ĐẠI HỌC HUẾ

TRƯỜNG ĐẠI HỌC KHOA HỌC

NGUYỄN ĐỨC HIỂN

XÂY DỰNG MÔ HÌNH LAI CHO BÀI TOÁN DỰ BÁO THEO TIẾP CẬN MỜ HƯỚNG DỮ LIỆU

LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH

HUẾ - NĂM 2019

ĐẠI HỌC HUẾ
TRƯỜNG ĐẠI HỌC KHOA HỌC

NGUYỄN ĐỨC HIỂN

XÂY DỰNG MÔ HÌNH LAI CHO BÀI TOÁN DỰ BÁO THEO TIẾP CẬN MỜ HƯỚNG DỮ LIỆU

CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH
MÃ SỐ: 9480101

LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH

Người hướng dẫn khoa học:

PGS.TS. Lê Mạnh Thạnh

HUẾ - NĂM 2019

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu do tôi thực hiện dưới sự hướng dẫn của PGS.TS. Lê Mạnh Thạnh. Các kết quả được viết chung với các tác giả khác đều được sự đồng ý của đồng tác giả trước khi đưa vào luận án. Các kết quả trong luận án là trung thực và chưa từng được công bố trong bất kỳ công trình nào khác.

Thừa Thiên Huế, ngày 20 tháng 06 năm 2019

Tác giả

Nguyễn Đức Hiển

LỜI CẢM ƠN

Luận án được thực hiện tại Khoa Công nghệ thông tin – Trường Đại học khoa học – Đại học Huế, dưới sự hướng dẫn của PGS.TS. Lê Mạnh Thạnh. Tôi xin bày tỏ lòng biết ơn sâu sắc đến Thầy về định hướng khoa học, người đã động viên, trao đổi nhiều kiến thức và chỉ bảo tôi vượt qua những khó khăn để hoàn thành luận án này.

Tôi cũng xin gửi lời cảm ơn chân thành đến các nhà khoa học, tác giả của các công trình công bố đã được trích dẫn trong luận án, đây là những tư liệu quý, kiến thức liên quan quan trọng giúp Nghiên cứu sinh hoàn thành luận án; Xin cảm ơn đến tất cả các Thầy, Cô tại Khoa Công nghệ thông tin – Trường Đại học Khoa học – Đại học Huế và các nhà khoa học đã góp ý, phản biện các công trình nghiên cứu của tôi.

Tôi trân trọng cảm ơn Khoa Công nghệ thông tin, Phòng đào tạo sau đại học thuộc Trường đại học Khoa học – Đại học Huế đã tạo điều kiện thuận lợi cho tôi trong suốt quá trình nghiên cứu thực hiện luận án.

Xin cảm ơn Ban giám hiệu Trường cao đẳng Công nghệ thông tin, các đồng nghiệp tại Khoa Công nghệ thông tin đã quan tâm giúp đỡ, tạo điều kiện để tôi có thể thực hiện kế hoạch nghiên cứu đảm bảo tiến độ.

Cuối cùng, tôi xin gửi lời cảm ơn sâu sắc tới gia đình, bạn bè, những người đã luôn ủng hộ, giúp đỡ và hỗ trợ tôi về mọi mặt để tôi yên tâm học tập đạt kết quả tốt. Luận án cũng là món quà tinh thần mà tác giả trân trọng gửi tặng đến các thành viên trong Gia đình.

DANH MỤC THUẬT NGỮ

Thuật ngữ Tiếng Anh	Viết tắt	Diễn giải Tiếng Việt
A priori knowledge		Tri thức tiên nghiệm
Adaptive-Network-based Fuzzy Inference System	ANFIS	Mạng thích nghi dựa trên cơ sở hệ suy luận mờ
Artificial Neural Networks	ANN	Mạng nơ-ron nhân tạo
Classification		Phân lớp
Clustering		Phân cụm
Data driven fuzzy models		Mô hình mờ hướng dữ liệu
Directional Symmetry	DS	Sự đổi hướng của dữ liệu thời gian
Exponential Moving Average	EMA	Đường trung bình động hàm mũ
Explanation-Based Learning	EBL	Học dựa trên sự giải thích
Forecasting		Dự báo
Fuzzy models		Mô hình mờ
Fuzzy rules-based models		Mô hình dựa trên luật mờ
Genetic Algorithms	GA	Giải thuật di truyền
Gross Domestic Product	GDP	Tổng sản phẩm quốc nội
Hierarchical Clustering	HC	Phân cụm theo thứ bậc
Interpretability		Tính có thể diễn dịch được
Knowledge-Based Inductive Learning	KBIL	Học quy nạp dựa trên tri thức
Magnetic Resonance Imaging	MRI	Hình ảnh đa phổ cộng hưởng từ
Mean Absolute Error	MAE	Sai số tuyệt đối trung bình
Mean Absolute Percent Error	MAPE	Sai số phần trăm tuyệt đối trung bình
Mean Square Error	MSE	Sai số bình phương trung bình
Multi Inputs and Single Output	MISO	Hệ thống nhiều đầu vào và một đầu ra
Normalize Mean Square Error	NMSE	Sai số bình phương trung bình chuẩn hóa
Prediction		Dự đoán
Radial Basis Functions	RBF	Hàm cơ sở hướng tâm
Radial Basis Network	RBN	Mạng nơ-ron RBN

Regression		Hồi quy
Relative Difference in Percentage of Price	RDP	Sai biệt tương đối (%) của giá
Relevance-Based Learning	RBL	Học dựa trên sự thích hợp
Root Mean Squared Error	RMSE	Sai số bình phương trung bình gốc
Self-Organizing Map	SOM	Mạng tự tổ chức / Bản đồ tự tổ chức
Support Vector	SV	Véc-tơ hỗ trợ
Support Vector Machine	SVM	Máy học véc-tơ hỗ trợ
ε-Support Vector Regression	ε-SVR	Máy học véc-tơ hỗ trợ hồi quy
SVM-based fuzzy models	f-SVM	Mô hình mờ dựa trên SVM
SVM-based Interpretable Fuzzy models	SVM-IF	Mô hình mờ có thể diễn dịch được dựa trên SVM
Takagi, Sugeno and Kang	TSK	Mô hình mờ TSK

MỤC LỤC

DANH MỤC BẢNG BIỂU

DANH MỤC HÌNH ẢNH

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Dự báo là một khoa học và nghệ thuật tiên đoán những sự việc sẽ xảy ra trong tương lai, trên cơ sở phân tích khoa học về các dữ liệu đã thu thập được. Thuật ngữ dự báo (forecasting) thường được sử dụng trong ngữ cảnh là quá trình đưa ra dự đoán (prediction) về tương lai dựa trên dữ liệu trong quá khứ và hiện tại, tuy nhiên các nguyên tắc của nó cũng hoàn toàn có thể ứng dụng để dự đoán các biến chéo. Chẳng hạn, người ta có thể dựa vào giá cổ phiếu trong quá khứ và hiện tại để dự đoán giá cổ phiếu trong tương lai. Tuy nhiên, người ta cũng có thể sử dụng những chỉ số của kinh tế vi mô để dự đoán được giá cổ phiếu, hay có thể dựa vào những đặc điểm cho trước của một ngôi nhà để dự đoán giá của ngôi nhà đó, … Có hai loại cơ bản của kỹ thuật dự báo [9]:

- Kỹ thuật dự báo định tính dựa trên các ý kiến, đánh giá, quan điểm, trực giác hay kinh nghiệm của chuyên gia. Kỹ thuật này thường được sử dụng khi dữ liệu quá khứ không đầy đủ hay đối tượng dự báo bị ảnh hưởng bởi những nhân tố không thể lượng hóa.
- Kỹ thuật dự báo định lượng, ngược lại với kỹ thuật định tính, dựa trên các phương pháp thống kê để phân tích dữ liệu lịch sử. Kỹ thuật này được sử dụng khi có đầy đủ dữ liệu lịch sử liên quan đến vấn đề dự báo, dữ liệu lượng hóa được và có một giả định về mối quan hệ giữa giá trị dữ liệu trong quá khứ hoặc giữa giá trị của các biến khác với biến cần dự báo.

Kỹ thuật dự báo định lượng sẽ dựa trên việc phân tích dữ liệu lịch sử để vẽ ra và mô hình hóa chiều hướng vận động của đối tượng phù hợp với một mô hình toán học nào đó, đồng thời sử dụng mô hình này cho việc dự báo xu hướng tương lai. Các kỹ thuật phân tích hồi quy cho phép xây dựng các mô hình hồi quy mô tả mối quan hệ giữa biến cần dự báo Y với các biến độc lập X [9][10]. Các mô hình máy học thống kê như máy học véc-tơ hỗ trợ, mạng nơ-ron nhân tạo, … cũng được nhiều nhà khoa

học nghiên cứu áp dụng với hy vọng xây dựng mô hình dự báo có độ chính xác cao hơn [6][26][42][52][54][71][87][90].

Những nghiên cứu xây dựng mô hình dựa trên luật mờ (fuzzy rule-based models) (có thể gọi ngắn gọn là mô hình mờ - fuzzy models) cũng là một trong những hướng tiếp cận để xây dựng các hệ thống hỗ trợ dự báo, dự báo điều khiển. Thành phần cốt lõi, cơ bản của một mô hình mờ là cơ sở tri thức của mô hình đó, mà cụ thể đó là tập luật mờ và lập luận hay suy diễn.

Về cơ bản có hai cách xây dựng cơ sở tri thức của mô hình mờ: Thứ nhất, thu thập tri thức dựa trên kinh nghiệm của các chuyên gia, được phát biểu dưới dạng các luật, các quy tắc, gọi chung là tri thức chuyên gia (Expert knowledge); Thứ hai là tích lũy, tổng hợp và hoàn thiện cơ sở tri thức dựa trên việc khám phá tri thức từ dữ liệu thực tế, gọi là tri thức dữ liệu (Data knowledge).

Theo hướng tiếp cận đầu tiên, chất lượng hoạt động của mô hình phụ thuộc vào chất lượng tri thức mà chuyên gia lĩnh vực cung cấp. Nếu người xây dựng mô hình khai thác tri thức từ một chuyên gia có kinh nghiệm, hiểu rõ lĩnh vực thì mô hình thu được sẽ có độ tin cậy cao. Tuy nhiên, tiêu chuẩn thế nào là một chuyên gia vẫn là chủ đề đang bàn cãi và do đó, giữa những chuyên gia có thể có những đánh giá khác nhau về cùng một vấn đề. Nghĩa là tính thống nhất trong các quy tắc, các luật từ kinh nghiệm con người là hạn chế, chưa kể tới tính đúng sai. Thêm vào đó, bản thân chuyên gia đôi khi gặp khó khăn khi diễn đạt tường minh tri thức của mình thành các luật, các quy tắc. Những điều trên khiến cho quá trình thu thập tri thức từ kinh nghiệm con người trở nên rất phức tạp mà bản thân người xây dựng mô hình phải gánh vác.

Hướng tiếp cận thứ hai có cách nhìn hoàn toàn khác khi xây dựng cơ sở tri thức cho mô hình. Dựa trên những dữ liệu thu thập từ thực nghiệm khách quan, các thuật toán khai phá dữ liệu như: phân cụm, phân lớp, các mô hình máy học thống kê, … sẽ được áp dụng để trích rút ra các tri thức, các quy luật hay khuynh hướng dữ liệu để xây dựng cơ sở tri thức. Rõ ràng, việc thu thập các số liệu thực nghiệm dễ dàng hơn nhiều so với thu thập tri thức của các chuyên gia. Xét đến cùng, kinh nghiệm của chuyên gia cũng được hình thành tích lũy từ chính những quan sát trên thực nghiệm.

Thêm vào đó, với một tập dữ liệu thực nghiệm đúng đắn, đầy đủ và toàn diện, tri thức thu được là khách quan và có tính nhất quán cao. Những mô hình mờ được xây dựng theo hướng tiếp cận này gọi là mô hình mờ hướng dữ liệu (data driven fuzzy models). Nhiều nghiên cứu đã được công bố chứng tỏ rằng những mô hình mờ hướng dữ liệu đã mang lại hiệu quả trong việc giải quyết các bài toán nhận dạng, điều khiển, phân tích dự đoán, … dựa vào các kỹ thuật phân cụm, phân lớp, hay hồi quy.

Hầu hết các mô hình mờ hướng dữ liệu đều được xây dựng dựa trên các thuật toán học tự động từ dữ liệu [2][15][17][19][21][24][29][49][56][78]. Hiệu quả của các mô hình này phụ thuộc nhiều vào tình trạng tập dữ liệu huấn luyện (training data set) và mô hình đầu ra đôi khi không phù hợp với thực tế, thiếu tính phổ quát,… Đồng thời, những mô hình mờ hướng dữ liệu được xem như những mô hình "hộp đen", con người khó có thể hiểu được một cách tường minh các quy tắc và lập luận bên trong mô hình. Chính vì những lý do đó, việc làm sáng tỏ tập quy tắc mờ của mô hình để các chuyên gia có thể hiểu và diễn dịch được các quy tắc, qua đó có thể lựa chọn, hiệu chỉnh, bổ sung để làm tăng hiệu quả sử dụng của các mô hình mờ hướng dữ liệu, cũng như tối ưu hóa những mô hình mờ hướng dữ liệu là thách thức mới của những nhà nghiên cứu xây dựng mô hình mờ.

Xu hướng nghiên cứu các giải pháp cho phép tích hợp các kiểu khác nhau của tri thức tiên nghiệm (a priori knowledge) vào mô hình hướng dữ liệu đã được nhiều nhà khoa học quan tâm nghiên cứu. Trong [74], Tulleken H. đã đề xuất sử dụng những kỹ thuật Bayes để tích hợp dữ liệu thô với tri thức liên quan đến điều khiển hệ thống. Tiếp theo đó, Abonyi J. và những đồng sự đã áp dụng kỹ thuật tương tự đối với việc tích hợp tri thức và mô hình mờ TSK [12]. Trong [33], Jang J.-S. R. cũng đã đề xuất một hướng tiếp cận cho việc tối ưu hóa mô hình mờ hướng dữ liệu bằng mạng nơ-ron. Kỹ thuật này có thể được xem như là một sự tích hợp của tri thức qui nạp (inductive knowledge) với mô hình hướng dữ liệu. Trong lĩnh vực dự báo u não, Weibei Dou và những đồng sự [89] đã đề xuất một phương pháp để tự động hóa việc phân đoạn các khối u não của con người từ hình ảnh đa phổ cộng hưởng từ (MRI). Phương pháp này cho phép kết hợp tri thức kinh nghiệm của các chuyên gia trong

lĩnh vực X quang để xây dựng một mô hình mờ. Trong lĩnh vực nhận dạng, V.A. Parasich cùng với đồng sự đã phân tích các loại tri thức tiên nghiệm và giải pháp tích hợp tri thức tiên nghiệm để tăng hiệu quả nhận dạng [75]. Những kết quả nghiên cứu trên chỉ mới đề cập đến một phương pháp học cụ thể, cho phép tích hợp một kiểu tri thức cụ thể hoặc thuộc một miền cụ thể nào đó; chưa có giải pháp cho việc tích hợp các kiểu khác nhau của tri thức nhằm cải thiện hiệu quả của mô hình mờ hay hiệu quả dự báo nói chung.

L. Martin và những đồng sự đã đề xuất kỹ thuật kết hợp mô hình số và mô hình ngữ nghĩa dưới dạng các luật mờ trong việc tạo quyết định nhóm [43]. Bằng cách xây dựng các mô-đun cho phép chuyển đổi qua lại giữa các luật mờ dạng số và dạng ngữ nghĩa, hệ thống cho phép các chuyên gia với sự hiểu biết ở dạng số hay dạng ngữ nghĩa và trong nhiều lĩnh vực khác nhau có thể kết hợp cùng nhau hình thành một hệ quyết định nhóm. Serge Guillaume và Luis Magdalena đã đề xuất một giải pháp cho phép tích hợp các luật mờ sản xuất từ dữ liệu thô với các luật mờ thu thập được từ các chuyên gia để hình thành một mô hình lai ghép [65]. Mô hình này cho phép các chuyên gia đồng bộ hóa các phân vùng mờ (fuzzy partitions) của mô hình mờ hướng dữ liệu với mô hình ngữ nghĩa của các chuyên gia, qua đó có thể tích hợp các luật của chuyên gia với tập luật mờ hướng dữ liệu để hình thành một mô hình mờ hợp nhất. Mark Steyver và những đồng sự đã đề xuất phương pháp kết hợp tri thức của chuyên gia với mô hình học thống kê dựa trên dữ liệu để xây dựng một mô hình lai ghép [49]. Những nghiên cứu trên nhằm hướng đến cải thiện được hiệu quả của mô hình nhờ kết hợp cả tri thức chuyên gia và việc học từ dữ liệu. Các mô hình tích hợp này cho phép khai thác ưu điểm của mỗi hướng tiếp cận trong việc xây dựng mô hình mờ, đó là sự chính xác, tinh túy của các luật thu thập được từ chuyên gia, sự đa dạng và nhanh chóng của các luật được trích xuất từ dữ liệu. Tuy nhiên, một trong những điểm hạn chế của mô hình mờ hướng dữ liệu gây nên sự khó khăn cho việc tích hợp với tri thức chuyên gia đó là mô hình mờ hướng dữ liệu thiếu "tính trong suốt" (transparence) hay "tính có thể diễn dịch được" (interpretability). Các chuyên gia không thể phân

tích, diễn giải được các tập luật mờ hướng dữ liệu, như vậy rất khó để có thể lựa chọn, kết hợp với luật chuyên gia.

Trong [65], Serge Guillaume và đồng sự đã đề cập đến việc cải thiện "tính có thể diễn dịch được" của mô hình mờ hướng dữ liệu bằng cách tích hợp phân vùng mờ với định nghĩa ngữ nghĩa của chuyên gia; tuy nhiên ngay cả các phân vùng mờ của mô hình mờ hướng dữ liệu cũng không rõ ràng và những chuyên gia cũng khó có thể nhận biết và phân tích được. Điều này cũng được J.L Castro đề cập đến trong [36] nhưng cũng chưa có giải pháp cụ thể để giải quyết.

Vấn đề phát triển mô hình mờ hướng dữ liệu từ máy học véc-tơ hỗ trợ (SVM - Support Vector Machines) cũng được nhiều tác giả quan tâm nghiên cứu. Trong [35], Chiang J. H. và Hao P.Y. đã giới thiệu một hướng tiếp cận cho phép tích hợp mô hình suy luận mờ với máy học véc- tơ hỗ trợ. Theo hướng tiếp cận này, nhiều công trình nghiên cứu đã đề xuất và ứng dụng các kỹ thuật rút trích các luật mờ từ SVM cho việc phát triển các mô hình mờ hướng dữ liệu cho các bài toán phân lớp, dự báo hồi quy [15][19][21][36][54][63][77]. Trong [24][36][56], các tác giả đã nghiên cứu cơ bản và chi tiết về kỹ thuật trích xuất các luật mờ dựa trên SVM. Việc kết hợp mô hình mờ với SVM đã phần nào khắc phục được điểm hạn chế ở tính chất "hộp đen" của các mô hình dự đoán bằng máy học thống kê, cụ thể ở đây là máy học véc-tơ hỗ trợ. Mô hình suy luận mờ đóng vai trò như là cầu nối trung gian giữa mô hình dự đoán theo máy học thống kê và các chuyên gia; kết quả học máy được chuyển sang trình bày ở dạng các luật mờ đã phần nào giúp cho các chuyên gia dễ hiểu hơn các mô hình dự đoán theo máy học thống kê. Tuy nhiên, cũng giống với các mô hình mờ hướng dữ liệu khác, một trong những thách thức đặt ra đối với mô hình mờ dựa trên SVM là làm thế nào đảm bảo tính diễn dịch. Bên cạnh đó, tính chính xác của mô hình máy học véc-tơ hỗ trợ tỷ lệ thuận với kích thước dữ liệu huấn luyện, đồng thời tính chất ngẫu nhiên của dữ liệu cũng quyết định hiệu quả huấn luyện, đây cũng là những thách thức đặt ra cho các nhà nghiên cứu.

Những nghiên cứu cải tiến nhằm nâng cao hiệu quả ứng dụng của SVM và mô hình mờ dựa trên SVM có thể tìm thấy trong [6][7][8][11][19][21][26][59][60]

[87][90]. Theo đó, bên cạnh những kỹ thuật cải tiến nhằm tối ưu hóa tham số, thuật toán học SVM, nhiều giải pháp đề xuất xây dựng mô hình lai ghép để cải thiện hiệu quả ứng dụng mô hình SVM và mô hình mờ dựa trên SVM. Trong [8], Nguyễn Đình Thuận và đồng sự đã đề xuất kết hợp mô hình ARIMA và SVM cho một bài toán dự báo cụ thể, hoặc trong [6][26][66][87] các tác giả đã đề xuất giải pháp kết hợp các kỹ thuật phân cụm với SVM để cải thiện hiệu quả của mô hình dự đoán dựa trên SVM. Với giải pháp sử dụng SOM để phân cụm dữ liệu đầu vào, sau đó áp dụng nhiều máy học SVM cho các phân cụm dữ liệu, kết quả thử nghiệm cho bài toán dự đoán giá chứng khoán đã cải thiện đáng kể cả về tốc độ và độ chính xác [26][66].

Bên cạnh đó, lý thuyết học dựa trên tri thức [71] cho thấy các kiểu khác nhau của tri thức tiên nghiệm (a priori knowledge) có thể sử dụng để cải thiện hiệu quả của mô hình máy học nói chung. Tùy thuộc vào vai trò của tri thức tiên nghiệm, việc học dựa trên tri thức có thể phân thành các kịch bản như sau: học dựa trên giải thích (explaination-based learning) hay còn gọi là EBL, học dựa trên sự phù hợp (relevance-based learning) hay còn gọi là RBL, và học qui nạp dựa trên tri thức (knowledge-based inductive learning) hay còn gọi là KBIL [71]. Lý thuyết này là cơ sở lý luận để chúng ta tin rằng có thể tích hợp các kiểu khác nhau của tri thức tiên nghiệm để cải thiện được hiệu quả của mô hình mờ học từ dữ liệu.

Qua tổng hợp và đánh giá những kết quả nghiên cứu về mô hình mờ hướng dữ liệu, giải pháp tích hợp các mô hình máy học khác nhau hoặc các kiểu khác nhau của tri thức tiên nghiệm để cải thiện mô hình, và vấn đề xây dựng mô hình mờ hướng dữ liệu dựa trên máy học Véc-tơ hỗ trợ, cho thấy: cần thiết phải nghiên cứu giải pháp tích hợp các kiểu khác nhau của tri thức tiên nghiệm vào mô hình mờ hướng dữ liệu trích xuất từ SVM, nhằm cải thiện những hạn chế của mô hình, và đồng thời nghiên cứu xây dựng một mô hình lai ghép dựa trên mô hình mờ hướng dữ liệu để giải quyết bài toán dự báo thực tế.

2. Mục tiêu nghiên cứu

Mục tiêu nghiên cứu của luận án là: Xây dựng mô hình mờ hướng dữ liệu lai ghép dựa trên việc tích hợp tri thức tiên nghiệm với mô hình mờ hướng dữ liệu cho bài toán dự báo hồi quy. Cụ thể, nghiên cứu những nội dung chủ yếu sau:

- Nghiên cứu phương pháp xây dựng mô hình mờ từ dữ liệu (mô hình mờ hướng dữ liệu), và cụ thể là xây dựng mô hình mờ dựa trên máy học véc-tơ hỗ trợ.
- Nghiên cứu phương thức cho phép tích hợp các kiểu khác nhau của tri thức tiên nghiệm trong mô hình mờ hướng dữ liệu dựa trên máy học véc-tơ hỗ trợ.
- Đề xuất mô hình lai ghép trên cơ sở mô hình mờ hướng dữ liệu trích xuất từ máy học véc-tơ hỗ trợ cho bài toán dự báo hồi quy và áp dụng để giải quyết bài toán dự báo dữ liệu chuỗi thời gian tài chính.

3. Cách tiếp cận và phương pháp nghiên cứu

Luận án tập trung tiếp cận trên 3 phương pháp chính:

- Phương pháp tổng hợp và phân tích: Tìm kiếm, thu thập, tổng hợp và phân tích các tài liệu về các công trình nghiên cứu đã công bố, các bài báo đăng ở các hội thảo và tạp chí lớn trong nước và quốc tế để đưa ra giải pháp xây dựng mô hình mờ hướng dữ liệu cho bài toán dự báo hồi quy và giải pháp cho phép tích hợp tri thức tiên nghiệm vào mô hình mờ hướng dữ liệu.
- Phương pháp mô hình hóa: Dựa trên kỹ thuật xây dựng mô hình mờ hướng dữ liệu và giải pháp tích hợp tri thức tiên nghiệm vào mô hình mờ hướng dữ liệu để đề xuất thuật toán xây dựng mô hình mờ giải quyết bài toán dự báo hồi quy.
- Phương pháp thực nghiệm, đánh giá kết quả và rút ra kết luận: Sử dụng phần mềm Matlab và các công cụ hỗ trợ về mô hình suy luận mờ và máy học véc-tơ hỗ trợ để cài đặt chương trình thực nghiệm; thực nghiệm trên dữ liệu thực tế, so sánh với kết quả của các mô hình khác đã được công bố để đánh giá và rút ra kết luận.

4. Phạm vi và đối tượng nghiên cứu

Luận án xác định phạm vi và những đối tượng nghiên cứu chính sau:

- Nghiên cứu về các phương pháp xây dựng mô hình mờ từ dữ liệu. Phân tích các vấn đề chi tiết bao gồm xác định cấu trúc mô hình, kỹ thuật xây dựng mô hình, tối ưu hóa mô hình, …
    - Các mô hình dựa trên luật mờ (Fuzzy rule-based models): Mamdani, TSK;
    - Trích xuất mô hình mờ TSK từ dữ liệu dựa vào máy học véc-tơ hỗ trợ - thuật toán f-SVM (SVM-based fuzzy models);
    - Tối ưu hóa các tham số của mô hình mờ hướng dữ liệu: thuật toán di truyền (GA), thuật toán Gradient decent;
    - Triển khai thực nghiệm và đánh giá hiệu quả mô hình mờ hướng dữ liệu dựa trên máy học véc-tơ hỗ trợ.
- Nghiên cứu giải pháp cải thiện hiệu quả của mô hình mờ hướng dữ liệu bằng cách tích hợp tri thức tiên nghiệm.
    - Các kịch bản tích hợp tri thức có trước vào mô hình máy học cho phép cải thiện hiệu quả mô hình máy học nói chung và mô hình mờ nói riêng:
        - Explanation-based learning (EBL);
        - Relevance-based learning (RBL);
        - Knowledge-based inductive learning (KBIL).
    - Phân tích các điều kiện để một hệ thống mờ "có thể diễn dịch được" (interpretable) và xét trong trường hợp cụ thể của mô hình mờ dựa trên máy học véc-tơ hỗ trợ. Từ đó xác định các tri thức tiên nghiệm cụ thể để tích hợp vào mô hình mờ dựa trên máy học véc-tơ hỗ trợ;
    - Đề xuất và triển khai thực nghiệm thuật toán trích xuất tập luật mờ dựa trên máy học véc-tơ hỗ trợ có tích hợp tri thức tiên nghiệm – thuật toán SVM-IF (SVM-based Interpretable Fuzzy models).

- Nghiên cứu giải pháp lai ghép kỹ thuật phân cụm (SOM, k-Means) với mô hình mờ hướng dữ liệu dựa trên máy học véc-tơ hỗ trợ để giải quyết bài toán dự báo dữ liệu chuỗi thời gian
    - Nghiên cứu xây dựng mô hình mờ dự báo hồi quy cho bài toán dự báo dữ liệu chuỗi thời gian;
    - Đề xuất mô hình mờ lai ghép kỹ thuật phân cụm với mô hình mờ hướng dữ liệu để giải quyết bài toán dự báo dữ liệu chuỗi thời gian;
    - Áp dụng mô hình lai ghép đề xuất để giải quyết bài toán dự báo dữ liệu chuỗi thời gian tài chính.

5. Đóng góp của luận án

Đóng góp của luận án tương ứng với 3 mục tiêu chính đã đề ra như sau:

Thứ nhất, đề xuất thuật toán f-SVM để trích xuất tập luật mờ từ dữ liệu huấn luyện dựa vào máy học vé-tơ hỗ trợ hồi quy. Quy trình trích xuất tập luật mờ có cho phép lựa chọn giá trị tham số epsilon phù hợp thông qua thực nghiệm bằng cách sử dụng tập dữ liệu xác thực.

Thứ hai, đề xuất thuật toán SVM-IF cho phép trích xuất tập luật mờ từ dữ liệu huấn luyện dựa vào máy học véc-tơ hỗ trợ hồi quy có tích hợp tri thức tiên nghiệm. Thuật toán là giải pháp tích hợp tri thức tiên nghiệm vào quá trình trích xuất tập luật mờ từ dữ liệu để đảm bảo tính có thể diễn dịch được của tập luật.

Thứ ba, đề xuất mô hình lai ghép kỹ thuật phân cụm với mô hình mờ hướng dữ liệu dựa trên máy học véc-tơ hỗ trợ hồi quy để giải quyết bài toán dự báo dữ liệu chuỗi thời gian. Mô hình đề xuất được áp dụng để giải quyết bài toán dự báo dữ liệu chuỗi thời gian tài chính.

6. Bố cục của luận án

Luận án gồm phần Mở đầu, 3 chương nội dung và phần Kết luận.

Phần Mở đầu của luận án trình bày tổng quan những nội dung nghiên cứu của luận án, bao gồm cả những nghiên cứu liên quan và những thách thức đặt ra trong vấn đề nghiên cứu. Ở 3 chương nội dung tiếp theo của luận án sẽ trình bày 3 kết quả

nghiên cứu của luận án, đồng thời trong đó có giới thiệu những kiến thức cơ sở có liên quan đến các nội dung nghiên cứu.

