toán tử sau:

1. Toán tử nối kết

Toán tử nối kết nhận vector đầu vào x(t) và nhân lần lượt từng thành phần của x(t) với từng thành phần của vector trọng số w(t). Kết quả của phép nối kết là vector u(t)

u(t) = x(t)@w(t)

1. Toán tử tính tổng

Toán tử tính tổng thực hiện việc cộng tất cả các thành phần của vector u(t) (u(t) là kết quả của toán tử nối kết)

v(t) = Σu(t)

1. Toán tử đặt ngưỡng

Toán tử đặt ngưỡng thực hiện việc cộng v(t) với ngưỡng θ(t) cho trước.

z(t) = v(t) Ξ θ(t)

Tóm lại phép toán hợp nhất được định nghĩa là phép toán đo lường quan hệ qua lại giữa vector nhập và vector trọng nối kết



Một số phép biến đổi hợp nhất thường sử dụng

Phép hợp nhất tuyến tính



Phép hợp nhất bậc hai



Phép hợp nhất cầu



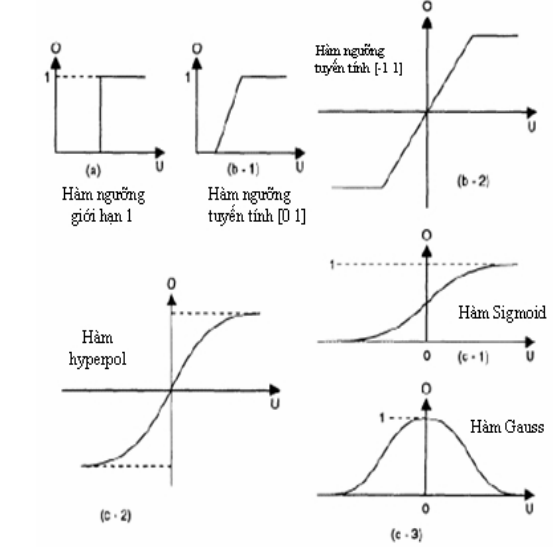
Phep hợp nhất đa mức



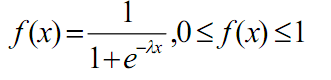
##### Hàm truyền phi tuyến

Hàm truyền phi tuyến có chức năng xếp hạng độ đo sao cho đảm bảo tính mềm dẻo và chặt chẽ của ánh xạ nơron và trả về độ đo tại đầu ra của nơron.

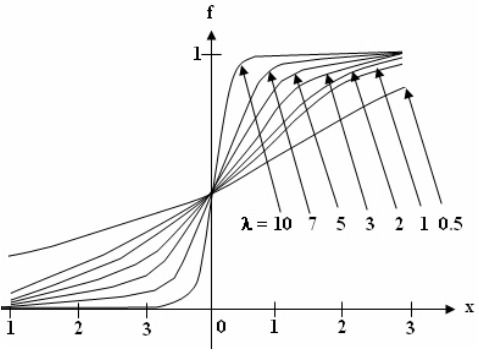
Một số hàm truyền phi tuyến



Hàm truyền thông dụng nhất là hàm sigmoid



Đồ thị có dạng



### Mạng nơron

#### Định nghĩa

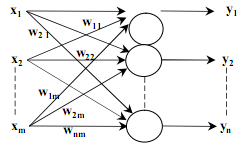
Mạng Nơron nhân tạo là sự mô phỏng sinh học bằng máy tính bộ não của con người. Nó có cấu trúc song song được cấu thành từ nhiều phần tử (Nơron nhân tạo) liên kết với nhau thông qua các trọng số, tậphợp các trọng số này tạo thành bộ trọng của mạng Nơron nhân tạo .Mỗi Nơron nhân tạo là một hệ thống động phi tuyến cókhả năngtự học .Các Nơron và bộ trọng cấu thành cấu trúc mạng. Do đó mạng Nơron có khả năng học từ kinh nghiệm hay từ tập mẫu Mạng Nơron được gọi là mô hình liên kết vì vai tròquan trọng của sự kết nối giữa chúng. Bộ trọng liên kết chính là “bộ nhớ” của hệ thống