Chương 1 trình bày kết quả nghiên cứu trích xuất mô hình mờ từ dữ liệu dựa trên máy học véc-tơ hỗ trợ hồi quy. Xây dựng thuật toán f-SVM cho phép trích xuất tập luật mờ hướng dữ liệu dựa vào máy học véc-tơ hỗ trợ hồi quy, trong đó có đề xuất giải pháp lựa chọn giá trị tham số epsilon tối bằng cách sử dụng tập dữ liệu xác thực. Những cơ sở lý thuyết về xây dựng mô hình mờ hướng dữ liệu, mô hình mờ TSK và máy học véc-tơ hỗ trợ hồi quy cũng được giới thiệu ở chương này, làm cơ sở để đưa ra thuật toán trích xuất tập luật mờ từ máy học véc-tơ hỗ trợ hồi quy. Thuật toán và mô hình đề xuất của Luận án được thực nghiệm và đánh giá kết quả trên một số bài toán ví dụ cụ thể.

Nội dung của Chương 2 liên quan đến kết quả nghiên cứu về giải pháp tích hợp tri thức tiên nghiệm để cải thiện mô hình mờ hướng dữ liệu và đề xuất thuật toán SVM-IF. Thuật toán này, với giải pháp tích hợp tri thức tiên nghiệm, cho phép trích xuất được tập luật mờ có thể diễn dịch được từ máy học véc-tơ hỗ trợ. Ở chương này, những kiểu khác nhau của tri thức tiên nghiệm và những kịch bản khác nhau của việc tích hợp tri thức tiên nghiệm trong việc huấn luyện một mô hình mờ được nghiên cứu trình bày. Cụ thể với trường hợp trích xuất mô hình mờ dựa vào máy học véc-tơ hỗ trợ, những tri thức tiên nghiệm cụ thể được xác định và lựa chọn để tích hợp, từ đó xây dựng thuật toán. Thuật toán đề xuất được thực nghiệm trên một số bài toán ví dụ, có so sánh kết quả với các thuật toán và mô hình trước đó.

Chương 3 trình bày giải pháp lai ghép kỹ thuật phân cụm với mô hình mờ trích xuất dựa vào máy học véc-tơ hỗ trợ để giải quyết bài toán dự báo dữ liệu chuỗi thời gian. Trong chương này bài toán dự báo dữ liệu chuỗi thời gian cụ thể được khảo sát và lựa chọn là bài toán dự báo giá cổ phiếu với các bước tiền xử ký dữ liệu, lựa chọn biến đầu vào và biến đầu ra, đồng thời xác định các tham số đánh giá hiệu quả của mô hình dự báo. Kỹ thuật phân cụm SOM được lựa chọn để phân cụm dữ liệu đầu vào nhằm giải quyết vấn đề dữ liệu lớn và hạn chế nhiễu trong dữ liệu huấn luyện. Mô hình thực nghiệm trên cả hai thuật toán f-SVM và SVM-IF với cùng tập dữ liệu

thực tế thu thập từ sàn chứng khoán quốc tế để có sự so sánh và đánh giá. Quá trình thực nghiệm cũng đồng thời được thực hiện trên cùng bộ dữ liệu với các mô hình đề xuất trước đó bởi các tác giả khác.

Phần kết luận trình bày tóm tắt những đóng góp chính của luận án về ý nghĩa khoa học và thực tiễn. Đồng thời chỉ ra những điểm tồn tại trong vấn đề nghiên cứu và một số định hướng nghiên cứu tiếp theo.

Chương 1. TRÍCH XUẤT MÔ HÌNH MỜ HƯỚNG DỮ LIỆU DỰA TRÊN MÁY HỌC VÉC-TƠ HỖ TRỢ

Chương này trình bày kết quả xây dựng thuật toán f-SVM và quy trình trích xuất mô hình mờ TSK từ dữ liệu dựa trên máy học véc-tơ hỗ trợ. Để làm cơ sở cho việc phân tích sự tương đương của máy học véc-tơ hỗ trợ hồi quy và mô hình mờ TSK, một số vấn đề cơ bản về lý thuyết tập mờ, đặc biệt là mô hình mờ TSK và lý thuyết cơ bản về máy học véc-tơ hỗ trợ phân lớp và hồi quy cũng được trình bày ở những mục đầu chương. Phần cuối chương là nội dung triển khai thực nghiệm cho thuật toán đề xuất.

1.1. Cơ bản về logic mờ

1.1.1. Lý thuyết tập mờ

Như chúng ta đã biết, tập hợp thường là kết hợp của một số phần tử có cùng một số tính chất chung nào đó. Ví dụ: tập các người giới tính nam. Ta có: $T = \{t/t\ là\ người\ giới\ tính\ nam\}$ Vậy, nếu một người nào đó có giới tính nam thì thuộc tập T, ngược lại là không thuộc tập T. Tuy nhiên, trong thực tế cuộc sống cũng như trong khoa học kỹ thuật có nhiều khái niệm không được định nghĩa một cách rõ ràng. Ví dụ, khi nói về một "nhóm những người già", thì thế nào là già? Khái niệm về già không rõ ràng vì có thể người có tuổi bằng 70 là già, cũng có thể tuổi bằng 80 cũng là già (dải tuổi là già có thể từ 70 trở lên), ... Nói cách khác, "nhóm những người già" không được định nghĩa một cách tách bạch rõ ràng như khái niệm thông thường về tập hợp. Các phần tử của nhóm trên không có một tiêu chuẩn rõ ràng về tính "thuộc về" (thuộc về một tập hợp nào đó). Đây chính là những khái niệm thuộc về tập mờ.

Lý thuyết tập mờ lần đầu tiên được Lotfi A. Zadeh, giáo sư thuộc trường Đại học California tại Berkley, giới thiệu trong một công trình nghiên cứu vào năm 1965 [5][84]. Ý tưởng nổi bật của Zadeh là đề nghị đánh giá khả năng một phần tử x là

thành viên của một tập A trong tập vũ trụ X, bằng cách xây dựng một ánh xạ hàm gọi là hàm thành viên (membership function) [5][84][85][86], ký hiệu như sau:

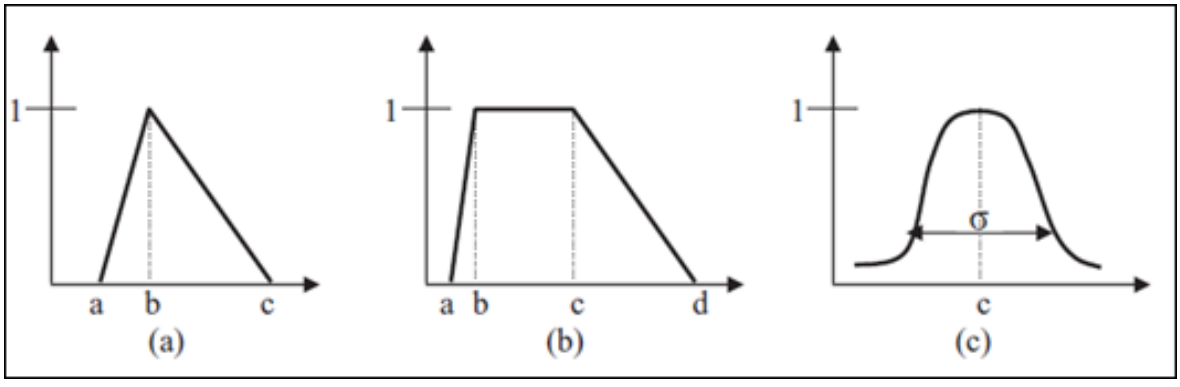
$$\mu_A: X \to [0,1]$$

Hàm thành viên $\mu_A(x)$ định nghĩa cho tập A trên tập vũ trụ X trong khái niệm tập hợp kinh điển chỉ có hai giá trị là 1 nếu $x \in A$ hoặc 0 nếu $x \notin A$. Tuy nhiên trong khái niệm tập mờ thì giá trị hàm thành viên chỉ mức độ thuộc về (membership degree) của phần tử x vào tập mờ A. Khoảng xác định của hàm $\mu_A(x)$ là đoạn [0, 1], trong đó giá trị 0 chỉ mức độ không thuộc về, còn giá trị 1 chỉ mức độ thuộc về hoàn toàn.

Theo đó, tập mờ được định nghĩa như sau [5][18][37]:

Định nghĩa 1.1. Cho một tập vũ trụ X với các phần tử ký hiệu bởi x, $X = \{x\}$. Một tập mờ A trên X là tập được đặc trưng bởi một hàm $\mu_A(x)$ mà nó liên kết mỗi phần tử $x \in X$ với một số thực trong đoạn $[0,1]$. Trong đó $\mu_A(x)$ là một ánh xạ từ X vào $[0,1]$ và được gọi là hàm thành viên của tập mờ A.

Kiểu của tập mờ phụ thuộc vào các kiểu hàm thành viên khác nhau. Đã có nhiều kiểu hàm thành viên khác nhau được đề xuất. Một số kiểu hàm thành viên sử dụng phổ biến trong logic mờ như sau (xem Hình 1.1) [18][37]:



Hình 1.1. Đồ thị của 3 hàm thành viên phổ biến:
(a) tam giác, (b) hình thang, (c) Gauss

Dạng tam giác (Triangles): Hàm thành viên này được xác định bởi 3 tham số là cận dưới a, cận trên c và giá trị b (ứng với đỉnh tam giác), với $a < b < c$. Hàm

thành viên này được gọi là đối xứng nếu nếu giá trị $b - a$ bằng giá trị $c - b$, hay $b = (a + b)/2$. Công thức xác định hàm thành viên tam giác như sau:

$$triangle(x; a, b, c) = \begin{cases} 0 & x < a \\ (x-a)/(c-a) & a \le x \le c \\ (b-x)/(b-c) & c \le x \le b \\ 0 & x > b \end{cases} \quad (1.1)$$

Dạng hình thang (Trapezoids): Hàm thành viên này được xác định bởi bộ 4 giá trị a, b, c, d, với $a < b < c < d$, theo công thức sau:

$$trapezoid(x; a, b, c, d) = \begin{cases} 0 & x < a \\ (x-a)/(b-a) & a \le x < b \\ 1 & b \le x < c \\ (d-x)/(d-c) & c \le x < d \\ 0 & x \ge d \end{cases} \quad (1.2)$$

Dạng Gauss: Hàm thành viên này được xác định bởi 2 tham số, gồm: giá trị c là giá trị trung bình (ứng với giá trị cực đại của hàm thành viên) và σ là độ lệch chuẩn (độ rộng của hàm). Chúng ta có thể điều chỉnh đồ thị hàm thành viên bằng cách thay đổi giá trị tham số σ. Công thức xác định hàm thành viên Gauss như sau:

$$gauss(x; c, \sigma) = exp\left(-\frac{(x-c)^2}{2\sigma^2}\right) \quad (1.3)$$

Bên cạnh đó, các khái niệm, tính chất, phép toán trong lý thuyết tập kinh điển cũng được mở rộng cho các tập mờ [4][5][37][84][85][86]. Theo đó, các phép toán như t-norm, t-conorm, negation và phép kéo theo (implication), ... trong logic mờ được đề xuất, nghiên cứu chi tiết cung cấp cho các mô hình ứng dụng giải các bài toán thực tế.

Trong hầu hết các kỹ thuật phát triển dựa trên lý thuyết tập mờ thì luật mờ "IF-THEN" phát triển và ứng dụng thành công trong khá nhiều lĩnh vực, như: điều khiển, xử lý ảnh, nhận dạng, mô hình hóa hệ thống, …

1.1.2. Luật mờ "IF-THEN"

Những luật mờ "IF-THEN" (hay có thể gọi ngắn gọn là luật mờ - fuzzy rules), là thành phần cơ bản của những hệ thống mờ. Mỗi luật mờ gồm có hai phần: phần IF

(tiền đề - antecedent) và phần THEN (mệnh đề kết luận – consequent), được biểu diễn như sau [37]:

$$IF\ <antecedent>\ THEN\ <consequent>$$

Phần tiền đề là những giá trị ngôn ngữ (linguistic terms) và thường được liên kết bởi liên từ "and". Phần mệnh đề kết luận có thể chia thành 3 kiểu như sau:

1) Kiểu kết luận mờ (fuzzy consequent):

$$IF\ x_1 is\ A_1\ and\ x_2 is\ A_2\ and\ \dots\ and\ x_p is\ A_p\ THEN\ y\ is\ B$$

trong đó A_i, B là những tập mờ.

2) Kiểu kết luận rõ (crisp consequent):

$$IF\ x_1 is\ A_1\ and\ x_2 is\ A_2\ and\ \dots\ and\ x_p is\ A_p\ THEN\ y = b$$

trong đó A_i, B là những tập mờ; b là một giá trị số không mờ hoặc là một giá trị dạng ký hiệu (gọi chung là giá trị rõ).

3) Kiểu kết luận hàm (functional consequent):

$$IF\ x_1 is\ A_1\ and\ x_2 is\ A_2\ and\ \dots\ and\ x_p is\ A_p\ THEN\ y = a_0 + \sum_{i=1}^{p} a_i x_i$$

trong đó A_i là những tập mờ; $a_0, a_1, \dots, a_p$ là những hằng số.

Trường hợp các luật mờ có kết luận là tập mờ thì hệ mờ thuộc dạng hệ mờ Mandani, ngược lại nếu các luật mờ có kết luận là giá trị rõ hoặc hàm thì hệ mờ thuộc dạng hệ mờ TSK [37].

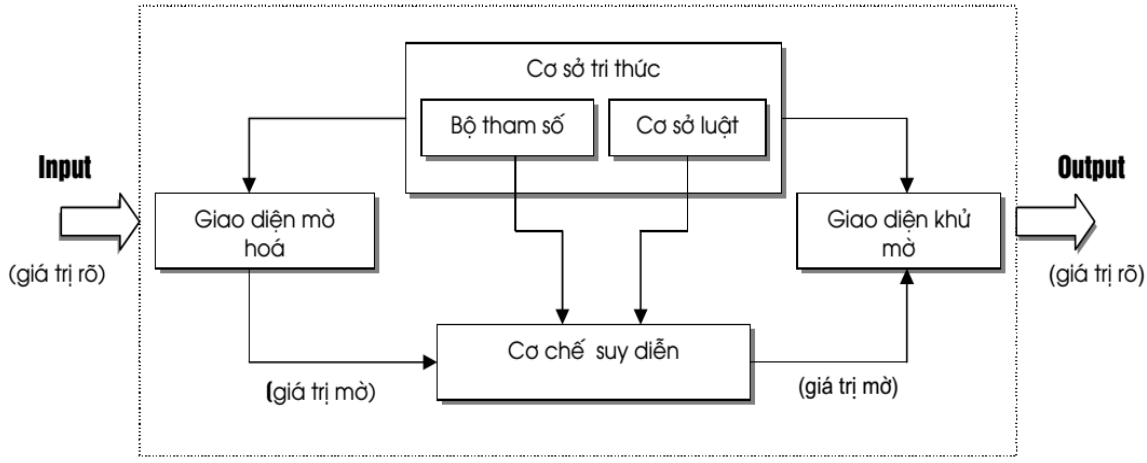
Về cơ bản có hai loại luật mờ đó là luật mờ ánh xạ và luật mờ kéo theo [37]. Luật mờ ánh xạ biểu diễn mối quan hệ ánh xạ hàm giữa những biến đầu vào với những đầu ra, phổ biến là nhiều đầu vào và một đầu ra (Multi Inputs and Single Output – MISO); trong khi luật mờ kéo theo biểu diễn mối quan hệ logic giữa hai biểu thức logic tiền đề và kết luận. Luật mờ kéo theo được thiết kế riêng lẻ và được ứng dụng chủ yếu trong chẩn đoán, ra quyết định ở trình độ cao. Luật mờ ánh xạ được thiết kế thành các tập luật và được ứng dụng phổ biến trong điều khiển, xử lý tín hiệu số, mô hình hóa hệ thống. Trong thực tế, gần như hầu hết các ứng dụng của logic mờ

trong tài chính, công nghiệp đều sử dụng luật mờ ánh xạ, mô hình mờ là hình thức ứng dụng đó.

1.2. Mô hình mờ hướng dữ liệu

Mô hình mờ (fuzzy models) hay cụ thể là mô hình dựa trên các luật mờ (fuzzy rule-based models) là cơ cấu tính toán dựa trên các khái niệm của lý thuyết tập mờ, các tập luật mờ "IF-THEN", cùng với cơ chế suy diễn mờ [4][18][37]. Lý thuyết tập mờ chính là công cụ toán học và logic để thiết lập nên các khâu cơ bản trong hoạt động của một mô hình mờ.

Về tổng thể, mỗi mô hình mờ nói chung đều bao gồm các đầu vào (input), đầu ra (output) cùng với một bộ xử lý. Bộ xử lý thực chất là một ánh xạ biểu diễn sự phụ thuộc của biến đầu ra hệ thống đối với các biến đầu vào. Các biến đầu vào nhận giá trị rõ, đầu ra có thể là một tập mờ hoặc một giá trị rõ. Quan hệ ánh xạ của đầu ra đối với các đầu vào mô hình mờ được mô tả bằng một tập luật mờ, thay vì một hàm số tường minh. Cụ thể hơn, cấu trúc cơ bản của một mô hình mờ bao gồm năm thành phần chủ yếu (Hình 1.2):



Hình 1.2. Cấu trúc cơ bản của một mô hình mờ

- Cơ sở luật (rule base) nơi chứa đựng tập các luật mờ "IF-THEN".
- Bộ tham số mô hình quy định hình dạng hàm thành viên của giá trị ngôn ngữ được dùng để biểu diễn biến mờ và các luật mờ.

- Cơ chế suy diễn (reasoning mechanism) có nhiệm vụ thực hiện thủ tục suy diễn mờ dựa trên cơ sở tri thức và các giá trị đầu vào để đưa ra một giá trị dự đoán ở đầu ra.
- Giao diện mờ hóa (fuzzification interface) thực hiện chuyển đổi các đầu vào rõ thành mức độ trực thuộc các giá trị ngôn ngữ.
- Giao diện khử mờ (defuzzification interface) thực hiện chuyển đổi kết quả suy diễn mờ thành giá trị đầu ra rõ khi cần.

Các mô hình mờ dạng luật có thể chia làm 2 dạng cơ bản tùy theo dạng luật mờ được sử dụng, đó là mô hình mờ dạng Mamdani và mô hình mờ dạng Takagi-Sugeno [4][37].

1.2.1. Mô hình mờ Mamdani

Mô hình mờ dạng Mamdani được đề xuất với mục tiêu ban đầu là điều khiển tổ hợp nồi hơi và động cơ hơi nước thông qua một tập luật dạng ngôn ngữ thu được từ những thao tác viên có kinh nghiệm [37][50][51]. Đây là dạng mô hình điển hình nhất, với bộ luật bao gồm các luật mờ mà phần tiền đề và phần kết luận đều là các tập mờ và biểu diễn bởi một hàm thành viên giải tích. Trong dạng này, có hai phương pháp lập luận được xây dựng: Phương pháp thứ nhất, theo truyền thống, xem mỗi luật là một quan hệ mờ và kết nhập chúng thành một quan hệ mờ chung R, đóng vai trò là một toán tử. Lập luận tức là tìm kiếm đầu ra B' cho mỗi đầu vào A', $B' = R(A')$. Với rất nhiều cách chọn các phép T-norm, T-conorm cho các kết nối AND, OR và phép kéo theo để tính toán, mỗi cách chọn như vậy sẽ cho kết quả B' khác nhau. Nhìn chung không thể nói cách chọn các phép toán như thế nào là tốt nhất mà phụ thuộc vào từng bài toán cụ thể và trực quan cảm nhận của người giải bài toán đó. Điều này rất phù hợp với lập luận xấp xỉ và tạo tính mềm dẻo trong ứng dụng của phương pháp. Trong phương pháp lập luận thứ hai, mỗi luật mờ được xem như một điểm trong không gian ngôn ngữ, xây dựng các ánh xạ định lượng ngữ nghĩa cho các giá trị ngôn ngữ để chuyển các điểm đó về không gian thực tạo thành một "siêu lưới". Thực hiện nội suy trên siêu lưới này để tìm kết quả đầu ra đối với một đầu vào cho trước.

Với mô hình mờ Mamdani, các luật mờ ngôn ngữ được biểu diễn như sau:

$$R_j: IF\ x_1\ is\ A_1^j\ and\ x_2\ is\ A_2^j\ and\ \ldots\ and\ x_p\ is\ A_p^j\ THEN\ y\ is\ C_j\ , j\ =\ 1, 2, \ldots, m$$

trong đó m là số lượng các luật mờ, $x_i \in U_i (i = 1,2, \ldots p)$ là các biến điều kiện đầu vào; $y\ \in V$ là các biến quyết định đầu ra; A_i^j và C_j là những tập mờ (cũng chính là những giá trị ngôn ngữ) được xác định bởi hàm thành viên tương ứng $\mu_{A_i^j}(x_i)$ và $\mu_{C_j}(y)$ tương ứng.

Giả sử các giá trị vào cho mô hình mờ Mamdani có dạng:

$$x_1\ is\ A_1', x_2\ is\ A_2', \ldots, x_p\ is\ A_p'$$

với $A_1', A_2', \ldots, A_p'$ là những tập mờ con của các tập nền $U_1, U_2, \ldots, U_p$.

Khi đó, đóng góp của luật mờ R_j trong đầu ra của mô hình mờ Mamdani là một tập mờ với hàm thành viên được tính bằng toán tử "min" theo công thức:

$$\mu_{c_j'}(y) = \left(\alpha_1^j \wedge \alpha_2^j \wedge \ldots \wedge \alpha_n^j\right) \wedge \mu_{C_j}(y) \qquad (1.4)$$

với α^j là độ phù hợp (matching degree) của luật R_j, và α_i^j là độ phù hợp giữa giá trị đầu vào x_i và biến điều kiện x_i của luật R_j.

$$\alpha_i^j = sup_{x_i}\left(\mu_{A_i'}(x_i) \wedge\ \mu_{A_i^j}(x_i)\right) \qquad (1.5)$$

với $\wedge$ là ký hiệu cho toán tử "min".

Cuối cùng đầu ra của mô hình mờ là tập hợp những đầu ra của tất cả các luật được tính bằng cách áp dụng toán tử "max", theo công thức sau:

$$\mu_{c'}(y) = max\{\mu_{c_1'}(y), \mu_{c_2'}(y), \ldots, \mu_{c_m'}(y)\} \qquad (1.6)$$

hay viết cách khác là:

$$\mu_{c'}(y) = \mu_{c_1'}(y) \vee\ \mu_{c_2'}(y) \vee\ \ldots \vee\ \mu_{c_m'}(y) \qquad (1.7)$$

với $\vee$ là ký hiệu cho toán tử "max".

Đầu ra của mô hình mờ Mamdani là một tập mờ và cần phải được giải mờ để có được kết quả là một giá trị rõ cần thiết.

1.2.2. Mô hình mờ TSK

Mô hình mờ dạng TSK (Takagi, Sugeno and Kang), còn được gọi là mô hình Takagi-Sugeno, được đề xuất bởi Takagi, Sugeno, và Kang nhằm phát triển cách tiếp cận mang tính hệ thống đối với quá trình sinh luật mờ từ tập dữ liệu vào-ra cho trước [37][70]. Mô hình mờ TSK được cấu thành từ một tập các luật mờ với phần kết luận của mỗi luật này là một hàm (không mờ), ánh xạ từ các tham số đầu vào của mô hình tới tham số đầu ra. Tham số của các hàm ánh xạ này có thể được đánh giá thông qua các thuật toán nhận dạng, như phương pháp bình phương nhỏ nhất hay bộ lọc Kalman. Các phương pháp lập luận cũng được xây dựng trong dạng này: Thứ nhất, luật nào phù hợp hơn với dữ liệu đầu sẽ được chọn và kết quả lập luận là phần kết luận của luật đó. Đây gọi là phương pháp lập luận single-winner-rule. Thứ hai, các luật đóng vai trò "bầu cử" (vote) cho mẫu dữ liệu đối với lớp của vế phải luật dựa trên độ phù hợp của luật đối với dữ liệu đó, lớp nào có tổng độ phù hợp cao nhất sẽ được dùng để phân lớp cho dữ liệu đầu vào tương ứng. Phương pháp lập luận này gọi là weighted-vote. Hệ luật mờ dạng Tagaki-Sugeno cùng với hai phương pháp lập luận single-winner-rule và weighted-vote khá trực quan, không phải khử mờ kết quả lập luận, rất phù hợp trong việc xây dựng các mô hình ứng dụng của một số bài toán trong khai phá dữ liệu như nhiều tác giả đã nghiên cứu [4][19][30][33][36][55][58] [77][79][80].

Với mô hình mờ TSK, các luật mờ "IF – THEN" dạng TSK, là cơ sở của phép suy luận mờ [37][69][70]. Luật mờ TSK được biểu diễn như sau:

$$R_j: IF\ x_1\ is\ A_1^j\ and\ x_2\ is\ A_2^j\ and\ \dots\ and\ x_p\ is\ A_p^j$$

$$THEN\ y = g_j(x_1, x_2, \dots, x_p)\ , với\ j\ =\ 1, 2, \dots, m$$

Trong đó $x_i (i = 1,2, \dots p)$ là các biến điều kiện đầu vào của luật mờ R_j; y là biến quyết định đầu ra, và được xác định bởi hàm không mờ $g_j(.)$ của các biến x_i;

A_i^j là những giá trị ngôn ngữ (những tập mờ) được xác định bởi các hàm thành viên tương ứng $\mu_{A_i^j}(x_i)$.

Việc tính toán giá trị đầu ra của mô hình mờ TSK khi thực hiện suy luận được thực hiện theo công thức sau:

$$y = \frac{\sum_{j=1}^{m} \alpha^j g_j(x_1, x_2, \dots, x_p)}{\sum_{j=1}^{m} \alpha^j}, \tag{1.8}$$

trong đó α^j là độ phù hợp của luật R_j và được tính toán tương tự như với mô hình mờ Mamdani bằng công thức (1.5).

Những giá trị đầu vào cho mô hình TSK là những giá trị số (không mờ), cụ thể là: $x_1 = a_1, x_2 = a_2, \dots, x_p = a_p$, như vậy độ so khớp của mỗi luật mờ R_j được tính toán bằng cách sử dụng toán tử "min" như sau:

$$\alpha^j = min\left(\mu_{A_1^j}(a_1), \mu_{A_2^j}(a_2), \dots, \mu_{A_p^j}(a_p)\right). \tag{1.9}$$

Tuy nhiên ta cũng có thể dùng toán tử nhân (phép tích) để tính độ so khớp như sau:

$$\alpha^j = \mu_{A_1^j}(a_1) \times \mu_{A_2^j}(a_2) \times \dots \times \mu_{A_p^j}(a_p) = \prod_{i=1}^{p} \mu_{A_i^j}(a_i). \tag{1.10}$$

Ví dụ xét mô hình mờ TSK gồm có 3 luật như sau:

$$IF\ x\ is\ Small\ THEN\ z = L1(x),$$

$$IF\ x\ is\ Medium\ THEN\ z = L2(x),$$

$$IF\ x\ is\ Large\ THEN\ z = L3(x).$$

Đầu ra của mô hình được tính toán như sau:

$$y = \frac{\mu_{Small}(x) \times L1(x) + \mu_{Medium}(x) \times L2(x) + \mu_{Large}(x) \times L3(x)}{\mu_{Small}(x) + \mu_{Medium}(x) + \mu_{Large}(x)}. \tag{1.11}$$

Về nguyên tắc, $g_j(.)$ có thể là một hàm dạng đa thức tùy ý. Tuy nhiên, trong thực tế $g_j(.)$ thường được chọn là một hàm tuyến tính, khi đó mô hình mờ được gọi

là mô hình mờ TSK bậc-1 (first-order TSK model) [37]. Hàm $g_j(.)$ tuyến tính có dạng:

$$g_j(x_1, x_2, \dots, x_p) = b_{j0} + b_{j1}x_1 + \cdots + b_{jp}x_p \,. \tag{1.12}$$

Một trường hợp đặc biệt là hàm $g_j(.)$ được chọn là hằng số, khi đó mô hình mờ được gọi là mô hình mờ TSK bậc-0 (zero-order TSK model), và hàm $g_j(.)$ có dạng:

$$g_j(.) = b_j \,. \tag{1.13}$$

Khi đó đầu ra của mô hình mờ TSK bậc-0 được tính toán bởi công thức:

$$y = \frac{\sum_{j=1}^{m} \alpha^j b_j}{\sum_{j=1}^{m} \alpha^j}. \tag{1.14}$$

Quá trình suy luận dựa trên mô hình mờ TSK được thực hiện như sau:

Bước 1. Kích hoạt các giá trị thành viên. Giá trị thành viên của các biến đầu vào được tính toán theo công thức nhân như sau:

$$\prod_{i=1}^{p} \mu_{A_i^j}(x_i) \,. \tag{1.15}$$

Bước 2. Tính kết quả đầu ra của hàm suy luận mờ theo công thức sau:

$$f(x) = \frac{\sum_{j=1}^{m} \overline{z}^j \prod_{i=1}^{p} \mu_{A_i^j}(x_i)}{\sum_{j=1}^{m} \prod_{i=1}^{p} \mu_{A_i^j}(x_i)}. \tag{1.16}$$

Trong đó, $\overline{z}^j$ là giá trị đầu ra của hàm $g_j(.)$ tương ứng với mỗi luật mờ. $f(x)$ được gọi là hàm quyết định đầu ra của mô hình mờ TSK.

Mô hình mờ TSK với ưu điểm có thể thể hiện các hành vi cục bộ của hệ thống được ứng dụng và không cần giải mờ sau khi lập luận bởi vì tập luật mờ của mô hình có phần kết luận của các luật là một hàm rõ [37]. Hơn nữa, trong các nghiên cứu của J.L. Castro, Ouahib Guenounoua, Volkan Uslan, … [36][58][77], cho thấy việc sử dụng các luật mờ có phần kết luận chỉ là các hàm rõ đã mang lại những kết quả rất khả quan. Đây là những lý do thúc đẩy những nghiên cứu tiếp tục về các mô hình ứng dụng hệ luật mờ TSK.

1.3. Sinh luật mờ từ dữ liệu

Vấn đề nghiên cứu xây dựng các mô hình mờ dạng luật dựa trên dữ liệu ứng dụng cho các bài toán nhận dạng mẫu và phân lớp (classification), dự báo và hồi quy (regression), phân cụm (clustering), … đã được rất nhiều tác giả quan tâm nghiên cứu. Từ năm 1985, Sugeno đã đề xuất phương pháp xây dựng mô hình mờ từ dữ liệu số hay còn gọi là dữ liệu thô [69][70], và phương pháp này đã thật sự chứng tỏ được hiệu quả trong việc phát triển các mô hình mờ. Đã có rất nhiều nghiên cứu đề xuất các kỹ thuật khác nhau để xây dựng mô hình mờ hướng dữ liệu.