#### Một số cấu trúc mạng nơron

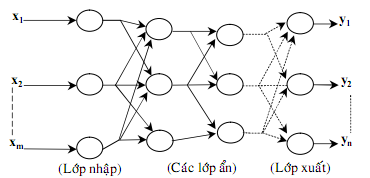
Mạng Nơron nhân tạo bao gồm tập hợp các liên kết qua lại bên trong giữa các Nơron trên nguyên tắc: “Đầu ra của mỗi Nơron được liên kết thông qua các trọng số đến các Nơron khác hoặc tới chính nó. Như vậy, việc bố trí các Nơron và sơ đồ liên kết qua lại giữa chúng sẽ hình thành một kiểu mạng Nơron nhân tạo

Một số cấu trúc mạng:

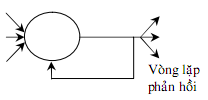
1. Mạng truyền thẳng một lớp



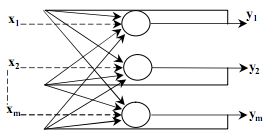
1. Mạng truyền thẳng đa lớp



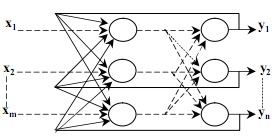
1. Một nơron liên kết phản hồi với chính nó



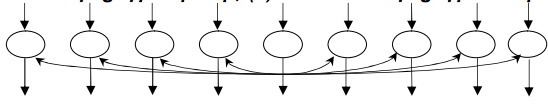
1. Mạng lặp một lớp



1. Mạng lặp đa lớp



1. Liên kết bên trong của phản hồi nhánh

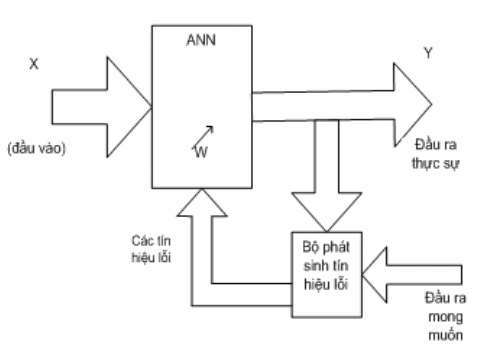


#### Phương pháp học

Chức năng của một mạng nơron được quyết định bởi các tính chất: cấu trúc mạng (số nơron của các lớp, cách thức các lớp liên kết với nhau) và các trọng số liên kết. Cấu trúc mạng thường là cố định, không thay đổi và các trọng số được quyết định bởi một thuật toán huấn luyện .Sự thay đổi giá trị các trọng số nhằm làm cho mạng có khả năng nhận biết mối quan hệ giữa đầu vào và đầu ra mong muốn .Tiến trình thay đổi các trọng số gọi là quá trình học mạng (learning) hay huấn luyện mạng (training). Các thuật toán học được chia làm ba nhóm chính: học có giám sát, học không giám sát và học tăng cường

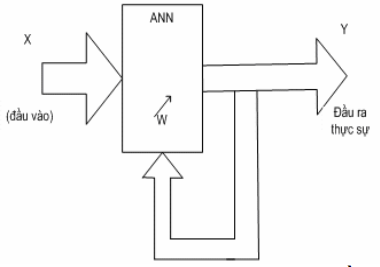
##### Học có giám sát

Mạng được học bằng cách cung cấp cho nó một mẫu gồm hai thành phần: vector đầu vào và vector đầu ra mong muốn tương ứng. Sai số giữa đầu ra thực tế và đầu ra mong muốn được thuật toán sử dụng để điều chỉnh trọng số trong mạng theo hướng giảm dần sai số, làm cho đầu ra thực tế gần với đầu ra mong muốn hơn



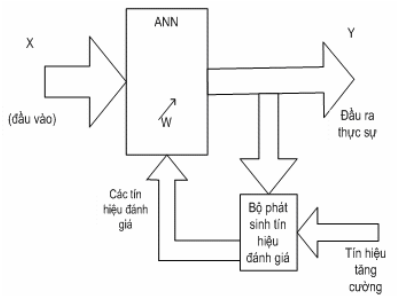
##### Học không giám sát

Với cách học không giám sát, không có đầu ra mong muốn làm chuần để điều chỉnh trọng số. Khi đó mạng buộc lòng phải tự khám phá ra các đặc trưng, mối quan hệ giữa đầu vào và đầu ra để tự động điều chỉnh cho đúng