Về cơ bản, bài toán sinh tập luật mờ từ dữ liệu vào - ra có thể mô tả tóm tắt như sau: Cho N cặp dữ liệu vào - ra $(x_i, y_i), với\ i = 1,2, \dots, N$. Cần sinh một tập luật mờ từ các cặp dữ liệu vào - ra trên, sao cho tập luật mờ này xác định ánh xạ $f: x \to y$.

Các bước cơ bản giải quyết bài toán này như sau [88]:

Bước 1: Xác định các tập mờ bao phủ các không gian dữ liệu đầu vào và đầu ra. Ví dụ với biến đầu vào x có các tập mờ $A_1, A_2, \dots, A_r$ có $\cup\, supp(A_i) = [\alpha_i, \beta_i]$ và mọi $x_i \in [\alpha_i, \beta_i]$. Các dạng tập mờ có thể chọn như: hình tam giác, hình thang, Gauss.

Bước 2: Với mỗi cặp dữ liệu (x_i, y_i), giả sử với biến đầu vào x, có $x_i \in\ supp(A_i)$ với độ thuộc $\mu_{ij}, với$ i $= 1,2, \dots,$ N, j $= 1,2, \dots,$ r, và biến đầu ra y có $y_i \in\ supp(B_i)$, với độ thuộc $\mu_i, với\ i = 1,2, \dots, N$ thì sinh được 1 luật:

$$IF\ x\ is\ A_i\ THEN\ y\ is\ B_i, \qquad \text{với độ thuộc } \prod \mu_{ij}$$

Bước 3: Với mỗi cặp dữ liệu (x_i, y_i), có thể có nhiều luật mờ được sinh ra, chỉ giữ lại luật có độ thuộc lớn nhất.

Kỹ thuật sinh luật mờ cơ bản trên là đơn giản, dễ thực hiện. Tuy nhiên, vì các hàm thành viên là cố định trong bước đầu tiên và không phụ thuộc vào các cặp dữ liệu vào-ra nên các hàm thành viên không đươc tối ưu hóa theo các cặp dữ liệu vào-ra và cơ sở luật mờ được tạo ra bởi phương pháp này có thể có số lượng khá lớn (theo kích thước bộ dữ liệu) do vậy đòi hỏi một khối lượng tính toán khổng lồ.

Phân cụm dữ liệu cũng là một giải pháp khá phổ biến để sinh luật mờ được nhiều tác giả quan tâm nghiên cứu và ứng dụng [34][64]. Với giải pháp này, tập dữ liệu vào – ra được phân cụm bằng các thuật toán phân cụm, khi đó một luật mờ sẽ được sinh ra tương ứng với mỗi phân cụm. Giải pháp này đã được chứng tỏ là khá hiệu quả trong những trường hợp không gian dữ liệu lớn. Tuy nhiên, do các tập mờ được tạo ra riêng cho mỗi luật (tương ứng với mỗi phân cụm) nên làm hạn chế tính diễn dịch của tập luật.

Ngoài ra, nhiều giải pháp khác như: mạng nơ-ron nhân tạo (Artificial Neural Networks – ANN) [38], [80], Mạng tự tổ chức SOM [40], Cây quyết định [78], Đại số gia tử [4], …. đã được nhiều tác giả nghiên cứu đề xuất, cải tiến và ứng dụng để trích xuất tập luật mờ giải quyết các bài toán phân lớp, dự báo, … Trong đó kỹ thuật trích xuất mô hình mờ dựa trên máy học Véc-tơ hỗ trợ đã được nhiều tác giả nghiên cứu và chứng minh tính hiệu quả của giải pháp, đặc biệt là hiệu quả ở tốc độ học của máy học véc-tơ hỗ trợ [15], [17], [24], [35], [36], [56], [63]. Đặc biệt trong [24], [36] và [56] đã tổng hợp những nghiên cứu và ứng dụng mô hình mờ trích xuất từ máy học véc-tơ hỗ trợ, ưu điểm nổi bật của mô hình mờ trích xuất từ SVM so với SVM nguyên thủy là "tính có thể diễn dịch được" của mô hình mờ. Tuy nhiên việc trích xuất một mô hình mờ đảm bảo "tính có thể diễn dịch được" vẫn là thách thức chưa được giải quyết của các nghiên cứu trích xuất mô hình mờ từ SVM.

1.4. Máy học véc-tơ hỗ trợ

1.4.1. Lý thuyết máy học Véc-tơ hỗ trợ

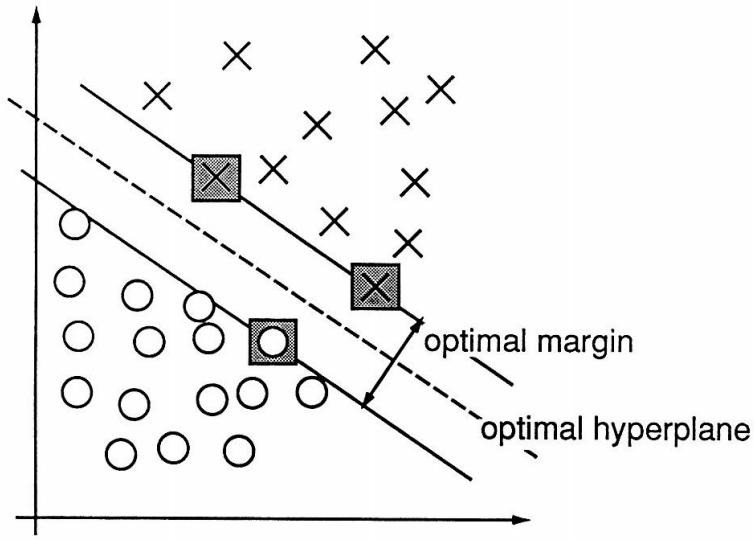
Thuật toán SVM ban đầu được tìm ra bởi Vladimir N. Vapnik và dạng chuẩn hiện nay sử dụng lề mềm được tìm ra bởi Corinna Cortes và Vapnik năm 1995 [23]. Đây là mô hình học dựa trên lý thuyết học thống kê (Statistical Learning), là một kỹ thuật được đề nghị để giải quyết cho các bài toán phân lớp. Lý thuyết cơ bản của máy học véc-tơ tựa cho vấn đề phân lớp có thể tóm tắt như sau:

Cho tập véctơ đầu vào $x_i \in R^n, i = 1,2,\dots,l$, và tập các giá trị nhãn lớp tương ứng $y_i \in \{-1; +1\}$ cho bộ phân lớp nhị phân. Hàm tuyến tính phân biệt hai lớp như sau:

$$f(x) = w^T.\Phi(x) + b\,, \tag{1.17}$$

trong đó, w là véc-tơ chuẩn (véc-tơ pháp tuyến) của siêu phẳng phân cách, b là độ lệch, và $\Phi(.)$ là hàm ánh xạ từ không gian đầu vào R^n sang không gian đặc trưng D, $\Phi(x): R^n \to D$.

Mục tiêu của SVM là tìm một siêu phẳng tối ưu sao cho khoảng cách lề giữa hai lớp đạt giá trị cực đại (Hình 1.3).



Hình 1.3. Hình ảnh phân lớp với SVM

Bên cạnh đó, để đảm bảo tính tổng quát hóa cao, một biến bù ξ, hay còn gọi là biến lỏng (slack variable) được đưa vào để nới lỏng điều kiện phân lớp. Bài toán đưa đến việc giải quyết tối ưu có ràng buộc:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i\,, \tag{1.18}$$

sao cho: $y_i(w^T.\Phi(x_i) + b) \geq 1 - \xi_i\,, \xi_i \geq 0, i = 1,2,\dots,l.$

trong đó, $C > 0$ là tham số chuẩn tắc (regularization parameter), ξ_i là biến lỏng.

Theo cách giải trong [23], việc giải bài toán (1.18) có thể chuyển thành giải bài toán đối ngẫu quy hoạch toàn phương (Quadratic Programming):

$$\max_{\alpha} L(\alpha) \equiv \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \Phi(x_i)^T . \Phi\left(x_j\right), \tag{1.19}$$

thỏa mãn: $0 \leq \alpha_i \leq C, i = 1,2, \dots, l$ và $\sum_{i=1}^{l} \alpha_i y_i = 0$, với α_i là các nhân tử Lagrange.

Sau khi xác định được các giá trị α_i từ bài toán (1.19), ta sẽ thu đươc các giá trị tối ưu w^* và b^* của siêu phẳng. Chỉ có các mẫu có $\alpha_i \geq 0$ mới tham gia vào các véc-tơ hỗ trợ (support vector). Cuối cùng, hàm quyết định phân lớp có dạng:

$$f(x) = sgn\left(\sum_{i=1}^{l} \alpha_i y_i \left(\Phi(x_i)^T . \Phi\left(x_j\right)\right) + b^*\right). \tag{1.20}$$

Gọi $K\left(x_i, x_j\right) = \Phi(x_i)^T . \Phi\left(x_j\right)$ là hàm nhân của không gian đầu vào. Khi đó hàm quyết định phân lớp (1.20) được viết lại như sau:

$$f(x) = sgn\left(\sum_{i=1}^{l} \alpha_i y_i K\left(x_i, x_j\right) + b^*\right). \tag{1.21}$$

Theo đó, tích vô hướng trong không gian đặc trưng tương đương với hàm nhân $K\left(x_i, x_j\right)$ ở không gian đầu vào. Như vậy, thay vì tính trực tiếp giá trị tích vô hướng, ta thực hiện gián tiếp thông qua hàm nhân $K\left(x_i, x_j\right)$.

1.4.2. Máy học Véc-tơ hỗ trợ cho vấn đề tối ưu hóa hồi qui

Với vai trò giải quyết vấn đề tối ưu hóa hồi quy, lý thuyết cơ bản của SVM có thể được vắn tắt như sau [13], [16], [87]:

Cho một tập dữ liệu huấn luyện $\{(x_1, y_1), \dots, (x_l, y_l)\} \subset R^n \times R$, trong đó R^n xác định miền dữ liệu đầu vào. Mục tiêu của máy học véc-tơ hỗ trợ hồi quy ε-SVR (ε-Support Vector Regression) là tìm một hàm quyết định siêu phẳng $f(x)$ tối ưu sao cho độ sai lệch trên tất cả các y_i của tập dữ liệu huấn luyện phải nhỏ hơn giá trị sai

số ε. Trong trường hợp hồi tuyến tính (linear regression), hàm quyết định $f(x)$ của máy học véc-tơ hỗ trợ hồi quy có dạng:

$$f(x) = \langle w, x \rangle + b \; với \; w \in R^n, b \in R, \tag{1.22}$$

trong đó $\langle ., . \rangle$ tích vô hướng trong không gian dữ liệu vào R^n; w là véc tơ pháp tuyến của siêu phẳng, và b là độ lệch.

Tìm hàm siêu phẳng tối ưu $f(x)$ trong (1.22) cũng có nghĩa là tìm w nhỏ. Một cách để đảm bảo w nhỏ là cực tiểu hóa chuẩn $\|w\|^2 = \langle w, w \rangle$. Chúng ta có thể viết lại thành bài toán tối ưu như sau:

$$\begin{gathered} min \; \frac{1}{2}\|w\|^2, \\ \text{sao cho: } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon \\ \langle w, x_i \rangle + b \; - y_i \leq \varepsilon \end{cases} \end{gathered} \tag{1.23}$$

Bằng cách đưa vào những biến lỏng (biến bù) ξ_i, ξ_i^* nhằm giải quyết vấn đề tìm hàm siêu phẳng $f(x)$ với "lề mềm", bài toán tối ưu (1.23) được viết thành:

$$\begin{gathered} min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*), \\ \text{Với tập ràng buộc: } \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i, \\ \langle w, x_i \rangle + b \; - y_i \leq \varepsilon + \xi_i^*, \\ \xi_i, \xi_i^* \; \geq 0, v\grave{a} \; i = 1, 2, \dots, l \end{cases} \end{gathered} \tag{1.24}$$

trong đó $C > 0$ là tham số chuẩn tắc, ε là sai số cho phép, và ξ_i, ξ_i^* là những biến lỏng.

Ý tưởng then chốt để giải quyết bài toán (1.24) là xây dựng hàm Lagrange từ hàm mục tiêu và các ràng buộc tương ứng, bằng cách đưa vào một tập kép các biến là nhân tử Lagrange. Hàm Lagrange được xây dựng như sau:

$$L \coloneqq \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) - \sum_{i=1}^{l}(\eta_i . \xi_i + \eta_i^* . \xi_i^*) \;, \tag{1.25}$$

$$-\sum_{i=1}^{l}\alpha_i(\varepsilon+\xi_i-y_i+w.\Phi(x)+b)\,,$$

$$-\sum_{i=1}^{l}\alpha_i^*(\varepsilon+\xi_i^*+y_i-w.\Phi(x)-b)\,,$$

trong đó $\eta_i,\eta_i^*,\alpha_i,\alpha_i^*\geq 0$ là những nhân tử Lagrange.

Các đạo hàm riêng của L đối với các biến w,b,ξ_i,ξ_i^* thỏa mãn các điều kiện sau:

$$\partial_b L=\sum_{i=1}^{l}(\alpha_i^*-\alpha_i)=0 \tag{1.26}$$

$$\partial_w L=w-\sum_{i=1}^{l}(\alpha_i^*-\alpha_i).x_i=0 \tag{1.27}$$

$$\partial_{\xi_i} L=C-\alpha_i-\eta_i=0 \tag{1.28}$$

$$\partial_{\xi_i^*} L=C-\alpha_i^*-\eta_i^*=0 \tag{1.29}$$

Bằng cách thế (1.26), (1.27), (1.28), và (1.29) vào (1.25), sẽ đưa đến bài toán tối ưu Quadratic Programming như sau:

$$max\ \ -\frac{1}{2}\Lambda^T H\Lambda+\ c^T\Lambda\,, \tag{1.30}$$

sao cho:

$$\textstyle\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0, và\ C\geq\ \alpha_i,\alpha_i^*\geq 0, với\ i=1,2,\dots,l$$

trong đó:

$$\Lambda^T=[\alpha_1,\alpha_2,\dots,\alpha_l,\alpha_1^*,\alpha_2^*,\dots,\alpha_l^*\,]$$

$$c^T=[\varepsilon+y_1,\varepsilon+y_2,\dots,\varepsilon+y_l,\varepsilon-y_1,\varepsilon-y_2,\dots,\varepsilon-y_l]$$

$$H=\begin{bmatrix}D & -D\\ -D & D\end{bmatrix}\text{ (ma trận kernel)}$$

với D là ma-trận vuông đối xứng $l \times l$ và các phần tử là $D_{i,j} = \langle x_i, x_j \rangle$.

Công thức (1.27) được viết lại thành:

$$w = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*).x_i \tag{1.31}$$

Và như vậy hàm quyết định (1.22) được viết thành:

$$f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)\langle x_i, x \rangle + b\ . \tag{1.32}$$

Những điểm đầu vào x_i tương ứng với $(\alpha_i - \alpha_i^*) \neq 0$ được gọi là những véc-tơ hỗ trợ (SV).

Mở rộng ra cho trường hợp hồi quy phi tuyến (nonliner regression), bằng cách ánh xạ dữ liệu đầu vào vào một không gian thuộc tính đa chiều như sau:

$$x_i \longmapsto \Phi(x_i) = (\Phi_1(x_i), \Phi_2(x_i), \dots, \Phi_n(x_i), \dots) \tag{1.33}$$

Hàm ánh xạ $\Phi(x_i)$ được xác định khi lựa chọn hàm nhân kernel:

$$K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle \tag{1.34}$$

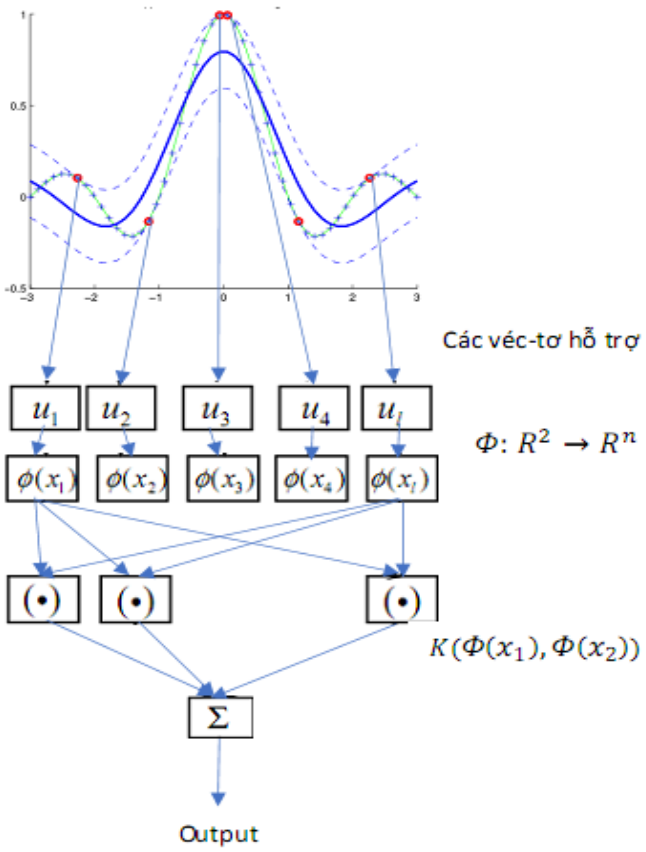
Và khi đó hàm quyết định (1.32) được viết thành:

$$f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*).K(x_i, x) + b \tag{1.35}$$

Sự khác biệt so với trường hợp hồi quy tuyến tính là véc-tơ w không còn được xác định một cách rõ ràng nữa. Và trong trường hợp hồi quy phi tuyến, vấn đề tối ưu hóa tương ứng với việc tìm hàm quyết định siêu phẳng $f(x)$ trong không qian thuộc tính đa chiều, không phải là trong không gian dữ liệu đầu vào.

1.5. Trích xuất mô hình mờ TSK dựa vào máy học véc-tơ hỗ trợ

Hình 1.4 thể hiện quá trình xác định hàm quyết định đầu ra của máy học véc-tơ hỗ trợ. Theo đó, các véc-tơ hỗ trợ u_i sẽ được ánh xạ vào không gian thuộc tính nhiều chiều, sau đó hàm quyết định đầu ra *(output)* sẽ được tính theo công thức (1.35) với hàm nhân $K(.)$ được chọn.



Hình 1.4. Quá trình xác định hàm quyết định đầu ra của máy học véc-tơ hỗ trợ

Bên cạnh đó, xét hàm đầu ra của hệ thống mờ TSK (1.16), đặt:

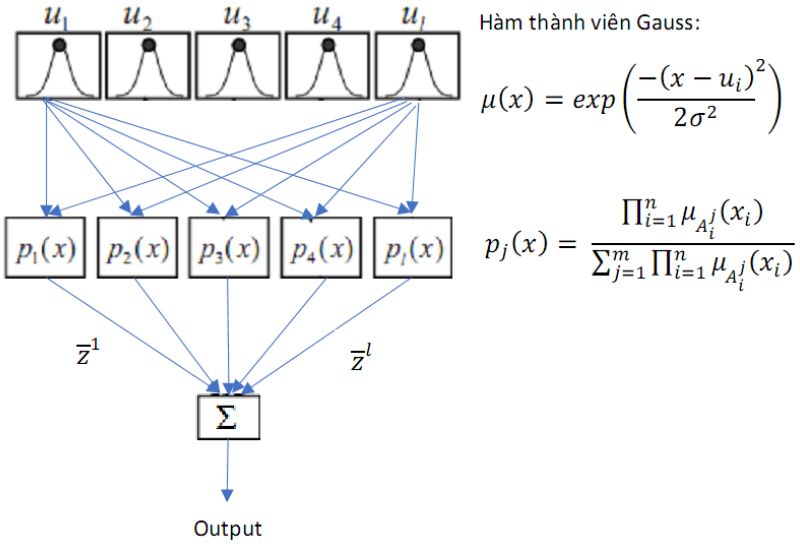
$$p_j(x) = \frac{\prod_{i=1}^{p} \mu_{A_i^j}(x_i)}{\sum_{j=1}^{m} \prod_{i=1}^{p} \mu_{A_i^j}(x_i)}$$

Khi đó (1.16) tương đương với:

$$f(x) = \sum_{j=1}^{m} \overline{z}^j \, p_j(x)$$

Trong trường hợp hàm thành viên Gauss được chọn, quá trình xác định hàm đầu ra của hệ thống mờ được thể hiện ở Hình 1.5.

Hình 1.4 và Hình 1.5 cho thấy sự tương đương giữa máy học véc-tơ hỗ trợ hồi quy và hệ thống mờ TSK. Vấn đề đặt ra là làm thế nào đồng nhất hàm đầu ra (1.16) của hệ thống mờ TSK và hàm quyết định (1.35) của máy học véc-tơ hỗ trợ hồi quy.



Hình 1.5. Quá trình xác định hàm đầu ra của hệ thống mờ TSK

Để (1.16) và (1.35) đồng nhất với nhau, trước tiên chúng ta phải đồng nhất giữa hàm nhân trong (1.35) và hàm thành viên trong (1.16). Ở đây, để thỏa mãn điều kiện Mercer [35], [61], hàm thành viên Gauss được chọn làm hàm nhân:

$$K(x_i, x) = exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right). \quad (1.36)$$

Đồng thời giá trị của độ lệnh *b* trong (1.35) phải bằng *0*.

Khi hàm Gauss được chọn làm hàm thành viên và hàm nhân kernel, đồng thời số luật mờ được thiết lập bằng với số véc-tơ hỗ trợ ($m = l$) và giá trị b trong (1.35) thiết lập bằng *0*, thì (1.35) và (1.16) tương ứng được viết lại thành:

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right) \quad (1.37)$$

và

$$f(x) = \frac{\sum_{j=1}^{l} \overline{z}^j exp\left(-\frac{1}{2}\left(\frac{x_j - x}{\sigma_j}\right)^2\right)}{\sum_{j=1}^{l} exp\left(-\frac{1}{2}\left(\frac{x_j - x}{\sigma_j}\right)^2\right)}. \quad (1.38)$$

Như cách biến đổi trong [35], hàm suy luận mờ có thể viết lại như sau:

$$f(x) = \sum_{j=1}^{l} \overline{z}^j exp\left(-\frac{1}{2}\left(\frac{x_j - x}{\sigma_j}\right)^2\right) \quad (1.39)$$

Nếu thiết lập $\overline{z}^j = (\alpha_i - \alpha_i^*)$ thì hàm đầu ra của mô hình mờ TSK (1.39) và hàm quyết định của máy học véc-tơ hỗ trợ hồi quy (1.37) là hoàn toàn bằng nhau.

Ngoài ra, có thể tiếp cận một cách khác như trong [35]. Theo cách tiếp cận này, hàm nhân của máy học véc-tơ hỗ trợ được thiết lập như sau:

$$K(x_i, x) = \frac{exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right)}{\sum_{i=1}^{l} exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right)}. \quad (1.40)$$

Khi đó hàm quyết định của máy học véc-tơ hỗ trợ (1.35) trở thành:

$$f(x) = \frac{\sum_{i=1}^{l} (\alpha_i - \alpha_i^*)\, exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right)}{\sum_{i=1}^{l} exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right)} + b \ . \quad (1.41)$$

Bên cạnh đó, để thỏa mãn điều kiện Mercer [35], [61], hàm thành viên Gauss được chọn:

$$\mu_{A_i^j}(x_i) = exp\left(-\frac{1}{2}\left(\frac{x_i - \bar{x}_i^j}{\sigma_i}\right)^2\right). \quad (1.42)$$

Khi đó hàm đầu ra của hệ thống mờ TSK (1.16) trở thành (1.38).

Nếu thiết lập $\overline{z}^j = (\alpha_i - \alpha_i^*)$ và chọn giá trị b trong (1.41) bằng 0 thì hàm quyết định (1.41) và hàm đầu ra của hệ thống mờ (1.38) sẽ bằng nhau. Tuy nhiên, biểu thức (1.40) chỉ có thể có nếu số lượng véc-tơ hỗ trợ l là biết trước.

Trong điều kiện của máy học véc-tơ hỗ trợ thì số lượng véc-tơ hỗ trợ không thể xác định trước khi huấn luyện, vì vậy hàm nhân của máy học véc-tơ hỗ trợ chỉ có thể chọn như sau [35]:

$$p_j(x) = \prod_{i=1}^{p} exp\left(-\frac{1}{2}\left(\frac{x_i - \bar{x}_i^j}{\sigma_i}\right)^2\right). \tag{1.43}$$

Tương đương với:

$$p_j(x) = \prod_{i=1}^{p} \mu_{A_i^j}(x_i)\,. \tag{1.44}$$

với $\bar{x}_i^j$ và σ_i là những tham số thực; giá trị của σ_i cho biết phương sai của mỗi hàm thành viên Gauss và được xác định như trong [88].

Sử dụng hàm nhân (1.42), hàm đầu ra của hệ thống mờ tương ứng nhận được trở thành:

$$\begin{aligned} f(x) &= \sum_{j=1}^{m} \overline{z}^j \; p_j(x) \\ &= \textstyle\sum_{j=1}^{m} \overline{z}^j \left(\prod_{i=1}^{p} \mu_{A_i^j}(x_i)\right). \end{aligned} \tag{1.45}$$

Lưu ý rằng hệ thống suy luận mờ phải được chuẩn hóa để trở thành (1.16). Để thực hiện việc chuẩn hóa, chúng ta phải điều chỉnh ma-trận kernel (ma-trận Hessian) như sau [35]:

$$H' = \begin{bmatrix} D' & -D' \\ -D' & D' \end{bmatrix}, \tag{1.46}$$

trong đó D' là một ma-trận đối xứng $l \times l$ với các phần tử là:

$$D'_{ij} = \frac{\langle \varphi(x_i), \varphi(x_j) \rangle}{\sum_{j=1}^{m} \langle \varphi(x_i), \varphi(x_j) \rangle} \tag{1.47}$$

khác với ma-trận D gồm các phần tử $D_{ij} = \langle \varphi(x_i), \varphi(x_j) \rangle$.

Khi ma-trận Hessian điều chỉnh áp dụng, thì các véc-tơ hỗ trợ β' cũng được điều chỉnh như sau:

$$\beta' = \beta_0 H_0 (H')^{-1}$$

với β_0 là những véc-tơ hỗ trợ và H_0 là ma-trận Hessian ban đầu.

Công thức tính toán hàm quyết định của máy học véc-tơ hỗ trợ trở thành:

$$f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) \langle \varphi(x_i), \varphi(x_j) \rangle + b, \quad (1.48)$$

trong đó những điểm đầu vào x_i ứng với $(\alpha_i - \alpha_i^*) \neq 0$ là những véc-tơ hỗ trợ và những hằng số b là sai số. Trong bài toán trích xuất mô hình mờ này, có thêm 1 ràng buộc đó là $b = 0$, và khi đó biểu thức (1.48) trở thành:

$$f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) \langle \varphi(x_i), \varphi(x_j) \rangle. \quad (1.49)$$

Cũng với cơ sở lập luận như trên, trong [35] các tác giả đã đề xuất giải pháp trích xuất tập luật mờ từ dữ liệu dựa vào máy học véc-tơ hỗ trợ. Tuy nhiên ở [35], các tác giả chưa đề cấp đến vấn đề tối ưu hóa tham số của các hàm thành viên mờ Gauss và giải pháp lựa chọn các tham số khi huấn luyện máy học véc-tơ hỗ trợ. Ở đây, luận án đề xuất thuật toán trích xuất tập luật mờ TSK từ kết quả huấn luyện máy học véc-tơ hỗ trợ hồi quy có kết hợp tối ưu hóa tham số các hàm thành viên mờ. Sơ đồ các bước thực hiện thuật toán được thể hiện ở Hình 1.6.

Input: Tập dữ liệu huấn luyện *H* và tham số lỗi ε

Output: Mô hình mờ TSK

Bước 1. Khởi tạo các tham số C, ε cho máy học véc-tơ hỗ trợ hồi quy ở công thức 1.30, và tham số σ cho hàm nhân ở công thức 1.43

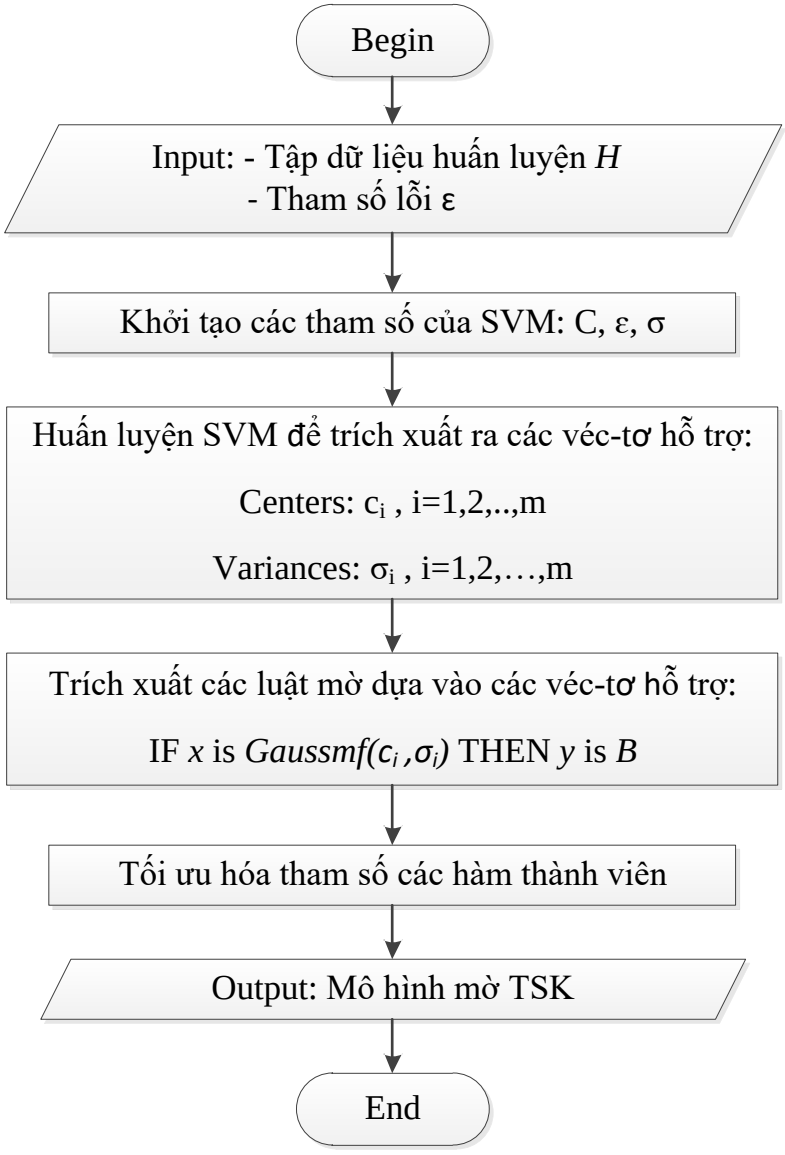
Bước 2. Huấn luyện máy học véc-tơ hỗ trợ bằng công thức 1.30 và hàm nhân được chọn theo công thuwcsc 1.43, để tìm ra các véc-tơ hỗ trợ (cũng chính là các giá trị trung bình của các hàm thành viên Gauss) và các giá trị độ lệch chuẩn tương ứng là c_i và σ_i, với $i = 1 = 1, 2, \ldots, m$

Bước 3. Trích xuất tập luật mờ dựa trên các cặp giá trị (c_i, σ_i), sử dụng hàm thành viên mờ Gauus.

Hàm đầu ra của hệ thống mờ được xác định bằng công thức:

$$f(x) = \frac{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\ \mathrm{K}(\mathrm{x_i}, \mathrm{x})}{\sum_{i=1}^{l}\mathrm{K}(\mathrm{x_i}, \mathrm{x})}. \quad (1.50)$$

Bước 4. Thực hiện tối ưu hóa các tham số của hàm thành viên mờ



Hình 1.6. Sơ đồ khối của thuật toán trích xuất tập luật mờ TSK dựa vào máy học véc-tơ hỗ trợ

1.6. Lựa chọn các tham số

1.6.1. Chọn các tham số của hàm thành viên

Những tham số của hàm thành viên có thể được tối ưu hóa dùng những thuật toán Gradient descent hoặc thuật toán di truyền (GA) [33][80]. Trong Luận án, để nhận được tập mờ tối ưu, trương tự phương pháp tối ưu hóa tham số (giảm lỗi) bằng thuật toán Gradient descent trong mô hình ANFIS được chuẩn hóa trong Matlab, giá trị các tham số của hàm thành viên được cập nhật theo các hàm thích nghi sau đây:

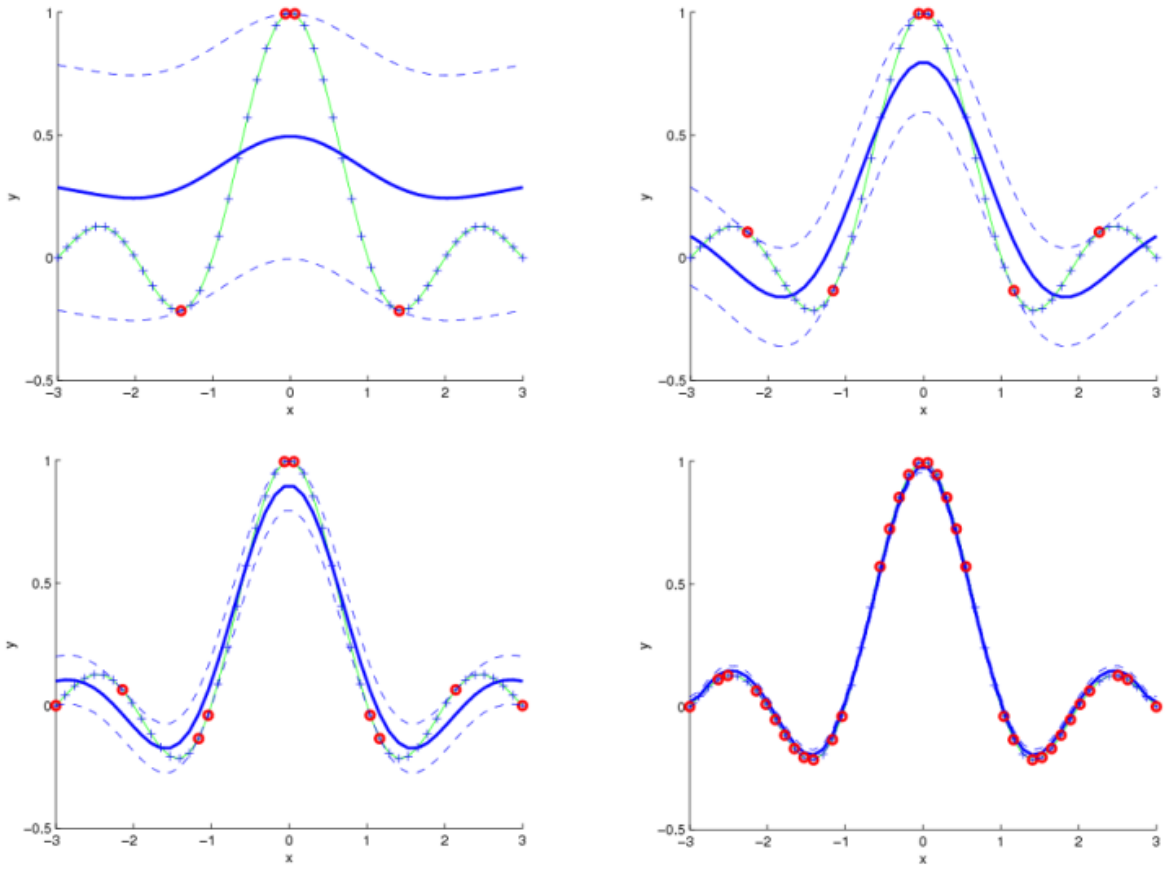
$$\sigma_i(t+1) = \sigma_i(t) + \delta\varepsilon_{1,i}\left[\frac{(x-c)^2}{\sigma^3}exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)\right], \quad (1.51)$$

$$c_i(t+1) = c_i(t) + \delta\varepsilon_{1,i}\left[\frac{-(x-c)}{\sigma^2}exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)\right]. \quad (1.52)$$

1.6.2. Vai trò của tham số ε

Một trong những đặc điểm nổi bật của mô hình mờ, cụ thể là mô hình mờ hướng dữ liệu, so với các mô hình máy học thống kê khác đó là "tính có thể diễn dịch được" (intepretability) [11], [18], [24], [36], [56], [80]. Tuy nhiên, đối với bài toán trích xuất mô hình mờ dựa vào máy học véc-tơ hỗ trợ, nếu tăng tính chính xác của mô hình thì số lượng véc-tơ hỗ trợ (SV) cũng tăng lên, đồng nghĩa với số lượng luật mờ trong mô hình trích xuất được cũng tăng lên. Điều này làm cho tính phức tạp của hệ thống tăng lên và đặc biệt là "tính có thể diễn dịch được" của hệ thống mờ giảm đi.

Xét kết quả thực nghiệm mô hình máy học véc-tơ hồi quy trên hàm hồi qui phi tuyến $Sinc(x)$ (bài toán được giới thiệu chi tiết ở mục 1.6.2). Theo kết quả thể hiện ở Hình 1.7, khi giá trị của tham số ε giảm đi thì số lượng véc-tơ hỗ trợ cũng tăng lên (các véc-tơ hỗ trợ được đánh dấu vòng tròn), đồng thời độ chính xác của kết quả dự đoán cũng tăng lên (đường đậm nét là đường dự đoán hồi quy, đường đánh dấu + là đường biểu diễn giá trị dữ liệu thực tế). Như vậy, với mỗi bài toán cụ thể, cần phải có sự lựa chọn số giá trị tham số ε phù hợp để có được số lượng luật mờ hợp lý, đảm bảo tính chính xác của mô hình đầu ra với ngưỡng sai số xác định.



Hình 1.7. Mối quan hệ giữa số lượng véc-tơ hỗ trợ và tham số ε (giá trị của ε tương ứng theo thứ tự các hình vẽ là 0.5, 0.2, 0.1 và 0.01)

Từ những phân tích trên, Luận án đề xuất thuật toán f-SVM cho phép trích xuất tập luật mờ TSK từ dữ liệu dựa vào máy học véc-tơ hỗ trợ, như thể hiện ở Hình 1.8.

Thuật toán **f-SVM**

Input: *Tập dữ liệu huấn luyện $\mathcal{H}$, Tham số lỗi ε.*

Output: *Mô hình mờ với hàm đầu ra $f(x)$.*

1. Khởi tạo các giá trị tham số: C, ε, σ;

2. Huấn luyện SVM: $f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\, K(x_i, x) + b$;

   Điều chỉnh ma trận kernel: $H' = \begin{bmatrix} D' & -D' \\ -D' & D' \end{bmatrix}$; (công thức 1.46)

   với $D'_{ij} = \frac{\langle \varphi(x_i), \varphi(x_j) \rangle}{\sum_j \langle \varphi(x_i), \varphi(x_j) \rangle}$; (công thức 1.47)

4. Trích xuất các $SV = \{c_i : (\alpha_i - \alpha_i^*) \neq 0, i \in \{0, \dots, l\}\}$;

5. Sinh ra tập luật mờ từ tập SV với hàm thành viên Gauss;

6. Tối ưu hóa tham số các hàm thành viên (công thức 1.51 và 1.52)

$$\sigma_i(t+1) = \sigma_i(t) + \delta\varepsilon_{1,i}\left[\frac{(x-c)^2}{\sigma^3} exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)\right],$$

$$c_i(t+1) = c_i(t) + \delta\varepsilon_{1,i}\left[\frac{-(x-c)}{\sigma^2} exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)\right];$$

7. **return** $f(x) = \frac{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(x_i,x)}{\sum_{i=1}^{l}K(x_i,x)}$

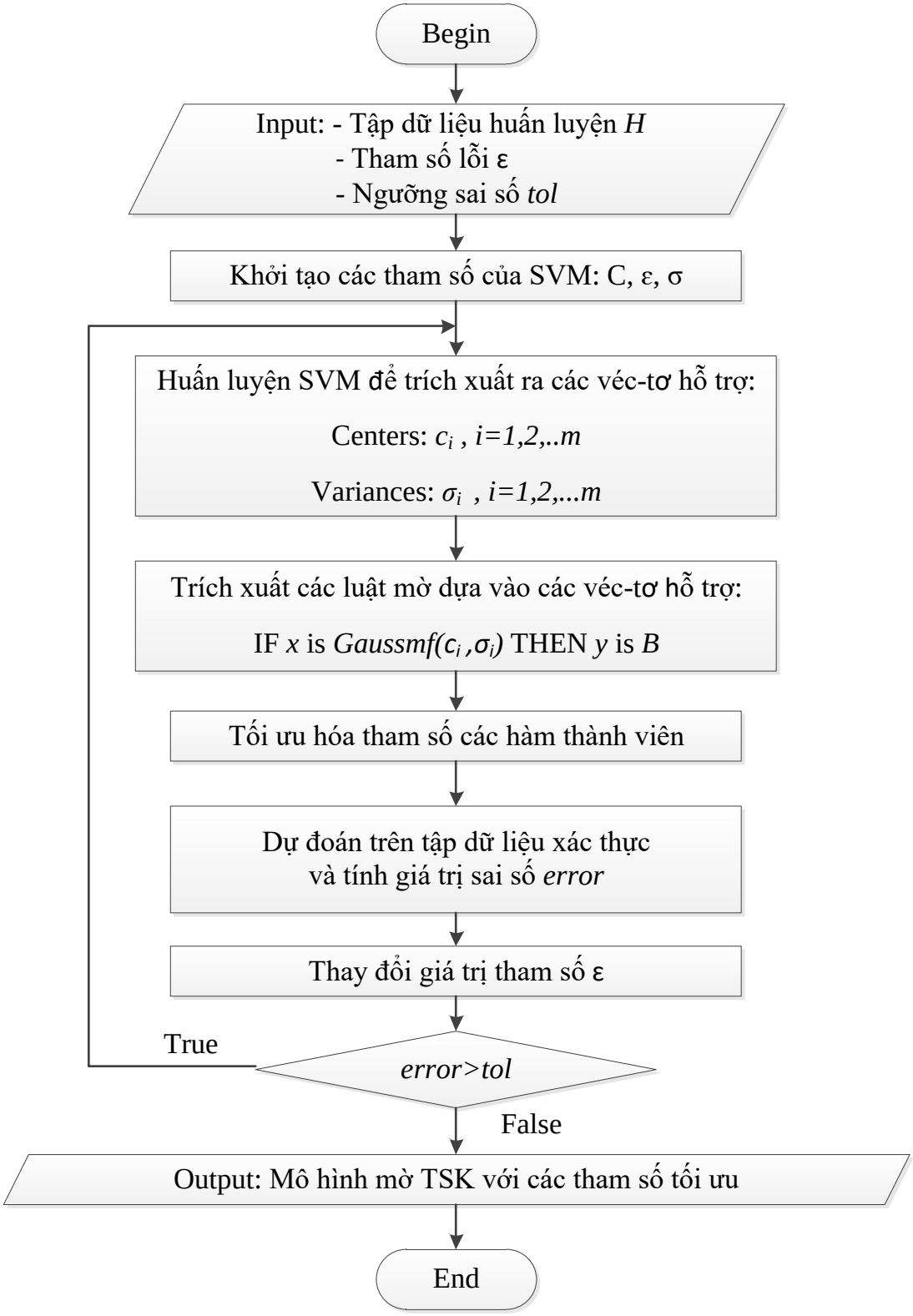
Hình 1.8. Thuật toán f-SVM

Trong thuật toán này, ngoài việc tối ưu hóa các tham số của hàm thành viên, giá trị tham số ε có thể được điều chỉnh để nhận được mô hình tối ưu. Độ phức tạp của thuật toán huấn luyện máy học véc-tơ hỗ trợ trong trường hợp tốt nhất là bình phương của số lượng phần tử dữ liệu huấn luyện [7][20]. Với kích thước tập dữ liệu huấn luyện là N thì độ phức tạp của thuật toán f-SVM là $O(N^2)$.

Việc lựa chọn giá trị tối ưu của tham số ε được thực hiện bằng cách sử dụng tập dữ liệu xác thực. Các bước thực hiện trích xuất tập luật mờ từ dữ liệu huấn luyện đầu vào, có tối ưu hóa các tham số của hàm thành viên bằng các hàm thích nghi (1.51) và (1.52); đồng thời lựa chọn giá trị tham số ε tối ưu được thể hiện ở Hình 1.9.

Theo đó, bước lựa chọn giá trị tham số ε tối ưu được thực hiện bằng cách thay đổi giá trị tham số ε, lặp lại việc thực hiện huấn luyện SVM để trích xuất tập luật mờ, sau đó tiến hành thực nghiệm dự báo trên tập dữ liệu xác thực để đánh giá sai số *error* giữa giá trị thực tế và giá trị dự đoán. Quá trình lặp lại sẽ kết thúc khi giá trị sai số *error* không lớn hơn giá trị ngưỡng sai số *tol* cho trước. Kết quả là với từng bài toán cụ thể, giá trị tam số ε được lựa chọn thích hợp để trích xuất được mô hình mờ TSK đầu ra đáp ứng yêu cầu dự đoán với ngưỡng sai số cho trước.

Với kích thước của tập dữ liệu xác thực là k, nhỏ hơn rất nhiều so với kích thước tập dữ liệu huấn luyện N, và T là số lần lặp lại để thực hiện dự đoán trên tập dữ liệu xác thực và đánh giá sai số *error*, thì độ phức tạp của thuật toán có lựa chọn tham số ε tối ưu sẽ là $O(T.N^2)$.



Hình 1.9. Thuật toán trích xuất tập luật mờ TSK dựa vào máy học véc-tơ hỗ trợ có lựa chọn giá trị tham số tối ưu

1.7. Tổ chức thực nghiệm

1.7.1. Mô tả thực nghiệm

Để đánh giá thuật toán f-SVM đã đề xuất, luận án xây dựng một hệ thống thử nghiệm dựa trên bộ công cụ Matlab. Trong thuật toán trích xuất tập luật mờ *f-SVM*, thuật toán học SVM của thư viện LibSVM được phát triển bởi nhóm của Chih-Chung Chang [20] được sử dụng để sản xuất ra các SV. Trong đó, hàm SVM*genfis()* được xây dựng để sinh ra mô hình mờ TSK ban đầu dựa vào những véc-tơ hỗ trợ nhận được từ kết quả huấn luyện SVM, theo đúng cấu trúc của hệ thống mờ ANFIS trong thư viện Matlab. Hàm *anfis()* của thư viện *Fuzzy Toolbox* của phần mềm Matlab được sử dụng tối ưu hóa các tham số hàm thành viên bằng phương pháp Gradient descent và trích xuất các luật mờ. Sau cùng, hàm *evalfis()* trong thư viện công cụ Matlab Fuzzy Logic được sử dụng để suy luận ra kết quả dự đoán sử dụng mô hình mờ TSK trích xuất được.

Các bài toán thực nghiệm được lựa chọn bao gồm một bài toán hồi quy phi tuyến và một bài toán chuỗi thời gian hỗn loạn. Những bài toán này được chọn dựa trên đề xuất của một số tác giả đã nghiên cứu đề xuất và thực nghiệm mô hình mờ hướng dữ liệu [35][80]; đồng thời những mẫu dữ liệu sinh ra từ những công thức này sẽ hạn chế nhiễu, điều đó sẽ thuận lợi hơn cho việc đánh giá hiệu quả của thuật toán (những kết quả thực nghiệm này đã được công bố trong công trình [A8]). Ngoài ra, trong quá trình thực hiện luận án, một mô hình thực nghiệm khác cũng được triển khai trên một bài toán thực tế là "Phân tích dữ liệu điểm sinh viên". Kết quả của trường hợp thực nghiệm này đã được công bổ ở công trình [A4].

Để đánh giá sai số *error* giữa giá trị thực tế của dữ liệu và giá trị dự đoán dựa trên mô hình mờ trích xuất được, sai số bình phương trung bình gốc - RMSE (Root Mean Squared Error) được chọn. Dựa trên sự so sánh giá trị của sai số RMSE giữa các trường hợp chọn giá trị ε khác nhau để có sự cân nhắc lựa chọn giá trị ε tối ưu nhất, đảm bảo số luật mờ (số véc-tơ hỗ trợ) đủ nhỏ và giá trị sai số RMSE trong ngưỡng cho phép (*tol*). Giá trị sai số RMSE được tính toán dựa vào công thức:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{k}(y_i - \hat{y}_i)^2}{k}} \quad (1.53)$$

trong đó k là tổng số mẫu dữ liệu, y_i và $\hat{y}_i$ là giá trị đúng và giá trị dự đoán được tương ứng.

1.7.2. Bài toán hồi quy phi tuyến

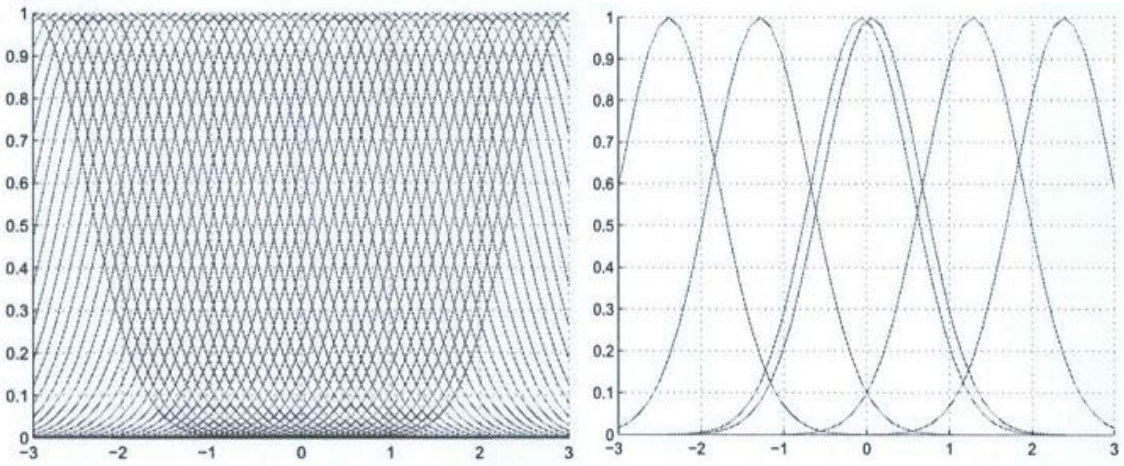
Mục tiêu của bài toán dự đoán hồi quy phi tuyến đơn giản là ước tính một hàm quyết định phù hợp với các mục tiêu mong muốn. Ở ví dụ dự doán hồi quy này, bài toán được chọn là dự đoán giá trị của hàm $Sinc(x)$ được xác định như sau [35]:

$$Sinc(x) = \begin{cases} \frac{sin(x)}{x} & if\ x \neq o \\ 1 & if\ x = 0 \end{cases} \quad (1.54)$$

Vùng dữ liệu được chọn làm dữ liệu huấn luyện xác định trong đoạn $x \in [-3\pi, 3\pi]$, và đây cũng cũng chính là vùng dữ liệu xác thực. Dựa vào công thức tính hàm $Sinc(x)$ ở trên để sinh ngẫu nhiên 50 mẫu dữ liệu huấn luyện và đồng thời cũng được dùng làm dữ liệu xác thực.

Trong quá trình huấn luyện mô hình máy học véc-tơ hỗ trợ, giá trị tham số ε được điều chỉnh thay đổi để các định số lượng véc-tơ hỗ trợ đầu ra. Trong trường hợp thực nghiệm này, tương tự như thực nghiệm trong [35][59], giá trị tham số C được thiết lập cố định bằng 10. Khi giá trị tham số ε được thiết lập bằng 0.0, sẽ có 50 véc-tơ hỗ trợ nhận được từ kết quả huấn luyện SVM, đồng nghĩa với việc trích xuất được 50 luật mờ (chú ý rằng, trong trường hợp này tất cả các mẫu dữ liệu huấn luyện đã được chọn làm các véc-tơ hỗ trợ đầu ra). Hình 1.10a thể hiện phân bố của 50 hàm thành viên mờ tương ứng trong trường hợp $\varepsilon = 0.0$.

Sau đó giá trị tham số ε được điều chỉnh tăng dần lên. Khi $\varepsilon = 0.1$, có 6 véc-tơ hỗ trợ nhận được, tương ứng các giá trị của x là -2.48, -1.48, -0.02, 0.02, 1.32, và 2.48. Bảng 1.1 thể hiện nội dung của 6 luật mờ trích xuất được. Hình 1.10b thể hiện phân bố của 6 hàm thành viên mờ tương ứng.



(a) (b)

Hình 1.10. Phân bố các hàm thành viên mờ: (a) trường hợp 50 luật ứng với $\varepsilon = 0.0$ và (b) trường hợp 6 luật ứng với $\varepsilon = 0.1$

Bảng 1.1. Tập 6 luật trích xuất được

Luật	Chi tiết
R1	IF *x is Gaussmf(0.66,-2.48)* THEN *y is 0.33*
R2	*IF x is Gaussmf(0.71,-1.32)* THEN *y is -0.36*
R3	IF *x is Gaussmf(0.78,-0.02)* THEN *y is 1.32*
R4	IF *x is Gaussmf(0.78,0.02)* THEN *y is 1.32*
R5	IF *x is Gaussmf(0.71,1.32)* THEN *y is -0.36*
R6	IF *x is Gaussmf(0.66,2.48)* THEN *y is 0.33*

Ở tất cả các trường hợp thay đổi giá trị tham số ε để điều chỉnh số lượng luật mờ được trích xuất, các mô hình mờ trích xuất được sẽ được thử nghiệm suy luận trên tập dữ liệu xác thực (cũng đồng thời là tập dữ liệu huấn luyện). Trong mỗi trường hợp, giá trị sai số RMSE sẽ được tính theo công thức (1.53). Kết quả các giá trị sai số RMSE tính được trong từng trường hợp điều chỉnh số lượng luật mờ trong mô hình thực nghiệm được thể hiện trong Bảng 1.2.

Bảng 1.2. Giá trị sai số RMSE trong các trường hợp thử nghiệm (C=10)

Tham số ε	Số luật	RMSE
0.0	50	$< 10^{-10}$
0.0001	30	$< 10^{-10}$
0.001	10	0.0015
0.01	8	0.0013
0.1	6	0.0197
0.5	4	0.0553

Qua thực nghiệm này có thể thấy rằng, có thể tối ưu hóa số lượng và vị trí của các véc-tơ hỗ trợ thông qua việc điều chỉnh giá trị tham số ε. Điều này cũng đồng nghĩa với việc tối ưu hóa phân bố và số lượng luật mờ trong mô hình trích xuất được thông qua điều chỉnh giá trị tham số ε. Đồng thời giá trị tham số ε được lựa chọn tối ưu cho từng bài toán cụ thể, thông qua việc đánh giá sai số RMSE (như trong Bảng 1.2). Với từng bài toán cụ thể, ngưỡng sai số RMSE được xác định, để từ đó có thể cân nhắc lựa chọn giá trị tham số ε hợp lý.

Bên cạnh thực nghiệm mô hình dự đoán dựa trên thuật toán f-SVM, luận án cũng tiến hành thực nghiệm trên cùng bộ dữ liệu đối với các mô hình tương tự khác, bao gồm: mô hình ANFIS được đề xuất bởi Jang J.S.R trong [30], [33] (đã được xây dựng thàm hàm tiêu chuẩn trong thư viên Matlab), mô hình SVM hồi quy nguyên thủy trong thư viện LibSVM trong [20], [54] và mô hình f-SVM chưa thực hiện tối ưu hóa tham số các hàm thành viên được đề xuất bởi J.-H Chiang và đồng sự trong [35]. Hiệu quả dự đoán trên 50 mẫu dữ liệu xác thực của các mô hình áp dụng được đánh giá thông qua sai số RMSE được thể hiện trong Bảng 1.3.

Bảng 1.3. Kết quả dự đoán trên 50 mẫu dữ liệu xác thực trong cho các trường hợp thực nghiệm của bài toán 1.7.2

Số luật mờ/Số véc-tơ hỗ trợ	Mô hình áp dụng			
	ANFIS	SVM	Mô hình f-SVM Chưa tối ưu hóa tham số hàm thành viên	Mô hình f-SVM
50	$<10^{-10}$	0.0074	0.0081	$< 10^{-10}$
30	$<10^{-10}$	0.0572	0.0501	$< 10^{-10}$
10	0.0017	0.0697	0.0611	0.0015
8	0.0018	0.0711	0.0785	0.0013
6	0.0248	0.2292	0.2312	0.0197
4	0.1894	0.2851	0.2901	0.0553

Các giá trị sai số RMSE trong Bảng 1.3 cho thấy trong trường hợp thực nghiệm này, kết quả dự đoán theo mô hình trích xuất được dựa vào thuật toán f-SVM đề xuất có độ chính xác tương đương và có phần tốt hơn so với mô hình ANFIS. So với mô hình SVM nguyên thủy và mô hình mờ trích xuất từ SVM được đề xuất trong [35] (chưa tối ưu hóa tham số các hàm thành viên), mô hình ứng dụng thuật toán f-SVM do luận án đề xuất cải tiến cho kết quả dự đoán chính xác hơn.

1.7.3. Bài toán dự báo dữ liệu chuỗi thời gian hỗn loạn Mackey-Glass

Để đánh giá hiệu quả ứng dụng của thuật toán f-SVM trong việc xây dựng một mô hình dự đoán, chúng tôi lựa chọn thực nghiệm trên dữ liệu chuỗi thời gian hỗn loạn Mackey-Glass (Mackey-Glass chaotic time series) [35][52][80]. Dữ liệu chuỗi thời gian Mackey-Glass được sinh theo công thức sau:

$$\dot{x} = \frac{ax(t-\tau)}{1+x^b(t-\tau)} - cx(t), \tag{1.55}$$

trong đó ta chọn $\tau = 30$, $a = 0.2$, $b = 10$, và $c = 0.1$.

Thuộc tính đầu vào được lựa chọn cho mô hình là giá trị $x(t-1)$, $x(t-2)$, thuộc tính đầu ra cần dự đoán là giá trị $x(t)$.

Sử dụng công thức (1.55) để sinh ra 1000 mẫu dữ liệu; trong đó trích 800 mẫu dữ liệu để huấn luyện cho máy học véc-tơ hỗ trợ và trích xuất ra mô hình mờ, 200 mẫu dữ liệu còn lại được sử dụng để xác thực mô hình mờ trích xuất được và chọn ra giá trị tham số tối ưu.

Tương tự với thực nghiệm ở 1.7.2, trong thực nghiệm này cố định giá trị tham số $C = 10$. Khi thiết lập giá trị tham số $\varepsilon = 0.0$ thì kết quả có 200 luật mờ nhận được. Giá trị tham số ε được điều chỉnh tăng dần. Khi $\varepsilon = 0.1$, hệ thống mờ trích xuất được rút gọn còn 9 luật. Kết quả dự đoán trên tập dữ liệu xác thực dựa vào tập luật trích xuất được trong các trường hợp thể hiện ở Bảng 1.4.

Bảng 1.4. Kết quả dự đoán trên 200 mẫu dữ liệu xác thực trong cho các trường hợp thực nghiệm của bài toán 1.7.3

Số véc-tơ hỗ trợ/ Số luật mờ	Mô hình áp dụng			
	ANFIS	SVM	Mô hình f-SVM Chưa tối ưu hóa tham số hàm thành viên	Mô hình f-SVM
170	<10-10	0.0540	0.0512	<10-10
36	0.0034	0.0509	0.0511	0.0086
25	0.0041	0.0635	0.0630	0.0092
16	0.0050	0.0748	0.0755	0.0095
9	0.0074	0.1466	0.1501	0.0098
4	0.0087	0.1955	0.1895	0.0102

Cùng với việc triển khai thực nghiệm mô hình sử dụng thuật toán f-SVM do luận án đề xuất, các mô hình ANFIS, SVM nguyên thủy và mô hình mờ trích xuất từ SVM được đề xuất trong [35] cũng được tiến hành trên cùng bộ dữ liệu thực nghiệm.

Bảng 1.4 thể hiện sự so sánh hiệu quả dự đoán trên 200 mẫu dữ liệu xác thực của các mô hình trong từng trường hợp thực nghiệm, thông qua giá trị của sai số RMSE.