##### Học tăng cường

Học tăng cường là một hình thức kết hợp giữa học có giám sát và học không giám sát. Nếu trong phương pháp học có giám sát, có hẳn đầu ra mong muốn để mạng điều chỉnh hướng các đầu ra thực tế gần với đầu ra mong muốn hơn thì trong học tăng cường, mạng chỉ có thể nhận được các phản hồi đơn giản như: “độ chính xác là 50%”, đầu ra “đúng” hay “sai”, “quá cao” hay “quá thấp”. Việc học tăng cường chỉ dựa trên những thông tin đánh giá mạng và các tín hiệu phản hồi gọi là tín hiệu tăng cường



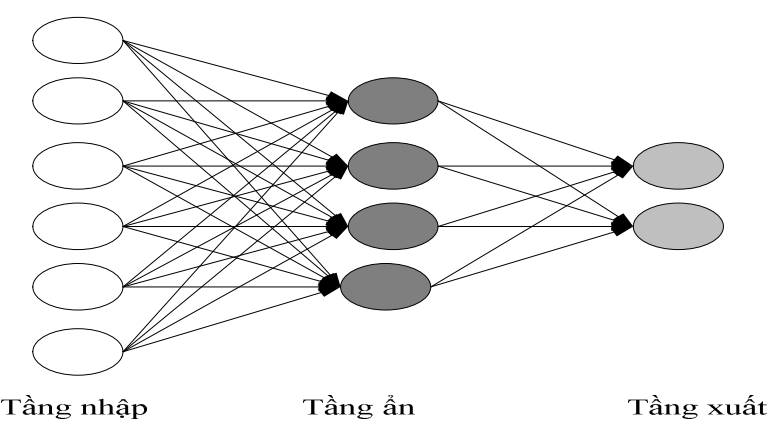
### Mạng truyền thẳng ba lớp

#### Cấu trúc mạng

Mạng nơron truyền thẳng được xem như một ánh xạ f giữa tập các giá trị đầu vào I và tập các giá trị đầu ra O

f: I 🡪 O hoặc Y = f(X) với X thuộc I và Y thuộc O

Một mạng truyền thẳng ba lớp bao gồm một lớp nhập, một lớp ẩn và mộp lớp xuất .Các nơron lớp nhập thực chất không phải là những nơron đúng nghĩa khi chúng không phải thực hiện các thao tác xử lý nào, chúng chỉ đơn thuần tiếp nhận các giá trị đầu vào và đưa vào trong lớp kế tiếp. Các nơron ở lớp ẩn mới thực sự thực hiện các tính toán và xuất giá trị đầu ra .Cụm từ “truyền thẳng” chỉ diễn tả tính chất của mạng là các nơron liên kết với nhau theo một hướng xác định



#### Lan truyền tiến

Quá trình lan truyền tiến là quá trình tính giá trị các nút xuất từ mẫu nhập vào mạng. Tiến trình này được sử dụng trong hai tình huống: học mạng và sử dụng mạng. Khi học, lan truyền tiến được sử dụng nhiều lần để cập nhật các giá trị trọng số. Ngược lại, khi sử dụng mạng, lan truyền tiến chỉ thực thi một lần cho từng mẫu nhập

Cho mạng truyền thẳng ba lớp có các thông số sau:

* Số nơron của lớp nhập là I
* Số nơron của lớp ẩn là H
* Số nơron của lớp xuất là K
* Vector đầu vào là xi (i = 1 🡪 I)

Phương trình tính kết xuất của các nút ẩn là

y = g() với j = 1, 2,…, H

Kết xuất của mạng với K nút xuất là

g(x) là hàm truyền phi tuyến

#### Thuật giải lan truyền ngược

* + Huấn luyện mạng nơron truyền thẳng ba lớp sử dụng thuật toán Lan truyền ngược gồm hai quá trình: Quá trình lan truyền tiến và quá trình lan truyền ngược:

Cho không gian các mẫu học (x, t) là giá trị cần huấn luyện; t là giá trị kết quả đích (đầu ra mong muốn) của quá trình huấn luyện .Hệ sốhọc .Qui định chỉ số lớp là tăng dần từ lớp đầu vào đến lớp đầu ra .Thuật giải lan truyền ngược được tóm tắt như sau:

Bước 1: Tạo mạng truyền thẳng gồm I nơron lớp nhập, H nơron lớp ẩn và K nơron lớp xuất