So sánh các giá trị của RMSE trong Bảng 1.4, có thể nhận thấy rằng mô hình ứng dụng thuật toán f-SVM đề xuất cho kết quả dự đoán gần tương đương với mô hình ANFIS. So với mô hình SVM nguyên thủy và mô hình trích xuất luật mờ dựa trên SVM đề xuất trong [35] thì mô hình đề xuất có độ chính xác của kết quả dự đoán cao hơn.

Qua các kết quả thực nghiệm có thể nhận thấy rằng thuật toán f-SVM đề xuất có thể được ứng dụng để trích xuất các mô hình mờ từ dữ liệu huấn luyện. Các mô hình mờ trích xuất được có thể ứng dụng cho các bài toán dự báo hồi quy và cho kết quả dự báo tốt hơn so với mô hình SVM nguyên thủy. Ngoài ra qua thực nghiệm cũng chứng tỏ rằng, việc điều chỉnh tăng giá trị của tham số ε để hạn chế cố luật mờ trong mô hình sẽ làm giảm độ phức tạp của mô hình và tốc độ suy luận dựa trên mô hình sẽ tăng lên. Tuy nhiên, song song với đó thì hiệu quả dự đoán dựa trên mô hình sẽ bị suy giảm. Việc cân nhắc lựa chọ giá trị tham số ε để có được số luật mờ thích hợp sẽ được quyết định theo từng bài toán ứng với từng giá trị ngưỡng sai số RMSE.

1.8. Tiểu kết Chương 1

Ở chương này, luận án đã trình bày sự tương đương giữa mô hình mờ TSK và máy học véc-tơ hỗ trợ hồi quy, từ đó đề xuất thuật toán f-SVM không những cho phép trích xuất tập luật mờ TSK từ máy học véc-tơ hỗ trợ hồi quy, mà còn cho phép tối ưu hóa các tham số của các hàm thành viên mờ, điều chỉnh và lựa chọn giá trị tham số ε tối ưu. Bằng thuật toán f-SVM có thể trích xuất được mô hình mờ TSK tối ưu cho từng bài toán dựa vào tập dữ liệu huấn luyện và tập dữ liệu xác thực. Với việc lựa chọn giá trị tham số ε thích hợp cho từng bài toán cụ thể, chúng ta có thể nhận được mô hình vừa đảm bảo tính chính xác khi dự đoán vừa đảm bảo giảm được số luật mờ trong mô hình. Chính việc giảm số luật mờ trong mô hình sẽ làm giảm thời gian thực hiện dự đoán và gia tăng "tính có thể diễn dịch" của mô hình (quan sát các phân bố hàm thành viên mờ ở Hình 1.9).

Kết quả thực nghiệm trên các dữ liệu thử nghiệm cho thấy giải pháp đề xuất thật sự mang lại hiệu quả dự đoán tốt trong sự so sánh với các mô hình như ANFIS, SVM nguyên thủy và mô hình mờ trích xuất từ SVM nhưng chưa tối ưu hóa các tham số; thể hiện qua các giá trị của thông số RMSE. Mặt khác, với mô hình mờ trích xuất được có số luật hạn chế, một trong những hiệu quả mang lại là các chuyên gia trong lĩnh vực dự báo có thể hiểu và phân tích được tập luật này một cách dễ dàng, từ đó có thể đánh giá tập luật mờ và qua đó có giải pháp để tối ưu hóa tập luật.

Tính "có thể diễn dịch được" là một trong những ưu điểm của mô hình mờ so với các mô hình máy học thống kê khác. Tuy nhiên, vấn đề đảm bảo trích xuất được một mô hình mờ "có thể diễn dịch được" từ dữ liệu trong thực tế, đặc biệt đối với mô hình mờ TSK, là một thách thức rất lớn [14]. Qua thực tế các phân bố hàm thành viên mờ của mô hình thực nghiệm trong Hình 1.9b, có thể nhận thấy rằng những phân bố này vẫn chưa thật sự đều và chưa có sự phân biệt rõ ràng, như vậy khả năng diễn dịch ý nghĩa của các luật mờ vẫn còn hạn chế. Đây chính là thách thức đặt ra cần phải được tiếp tục nghiên cứu nhằm tìm ra những giải pháp để có thể cải thiện tính "có thể diễn dịch được" của mô hình. Ở chương tiếp theo của luận án sẽ trình bày những kết quả nghiên cứu về giải pháp tích hợp các tri thức tiên nghiệm (a priori knowledge) vào quá trình học mô hình mờ để có thể cải thiện tính diễn dịch của mô hình mờ.

Chương 2. TÍCH HỢP TRI THỨC TIÊN NGHIỆM VÀO MÔ HÌNH MỜ HƯỚNG DỮ LIỆU

Chương này trước tiên trình bày vai trò của tri thức tiên nghiệm trong việc học một mô hình mờ từ dữ liệu, những kịch bản mà tri thức tiên nghiệm có thể được tích hợp để cải thiện hiệu quả mô hình học được. Tiếp theo là việc xác định và lựa chọn những tri thức tiên nghiệm, đặc biệt là tri thức liên quan đến vấn đề đảm bảo tính "có thể diễn dịch được" của mô hình mờ và cụ thể là tri thức tiên nghiệm trong trường hợp học mô hình mờ dựa trên máy học véc-tơ hỗ trợ. Từ đó đưa đến việc xây dựng thuật toán SVM-IF, trích xuất mô hình mờ có tích hợp tri thức tiên nghiệm.

2.1. Tri thức tiên nghiệm

Tri thức tiên nghiệm (a priori knowledge) được hiểu là tri thức có được trước khi học, nó thường được dùng theo nghĩa đối lập với tri thức hậu nghiệm (a posteriori knowledge), là những tri thức có được sau khi học [65][68][71][75]. Tri thức tiên nghiệm về hệ thống được nghiên cứu có thể ở dưới nhiều dạng khác nhau. Một khác biệt đầu tiên trong tri thức của một mô hình là tri thức mô tả cơ chế hoạt động của mô hình và tri thức tinh túy có từ kinh nghiệm của chuyên gia. Cả hai kiểu khác nhau của tri thức này đều có thể kết hợp với nhau trong một mô hình mờ [49][65]. Việc tích hợp tri thức tiên nghiệm để có thể tạo ra và cải thiện hiệu quả các mô hình thực tế đã được nhiều tác giả nghiên cứu và đề xuất [43][47][65][74][75][80].

Tri thức về qui trình sẵn có, có thể được sử dụng để mô tả hệ thống phi tuyến phức tạp như là một bộ thu thập giản đơn, ví dụ như các hệ thống tuyến tính chỉ có giá trị trong chế độ hoạt động nhất định nào đó. Những thông tin này có thể biểu diễn dưới dạng các qui tắc mờ. Các biến đặc trưng cho sự thay đổi các chế độ hoạt động trở thành một phần của các đối tượng trong hệ thống các qui tắc mờ và hàm thành viên được định nghĩa để xác định cho mỗi mô hình thành phần của một miền nhất định. Trong thực tế, tri thức có trước về lĩnh vực của bài toán là khá quan trọng và cần thiết trong các bài toán như: nhận dạng mẫu, phân lớp, phân cụm, hồi quy, dự

báo, … Những tri thức tiên nghiệm có nhiều dạng khác nhau, từ thông tin của các thuộc tính dữ liệu, chất lượng các mẫu, sự phụ thuộc của các biến, cho đến tính năng của mô hình, …. Nếu biết vận dụng và khai thác đúng cách thì các tri thức tiên nghiệm này có thể giúp cải thiện đáng kể hệ thống ở mọi giai đoạn, bao gồm: chuẩn bị dữ liệu, lựa chọn dữ liệu, lựa chọn tính năng, thiết kế mô hình, diễn giải kết quả và đánh giá hiệu suất mô hình. Đơn giản như trong [45], L.J. Cao và Francis E.H. Tay đã sử dụng tri thức có liên quan về thuộc tính dữ liệu để nâng cao hiệu quả dự đoán giá cổ phiếu. Hay trong [75], V.A. Parasich và các đồng sự đã sử dụng những tri thức tiên nghiệm liên quan đến vấn đề nhận dạng hình ảnh, như việc phân tách các bộ phận của hình ảnh, sự liên quan của các bộ phận hình ảnh, … để tăng hiệu quả nhận dạng hình ảnh. Và nhiều trường hợp khác nữa của giải pháp tích hợp tri thức tiên nghiệm nhằm tăng hiệu quả mô hình được đề xuất trong [11][34][47][68][74][89].

Đối với vấn đề mô hình hóa hệ thống nói chung và vấn đề xây dựng các mô hình mờ nói riêng, các tri thức tiên nghiệm thường liên quan đến các vấn đề như:

- Tầm quan trọng của dữ liệu: trong nhiều ứng dụng thực tế, những mẫu dữ liệu nhất định có thể là ngoại lai và một số có thể bị nhiễu. Do vậy, mô hình xây dựng được từ dữ liệu có thể bị nhiễu hay mất ổn định.

- Hành vi của các máy học: trong một quá trình học tập, không gian giả thuyết của máy học cần được hạn chế trước. Ví dụ, đối với mô hình mạng nơ-ron hồi quy, người ta phải xác định các nguyên mẫu của một vấn đề hồi quy và thiết kế trước các cấu trúc liên kết mạng của một mạng nơ-ron.

- Mục tiêu của các máy học: các tiêu chí như sự ổn định, độ bền vững, thời gian thiết lập là những tri thức phải có trước cho một nhà thiết kế hệ thống.

2.2. Vai trò của tri thức tiên nghiệm trong học mô hình mờ

Trong phần này chúng ta sẽ chứng tỏ vai trò của tri thức tiên nghiệm với việc học một mô hình mờ. Vấn đề xây dựng mô hình mờ, hay mô hình hóa mờ (fuzzy modelling), là gắn liền với việc học từ nhiều nguồn tri thức khác nhau để xác lập các giả thuyết sao cho mô hình nhận được phù hợp với tất cả các ràng buộc cho trước.

Do vậy, một số thuật ngữ trong máy học, đặc biệt là học có sử dụng tri thức tiên nghiệm, có thể được đưa vào trong mô hình hóa mờ để làm rõ vai trò của tri thức tiên nghiệm trong mô hình hóa mờ. Vấn đề đặt ra là cần làm rõ mối quan hệ logic giữa giả thuyết (*Hypothesis*), những mẫu dữ liệu (*Descriptions*) (dưới dạng các thuộc tính), và kết quả dự đoán (*Predictions*). Cho *Descriptions* là hội của tất cả các mẫu dữ liệu trong tập huấn luyện và cho *Predictions* là hội của tất cả các tiên đoán. Khi đó, *Hypothesis* "giải thích các dữ liệu quan sát được" phải thỏa mãn điều kiện sau (ký hiệu $\models$ có nghĩa là suy dẫn logic) [71]:

$$Hypothesis \wedge Descriptions \models Predictions$$

Xét trong trường hợp học mô hình mờ, khái niệm *Hypothesis* có thể được định nghĩa như sau:

Định nghĩa 2.1 (*Hypothesis*). Cho $D = \{(x_i, y_i), x_i \in X, y_i \in Y\}$ với $X, Y \in R$ là một tập dữ liệu huấn luyện (các quan sát thực tế), một mô hình mờ M là được gọi là *Hypothesis* nếu điều kiện sau thỏa mãn:

$$(\forall x_i \in X)(M(x_i) = y_i \in Y)\,.$$

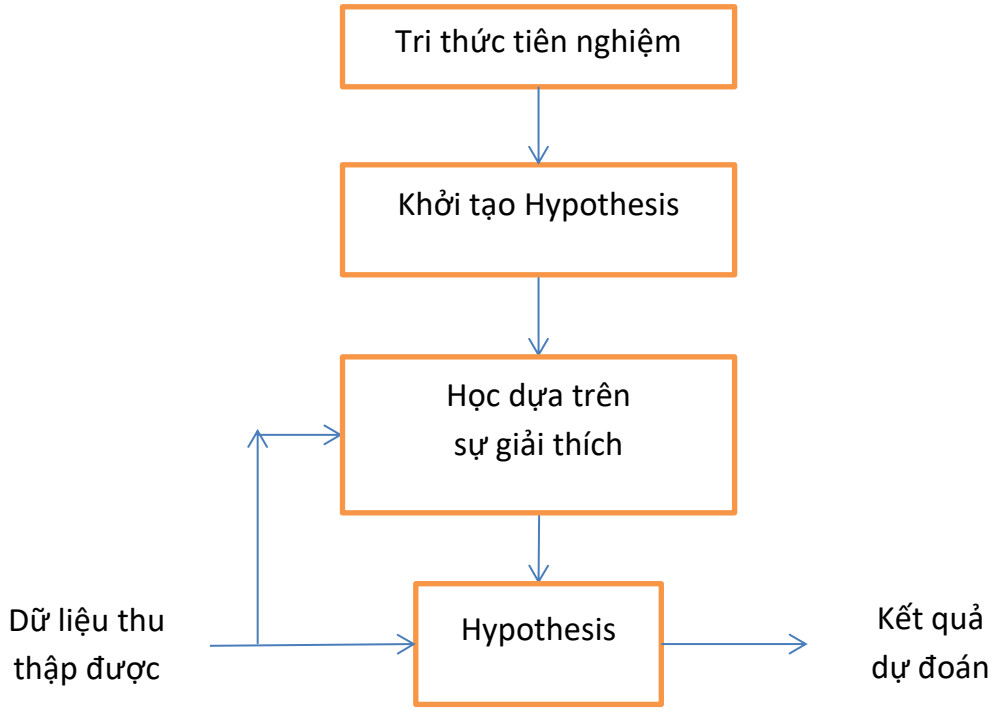
Lý thuyết máy học trong [71] đã định nghĩa 3 kịch bản học với những tri thức tiên nghiệm, gồm: Học dựa trên giải thích (EBL – Explanation-bassed learning), Học dựa trên sự thích hợp (RBL – Relevance-based learning) và Học quy nạp dựa trên tri thức (KBIL – Knowledge-based inductive learning). Các nội dung tiếp theo của Chương này sẽ trình bày các kịch bản học trên trong trường hợp mô hình hóa mờ.

2.2.1. Học dựa trên sự giải thích (EBL)

Kịch bản học EBL là một phương thức trích xuất những luật chung từ các quan sát riêng lẻ. Ý tưởng cơ bản của EBL là sử dụng tri thức tiên nghiệm để xây dựng cấu trúc ban đầu của Hypothesis, rồi sau đó xác lập Hypothesis chính thức dựa vào các quan sát thực nghiệm. Hình 2.1 biểu diễn kịch bản học EBL. Cụ thể trong [71] kịch bản học EBL được mô tả như sau (trong đó $Background$ đại diện cho tri thức tiên nghiệm trong các kịch bản học tương ứng):

$$Background \models Hypothesis$$

$$Hypothesis \wedge Descriptions \models Predictions$$



Hình 2.1. Kịch bản học EBL

Xét một ví dụ học trong thực tế như sau: Trong một phim hoạt hình nổi tiếng của Gary Larson có cảnh một người thượng cổ lớn tuổi, tên là Jog, đang xiên một con thằn lằn trên đầu một chiếc cây nhọn. Ông ta được quan sát bởi một đám đông ngơ ngác những người có trí tuệ hạn chế như thời của ông ta. Những người này chỉ có thói quen cầm nắm những con mồi bắt được trên tay và nướng chúng trên ngọn lửa để làm thức ăn. Chỉ bằng một trải nghiệm trực quan như trên là đủ để thuyết phục những người theo dõi về một nguyên tắc chung trong việc nướng thức ăn mà không làm tổn hại tay. Trong trường hợp ví dụ này, người thượng cổ đã khái quát hóa bằng cách giải thích sự thành công của cây nhọn: nó đỡ được con thằn lằn trong khi tay họ được giữ tránh xa con thằn lằn. Từ sự giải thích quan sát thực nghiệm này, họ có thể rút ra một quy tắc chung là: bất kỳ vật nhọn, dài và cứng nào cũng có thể được dùng để nướng những mảnh thức ăn nhỏ, mềm. Kiểu quy trình trích xuất ra quy tắc chung

từ sự giải thích các quan sát thực nghiệm này được gọi là học dựa trên sự giải thích, hay EBL. Cần lưu ý rằng, quy tắc chung trong trường hợp này tuân theo logic được sở hữu bởi những người thượng cổ.

Xét trong trường hợp học mô hình mờ, kịch bản học EBL được mô tả như sau (xem Hình 2.1):

Cho A là một tri thức có trước về một mô hình mờ M. Việc học mô hình mờ M từ tập dữ liệu quan sát $D = \{(x_i, y_i), x_i \in X, y_i \in Y\}$ với $X, Y \in R$ và tri thức tiên nghiệm A được gọi là học theo kịch bản học EBL nếu thỏa mãn điều kiện sau:

$$A \models M,$$

$$((\forall x_i \in X)(M(x_i) = y_i \in Y)\ .$$

Theo kịch bản học này, tri thức tiên nghiệm có vai trò xác định khuôn mẫu ban đầu của mô hình và mô hình thì được trích xuất từ dữ liệu huấn luyện. Đối với việc học mô hình mờ từ dữ liệu huấn luyện thì việc lựa chọn trước các nguyên mẫu về phân lớp hay hồi quy cho mô hình, rồi sau đó tiến hành huấn luyện mô hình bằng dữ liệu thu thập được, được xem là hình thức học EBL.

Ví dụ, xét bài toán hồi quy sau: Cho một tập dữ liệu quan sát được $D = \{(x_i, y_i), x_i \in X, y_i \in Y\}$ với $X, Y \in R$. Mục đích của bài toán hồi quy là xác định một giả thuyết:

$$y = M(x; \theta), x \in X, y \in Y, \theta \in R\ ,$$

sao cho mô hình mờ M có thể giải thích được tập dữ liệu D. Những tri thức tiên nghiệm được tổng hợp như sau:

- Kiểu của mô hình hồi quy: Kiểu của mô hình hồi quy là vấn đề quan trọng của bài toán hồi quy. Việc xác định kiểu của mô hình hồi quy cho chúng ta xác định được những tham số liên quan của mô hình. Các nguyên mẫu của mô hình hồi quy phổ biến có thể được kể đến là hồi quy tuyến tính, hồi quy đa thức hoặc hàm cơ sở hướng tâm (Radial Basis Functions- RBF).
- Ý nghĩa của "Similar": Phổ biến nhất được dùng để đo lường sự giống nhau giữa các giá trị dữ liệu và đường hồi quy là "sai số khoảng cách" và "sự hợp

lý". Sai số khoảng cách thường dùng là khoảng cách Euclid tiêu chuẩn.

2.2.2. Học dựa trên sự thích hợp (RBL)

Theo kịch bản học này, tri thức tiên nghiệm sẽ kết hợp với những quan sát thực nghiệm để cho phép máy học có thể rút ra những qui tắc mới giải thích cho các thực nghiệm trên. Hình 2.2 biểu diễn kịch bản học RBL. Cụ thể trong [71] kịch bản học RBL được mô tả như sau:

$$Background \wedge Descriptions \wedge Predictions \models Hypothesis$$

$$Hypothesis \wedge Descriptions \models Predictions$$

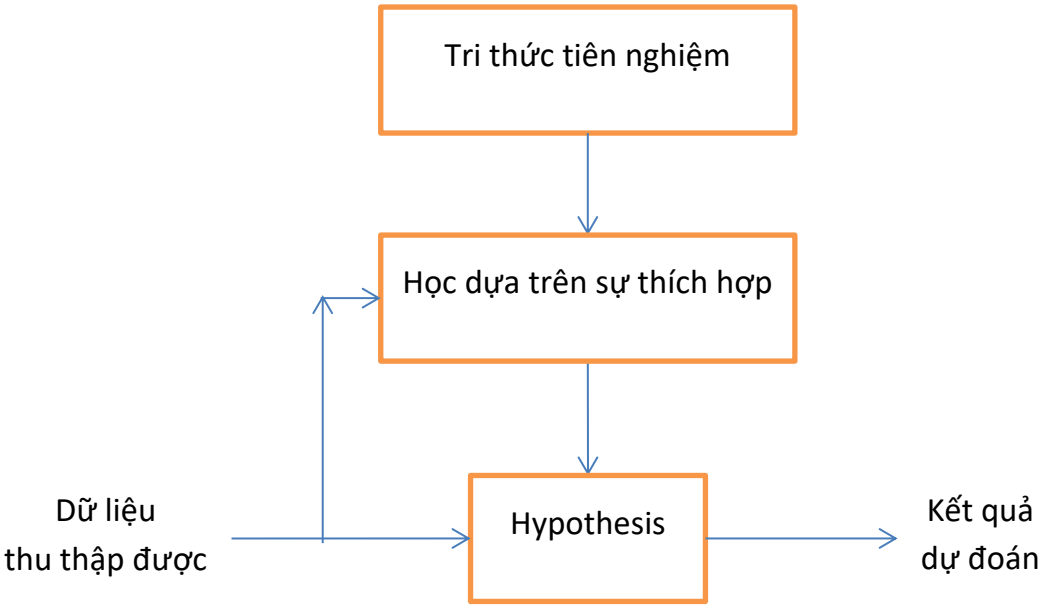
Xét một ví dụ học trong thực tế như sau: Một khách du lịch đến Brazil, khi bắt gặp người Brazil đầu tiên, tên là Ronaldo, nói tiếng Tây ban nha, ngay lập tức vị khách rút ra kết luận là "Người Barzil nói tiếng Tây ban nha". Kết luận tương tự cũng chắc chắn được rút ra ngay cả đối với những vị khách hoàn toàn không có kiến thức về thuộc địa. Tuy nhiên, vị khách hoàn toàn không có kết luận là "Người Brazil có tên là Ronaldo". Trong trường hợp này, vị khách du lịch chỉ thu thập được một mẫu dữ liệu là một người Brazil nói tiếng Tây ban nha, tuy nhiên vị khách đã rút ra kết luận dựa trên tri thức có trước có liên quan trong trường hợp này là "Những người trong cùng một quốc gia thì nói chung một thứ tiếng". Ngược lại, giả thuyết Ronaldo là tên của người Brazil thì không thể rút ra được, vì hoàn toàn không có tri thức tiên nghiệm liên quan đến vấn đề tên riêng của người. Tri thức có trước trong trường hợp này của khách du lịch là tri thức có liên quan đến kết luận về tiếng nói của một cộng đồng người. Kịch bản học để trích xuất ra quy tắc về tiếng nói của người Brazil của khách du lịch trong trường hợp này chính là RBL.

Xét trong trường hợp học mô hình mờ, kịch bản học EBL được mô tả như sau (xem Hình 2.2):

Cho A là tri thức có trước về một mô hình mờ M. Việc học mô hình mờ M từ tập dữ liệu quan sát $D = \{(x_i, y_i), x_i \in X, y_i \in Y\}$ với $X, Y \in R$ và tri thức tiên nghiệm A được gọi là học theo kịch bản học RBL nếu thỏa mãn điều kiện sau:

$$A \wedge D \models M,$$

$$((\forall x_i \in X)(M(x_i) = y_i \in Y) .$$



Hình 2.2. Kịch bản học RBL

Theo phương pháp học này, đối với việc học mô hình mờ thì việc sử dụng những tri thức liên quan về việc xác định cấu trúc mô hình như số lượng biến đầu vào, số lượng quy tắc trong mô hình, … hoặc các tri thức liên quan đến các thuộc tính về chức năng của mô hình đề gia tăng độ vững chắc của mô hình.

Trong môi trường dữ liệu thu thập được bị nhiễu, tri thức tiên nghiệm có thể được tích hợp để gia cố mô hình theo những cách sau:

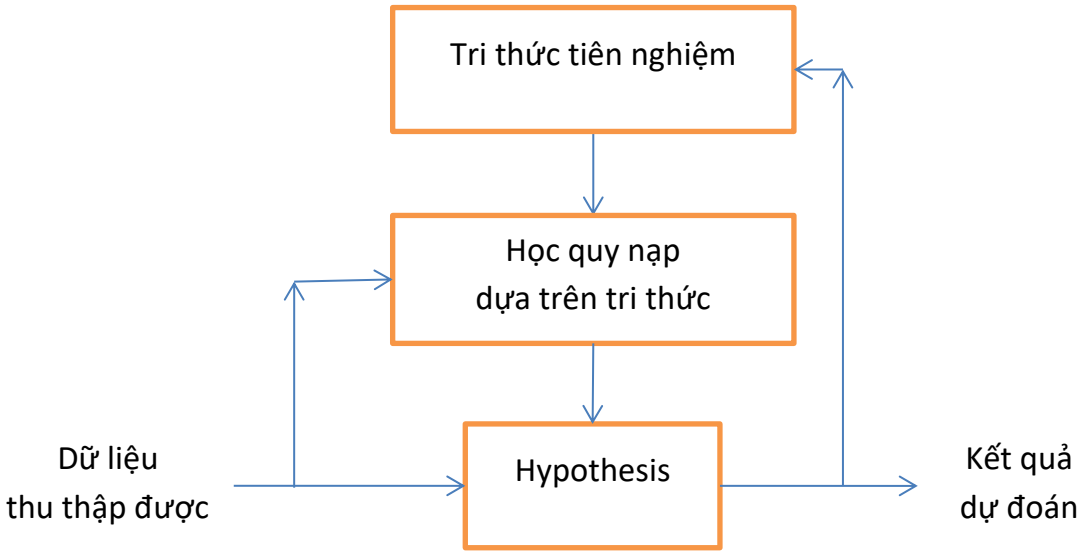
- **Bổ sung dữ liệu huấn luyện đúng:** Một trong những vấn đề mấu chốt của các phương pháp học dựa trên dữ liệu là dữ liệu thu thập được thiếu tính bao phủ, cũng có thể là do kích thước dữ liệu hạn chế. Một cách tự nhiên để vượt qua vấn đề này là sinh ra những dữ liệu đúng trong vùng thích hợp. Những dữ liệu bổ sung đó được xem như là những thông tin liên quan để gia cố mô hình.
- **Bổ sung những ràng buộc liên quan mà mô hình phải đảm bảo:** Các ràng buộc liên quan được bổ sung để định rõ mô hình phải tuân theo được xem là tri thức tiên nghiệm trong kịch bản học RBL. Chẳng hạn trong trường hợp mô

hình hóa các hệ thống điều khiển thì những ràng buộc liên quan như thời gian setting time, rise time, hay tính ổn định của mô hình được xem là những tri thức tiên nghiệm trong trường hợp này.

2.2.3. Học quy nạp dựa trên tri thức (KBIL)

KBIL là một phương thức học theo kiểu tăng cường, trong đó tri thức tiên nghiệm và giả thuyết mới học được sẽ kết hợp với nhau để giải thích cho các quan sát thực nghiệm. Hình 2.3 biểu diễn kịch bản học KBIL. Cụ thể trong [71] kịch bản học KBIL được mô tả như sau:

$$Background \wedge Hypothesis \wedge Descriptions \models Predictions$$



Hình 2.3. Mô hình học KBIL

Theo kịch bản học KBIL, ban đầu tri thức tiên nghiệm và những quan sát thực nghiệm được dùng để xác định *Hypotheses* ban đầu. Sau đó, kết hợp các quan sát thực nghiệm và tri thức tiên nghiệm để củng cố *Hypotheses* và gia tăng tính chính xác của dự đoán.

Xét một ví dụ học trong thực tế như sau: Một sinh viên y khoa có kiến thức chẩn đoán bệnh tốt, nhưng không có kiến thức gì về dược học. Anh ta đang theo dõi một phiên tư vấn giữa một chuyên gia nội khoa với một bệnh nhân. Sau một loạt các câu hỏi và trả lời, chuyên gia chỉ cho bệnh nhân tham gia một khóa học về một loại kháng

sinh đặc biệt M. Sinh viên y khoa lập tức rút ra một quy tắc chung là loại kháng sinh đặc biệt M có hiệu quả cho một loại cụ thể của nhiễm trùng. Trong trường hợp này, giả định là tri thức tiên nghiệm của sinh viên y khoa là đủ để chẩn đoán bệnh của bệnh nhân là D. Tuy nhiên tri thức đó là không đủ để giải thích cho quan sát thực nghiệm lúc này đó là chuyên gia nội khoa kê toa thuốc cụ thể là M. Sinh viên y khoa này phải đề xuất một quy tắc khác, cụ thể là "M là có hiệu quả kháng lại D". Bằng quy tắc này, kết hợp với tri thức tiên nghiệm của mình về chẩn đoán bệnh cho bệnh nhân, sinh viên ý khoa có thể giải thích được tại sao chuyên gia nội khoa đã kê toa thuốc M cho trường hợp bệnh nhân cụ thể này. Quy trình rút ra quy tắc và giải thích quan sát thực nghiệm trong trường hợp này của sinh viên y khoa chính là kịch bản học quy nạp dựa trên tri thức, hay KBIL.

Xét trong trường hợp học mô hình mờ, kịch bản học KBIL được mô tả như sau (xem Hình 2.3):

Cho A là tri thức có trước về một mô hình mờ M. Việc học mô hình mờ M từ tập dữ liệu quan sát $D = \{(x_i, y_i), x_i \in X, y_i \in Y\}$ với $X, Y \in R$ và tri thức tiên nghiệm A được gọi là học theo kịch bản học KBIL nếu thỏa mãn điều kiện sau:

$$((\forall x_i \in X)(A \wedge M(x_i) = y_i \in Y)\ .$$

Lưu ý rằng, theo kịch bản học KBIL, tri thức tiên nghiệm đóng hai vai trò trong việc giảm độ phức tạp học máy:

- Bất kỳ *Hypotheses* nào sinh ra dựa vào KBIL cũng phải phù hợp với tri thức tiên nghiệm cũng như với những quan sát thực nghiệm mới, phạm vi của *Hypotheses* sẽ được thu gọn để chỉ chứa quy tắc thật sự đã biết.
- Với tập dữ liệu quan sát thực tế bất kỳ, phạm vi của *Hypotheses* được rút ra để giải thích cho các quan sát thực tế có thể được rút gọn đáng kể, bởi vì các tri thức tiên nghiệm sẽ giúp cho việc đưa ra những quy tắc mới để giải thích cho các quan sát thực tế. Phạm vi *Hypotheses* càng nhỏ thì càng dễ tìm.