Bước 2: Khởi tạo bộ trọng cho mạng với giá trị nhỏ

Bước 3: Trong <điều kiện chưa kết thúc>

+ Tiến hành quá trình lan truyền tiến của mẫu qua mạng

+ Với mỗi đầu ra của lớp xuất tính sai số

Chuyển sang lớp ẩn

+ Với mỗi nơron trên lớp ẩn, tính sai số

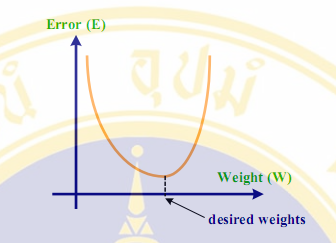
+ Cập nhật lại trọng số

+ Chọn mẫu (x,t) mới và thực hiện lại bước 3

# HƯỚNG TIẾP CẬN

## Tối thiểu hóa độ lỗi:

* Để tối thiểu hóa độ lỗi của mô hình, ta sẽ tính độ lỗi ở mỗi lần train (hay test) mẫu, người ta đưa ra mối tương quan giữa độ lỗi và trọng số qua đồ thị sau:



* Mục tiêu của chúng ta là cần tìm ra một trọng số sao cho ở đấy độ lỗi đạt cực tiểu.
* Mỗi mẫu p, chúng ta gán tương ứng độ lỗi Ep. Ta có một trong những công thức tính độ lỗi đơn giản như sau:
* Độ lỗi cho cả mô hình chính là tổng độ lỗi khi train (hay test) một mẫu
* Như vậy để mô hình dự đoán chính xác, ta cần tìm ra một bộ trọng sao cho ở đấy độ lỗi đạt cực tiểu.
* Mean of square error (MSE) được dùng như là một điều kiện đề ngừng training mạng.

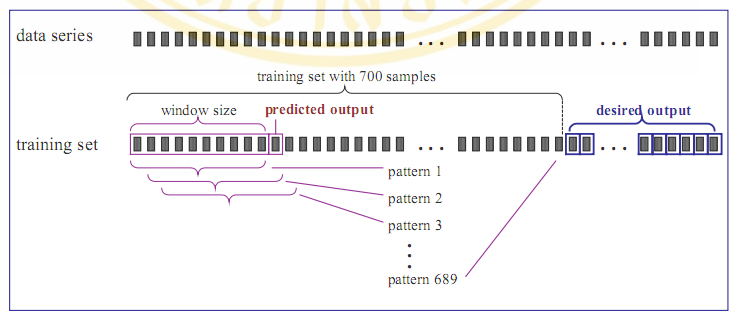
Ở đây N là tổng số mẫu.

* Như vậy khi MSE đạt đến một ngưỡng chấp nhận được (do ta qui định), thì mạng coi như training thành công.

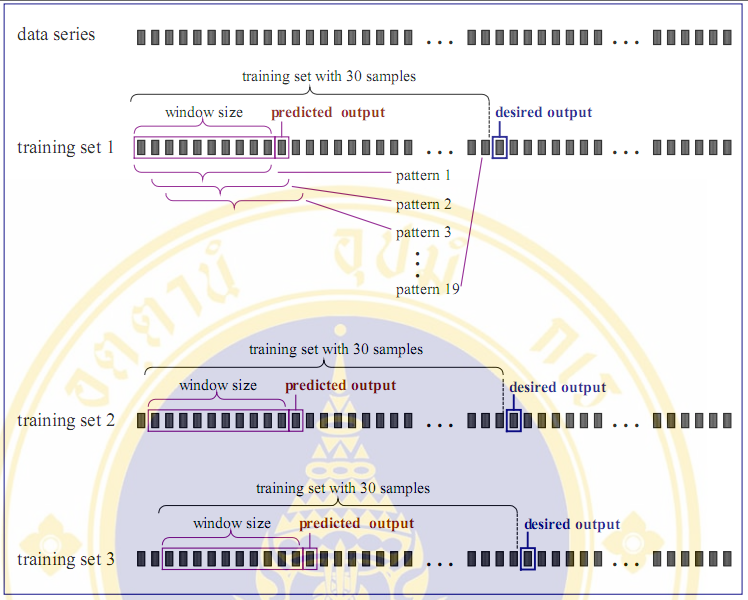
## Các chiến lược dự đoán:

### Cách tổ chức dữ liệu đưa vào mô hình:

Cách truyền thống:



Cách cải tiến:



**Nhận xét:**

Như đã nói, một trong những nhược điểm lớn của ANN là cần một lượng lớn dữ liệu mẫu cho quá trình training, đó là một trong những nguyên nhân làm quá trình training chậm đi, và hơn nữa chúng ta tiến hành xử lý dữ liệu chuỗi thời gian nên sẽ có một số mẫu sau một thời gian nó không còn mang tính hợp lệ (như giai đoạn đầu mới mở thì nó không theo một quy luật nào, nhưng dần dần tuy vẫn còn nhiễu nhưng nó sẽ tiến đến một xu hướng chung của quốc tế). Đó cũng là nhược điểm ở cách truyền thống.

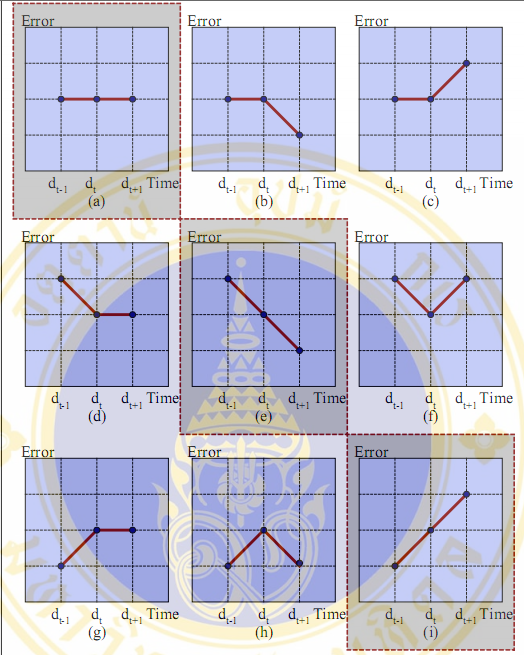
Ở cách cải tiến ta sẽ phân đoạn bộ training ra thành những bộ dữ liệu nhỏ hơn để đạt được những kết quả tốt hơn bởi vì mạng neuron bây giờ sẽ được train dần dần theo thời gian.

### Tiếp cận theo hướng dùng hàm TSCFD

Trong dự đoán dữ liệu chuỗi thời gian, độ chính xác không chỉ phụ thuộc vào việc mô hình cho ra kết quả dự đoán gần với giá trị thực, mà nó còn bị chi phối bởi yếu tố xu hướng biến động của giá trị dự đoán.

Do đó hàm chi phí đáp ứng vấn đề này được đưa ra có tên Tow-Step Continuous Fluctuation Direction-based (TSCFD cost function)

Hình bên dưới đây sẽ mô tả các trường hợp có thể xảy của giá trị thực:



Ở đây dt chính là giá trị thực của ngày dự đoán, dt+1 là giá trị của ngày tiếp theo.

Chúng ta chỉ quan tâm đến trường hợp (a), (e), (i) vì nó biểu lộ xu hướng rõ ràng tăng, giảm hoặc giữ nguyên.

Hàm này sẽ được cải tiến ở hàm (\*\*\*\*) ở mục 1.1

Giá trị x ở đây đã thay thế cho dk ở hàm (\*\*\*\*)

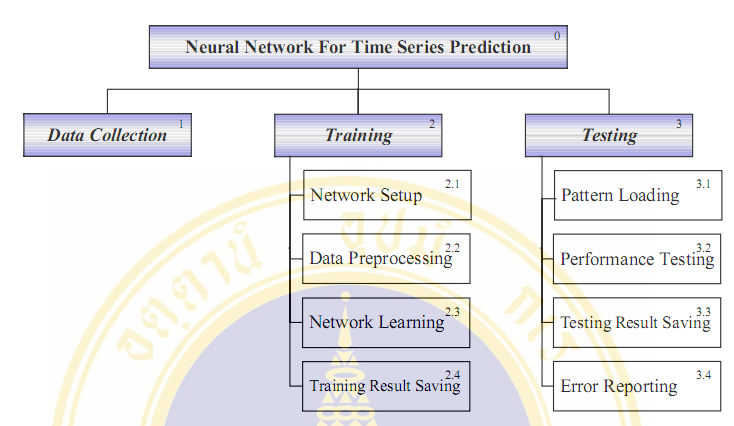
Và x sẽ được tính như sau:

### Một số độ đo lỗi đánh giá mô hình:

MSE, MAE, MAPE,...: để đo sự khác biệt giữa giá trị đoán và giá trị thực.