Trong trường hợp học mô hình mờ, việc áp dụng các thuật toán tối ưu hóa mô hình như Gradient descent chính là một hình thức học dựa vào KBIL, bởi vì

Hypotheses (mô hình mờ) sẽ tăng dần sự thích nghi theo quá trình học và phạm vi của *Hypotheses* sẽ thu gọn để phù hợp với những phản hồi từ quan sát thực nghiệm.

2.3. Xác định tri thức tiên nghiệm để tích hợp vào mô hình mờ trích xuất từ máy học véc-tơ hỗ trợ

Đối với máy học véc-tơ hỗ trợ hồi quy, khi số lượng SV tăng lên thì có thể nhận được kết quả đường hồi quy có độ chính xác cao hơn. Tuy nhiên, điều này đồng nghĩa với việc số lượng SV sẽ tăng lên, dẫn đến số luật mờ của mô hình trích xuất được cũng sẽ tăng lên tương ứng, làm cho tính "có thể diễn dịch được" của mô hình giảm đi. Ở phần này chúng ta sẽ bàn luận kỹ hơn về vấn đề "có thể diễn dịch được" của hệ thống mờ và từ đó xác định những tri thức tiên nghiệm có thể tích hợp để có thể trích xuất tập luật mờ "có thể diễn dịch được" từ máy học véc-tơ hỗ trợ.

Tính "có thể diễn dịch được" (interpretability) là một trong những điểm khác biệt cơ bản giữa mô hình máy học thống kê, ví dụ là SVM, và hệ thống mờ [14][37][65]. Một hệ thống mờ yêu cầu phải có đặc tính "có thể diễn dịch được", điều này là khá rõ ràng nếu các luật mờ là được thu thập từ các chuyên gia. Tuy nhiên, đối với một hệ thống mờ được xây dựng dựa vào kết quả của việc học từ dữ liệu thông qua các thuật toán học tự động thì không dễ để có thể đảm bảo được đặc tính có thể hiểu được. Với xu hướng hiện nay, các hệ thống luật mờ được xây dựng dựa vào kết quả học từ dữ liệu trở nên tất yếu và khá phổ biến, chẳng hạn như dựa vào kết quả học của máy học Véc-tơ hỗ trợ; vấn đề đặt ra là làm thế nào để đảm bảo tính có thể diễn dịch được của hệ thống mờ dựa trên dữ liệu.

Trong nội dung tiếp theo, luận án sẽ tập trung phân tích một vài điều kiện, được xem như là những ràng buộc có liên quan, để đảm bảo tính "có thể diễn dịch được" của hệ thống mờ như sau:

Đầy đủ và đa dạng (Completeness and Diversity): Các phân hoạch mờ (fuzzy partition) của tất cả các biến trong hệ thống mờ phải đảm bảo tính đầy đủ và tính phân biệt được. Ngoài ra, số tập mờ trong một phân hoạch mờ cũng cần phải giới hạn. Điều kiện đảm bảo tính đầy đủ và phân biệt được sẽ cho phép gán một giá trị ngôn ngữ rõ

ràng cho mỗi tập mờ trong một phân hoạch mờ. Và thông thường thì điều này sẽ kéo theo số lượng tập mờ trong một phân hoạch mờ sẽ là số nhỏ. Sự phân bố của các tập mờ có thể lượng hóa bằng độ đo sự tương tự (similarity) giữa các tập mờ láng giềng, được định nghĩa như sau [81].

Định nghĩa 2.2 (Độ đo tương tự). Cho A_i và A_j là 2 tập mờ trong tập vũ trụ X. Độ đo sự tương tự giữa 2 tập mờ A_i và A_j là số đo khoảng cách được xác định theo công thức sau:

$$S(A_i, A_j) = \frac{\mathfrak{M}(A_i \cap A_j)}{\mathfrak{M}(A_i) + \mathfrak{M}(A_j) - \mathfrak{M}(A_i \cap A_j)} \tag{2.1}$$

trong đó $\mathfrak{M}(A) = \int_{x \in X} A(x)dx$.

Định nghĩa trên chỉ mang ý nghĩa khái niệm về độ đo tương tự. Về thực tế thì không dễ áp dụng để tính toán cho một kiểu hàm thành viên mờ bất kỳ. Như ở chương trước đã đề cập, hàm thành viên Gauss được chọn để chuyển đổi đầu ra của máy học véc-tơ hỗ trợ thành hệ thống mờ. Như vậy, ở đây chúng ta sẽ xem xét việc tính toán độ đo tương tự cho trường hợp hàm thành viên Gauss.

Mệnh đề 2.1. Nếu hàm thành viên là hàm Gauss $\mu_A(x) = exp\left(-\frac{1}{2}\frac{\|x-\mu\|^2}{\sigma^2}\right)$, với μ là trung tâm và σ xác định phương sai của hàm thành viên mờ, thì độ đo sự tương tự giữa các hàm thành viên được xác định:

$$S^G(A_i, A_j) = \frac{e^{-\frac{d^2}{\sigma^2}}}{2 - e^{-\frac{d^2}{\sigma^2}}} \tag{2.2}$$

là nhất quán với

$$S(A_i, A_j) = \frac{\mathfrak{M}(A_i \cap A_j)}{\mathfrak{M}(A_i) + \mathfrak{M}(A_j) - \mathfrak{M}(A_i \cap A_j)}$$

Tức là, $S_1^G > S_2^G$ nếu và chỉ nếu $S_1 > S_2$ hoặc ngược lại.

Chứng minh. Trước tiên, chúng ta thấy rằng phần giao nhau của hai hàm thành viên Gauss là tỷ lệ với $\sigma e^{-\frac{d^2}{\sigma^2}}$. Sẽ không mất tính tổng quát khi ta giả sử rằng một hàm thành viên xác định tại 0 và hàm còn lại xác định tại d. Khi đó phần giao nhau của hai hàm thành viên A_i, A_j sẽ là $2.I$. Với I được xác định như sau:

$$I = \int_{d/2}^{\infty} exp\left(-\frac{1}{2}\frac{x^2}{\sigma^2}\right) dx \tag{2.3}$$

Đổi biến tích phân trong (2.3), với biến $z = \frac{x}{\sigma}$, ta có công thức tính mới của I:

$$I = \int_{d/2\sigma}^{\infty} \sigma\, exp\left(-\frac{z^2}{2}\right) dz \tag{2.4}$$

Vì $I > 0$, nếu đặt $I^2 = a$ thì $I = \sqrt{a}$. Như vậy ta có:

$$I^2 = \sigma^2 \left[\int_{d/2\sigma}^{\infty} exp\left(-\frac{x^2}{2}\right) dx\right]\left[\int_{d/2\sigma}^{\infty} exp\left(-\frac{y^2}{2}\right) dy\right] \tag{2.5}$$

Tương đương với:

$$I^2 = \sigma^2 \int_{d/2\sigma}^{\infty}\int_{d/2\sigma}^{\infty} exp\left(-\frac{x^2 + y^2}{2}\right) dxdy \tag{2.6}$$

Cho $x = r\, cos\theta$ và $x = r\, sin\theta$, ta có:

$$I^2 = \sigma^2 \int_0^{\pi/2}\int_{d/2\sigma}^{\infty} e^{-r^2/2}\, r\, drd\theta \tag{2.7}$$

Tương đương với:

$$I^2 = \sigma^2 \int_0^{\pi/2} e^{-\frac{1d^2}{2\sigma^2}}\, d\theta \tag{2.8}$$

Hay:

$$I^2 = \frac{1}{2}\, \pi\, \sigma^2 e^{-\frac{1d^2}{8\sigma^2}} \tag{2.9}$$

Như vậy:

$$I = \frac{\sqrt{2\pi}\,\sigma}{2}\, exp\left(-\left(\frac{d}{4\sigma}\right)^2\right) \tag{2.10}$$

Và như vậy, phần giao của hai hàm thành viên Gauss được xác định là:

$$2I = \sqrt{2\pi}\,\sigma\, exp\left(-\left(\frac{d}{4\sigma}\right)^2\right) \tag{2.11}$$

Bằng cách chứng minh tương tự, ta xác định được khu vực của riêng từng hàm thành viên là $\sqrt{2\pi}\,\sigma$.

Như vậy, khi thay vào công thức (2.1) ta có độ đo tương tự của hai hàm thành viên Gauss là:

$$S(A_i, A_j) = \frac{\sqrt{2\pi}\,\sigma\, exp\left(-\left(\frac{d}{4\sigma}\right)^2\right)}{\sqrt{2\pi}\,\sigma + \sqrt{2\pi}\,\sigma - \sqrt{2\pi}\,\sigma\, exp\left(-\left(\frac{d}{4\sigma}\right)^2\right)} \tag{2.12}$$

tương đương với:

$$S(A_i, A_j) = \frac{exp\left(-\left(\frac{d}{4\sigma}\right)^2\right)}{2 - exp\left(-\left(\frac{d}{4\sigma}\right)^2\right)} \tag{2.13}$$

Công thức (2.13) chứng tỏ trong trường hợp hàm thành viên Gauss được chọn thì $S(A_i, A_j)$ nhất quán với $S^G(A_i, A_j)$. Như vậy $S_1^G > S_1^G$ khi và chỉ khi $S_1 > S_2$ và ngược lại.

Hiệu quả (Efficiency): Cấu trúc của hệ thống mờ càng nhỏ gọn càng tốt. Điều này có nghĩa rằng số lượng các biến ngôn ngữ sử dụng trong các luật mờ càng ít càng tốt. Thông qua các kỹ thuật lựa chọn các thuộc tính đầu vào ta có thể xây dựng một hệ thống mờ đảm bảo tính hiệu quả [62]. Bên cạnh đó, số lượng luật mờ trong một hệ thống cũng phải nhỏ. Mối quan hệ giữa độ đo sự tương tự và số lượng luật mờ được xác định theo mệnh đề sau:

Mệnh đề 5.2. Xét một miền D có độ dài L và một tập các hàm thành viên Gauss xác định trên miền D. Nếu mỗi hàm thành viên Gauss có độ lệch chuẩn là σ và độ đo sự tương tự giữa 2 hàm thành viên bất kỳ là nhỏ hơn k, với $0 \leq k \leq 1$, thì số lượng hàm thành viên Gauss thỏa mãn:

$$n < \frac{L}{\sigma\sqrt{ln\frac{1+k}{2k}}} \tag{2.14}$$

Chứng minh. Vì độ đo sự tương tự giữa hai hàm thành viên Gauss không được lớn hơn k, với $0 \leq k \leq 1$, nên ta có:

$$S^G(A_i, A_j) = \frac{e^{-\frac{\left(\frac{L}{n}\right)^2}{\sigma^2}}}{2 - e^{-\frac{\left(\frac{L}{n}\right)^2}{\sigma^2}}} < k \tag{2.15}$$

Với A_i, A_j là hai tập mờ láng giềng xác định cho hai tập mờ tương ứng. Bất đẳng thức trên được viết lại như sau:

$$e^{-\frac{\left(\frac{L}{n}\right)^2}{\sigma^2}} < 2k - ke^{-\frac{\left(\frac{L}{n}\right)^2}{\sigma^2}} \tag{2.16}$$

Tương đương với:

$$e^{-\frac{\left(\frac{L}{n}\right)^2}{\sigma^2}} < \frac{2k}{1+k} \tag{2.17}$$

Lấy logarit cả 2 vế của bất đẳng thức trên, ta có:

$$lne^{-\frac{\left(\frac{L}{n}\right)^2}{\sigma^2}} < ln\frac{2k}{1+k} \tag{2.18}$$

Tương đương với:

$$n < \frac{L}{\sigma\sqrt{ln\frac{1+k}{2k}}} \quad . \tag{2.19}$$

Điều này có nghĩa rằng, khi độ đo sựu tương tự giữa hai tập mờ là nhỏ hơn k, thì số lượng hàm thành viên mờ không vượt quá $\frac{L}{\sigma\sqrt{ln\frac{1+k}{2k}}}$.

Nhất quán (Consistency): Các luật mờ trong một cơ sở luật phải có sự phù hợp với nhau và phù hợp với những tri thức tiên nghiệm sẵn có. Vấn đề không nhất quán của các luật mờ có thể rơi vào các trường hợp như sau:

- Tồn tại hai hoặc nhiều hơn hai quy tắc mờ được định nghĩa trên các sự kiện vào tương tự nhau, nhưng kết luận thì khác nhau.

Ví dụ có 2 quy tắc mờ được xác định tương ứng trên 2 tập mờ A_1 và A_2; kết luận của chúng tương ứng là B_1 và B_2. Nếu $S(A_1, A_2)$ lớn hơn rất nhiều so với $S(B_1, B_2)$ thì 2 luật này có sự kiện vào tương tự nhau, nhưng kết luận thì rất khác nhau. Hình thức không nhất quán này thường xuyên xảy ra đối với mô hình mờ hướng dữ liệu.

- Tồn tại các quy tắc mờ với các phần kết luận của chúng trái ngược nhau. Ví dụ các phần kết luận của các luật mờ không thể xảy ra đồng thời.

Thông thường đối với các hệ thống mờ trích xuất từ dữ liệu, các điều kiện đảm bảo tính "có thể diễn dịch được" của hệ thống ở trên, sẽ bị suy giảm. Trong phần tiếp theo chúng ta sẽ xem xét những điều kiện nào có thể hỗ trợ được trong quá trình học của máy học véc-tơ hỗ trợ khi trích xuất hệ thống mờ.

2.4. Tích hợp tri thức tiên nghiệm vào mô hình mờ trích xuất từ máy học véc-tơ hỗ trợ

2.4.1. Đặt vấn đề

Trong quá trình huấn luyện mô hình mờ từ dữ liệu nói chung, hay quá trình trích xuất mô hình mờ từ dữ liệu dựa trên máy học véc-tơ hỗ trợ nói riêng, việc tích hợp tri thức tiên nghiệm sẽ làm tăng hiệu quả học mô hình [68][71]. Đối với vấn đề trích xuất mô hình mờ dựa vào máy học véc-tơ hỗ trợ, các tri thức tiên nghiệm đã được tích hợp theo các kịch bản học khác nhau.

Trước tiên, phải nhận định rằng, trong quá trình học mô hình mờ dựa vào máy học véc-tơ hỗ trợ, mô hình kết quả là phân lớp hay hồi quy đã được chọn trước khi

huấn luyện mô hình bằng dữ liệu huấn luyện. Điều này chứng tỏ tri thức tiên nghiệm về cấu trúc mô hình đã được tích hợp trong quá trình học theo kịch bản học EBL.

Tiếp theo, với việc lựa chọn tập dữ liệu huấn luyện, lựa chọn các thuộc tính của dữ liệu đầu vào, kết hợp với các kỹ thuật tiền xử lý dữ liệu theo kinh nghiệm của các chuyên gia, thì một kiểu nữa của tri thức tiên nghiệm đó là tri thức có liên quan cũng đã được tính hợp trong việc học mô hình mờ theo kịch bản học RBL.

Sau đó là bước tối ưu hóa tham số các hàm thành viên mờ bằng thuật toán Gradient descent. Việc áp dụng thuật toán Gradient descent trong quá trình trích xuất mô hình mờ đã làm tăng dần độ thích nghi theo quá trình học và mô hình trích xuất sẽ được thu gọn để phù hợp với những phản hồi từ những dữ liệu thu thập được từ thực tế. Trường hợp tối ưu hóa này chính là giải pháp tích hợp tri thức tiên nghiệm vào quá trình học mô hình theo kịch bản KBIL.

Vấn đề quan tâm tiếp theo của chúng ta là làm thế nào để có thể trích xuất được tập luật mờ từ máy học véc-tơ hỗ trợ, sao cho tập luật mờ trích xuất được vẫn đảm bảo đặc tính "có thể diễn dịch được". Chúng ta biết rằng, trong quá trình học mô hình mờ từ dữ liệu, máy học SVM đóng 2 vai trò: xác định cấu trúc của mô hình mờ và các tham số tương ứng. Cấu trúc của mô hình bao gồm: số hàm thành viên, trung tâm của các hàm thành viên; các thành phần này được chuyển đổi trực tiếp từ số lượng và vị trí của các SV.

Từ những điều kiện đảm bảo đặc tính "có thể diễn dịch được" cho hệ thống mờ ở trên, khi xét trong trường hợp cụ thể là hệ thống mờ được trích xuất từ máy học véc-tơ hỗ trợ, các điều kiện sau đây cần phải được thỏa mãn:

- Số lượng luật mờ phải được hạn chế. Điều này cũng đồng nghĩa với việc phải hạn chế số lượng SV. Như đã đề cập ở Mục 2.3, số lượng SV sẽ quyết định số lượng luật mờ được tạo ra. Chính vì vậy số lượng SV cần được hạn chế để đảm bảo trích xuất được một hệ thống mờ "có thể diễn dịch được".
- Những luật mờ dư thừa phải được loại bỏ. Điều kiện để xác định được luật mờ dư thừa là: nếu tồn tại hai hoặc nhiều hơn hai luật mờ trong cùng một vùng mà có độ đo sự tương tự các tập mờ là cao.

Đối với mô hình máy học véc-tơ hỗ trợ, số lượng và vị trí của các SV là không thể xác định được trước khi huấn luyện mô hình. Như vậy để điều khiển số lượng và vị trí của các SV, chúng ta sẽ phải điều chỉnh các tham số liên quan trong mô hình máy học véc-tơ hỗ trợ.

2.4.2. Thuật toán SVM-IF

Nhằm mục tiêu thỏa mãn hai điều kiện ở trên, luận án đề xuất thuật toán SVM-IF, trong Hình 2.4, cho phép trích xuất được hệ thống mờ "có thể diễn dịch được" từ dữ liệu huấn luyện dựa vào máy học véc-tơ hỗ trợ.

Thuật toán SVM-IF($\mathcal{H}$, *k*, ε, *tol*)

Input: *Tập dữ liệu huấn luyện* $\mathcal{H}$;
Ngưỡng độ đo tương tự giữa 2 hàm thành viên sim;
Tham số lỗi ε;

Output: *Mô hình mờ với hàm quyết định đầu ra là* $f(x)$;

1. Khởi tạo các giá trị tham số: $C, \varepsilon, \sigma, step$;
2. Huấn luyện SVM: $f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\, K(x_i, x) + b$;
3. Trích xuất các $SV = \{(\alpha_i - \alpha_i^*): (\alpha_i - \alpha_i^*) \neq 0, i \in \{0, \dots, l\}\}$;
4. InterpretabilityTest(*c, σ, sim*);
5. Điều chỉnh ma trận kernel: $H' = \begin{bmatrix} D' & -D' \\ -D' & D' \end{bmatrix}$, (công thức 1.46)

   với $D'_{ij} = \frac{\langle \varphi(x_i), \varphi(x_j) \rangle}{\sum_j \langle \varphi(x_i), \varphi(x_j) \rangle}$; (công thức 1.47)
6. Sinh ra tập luật mờ từ tập SV với hàm thành viên Gauss;
7. Tối ưu hóa tham số các hàm thành viên (công thức 1.51 và 1.52)

$$\sigma_i(t+1) = \sigma_i(t) + \delta\varepsilon_{1,i}\left[\frac{(x-c)^2}{\sigma^3} exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)\right]$$

$$c_i(t+1) = c_i(t) + \delta\varepsilon_{1,i}\left[\frac{-(x-c)}{\sigma^2} exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)\right]$$

8. **return** $f(x) = \frac{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(x_i, x)}{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)}$

Hình 2.4. Thuật toán SVM-IF

Thuật toán *SVM-IF* có sử dụng đến thuật toán *InterpretabilityTest* (Hình 2.5) để kiểm tra độ đo độ tương tự, $S^G(A_i, A_j)$ của hai tập mờ và gộp chúng lại nếu $S^G(A_i, A_j) \leq sim$. Dòng 2 và 3 cho phép tính toán độ đo độ tương tự và xác định hai tập mờ có độ tương tự lớn nhất. Dòng 4 đến dòng 6 cho phép kiểm tra để gộp các tập mờ có độ tương tự lớn nhất và lơn hơn giá trị sim cho trước, đồng thời cập nhật lại tập mờ mới. Việc kiểm tra và gộp các tập mờ sẽ được thực hiện lặp lại, mỗi lần gộp một cặp cho đến khi độ đo sự tương tự giữa tất cả các tập mờ không lớn hơn giá trị sim cho trước.

Thuật toán InterpretabilityTest(*c, σ, sim*)

Input: *Tập các véc-tơ hỗ trợ (trung điểm của hàm thành viên Gauss) c;*
Tham số xác định độ lệch chuẩn σ;
Tham số ngưỡng độ tương tự cho trước sim;

Output: *Tập các véc-tơ hỗ trợ đã được rút gọn;*

1. **repeat**
2. Tính độ đo sự tương tự giữa các cặp tập mờ A_i, A_j (theo công thức 2.2):

$$S^G(A_i, A_j) = \frac{e^{-\frac{d^2}{\sigma^2}}}{2 - e^{-\frac{d^2}{\sigma^2}}}$$

$$với\ d = \sqrt{(c_i - c_j)^2 + (\sigma_i - \sigma_j)^2}$$

3. Lựa chọn một cặp tập mờ A_i^* và A_j^* sao cho:

$$S^G(A_i^*, A_j^*) = max_{i,j}\{S^G(A_i, A_j)\}$$

4. **if** $S^G(A_i^*, A_j^*) > sim$ **then**
5. Gộp cặp tập mờ A_i^* và A_j^* thành một tập mờ mới A_k;
6. **end if**
7. **until** không còn căp tập mờ nào có độ đo sự tương tự $S^G(A_i, A_j) > sim$;
8. **Return**

Hình 2.5. Thuật toán *InterpretabilityTest*

Cho số lượng véc-tơ hỗ trợ là l, ta có độ phức tạp tính toán của thuật toán *InterpretabilityTest* là $O(l^2)$. Trong khi độ phức tạp của thuật toán huấn luyện máy học véc-tơ hỗ trợ có độ phức tạp là $O(N^2)$ (với N là kích thước tập dữ liệu huấn luyện). Như vậy độ phức tạp của thuật toán SVM-IF là $O(N^2 + l^2) = O(N^2)$.

2.4.3. Qui trình trích xuất mô hình mờ dựa trên thuật toán SVM-IF có lựa chọn giá trị tối ưu cho các tham số

Theo thuật toán SVM-IF đề xuất, quá trình thực hiện trích xuất tập luật mờ từ dữ liệu huấn luyện đầu vào có tich hợp tri thức tiên nghiệm để tối ưu hóa số lượng cũng như phân bố các hàm thành viên, đồng thời thay đổi để chọn lựa giá trị tham số ε thông qua thực nghiệm dự đoán trên tập dữ liệu xác thực, được thể hiện ở Hình 2.6. Các bước được thực hiện như sau:

Input: Tập dữ liệu huấn luyện $\mathcal{H}$ và các tham số *k, ε, tol*

Output: Mô hình mờ với các tham số tối ưu

Bước 1. Khởi tạo các tham số cho máy học véc-tơ hỗ trợ hồi quy: C, ε, σ

Bước 2. Huấn luyện máy học véc-tơ hỗ trợ để xác định các véc-tơ hỗ trợ (cũng chính là các giá trị trung bình của các hàm thành viên) và các giá trị tham số xác định phương sai tương ứng: $c_i, \sigma_i, với\ i = 1,2,..l$

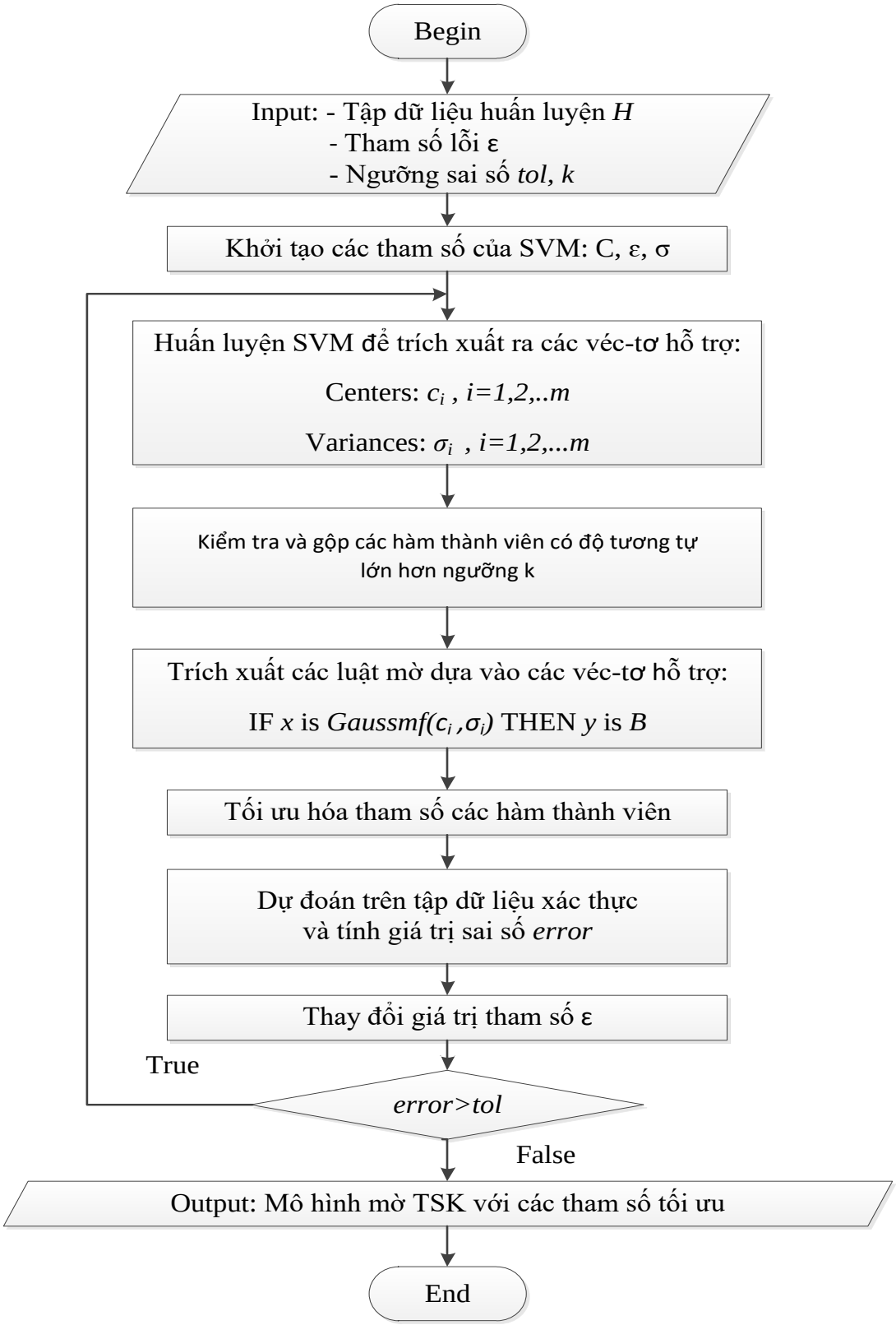
Bước 3. Lặp lại việc kiểm tra và gộp các tập mờ, mỗi lần gộp một cặp cho đến khi độ đo sự tương tự giữa tất cả các tập mờ không lớn hơn giá trị sim cho trước.

Bước 4. Trích xuất tập luật mờ dựa trên các cặp giá trị (c_i, σ_i), sử dụng hàm thành viên mờ Gauus. Hàm đầu ra của hệ thống được xác định bằng công thức:

$$f(x) = \frac{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\mathrm{K}(\mathrm{x_i}, \mathrm{x})}{\sum_{i=1}^{l}\mathrm{K}(\mathrm{x_i}, \mathrm{x})}$$

Bước 5. Thực hiện tối ưu hóa các tham số của hàm thành viên mờ

Bước 6. Lựa chọn giá trị tham số $\boldsymbol{\varepsilon}$ tối ưu bằng cách lặp lại việc huấn luyện mô hình và thực nghiệm dự báo trên tập dữ liệu xác thực



Hình 2.6. Quy trình trích xuất tập luật mờ TSK từ máy học véc-tơ hỗ trợ có tích hợp tri thức tiên nghiệm

Bước 7. Trích xuất tập luật mờ TSK với các tham số và phân bố các hàm thành viên mờ đã tối ưu hóa.

Với kích thước của tập dữ liệu xác thực là $k << N$ và T là số lần lặp lại việc huấn luyện mô hình và thực hiện dự báo trên tập dữ liệu xác thực để đánh giá sai số *error*, thì độ phức tạp của thuật toán trích xuất tập luật mờ TSK có cho phép lựa chọn giá trị tối ưu cho các tham sẽ là $O(T.N^2)$.

2.5. Tổ chức thực nghiệm

2.5.1. Mô tả thực nghiệm

Để đánh giá thuật toán SVM-IF đã đề xuất, chúng tôi xây dựng một hệ thống thử nghiệm dựa trên bộ công cụ Matlab. Tương như trong thực nghiệm thuật toán f-SVM ở Chương 1, thuật toán học SVM của thư viện LIBSVM được phát triển bởi nhóm của Chih-Chung Chang [20], được sử dụng để sản xuất ra các SV, làm cơ sở để trích xuất các luật mờ trong thuật toán SVM-IF. Sau cùng, hàm AVALFIS trong thư viện công cụ Matlab Fuzzy Logic được sử dụng để suy luận dựa trên mô hình mờ trích xuất được.

Bên cạnh việc thực đối với mô hình mờ trích xuất theo thuật toán SVM-IF, các thực nghiệm trên các mô hình f-SVM, ANFIS và SVM nguyên thủy như trong Chương 1 cũng được tiến hành thực nghiệm trên cùng bộ dữ liệu để có sự so sánh, đánh giá hiệu quả của mô hình. Thuật toán SVM-IF cùng với một số kết quả thực nghiệm trên một số bài toán ví dụ đã được công bố ở công trình [A1], [A9].

Trong trường hợp thực nghiệm này, hai bài toán ví dụ là hàm hồi quy phi tuyến $Sinc(x)$ và chuỗi thời gian hỗn loạn *Mackey-Glass* đã thực nghiệm ở Chương 1 sẽ được thực nghiệm lại với thuật toán SVM-IF để có cơ sở so sánh, đánh giá hiệu quả của thuật toán. Ngoài ra, hệ thống Lorenz được đề xuất bởi Lorenz E.N [46], với mục tiêu là dùng để mô tả những hành vi bất thường của thời tiết cũng được chọn để thực nghiệm theo như đề xuất của Y. Jin [80].