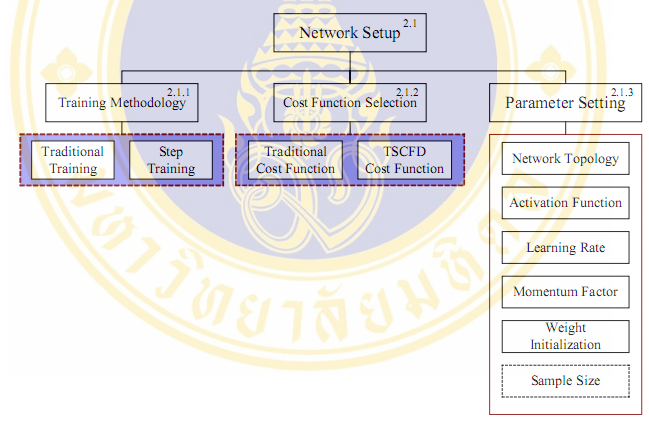
# CÀI ĐẶT

## Sơ đồ tổng quan các chức năng trong mô hình:



## Training:

### Cài đặt Mạng:



#### Tổ chức dữ liệu training:

* Như đã trình bày ở trên, ta sẽ có 2 cách là cách truyền thống và cải tiến (step-training)

#### Lựa chọn hàm chi phí:

* Ta cũng sẽ có 2 hàm là truyền thống và hàm TSCFD.

#### Các tham số:

* Network Topology: Ở đây ta chọn mạng truyền thẳng 3 lớp với thuật toán học lan truyền ngược.
* Trong đó lớp output sẽ có 1 node xuất ra giá dự đoán.
* Số node ở lớp input và lớp ẩn là không biết trước, sẽ phải qua thực nghiệm để quyết định. (Bởi vì không có một giả thiết nào có trước đưa ra số node gần đúng cho mô hình.)
* Cấu trúc mạng mà một số bài báo đã áp dụng:
* Input (5,6,10,15,20) – Hiden ()

5-2-1, 5-4-1,6-3-1,10-3-1,15-3-1,**20-2-1**

* 64-8-1, 64-16-1, 64-32-1, **64-64-1**, 64-100-1
* 28-60-1 (Cố định số node ẩn = số node input = n, khảo sát n, rồi sao đó khảo sát nó node ẩn)
* ?-(9-14)-1
* Hàm truyền phi tuyến: Nhìn chung đều là dùng hàm sigmoid, nhưng có một số dạng như sau
* Hệ số học : Hệ số lớn sẽ giúp đẩy nhanh tốc độ học, nhưng nếu quá cao sẽ làm cho kết quả không thể hội tụ như mong muốn (step over).Miền giá trị một số bài báo đưa ra là 0.1, 0.01-0.05
* Momentum Factor : Giúp cho việc tránh rơi vào bẫy tối ưu cục bộ. Cần qua thực nghiệm để lựa chọn.
* Khởi tạo bộ trọng số w ngẫu nhiên trong đoạn [0,1].

### Tiền xử lý dữ liệu:

* Đây được đánh giá là một bước khá quan trọng, vì nó sẽ giúp cho mô hình có thể biết được mối quan hệ tưởng tiềm ẩn trong dữ liệu. Trong đó gồm 2 vấn đề chính là khử nhiễu và chuẩn hóa giá trị.
* Về khử nhiễu hiện tại có một số phương pháp: dùng giá trị trung bình trượt (mô hình ARIMA), rút đặc trưng thành phần chính (ICA, PCA, KICA, KPCA), dùng Kohonen.

Ví dụ: Một công thức ARIMA:

* Về chuẩn hóa giá trị, ta sẽ chuẩn hóa về [0,1] cho phù hợp với miền giá trị của hàm truyền phi tuyến sigmoid trong mô hình. Hiện tại có 2 công thức:

Đối với dự đón xu hướng thì dữ liệu ở đầu vào được chuẩn hoá về dạng return [-1,1] theo công thức sau:

Giá đóng cửa được scale về dạng return theo công thức

 với I là giá

Và khảo sát với số node là 5.

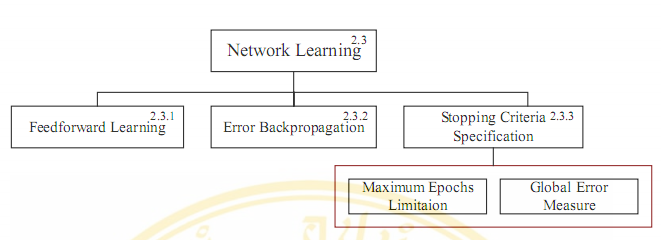
Đối với từng period khác nhau, giá trị từng node có sự điều chỉnh.

* Period = 1 và 5: mỗi node là return của 1 ngày trước đó.
* Period = 10: mỗi node là return của từng bộ 2 ngày trước.
* Period = 30: mỗi node là return của từng bộ 6 ngày trước.

Ví dụ với period = 10, thời điểm hiện tại là ngày 10, thời điểm cần dự đoán là ngày 20. Nhãn dự đoán sẽ là return 20 – 10 (của ngày 20 so với ngày 10), các node bên trong lần lượt là:

1: return 2 – 0, 2: return 4 – 2, 3: return 6 – 4, 4: return 8 – 6, 5 : return 10 – 8.

### Quá trình training:



* Global Error Measure: Ta sẽ dùng MSE, MAE, MAPE, Tolerance và so sánh kết quả xem cái nào tốt hơn.

# THỰC NGHIỆM

## Dữ liệu

Dữ liệu thực nghiệm đối với mô hình ANN là 5 mã chứng khoán Việt Nam: BT6, DHG, FPT, VIS và VNM

Mã BT6 được xét trên toàn bộ các giao dịch từ 18/04/2002 đến 10/12/2010

Mã DHG được xét trên toàn bộ các giao dịch từ 21/12/2006 đến 10/12/2010

Mã FPT được xét trên toàn bộ các giao dịch từ 13/12/2006 đến 10/12/2010

Mã VIS được xét trên toàn bộ các giao dịch từ 25/12/2006 đến 10/12/2010

Mã VNM được xét trên toàn bộ các giao dịch từ 19/1/2006 đến 10/12/2010

Tất cả dữ liệu được chi theo tỷ lệ huấn luyện : kiểm thử là 9:1

## Kịch bản

## Phần ANN

Các tham số của mạng như sau:

* Số node tầng nhập: 5.
* Số node tầng ẩn: 4.
* Số node tầng xuất: 1.
* Hệ số học 0.3
* Số vòng lập tối đa 2000
* Bias 0
* Độ chính xác khi train 90%

Thực hiện kiểm thử với các period lần lượt 1, 5, 10 và 30 tương ứng với dự đón xu hướng của 1 ngày, 5 ngày, 10 ngày và 30 ngày trong tương lai

Kết quả khi thực hiện kiểm thử với từng Period

Period = 1:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | NMSE | SIGN(%) | DM4Price(%) | | | Độ lỗi khi train | |
| Right | Wrong | Cannot measure | NMSE | Max Loop |
| BT6 | 1.28 | 42 | 72.09 | 19.06 | 8.83 | 1.65 | 2000 |
| DHG | 1.34 | 25 | 51.02 | 13.27 | 35.71 | 1.03 | 2000 |
| FPT | 1.3 | 43 | 72.73 | 23.23 | 4.04 | 1.18 | 2000 |
| VIS | 1.03 | 41 | 65.31 | 34.69 | 0 | 1.04 | 2000 |
| VNM | 1.15 | 32 | 57.85 | 21.48 | 20.66 | 0.98 | 2000 |

Period = 5:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | NMSE | SIGN(%) | DM4Price(%) | | | Độ lỗi khi train | |
| Right | Wrong | Cannot measure | NMSE | Max Loop |
| BT6 | 1.4 | 44 | 58.69 | 32.39 | 8.92 | 1.38 | 2000 |
| DHG | 1.17 | 25 | 41.67 | 27.08 | 31.25 | 1.01 | 2000 |
| FPT | 1.21 | 38 | 56.7 | 42.27 | 1.03 | 1.78 | 2000 |
| VIS | 1.03 | 70 | 62.11 | 37.89 | 0 | 1 | 2000 |
| VNM | 1.78 | 50 | 48.74 | 30.25 | 21 | 1.04 | 2000 |

Period = 10:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | NMSE | SIGN(%) | DM4Price(%) | | | Độ lỗi khi train | |
| Right | Wrong | Cannot measure | NMSE | Max Loop |
| BT6 | 2.32 | 39 | 51.42 | 42.86 | 5.71 | 1.03 | 2000 |
| DHG | 1.09 | 25 | 38.7 | 29.03 | 32.25 | 0.98 | 2000 |
| FPT | 1.41 | 61 | 58.51 | 36.17 | 5.31 | 2.18 | 2000 |
| VIS | 0.93 | 73 | 50 | 50 | 0 | 1.07 | 2000 |
| VNM | 1.35 | 25 | 38.79 | 46.55 | 14.66 | 1.12 | 2000 |