2.5.2. Bài toán hồi quy phi tuyến

Để chứng tỏ hiệu quả của mô hình mờ dựa trên SVM và việc phát hiện tri thức tiên nghiệm về cấu trúc mô hình, cụ thể là số lượng luật mờ, thực nghiệm hồi quy phi tuyến được triển khai trên tập dữ liệu sinh ra từ hàm $Sinc(x)$ (công thức 1.54). Tập dữ liệu huấn luyện được xác định trong phạm vi từ -3π đến $+3\pi$. Với 50 mẫu dữ liệu sinh ra ngẫu nhiên vừa được dùng để là dữ liệu huấn luyện vừa được dùng làm dữ liệu xác thực.

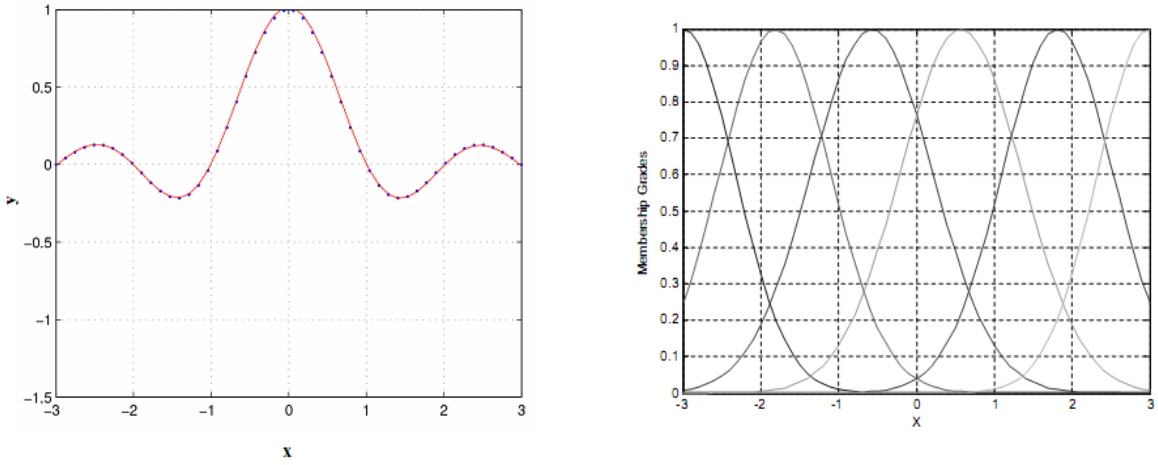
Trong quá trình xác định cấu trúc SVM, tham số ε được thay đổi để điều chỉnh số lượng SV. Trước tiên, cố định tham số C =10. Sau đó, giá trị tham số được chọn là $\varepsilon = 0.001$ và thay đổi tăng dần đến 0.5. Khi giá trị tham số $\varepsilon = 0.08$ thì có 6 SV nhận được tương ứng với 6 luật mờ thể hiện ở Bảng 2.1. Hình 2.7 thể hiện đường kết quả dự đoán trên dữ liệu xác thực và phân bố các hàm thành viên của của mô hình thực nghiệm với $\varepsilon = 0.08$ và 6 luật mờ trích xuất được đã tối ưu hóa.

Bảng 2.1. Tập 6 luật trích xuất được từ mô hình đã tối ưu hóa

Thứ tự	Luật
R1	IF x is Gaussmf(0.66, -2.99) THEN y is 0.418
R2	IF x is Gaussmf(0.71, -1.813) THEN y is -1.741
R3	IF x is Gaussmf(0.78, -0.572) THEN y is 1.32
R4	IF x is Gaussmf(0.78, 0.572) THEN y is 1.32
R5	IF x is Gaussmf(0.71, 1.813) THEN y is -1.741
R6	IF x is Gaussmf(0.66, 2.99) THEN y is 0.418

So sánh phân bố các hàm thành viên mờ của 6 luật trong Hình 2.6b với phân bố 6 luật ở Hình 1.7b, ta thấy phân bố ở Hình 2.7b đã được tối ưu, các hàm thành viên có sự phân bố đều hơn. Như vậy chúng ta có thể điều chỉnh giá trị tham số ε, tức điều chỉnh số lượng SV để tối ưu hóa vị trí của các SV, đồng nghĩa với việc tối ưu hóa phân bố và số lượng luật mờ. Với việc tích hợp tri thức tiên ngiệm dựa trên độ đo

tương tự của các hàm thành viên mờ, kết quả có được phân bố các hàm thành viên rõ hơn, tức là tính diễn dịch của mô hình mờ tăng lên.



(a) (b)

Hình 2.7. Kết quả mô hình đã tối ưu hóa (RMSE = 0.0183)

Bảng 2.2. So sánh kết quả các mô hình qua thông số RMSE

Số luật mờ/Số véc-tơ hỗ trợ	Mô hình áp dụng			
	ANFIS	SVM	Mô hình f-SVM	Mô hình SVM-IF
50	$<10^{-10}$	0.0074	$< 10^{-10}$	---
30	$<10^{-10}$	0.0572	$< 10^{-10}$	---
10	0.0017	0.0697	0.0015	0.0011
8	0.0018	0.0711	0.0013	0.0010
6	0.0248	0.2292	0.0197	0.0183
4	0.1894	0.2851	0.0553	0.0553

Bảng 2.2 thể hiện kết quả so sánh hiệu quả của mô hình đề xuất sử dụng thuật toán SVM-IF với các mô hình ANFIS, mô hình SVM nguyên thủy, và mô hình sử dụng thuật toán f-SVM ở Chương 1. Tất cả các thực nghiệm đều dùng chung một bộ

dữ liệu huấn luyện (cũng đồng thời là dữ liệu xác thực). Kết quả dự đoán với dữ liệu xác thực trong các trường hợp đều được tính sai số RMSE. Kết quả so sánh cho thấy, đối với trường hợp bài toán cụ thể này, mô hình đề xuất khi đã tối ưu hóa (6 luật) có kết quả tốt hơn các mô hình khác. So sánh 2 cột giá trị của RMSE trong trường hợp áp dụng thuật toán f-SVM và SVM-IF, ta thấy với cùng số luật mờ trích xuất được thì kết quả dự đoán với mô hình áp dụng thuật toán SVM-IF có giá trị sai số nhỏ hơn. Điều này cho thấy, việc tích hợp tri thức tiên nghiệm, cụ thể ở đây là tri thức về độ đo tương tự của các tập mờ đã giúp tối ưu hóa vị trí các hàm thành viên trong mô hình mờ và từ đó cải thiện được hiệu quả áp dụng mô hình.

Ngoài ra, với tập luật mờ trích xuất được đã được rút gọn và tối ưu hóa phân bố, con người có thể dễ dàng diễn dịch ngữ nghĩa cho tập luật. Bảng 2.3 thể hiện các luật đã được diễn dịch ngữ nghĩa cho tập luật trích xuất trong Bảng 2.1.

Bảng 2.3. Diễn dịch ngữ nghĩa cho các luật ở Bảng 2.1

Thứ tự	Luật
R_1	*IF x xấp xỉ -2.99 THEN y = 0.418*
R_2	*IF x xấp xỉ -1.813 THEN y = -1.741*
R_3	*IF x xấp xỉ -0.572 THEN y = 1.32*
R_4	*IF x xấp xỉ 0,572 THEN y = 1.32*
R_5	*IF x xấp xỉ 1.813 THEN y = -1.741*
R_6	*IF x xấp xỉ 2.99 THEN y = 0.418*

2.5.3. Bài toán dự báo dữ liệu chuỗi thời gian hỗn loạn Mackey-Glass

Thực nghiệm này nhằm chứng tỏ mô hình mờ trích xuất từ dữ liệu huấn luyện dựa trên SVM với sự tích hợp tri thức tiên nghiệm thật sự mang lại hiệu quả. Dữ liệu được lựa chọn để thử nghiệm là dữ liệu chuỗi thời gian Mackey-Glass. Tập dữ liệu sử dụng được sinh ra từ công thức (1.55).

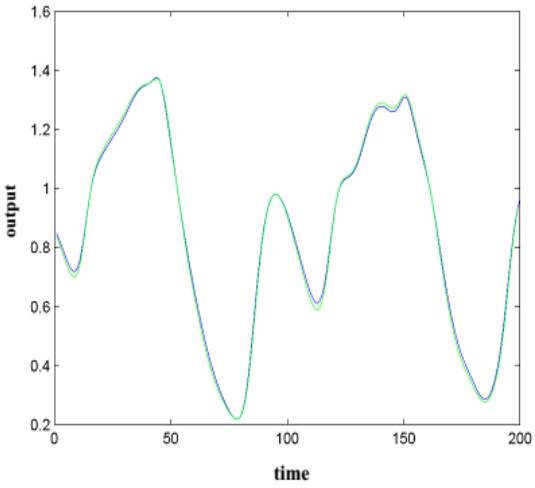
Trong đó ta chọn $\tau = 30$, $a = 0.2$, $b = 10$, và $c = 0.1$. Với 1000 mẫu dữ liệu sinh ra, 800 mẫu dữ liệu được sử dụng để huấn luyện cho máy học véc-tơ hỗ trợ và sinh ra các luật mờ, 200 mẫu dữ liệu còn lại được sử dụng để thử nghiệm suy luận dựa trên tập luật mờ trích xuất được. Thuộc tính đầu vào được lựa chọn là các giá trị $x(t-1)$, $x(t-2)$, thuộc tính đầu ra cần dự đoán là giá trị $x(t)$. Như vậy mô hình có 2 đầu vào và 1 đầu ra.

Tương tự với ví dụ trước, trong thực nghiệm này giá trị tham số $C = 10$. Khi thiết lập giá trị cho tham số $\varepsilon = 0.0$ thì kết quả có 200 luật mờ nhận được. Giá trị tham số ε được điều chỉnh tăng dần. Khi $\varepsilon = 0.1$, hệ thống mờ thu được gồm có 9 luật như thể hiện ở Bảng 2.4. Trong trường hợp này, với 9 luật mờ trích xuất được, chỉ có 3 hàm thành viên tương ứng với biến $x(t-1)$ đó là: *Gaussmf(0.52, 0.51)*, *Gaussmf(0.66, 1.09*) và *Gaussmf(0.52, 0.51)*. Tương tự cũng chỉ 3 hàm thành viên tương ứng với biến $x(t-2)$, đó là: *Gaussmf(0.56, 0.48), Gaussmf(0.56, 0.38)* và *Gaussmf(0.65, 1.07)*. Hình 2.8b thể hiện phân bố của 3 hàm thành viên tương ứng với biến $(t-2)$ trong trường hợp này.

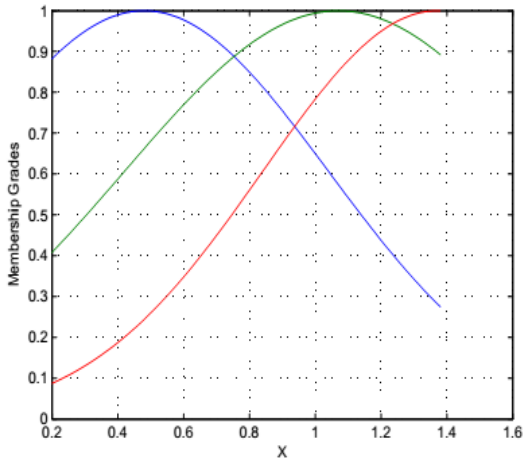
Bảng 2.4. Tập 9 luật trích xuất được từ 800 mẫu dữ liệu huấn luyện của thực nghiệm 2.5.3

Thứ tự	Luật
R_1	*IF x(t-2) is Gaussmf(0.56, 0.48) and x(t-1) is Gaussmf(0.52, 0.51) THEN x(t) is 1.12*
R_2	*IF x(t-2) is Gaussmf(0.56, 0.48) and x(t-1) is Gaussmf(0.66, 1.09) THEN x(t) is 1.08*
R_3	*IF x(t-2) is Gaussmf(0.56, 0.38) and x(t-1) is Gaussmf(0.53, 1.39) THEN x(t) is 0.97*
R4	*IF x(t-2) is Gaussmf(0.65, 1.07) and x(t-1) is Gaussmf(0.52, 0.51) THEN x(t) is 1.32*

R5	*IF x(t-2) is Gaussmf(0.65, 1.07) and x(t-1) is Gaussmf(0.66, 1.09) THEN x(t) is 0.94*
R6	*IF x(t-2) is Gaussmf(0.65, 1.07) and x(t-1) is Gaussmf(0.53, 1.39) THEN x(t) is 1.11*
R7	*IF x(t-2) is Gaussmf(0.53, 1.37) and x(t-1) is Gaussmf(0.52, 0.51) THEN x(t) is 1.11*
R8	*IF x(t-2) is Gaussmf(0.53, 1.37) and x(t-1) is Gaussmf(0.66, 1.09) THEN x(t) is 1.09*
R9	*IF x(t-2) is Gaussmf(0.53, 1.37) and x(t-1) is Gaussmf(0.53, 1.39) THEN x(t) is 0.98*



(a)



(b)

Hình 2.8. Kết quả dự đoán trên 200 mẫu dữ liệu xác thực của thực nghiệm 2.5.3 (trường hợp RMSE = 0.0092)

Bên cạnh việc thực nghiệm dự đoán giá trị $x(t)$ trên bộ dữ liệu thử nghiệm (200 mẫu dữ liệu) sử dụng mô hình mờ trích xuất được từ dữ liệu huấn luyện với các thuật toán f-SVM và SVM-IF; các thực nghiệm với các mô hình ANFIS và SVM nguyên thủy cũng được tiến hành trên cùng bộ dữ liệu. Hiệu quả dự đoán của các mô hình trên 200 mẫu dữ xác thực được so sánh và đánh giá dựa trên thông số RMSE.

So sánh các giá trị của RMSE trong Bảng 2.5 ta có thể nhận thấy rằng mô hình ứng dụng thuật toán SVM-IF cho kết quả dự đoán tương đương với mô hình ANFIS và tốt hơn so với mô hình SVM. So sánh giá trị của RMSE trên hai cột tương ứng là mô hình f-SVM và mô hình SVM-IF, ta thấy: với cùng số lượng luật mờ trong mô hình, giá trị sai số RMSE của mô hình SVM-IF là bé hơn so với mô hình f-SVM.

Bảng 2.5. So sánh kết quả các mô hình qua thông số RMSE

Số luật mờ	Mô hình áp dụng			
	ANFIS	SVM	Mô hình f-SVM	Mô hình SVM-IF
170	$<10^{-10}$	0.0540	$<10^{-10}$	$<10^{-10}$
36	0.0034	0.0509	0.0086	0.0076
25	0.0041	0.0635	0.0092	0.0090
14	0.0050	0.0748	0.0095	0.0091
9	0.0074	0.1466	0.0098	0.0092
4	0.0087	0.1955	0.0102	0.0088

2.5.4. Hệ thống Lorenz

Hệ thống Lorenz lần đầu tiên được đề xuất bởi E. N. Lorenz năm 1963, được mô tả bằng công thức sau [46][76]:

$$\begin{cases} \frac{dx}{dt} = -\delta(x - y) \\ \frac{dy}{dt} = -xz + \gamma x - y \\ \frac{dz}{dt} = xy - bz \end{cases} \quad (2.20)$$

Trong đó các thành phần δ, γ, và b có giá trị tương ứng là $\delta = 10$, $\gamma = 28$, và $b = 8/3$. Trong thực nghiệm này, chúng tôi dự đoán các giá trị $x(t)$, $y(t)$ và $z(t)$ dựa

vào các giá trị $x(t-1)$, $y(t-1)$ và $z(t-1)$. Bằng cách sử dụng phương pháp Runge-Kutta bậc 4 với khoảng cách bước là 0.05, chúng ta tạo ta 2000 mẫu dữ liệu; trong đó 1000 mẫu dữ liệu được dùng để huấn luyện và các mẫu còn lại dùng để thử nghiệm xác thực mô hình.

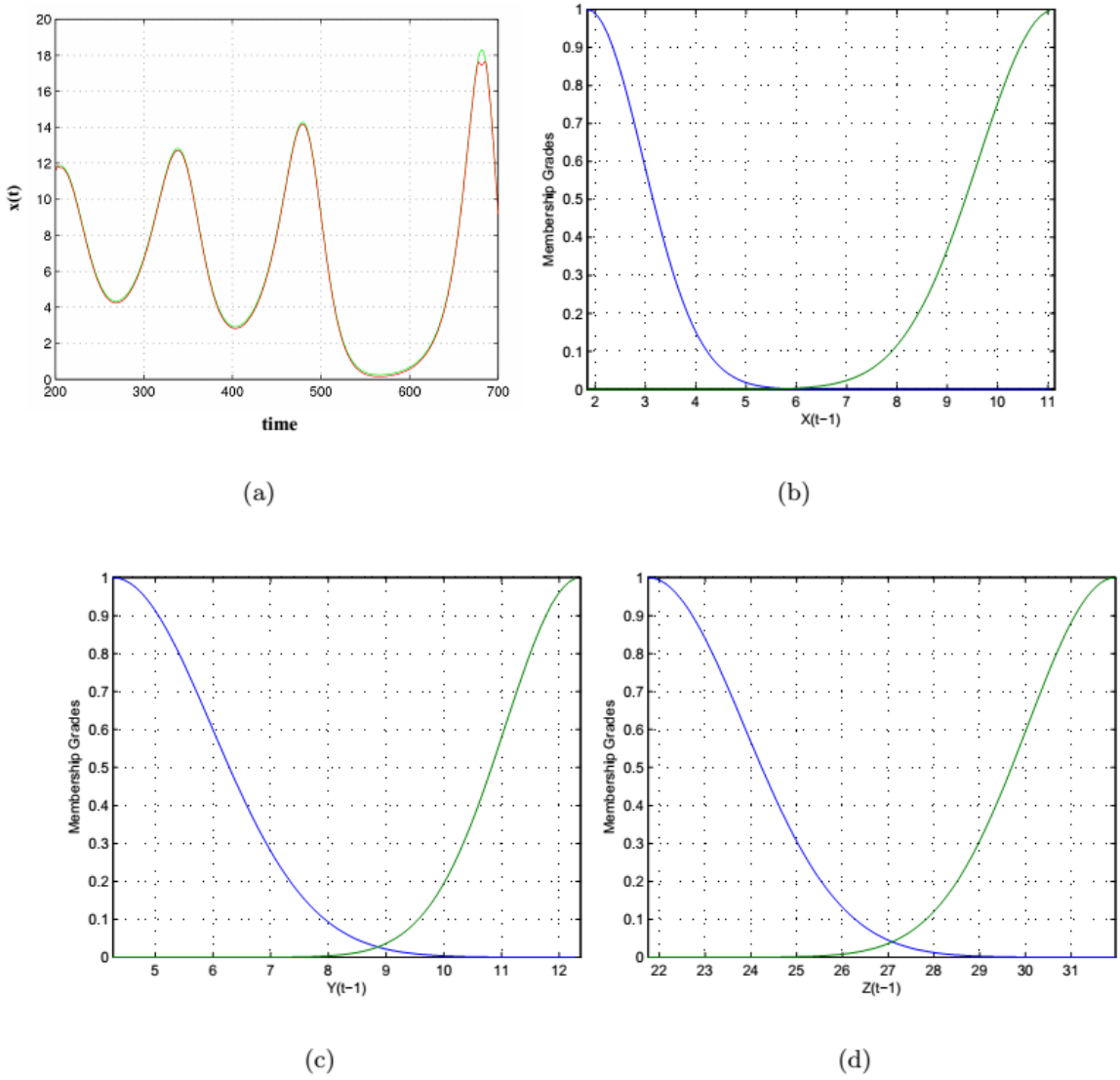
Bảng 2.6. Tập luật trích xuất được từ 1000 mẫu dữ liệu huấn luyện

Thứ tự	Luật
R_1	*IF x(t-1) is Gaussmf(0.56, 0.48) and y(t-1) is Gaussmf(8.63, -21.98) and z(t-1) is Gaussmf(12.52, -12.51) THEN x(t) is 11.27*
R_2	*IF x(t-1) is Gaussmf(0.56, 0.48) and y(t-1) is Gaussmf(8.63, -21.98) and z(t-1) is Gaussmf(15.11, 32.77) THEN x(t) is 4.32*
R_3	*IF x(t-1) is Gaussmf(0.56, 0.48) and y(t-1) is Gaussmf(7.88, 25.14) and z(t-1) is Gaussmf(12.52, -12.51) THEN x(t) is 6.77*
R4	*IF x(t-1) is Gaussmf(0.56, 0.48) and y(t-1) is Gaussmf(7.88, 25.14) and z(t-1) is Gaussmf(15.11, 32.77) THEN x(t) is 10.89*
R5	*IF x(t-1) is Gaussmf(4.51, 21.55) and y(t-1) is Gaussmf(8.63, -21.98) and z(t-1) is Gaussmf(12.52, -12.51) THEN x(t) is 4.61*
R6	*IF x(t-1) is Gaussmf(4.51, 21.55) and y(t-1) is Gaussmf(8.63, -21.98) and z(t-1) is Gaussmf(15.11, 32.77) THEN x(t) is 10.02*
R7	*IF x(t-1) is Gaussmf(4.51, 21.55) and y(t-1) is Gaussmf(7.88, 25.14) and z(t-1) is Gaussmf(12.52, -12.51) THEN x(t) is 16.38*
R8	*IF x(t-2) is Gaussmf(4.51, 21.55) and y(t-1) is Gaussmf(7.88, 25.14) and z(t-1) is Gaussmf(15.11, 32.77) THEN x(t) is 17.33*

Tập luật tối ưu khi huấn luyện mô hình sử dụng thuật toán SVM-IF bằng 1000 mẫu dữ liệu huấn luyện được thể hiện ở Bảng 2.6. Với tập 8 luật có sự phân biệt rõ ràng này sẽ dễ dàng diễn dịch ngữ nghĩa, điều này thể hiện ưu điểm của tính "có thể

diễn dịch được" của mô hình mờ. Các Hình 2.9b,c,d thể hiện sự phân bố của các hàm thành viên tương ứng với các biến đầu vào x(t-1), y(t-1) và x(t-1). Số hàm thành viên tương ứng của mỗi biến đầu vào chỉ là 2 hàm, đồng thời phân bố cũng đều và khá rõ ràng.

Kết quả thực nghiệm dự đoán trên tập dữ liệu xác thực của mô hình sau khi đã tối ưu hóa thể hiện bằng đồ thị ở Hình 2.9a.



Hình 2.9. (a) Kết quả mô hình đã tối ưu hóa (RMSE = 0.0043), (b)(c)(d) Phân bố các hàm thành viên tương ứng với x(t-1), y(t-1) và z(t-1)

Bảng 2.7. So sánh kết quả các mô hình qua thông số RMSE

Số luật mờ / Số véc-tơ hỗ trợ	Mô hình áp dụng		
	ANFIS	Mô hình f-SVM	Mô hình SVM-IF
150	---	0.0110	$<10^{-10}$
144	---	0.9966	$2.05*10^{-8}$
142	---	1.9970	$2.10*10^{-8}$
139	---	2.9837	$4.74*10^{-8}$
134	---	3.9431	$3.55*10^{-8}$
127	---	4.8669	$4.64*10^{-8}$
89	---	5.6453	$5.70*10^{-8}$
72	---	6.2638	$1.47*10^{-5}$
56	---	6.7905	$8.57*10^{-5}$
44	---	7.2302	$9.44*10^{-5}$
27	0.0033	7.5741	$1.32*10^{-5}$
8	0.0515	7.7502	0.0043
7	---	7.7857	0.3603

Bảng 2.7 thể hiện kết quả dự đoán thử nghiệm trên 500 mẫu dữ liệu thử nghiệm của mô hình đề xuất và các mô hình đối sánh khác thông qua thông số RMSE. Trong thực nghiệm này, mô hình ANFIS chỉ thực nghiệm với số luật tương ứng là 27 và 8 luật để so sánh giá trị sai số RMSE với hai mô hình f-SVM và SVM-IF. Đặc biệt với trường hợp rút gọn còn 8 luật, sai số của kết quả dự đoán của mô hình SVM-IF là 0.0043, nhỏ hơn so với sai số tương ứng của mô hình ANFIS và nhỏ hơn rất nhiều so với mô hình ứng dụng thuật toán f-SVM; trong khi đó số hàm thành viên tương ứng

với mỗi biến đầu vào được rút gọn thành 2 hàm (Hình 2.9b,c,d). Kết quả đánh giá sai số RMSE ở Bảng 2.7 cho thấy hiệu quả của mô hình ứng dụng thuật toán SVM-IF so với các mô hình khác, đặc biệt là khi đã tối ưu hóa tập luật với số luật là 8.

2.6. Tiểu kết Chương 2

Trong chương này Luận án đã nghiên cứu một hướng tiếp cận tích hợp tri thức tiên nghiệm với việc học mô hình mờ để có thể trích xuất được hệ thống mờ tốt hơn từ dữ liệu huấn luyện. Với các kịch bản khác nhau của việc học mô hình có sự kết hợp của tri thức tên nghiệm, chúng ta có thể xây dựng các mô hình mờ hướng dữ liệu trong đó có sự tích hợp các kiểu tri thức tiên nghiệm để cải tiến hiệu quả của mô hình. Cụ thể trong trường hợp trích xuất mô hình mờ TSK từ máy học véc-tơ hỗ trợ hồi quy, các tri thức tiên nghiệm về cấu trúc mô hình (số lượng và phân bố các hàm thành viên) đã được tích hợp trong quá trình học mô hình thông qua thuật toán SVM-IF.

Với những kết quả thực nghiệm mô hình sử dụng thuật toán SVM-IF trên một số ví dụ cụ thể, kết hợp so sánh với các kết quả thực nghiệm trên các mô hình khác, như ANFIS, SVM và f-SVM đã chứng tỏ được tính khả thi và hiệu quả của việc tích hợp tri thức tiên nghiệm cải tiến hiệu quả mô hình mờ hướng dữ liệu.

Với tập luật mờ trích xuất được bằng cách sử dụng thuật toán SVM-IF đề xuất có số lượng hạn chế, đã được tối ưu hóa phân bố, đảm bảo tính "có thể diễn dịch được", những chuyên gia trong lĩnh vực ứng dụng cụ thể dễ dàng diễn dịch ngữ nghĩa cho các luật này. Trên cơ sở phân tích ngữ nghĩa các tập luật mờ trích xuất từ dữ liệu, các chuyên gia trong lĩnh vực dự báo có thể điều chỉnh loại bỏ các luật không phù hợp, bổ sung các luật chuyên gia thích hợp để tăng hiệu quả dự báo của mô hình.

Vấn đề xây dựng mô hình dự báo sử dụng thuật toán trích xuất mô hình mờ từ dữ liệu ứng dụng cho một bài toán dự báo trong thực tế sẽ được trình bày ở chương tiếp theo. Những vấn đề phát sinh trong bài dự báo thực tế như kích thước dữ liệu lớn, tính nhiễu của các tập dữ liệu sẽ được đề cập và giải quyết.

Chương 3. LAI GHÉP KỸ THUẬT PHÂN CỤM VỚI MÔ HÌNH MỜ HƯỚNG DỮ LIỆU

Chương này trình bày bài toán dự báo, đặc biệt là bài toán dự báo dữ liệu chuỗi thời gian, những giải pháp cho bài toán dự báo chuỗi thời gian. Trên cơ sở giải pháp xây dựng một mô hình lai ghép dựa trên kỹ thuật phân cụm và mô hình mờ trích xuất từ máy học véc-tơ hỗ trợ cho bài toán dự báo dữ liệu chuỗi thời gian, Luận án trình bày thực nghiệm cho bài toán dự báo giá cổ phiếu.

3.1. Bài toán dự báo

Dự báo đã hình thành từ đầu những năm 60 của thế kỉ 20. Khoa học dự báo với tư cách một ngành khoa học độc lập có hệ thống lí luận, phương pháp luận và phương pháp hệ riêng nhằm nâng cao tính hiệu quả của dự báo. Ngày nay dự báo là một nhu cầu không thể thiếu được của mọi hoạt động kinh tế - xã hội, khoa học- kỹ thuật, được tất cả các ngành khoa học quan tâm nghiên cứu. Dự báo là một chủ đề rất rộng, dưới góc nhìn tổng quát thì dự báo là kỹ thuật phân tích dữ liệu trong quá khứ và hiện tại để dự đoán giá trị dữ liệu hay vấn đề, sự kiện có thể xảy ra trong tương lai.

Dựa vào phương pháp dự báo có thể chia dự báo làm 3 nhóm [10]:

- Dự báo bằng phương pháp chuyên gia: Loại dự báo này được tiến hành trên cơ sở tổng hợp, xử lý ý kiến của các chuyên gia thông thạo với hiện tượng được nghiên cứu, từ đó có phương pháp xử lý thích hợp đề ra các dự đoán, các dự đoán này được cân nhắc và đánh giá chủ quan từ các chuyên gia. Phương pháp này có ưu thế trong trường hợp dự đoán những hiện tượng hay quá trình bao quát rộng, phức tạp, chịu sự chi phối của nhiều yếu tố. Ví dụ như dự báo về sự phát triển của khoa học - kỹ thuật, sự thay đổi của môi trường, thời tiết, chiến tranh trong khoảng thời gian dài. Một cải tiến của phương pháp lấy ý kiến chuyên gia là phương pháp Delphi - Là phương pháp dự báo dựa trên cơ sở sử dụng một tập hợp những đánh giá của một nhóm chuyên gia. Mỗi chuyên gia được hỏi ý kiến và rồi dự báo của họ được trình bày dưới dạng thống kê tóm

tắt. Việc trình bày những ý kiến này được thực hiện một cách gián tiếp (không có sự tiếp xúc trực tiếp) để tránh những sự tương tác trong nhóm nhỏ qua đó tạo nên những sai lệch nhất định trong kết quả dự báo. Sau đó người ta yêu cầu các chuyên gia duyệt xét lại những dự báo của họ trên cơ sở tóm tắt tất cả các dự báo có thể có những bổ sung thêm.