Period = 30:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | NMSE | SIGN(%) | DM4Price(%) | | | Độ lỗi khi train | |
| Right | Wrong | Cannot measure | NMSE | Max Loop |
| BT6 | 0.91 | 46 | 51.01 | 42.93 | 6.06 | 0.98 | 2000 |
| DHG | 2.57 | 68 | 39.5 | 29.63 | 30.86 | 1.15 | 2000 |
| FPT | 1.89 | 55 | 57.31 | 40.24 | 2.43 | 1.04 | 2000 |
| VIS | 1.18 | 79 | 48.75 | 51.25 | 0 | 1.04 | 2000 |
| VNM | 1.41 |  | 44.23 | 35.58 | 20.19 | 1.17 | 2000 |

Nhận xét:

Với period nhỏ thì kết quả dự đoán nhìn chung rất khả quan ví dụ với mã chứng khoán BT6 với Period =1 thì độ đo DM4Price cho thấy kết quả dự đoán đúng xu hương lên đến 72.09% hay mã FPT với 72.73% một kết quả tương đối cao. Và nhìn qua các mã khác ta cũng vậy.

Tuy nhiên với số period càng lớn thì kết quả dự đoán xu hướng cũng giảm dần. chẳng hạng như mã VIS ta thấy với period lần lược 1, 5, 10, 30 thì dự đón đúng theo DM4Price tương ứng 65.31% 🡪 62.11% 🡪 50% 🡪 48.75%

* Với chu kỳ

Và dưới đây là đồ thị thể hiện kết quả dự đón so với đường thực của mã chứng khoán FPT với từng period:

Period = 1:

Period = 5:

Period = 10:

Period 30:

# KẾT LUẬN

## Nhận xét về mô hình

### Ưu điểm

* Phần lớn các mô hình ANN dùng dữ liệu trong quá khứ và ngày hiện tại để dự đoán giá trong tương lai. ANN được dùng rất phổ biến bởi vì nó cho ra kết quả gần đúng với bất kỳ hàm phi tuyến nào với mức độ chính xác cao.
* Ở những giai đoạn trước, phần lớn nghiên cứu tập trung vào các ứng dụng chỉ dùng mô hình ANN chuẩn để dự đoán như: dùng recurrent neural networks, mạng lan truyền ngược với luật tổng quát delta…
* ANN có thể dùng hiệu quả trong việc nhận dạng xu hướng thông qua xử lý các dữ liệu trong quá khứ mà không cần sự giám sát của con người. ANN có một thời gian dài được xem là công cụ lý tưởng cho dự đoán chứng khoán bởi một số ưu điểm của nó như dễ cài đặt, phân lớp tốt và sử dụng bộ nhớ thấp. Mô hình Multilayer Perceptron (MLP) được cài trong nhiều ứng dụng, vì tính đơn giản trong cài đặt và hiệu quả của hệ thống lan truyền ngược.

### Khuyết điểm

* Tuy nhiên, ANN mà cụ thể là Back propagation (BP) ANN – mạng lan truyền ngược bị hạn chế vì tín hiệu nhiễu, biến động dữ dội của dữ liệu chứng khoán. Khi training mạng lan truyền ngược trước những tín hiệu nhiễu của dữ liệu thường cho ra những kết quả dự đoán thường hay xuất hiện, giải pháp mà BP sử dụng thường tập trung vào sự tối ưu cục bộ cũng như kiến trúc của mạng (số lớp ẩn, số node từng lớp) thường phải qua thực nghiệm để quyết định, khó đưa ra tối ưu toàn cục, chịu sự tác động tăng thời gian xử lý khi dữ liệu lớn dần, trở nên phức tạp khi gặp vấn đề số chiều thay đổi.
* Thị trường chứng khoán về bản chất là luôn biến động, phi tuyến, phức tạp, khó tham số hóa và thật sự hỗn loạn. Phụ thuộc vào nhiều yếu tố tác động như sự kiện chính trị, điều kiện kinh tế toàn cầu, lãi suất ngân hàng, nhà đầu tư…
* Việc xác định bộ tham số cho ANN là điều cần phải thực nghiệm nhiều, và có nên sử dụng phương pháp cải tiến để xác định bộ tham số thích hợp.

## Hướng phát triển

# PHỤ LỤC. HƯỚNG DẪN SỬ DỤNG CHƯƠNG TRÌNH

# TÀI LIỆU THAM KHẢO

[1]

[2]

[3]