- Dự báo theo phương trình hồi quy: Theo phương pháp này, mức độ cần dự báo phải được xây dựng trên cơ sở xây dựng mô hình hồi quy, mô hình này được xây dựng phù hợp với đặc điểm và xu thế phát triển của hiện tượng nghiên cứu. Để xây dựng mô hình hồi quy, đòi hỏi phải có tài liệu về hiện tượng cần dự báo và các hiện tượng có liên quan. Loại dự báo này thường được sử dụng để dự báo trung hạn và dài hạn ở tầm vĩ mô.
- Dự báo dựa vào dữ liệu dãy thời gian: là dựa trên cơ sở dãy số thời gian phản ánh sự biến động của hiện tượng ở những thời gian đã qua để xác định mức độ của hiện tượng trong tương lai.

Nhóm phương pháp thứ nhất, phương pháp chuyên gia, là nhóm sử dụng phương pháp dự báo định tính. Phương pháp dự báo này chủ yếu dựa trên phán đoán chủ quan và trực giác của người tham gia dự báo. Người tham gia có thể là người trực tiếp tham gia vào các công việc thuộc lĩnh vực dự báo hoặc là những người có chuyên môn sâu, kinh nghiệm rộng trong lĩnh vực cần nghiên cứu. Với phương pháp dự báo định tính, những khó khăn gặp phải chủ yếu là lựa chọn chuyên gia, trưng cầu ý kiến chuyên gia và xử lý ý kiến chuyên gia. Phương pháp này chủ yếu được sử dụng trong các trường hợp đối tượng dự báo thiếu thông tin, thiếu thống kê đầy đủ, toàn diện và đáng tin cậy về quy luật vận động của đối tượng dự báo trong quá khứ và hiện tại. Cũng có thể là trường hợp thiếu hoặc không có cơ sở lý luận thực tiễn chắc chắn đảm bảo cho việc mô tả quy luật vận động của đối tượng bằng cách sử dụng các mô hình toán học, hoặc đối tượng dự báo có độ bất định lớn, độ tin cậy thấp về hình thức thể hiện, về chiều hướng biến thiên, …

Hai nhóm phương pháp còn lại là các nhóm sử dụng phương pháp dự báo định lượng. Các phương pháp định lượng dựa vào các mô hình toán, các dữ liệu trong quá

khứ cùng với các yếu tố khác. Bằng việc sử dụng các dữ liệu trong quá khứ để tìm ra xu hướng, quy luật vận động của đối tượng nghiên cứu theo một mô hình nào đó và sử dụng mô hình tối ưu nhất để thực hiện ước lượng chúng thông qua các kiểm định tin cậy.

Nhóm phương pháp dự báo theo phương trình hồi quy, còn gọi là phân tích quan hệ nguyên nhân – kết quả, chủ yếu phân tích mối liên hệ nhân quả liên quan đến việc xác định các yếu tổ ảnh hưởng đến yếu tố ta muốn dự đoán, như phân tích hồi quy xem GDP phụ thuộc vào lượng đầu tư trong nước, lượng đầu tư nước ngoài, dân số,… hay sự phụ thuộc của giá cổ phiếu vào các chỉ số của kinh tế vi mô, giá vàng,… Trong đó, biến biểu diễn yếu tố muốn dự đoán gọi là biến phụ thuộc và biểu diễn cho các yếu tố ảnh hưởng đến yếu tố muốn dự đoán gọi là các biến độc lập [9].

Xét trường hợp cần dự báo giá trị của biến phụ thuộc Y dựa vào các biến độc lập $X_i, i = 1,2, \dots, p$. Phương trình hồi quy biểu diễn sự phụ thuộc của biến phụ thuộc Y vào các biến độc lập X_i có dạng:

$$Y = f(X_i), i = 1,2, \dots, p \quad (3.1)$$

Nếu hàm phụ thuộc $f(.)$ có dạng bậc nhất, thì mô hình có được là mô hình hồi quy tuyến tính. Ngược lại, mô hình được gọi là mô hình hồi quy phi tuyến nếu hàm phụ thuộc có dạng không phải bậc nhất, như Parabol, Hypebol, hàm mũ, … Các phương pháp dự báo theo phương trình hồi quy có thể dễ dàng triển khai thực hiện trên phần mềm Excel hoặc thực hiện bằng những thao tác đơn giản trên các phần mềm hỗ trợ phân tích định lượng như SPSS, Eviews.

Nhóm phương pháp dựa vào dữ liệu chuỗi thời gian dựa trên giả định cơ bản là quy luật vận động của hiện tượng trong quá khứ sẽ tiếp tục trong hiện tại và tương lai. Đây được xem như là một sự thừa nhận về tính liên tục, nó là một giả thuyết cơ bản của các phương pháp dự báo định lượng nói chung và phương pháp dự báo theo dữ liệu chuỗi thời gian nói riêng. Về cơ bản, phương pháp dự báo dữ liệu chuỗi thời gian cũng dựa trên ý tưởng phân tích hồi quy, tuy nhiên, không giống như dự báo nhân quả, dự báo chuỗi thời gian cố gắng dự báo tương lai dựa vào các giá trị quá

khứ của chính biến đang cần dự báo và các sai số trong quá khứ để tìm ra kiểu thức vận động của biến trong giai đoạn đã qua và ngoại suy tiếp kiểu đó cho tương lai.

Theo thống kê của Tufte, có khoảng 75% dữ liệu hình ảnh trên các tờ báo ở dạng chuỗi thời gian và kích thước của dữ liệu chuỗi thời gian tăng theo cấp số nhân [57]. Gần như phần lớn các nguồn cung cấp dữ liệu lớn đều ở dạng dữ liệu chuỗi thời gian. Vấn đề phân tích, trong đó đặt biệt là phân tích dự báo trên dữ liệu chuỗi thời gian đã và đang thu hút rất nhiều sự quan tâm, nghiên cứu. Bài toán dự báo dữ liệu chuỗi thời gian được ứng dụng trong nhiều lĩnh vực như dự báo giá cổ phiếu, dự báo thời tiết, dự báo sản lượng sản xuất, … [1], [3], [8], [9], [22], [25], [26], [27], [28], [42], [47], [54], [82], [87], [90]. Phương pháp xây dựng mô hình dự báo chuỗi thời gian dựa trên kỹ thuật phân tích hồi quy tuyến tính đã được chuẩn hóa thành những công cụ được sử dụng khá phổ biến như SPSS, Eviews. Những nghiên cứu mới về các giải pháp xây dựng, cải tiến và nâng cao hiệu quả dự báo của mô hình dự báo dữ liệu chuỗi thời gian vẫn đang thu hút sự quan tâm của rất nhiều nhà nghiên cứu trên thế giới; đây cũng chính là mục tiêu hướng đến của đề tài luận án này.

3.2. Dự báo dữ liệu chuỗi thời gian

3.2.1. Bài toán dự báo dữ liệu chuỗi thời gian

Chuỗi thời gian là một chuỗi các giá trị của một chỉ tiêu nghiên cứu được sắp xếp theo thứ tự thời gian. Ví dụ như giá đóng phiên hàng ngày của một mã cổ phiếu náo đó ở thị trường chứng khoán, chỉ số giá tiêu dùng hàng tháng của cả nước. lượng tiêu thụ điện hàng tháng ở một thành phố, số vụ tai nạn giao thông đường bộ, số vụ tự tử hàng năm, … Một chuỗi thời gian có dạng tổng quát như sau [9]:

t_i	t_1	t_2	…	t_N
x_i	x_1	x_2		x_N

Trong đó: $t_i, i = 1,2,..N$ chỉ mốc thời gian thứ i; và $x_i, i = 1,2,..N$ là giá trị của chỉ tiêu tương ứng với thời gian thứ i.

Về cơ bản, mục tiêu của dự báo dữ liệu chuỗi thời gian là để ước tính một số giá trị trong tương lai dựa vào mẫu dữ liệu hiện tại và trong quá khứ. Về mặt toán học có thể biểu diễn như sau:

$$\hat{x}_{(t+\Delta_t)} = f\left(x_{(t-\Delta_{t1})}, x_{(t-\Delta_{t2}),}, x_{(t-\Delta_{t3})}, \dots\right) \quad (3.2)$$

trong đó, với ví dụ cụ thể này, $\hat{x}_{(t+\Delta_t)}$ là giá trị dự đoán tại mốc thời gian $(t + \Delta_t)$ của một chuỗi thời gian rời rạc x.

Mục tiêu của dự báo chuỗi thời gian là tìm một hàm $f(.)$ sao cho giá trị dự đoán $\hat{x}$ của chuỗi thời gian tại một thời điểm trong tương lai là không thiên lệch (unbiased) và nhất quán (consistent). Lưu ý rằng thước đo độ tốt của mô hình dự báo chính là hiệu quả và độ sai lệch (bias). Giới hạn Cramér-Rao cho biết giới hạn dưới cho phương sai của ước lượng độ không thiên lệch. Nếu ước lượng độ không thiên lệch đạt đến giới hạn này thì có thể nói mô hình dự đoán là hiệu quả [9].

Ước lượng thường rơi vào 2 loại là tuyến tính (linear) và không tuyến tính (nonlinear). Trong nhiều thập niên qua, rất nhiều tài liệu viết về kỹ thuật dự đoán tuyến tính: dự đoán ước lượng một giá trị trong tương lai dựa vào sự kết hợp tuyến tính của các giá trị trong quá khứ và hiện tại. Thực tế thì việc dự đoán chuỗi thời gian trong thế giới thực thường không rơi vào kiểu dự đoán tuyến tính mà lại là mô hình dự đoán không tuyến tính.

Vấn đề dự báo theo chuỗi thời gian, mà đặc biệt là vấn đề dự báo giá cổ phiếu đã và đang thu hút được nhiều sự quan tâm nghiên cứu của các nhà khoa học. Bài toán dự báo giá cổ phiếu hiện nay chủ yếu được tiếp cận dưới hai dạng, đó là dự báo giá cổ phiếu sau n-ngày hoặc dự báo xu hướng của giá cổ phiếu sau n-ngày [6], [22], [26], [27], [28], [31], [32], [45], [53], [67], [87], [90]. Nhiều mô hình và giải pháp đã được đề xuất, như mạng nơ-ron nhân tạo [31], [42], [44], máy học véc-tơ hỗ trợ [6], [45], [54], [87], [90], mô hình chuỗi Markov ẩn [3], ứng dụng Đại số gia tử [1], [2], [3],…. Đồng thời cũng có nhiều giải pháp đề xuất cải tiến và tích hợp các mô hình, với mục tiêu cuối cùng là nâng cao độ chính xác của kết quả dự báo [6], [26], [53], [66].

Những nghiên cứu gần đây chủ yếu tập trung vào hướng cải tiến và kết hợp nhiều phương thức học khác nhau để nâng cao hiệu quả dự báo, như mô hình kết hợp SVM và SOM (Self-Organizing Map) [26], [66], kết hợp HNN, AMN và GA [53], kết hợp K-means và SVM [6], mô hình kết hợp chuỗi Markov bậc cao và chuỗi thời gian mờ [3]. Hầu hết đa số các nghiên cứu đề xuất mô hình dự báo dữ liệu chuỗi thời gian tài chính đều sử dụng các mô hình máy học. Một trong những điểm hạn chế của mô hình máy học là chính là mô hình số, là dạng "hộp đen" đối với người sử dụng cũng như các chuyên gia.

Những nghiên cứu trích xuất mô hình mờ cho bài toán dự báo từ các máy học thống kê như mạng nơ-ron, máy học véc-tơ hỗ trợ, SOM, … đã phần nào giải quyết được vấn đề "hộp đen" của mô hình máy học thống kê [24], [35], [38], [40], [56], [80]. Tập luật mờ trích xuất được sẽ là cơ sở luật cho hệ thống dự báo mờ. Nếu tập luật đảm bảo tính "có thể diễn dịch" thì các chuyên gia có thể hiểu và phân tích ngữ nghĩa tập luật, trên cơ sở đó có thể chọn lọc hoặc bổ sung luật nếu cần thiết.

3.2.2. Đánh giá độ phù hợp của mô hình dự báo

Cần nhận thức được rằng đối với một bộ dữ liệu lịch sử thu thập được liên quan đến đối tượng cần dự báo, người ta có thể vận dụng không chỉ một mà là một vài phương pháp dự báo khác nhau để thực hiện mục tiêu dự báo trong tương lai. Không có phương pháp dự báo nào là hoàn hảo nhất mà tùy vào bản chất của hiện tượng, độ dài dự báo, độ dài của chuỗi thời gian, cùng với kinh nghiệm thực tế là những yếu tố cần thiết để cân nhắc xem trong từng bài toán dự báo thì mô hình dự báo nào là phù hợp hơn cả. Mức độ phù hợp này được xem xét trên khía cạnh mô hình dự báo nào cho ra kết quả dự báo chính xác hơn, trong phần lớn tình huống sự chính xác được xem như tiêu chuẩn cơ bản để chọn lựa một phương pháp dự báo phù hợp, vì thế hai thuật ngữ "chính xác" và "phù hợp" có thể được dùng lẫn nhau để chỉ việc mô hình dự báo đã xây dựng được có thể dự báo gần đúng đến mức nào so với dữ liệu thật khi thử nghiệm [9].

Có nhiều chỉ tiêu đo lường mức độ chính xác của mô hình dự báo. Trong nội dung này của luận án sẽ tập trung nghiên cứu một số chỉ tiêu tiêu biểu. Các chỉ tiêu

này đều được xây dựng dựa trên thông tin về sai số dự báo, ký hiệu là e_t, đó là chênh lệch giữa giá trị thực tế và giá trị dự báo ở cùng thời điểm t. Về mặt công thức nếu y_t là ký hiệu cho giá trị quan sát thực tế và $\hat{y}_t$ là ký hiệu cho giá trị dự báo ở cùng thời điểm thì sai số dự báo được hình thành như sau: $e_t = (y_t - \hat{y}_t)$.

Nếu chuỗi thời gian dùng để thử nghiệm mô hình dự báo có độ dài thời gian là k, tức có k giá trị quan sát y_t, khi áp dụng thử nghiệm mô hình dự báo sẽ có k giá trị $\hat{y}_t$ dự báo được và khi đó sẽ tính được k giá trị sai số $e_t = (y_t - \hat{y}_t)$. Dựa trên các giá trị sai số e_t này có thể tính toán các đại lượng đo lường sai số dự báo phổ biến sau [9]:

- **Sai số tuyệt đối trung bình (Mean Absolute Error – MAE)**:

  Công thức tính sai số này như sau:

$$MAE = \frac{\sum_{t=1}^{k}|e_t|}{k} \tag{3.3}$$

Chú ý là khi tính các đại lượng đo độ chính xác của mô hình dự báo thì các xử lý đối với e_t phải lấy trị tuyệt đối hoặc bình phương để tránh triệt tiêu do trái dấu.

- **Sai số phần trăm tuyệt đối trung bình (Mean Absolute Percent Error – MAPE):**

  Công thức tính sai số này như sau:

$$MAPE = \frac{\sum_{t=1}^{k}(|e_t|/y_t)}{k}100\% \tag{3.4}$$

Công thức này giúp ta khử đơn vị tính trong tử số của công thức MAE để có một đại lượng có đơn vị tính là %, giúp dễ so sánh giữa MAPE của các mô hình dự báo trên các chuỗi dữ liệu khác về đơn vị tính.

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

  Công thức tính cho sai số này như sau:

$$MSE = \frac{\sum_{t=1}^{k} {e_t}^2}{k} \tag{3.5}$$

So sánh 2 công thức của MAE và MSE thì công thức MSE có nhược điểm là nó làm sai số bị bình phương lên, nên giá trị cuối cùng của MSE rất lớn, tuy nhiên ưu điểm của MSE là nó giúp cho các phép tính toán liên quan trở nên dễ xử lý hơn so với khi dùng MAE, nên MSE thông dụng hơn MAE.

Để khắc phục nhược điểm phóng đại (bình phương) của MSE, có thể sử dụng sai số thay thế là $RMSE = \sqrt{MSE}$. Công thức tính của RMSE là công thức (1.53).

- **Sai số bình phương trung bình chuẩn hóa (Normalize Mean Square Error – NMSE):**

NMSE là dạng chuẩn hóa của MSE trong đó đã xử lý trường hợp dữ liệu giống nhau. Công thức tính của sai số này như sau:

$$NMSE = \frac{1}{k\sigma^2}\sum_{i=1}^{k} {e_t}^2, \tag{3.6}$$

$$với\ \sigma^2 = \frac{1}{k-1}\sum_{i=1}^{k}(y_i - \bar{y})^2$$

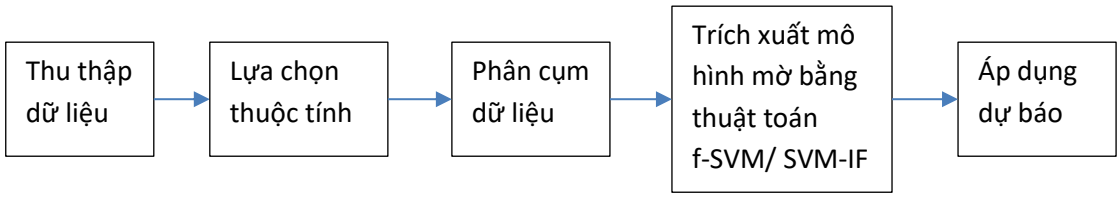
$$và\ \bar{y} = \frac{1}{k}\sum_{i=1}^{k} y_i$$

3.3. Đề xuất mô hình mờ dự báo dữ liệu chuỗi thời gian

Trong những trường hợp xây dựng mô hình mờ hướng dữ liệu dự báo dữ liệu chuỗi thời gian cho các bài toán dự báo thực tế nói riêng và xây dựng mô hình hướng dữ liệu nói chung, có rất nhiều thách thức đặt ra. Tuy nhiên, cơ bản nhất vẫn là những thách thức đến từ dữ liệu huấn luyện mô hình [26], [66], [73]. Cụ thể những thách thức đó là:

1) Kích thước dữ liệu huấn luyện lớn, thiếu tính đặc trưng, và tính ngẫu nhiên (hay tính nhiễu) của dữ liệu khá cao,
2) Việc lựa chọn thuộc tính dữ liệu vào trong rất nhiều thuộc tính dữ liệu sẵn có
3) Và tiếp đến là sự bùng nổ tập luật mờ học được

Với mục tiêu vượt qua những thách thức nêu trên, luận án đề xuất xây dựng mô hình mờ nhiều giai đoạn cho bài toán dự báo dữ liệu chuỗi thời gian. Mô hình lai ghép đề xuất gồm 5 giai đoạn, được thể hiện ở Hình 3.1.



Hình 3.1. Mô hình nhiều giai đoạn cho bài toán dự báo dữ liệu chuỗi thời gian

Giai đoạn 1. Thu thập dữ liệu lịch sử của chỉ tiêu cần dự báo. Dữ liệu thu thập được phải đủ lớn, đảm bảo tốt nhất tính đặc trưng và bao phủ.

Giai đoạn 2. Lựa chọn thuộc tính dữ liệu vào dựa vào ý kiến của các chuyên gia trong lĩnh vực dự báo. Tập thuộc tính phải được rút gọn để đảm bảo tính diễn dịch được của mô hình mờ như đã trình bày ở Chương 3.

Giai đoạn 3. Phân cụm dữ liệu đầu vào để thu gọn kích thước tập dữ liệu, giảm tính nhiễu của dữ liệu theo từng phân cụm. Giải pháp phân cụm dữ liệu sẽ được trình bày chi tiết ở mục tiếp theo.

Giai đoạn 4. Trích xuất các mô hình mờ cho từng phân cụm bằng cách sử dụng thuật toán f-SVM hoặc SVM-IF đã đề xuất ở Chương 1 và Chương 2.

Giai đoạn 5. Thực hiện dự báo dựa trên mô hình mờ trích xuất được.

3.4. Phân cụm dữ liệu đầu vào

Một trong những thách thức của các ứng dụng khai phá dữ liệu là dữ liệu đầu vào thường rất lớn, trong khi đó có nhiều thuật toán học là không hiệu quả với kích thước dữ liệu lớn. Với vấn đề trích xuất tập luật từ dữ liệu thì tập dữ liệu huấn luyện với kích thước lớn cũng dẫn đến việc bùng nổ tập luật trích xuất được. Một trong những hướng tiếp cận để giải quyết vấn đề tập dữ liệu lớn này là phân dữ liệu đầu vào thành các cụm nhỏ và chuyển bài toán thành các bài toán với kích thước dữ liệu

nhỏ hơn. Các thuật toán học sẽ được áp dụng trên từng cụm dữ liệu nhỏ và sau đó tổng hợp các kết quả học lại.

Ngoài ra, một thách thức nữa nảy sinh trong bài toán dự báo dữ liệu chuỗi thời gian đó là dữ liệu có thể không ổn định theo thời gian do nhiều yếu tố khác tác động. Ví dụ như trong dữ liệu chuỗi thời gian giá cổ phiếu, phân bố thống kê của giá cổ phiếu theo thời gian phụ thuộc vào nhiều yếu tố khác nhau như sự tăng trưởng hay suy thoái của kinh tế, tình hình chính trị, môi trường, thiên tai, … Điều đó gây nên tình trạng bất ổn định trong dữ liệu, gọi là nhiễu. Tình trạng nhiễu của tập dữ liệu huấn luyện gây nên nhiều hạn chế cho việc tìm ra những qui tắc dự báo dựa trên dữ liệu quá khứ. Giải pháp phân dữ liệu thành các cụm khác nhau tương ứng với các phân bố thống kê của các điểm dữ liệu, sẽ là một cách để khắc phục đặc điểm không ổn định của dữ liệu chuỗi thời gian.

Phân cụm là một kỹ thuật rất quan trọng trong khai phá dữ liệu, nó thuộc lớp các phương pháp học không giám sát trong học máy. Có rất nhiều định nghĩa khác nhau về kỹ thuật này, nhưng về bản chất ta có thể hiểu phân cụm là các qui trình tìm cách nhóm các đối tượng đã cho vào các cụm (clusters), sao cho các đối tượng trong cùng 1 cụm thì tương tự nhau và các đối tượng khác cụm thì không tương tự nhau.

Những kỹ thuật phân cụm dữ liệu thường được đề xuất sử dụng và được chứng tỏ mang lại hiệu quả, như k-Means, SOM, HC, … [6], [26], [66], [87]. Không có một thuật toán phân cụm nào là tốt nhất và thích hợp cho tất cả mọi ứng dụng. Với mỗi ứng dụng khác nhau thì người sử dụng phải lựa chọn ra một thuật toán phân cụm cụ thể thích ứng với ứng dụng đó. Kết quả đánh giá cho từng thuật toán cũng phụ thuộc vào những yêu cầu của từng ứng dụng. Những nghiên cứu trong [41] và [57] đã khẳng định k-Means và SOM hiệu quả hơn so với các kỹ thuật phân cụm khác trong trường hợp giải quyết bài toán khai phá dữ liệu với các tập dữ liệu lớn.

Với mục tiêu phân cụm dữ liệu ở Giai đoạn 3 trong mô hình đề xuất ở Hình 3.1, là bước tiền xử lý tập dữ liệu đầu vào có kích thước lớn, như vậy k-Means và SOM sẽ là những kỹ thuật phân cụm dữ liệu phù hợp để lựa chọn áp dụng trong trường hợp bài toán dự báo dữ liệu chuỗi thời gian của luận án. Kỹ thuật phân cụm k-Means và

SOM cũng chính là các kỹ thuật được đề xuất ứng dụng phân cụm dữ liệu chuỗi thời gian tài chính trong các nghiên cứu ở [6], [26], [66].

3.4.1. Kỹ thuật phân cụm k-Means

K-Means là một trong những thuật toán cơ bản nhất của lớp thuật toán học không giám sát được sử dụng phổ biến trong kỹ thuật phân cụm. Thuật ngữ k-Means lần đầu tiên được sử dụng bởi MacQueen J.B. vào năm 1967 [48]. Tư tưởng chính của thuật toán k-Means là tìm cách phân nhóm các đối tượng đã cho vào K cụm (K là một số nguyên dương xác định số các cụm được phân chia) sao cho tổng bình phương khoảng cách giữa các đối tượng đến tâm nhóm (centroid) là nhỏ nhất.

Kỹ thuật phân cụm k-Means có thể được vắn tắt như sau:

Cho một tập dữ liệu ban đầu gồm N đối tượng là những véc-tơ trong không gian d chiều $X_i = \{x_{i1}, x_{i2}, \dots, x_{id}\}$ với $i = 1,2, \dots, N$. Cần phân tập dữ liệu ban đầu thành k phân cụm $\{C_1, C_2, \dots, C_k\}$, sao cho hàm tiêu chuẩn: $E = \sum_{i=1}^{k} \sum_{x \in C_i} D^2(x - m_i)$ đạt giá trị tối thiểu. Trong đó: m_i là trọng tâm của cụm C_i và D là khoảng cách giữa hai đối tượng.

Trọng tâm của một cụm là một véc-tơ, trong đó giá trị của mỗi phần tử của nó là trung bình cộng các thành phần tương ứng của các đối tượng véc-tơ dữ liệu trong cụm đang xét. Tham số đầu vào của thuật toán phân cụm k-Means là số cụm k, tập dữ liệu gồm N phần tử và tham số đầu ra của thuật toán là các trọng tâm của các cụm dữ liệu. Độ đo khoảng cách D giữa các đối tượng dữ liệu thường được sử dụng là khoảng cách Euclid, bởi vì đây là mô hình khoảng cách dễ để lấy đạo hàm và xác định các cực trị tối thiểu. Hàm tiêu chuẩn và độ đo khoảng cách có thể được xác định cụ thể hơn tuỳ vào ứng dụng hoặc các quan điểm của người dùng.

Thuật toán phân cụm k-Means với k phân cụm cho trước bao gồm các bước cơ bản như sau:

Input: *Tập dữ liệu gồm N đối tượng* $X_i, i = 1,2,..,N;$
Số các phân cụm $k;$

Output: *Tập các phân cụm* $C_i, i = 1,2, \dots, k;$

Bước 1. Chọn k đối tượng m_j với $j = 1,2,\ldots,k$ là trọng tâm ban đầu của k cụm từ tập dữ liệu (việc lựa chọn này có thể là ngẫu nhiên hoặc theo kinh nghiệm).

Bước 2. Đối với mỗi đối tượng $X_i, i = 1,2,..,N$, tính toán khoảng cách từ nó tới mỗi trọng tâm m_j với $j = 1,2,\ldots,k$, sau đó tìm trọng tâm gần nhất đối với mỗi đối tượng.

Bước 3. Đối với mỗi $j = 1,2,\ldots,k$, cập nhật trọng tâm cụm m_j bằng cách xác định trung bình cộng của các véc-tơ đối tượng dữ liệu.

Bước 4. Lặp các bước 2 và 3 cho đến khi các trọng tâm của cụm không thay đổi.

Kỹ thuật phân cụm k-Means phân tích phân cụm đơn giản nên có thể áp dụng đối với tập dữ liệu lớn. Tuy nhiên, nhược điểm của k-Means là chỉ áp dụng với dữ liệu có thuộc tính số và khám phá ra các cụm có dạng hình cầu, k-Means còn rất nhạy cảm với nhiễu và các phần tử ngoại lai trong dữ liệu.

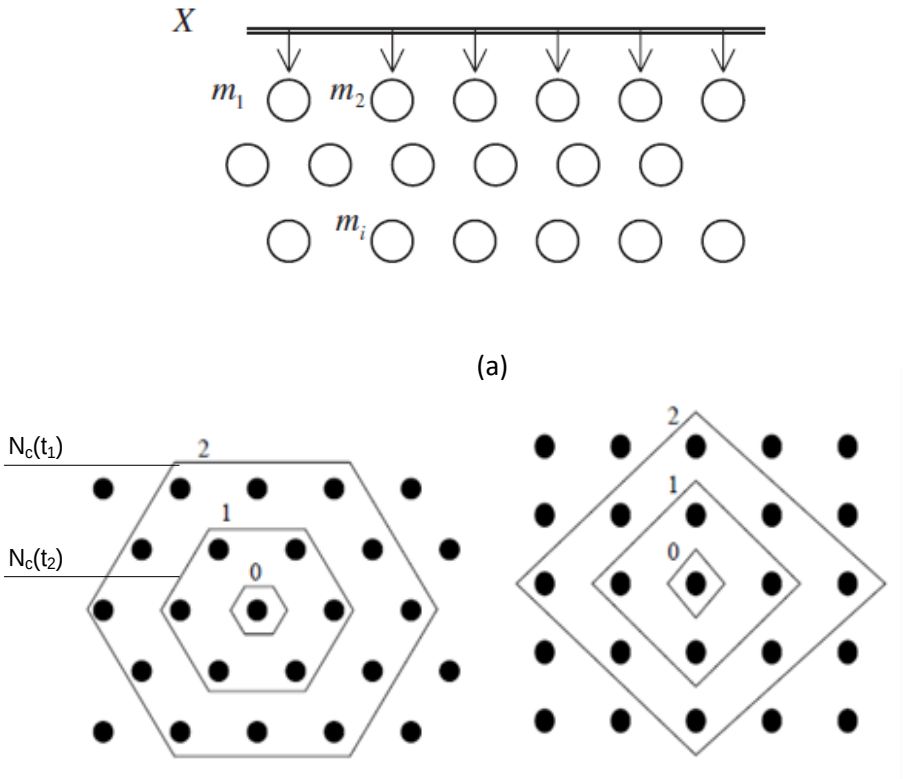
Hơn nữa, chất lượng phân cụm dữ liệu của thuật toán k-Means phụ thuộc nhiều vào các tham số đầu vào như: số cụm k và k trọng tâm khởi tạo ban đầu (m_j). Trong trường hợp, các trọng tâm khởi tạo ban đầu mà quá lệch so với các trọng tâm cụm tự nhiên thì kết quả phân cụm của k-Means là rất thấp, nghĩa là các cụm dữ liệu được khám phá rất lệch so với các cụm trong thực tế. Trên thực tế người ta chưa có một giải pháp tối ưu nào để chọn các tham số đầu vào, giải pháp thường được sử dụng nhất là thử nghiệm với các giá trị đầu vào k khác nhau rồi sau đó chọn giải pháp tốt nhất.

3.4.2. Kỹ thuật phân cụm SOM

Bản đồ tự tổ chức SOM (Self-Organizing Map) hay SOFM (Self-Organizing Feature Map) lần đầu tiên được giới thiệu bởi C. von der Malsburg năm 1973 và được phát triển bởi T. Kohonen (Finland) năm 1982 [39][72], nên còn được gọi là mạng Kohonen. Ban đầu SOM được đề xuất như là một giải pháp hiệu quả cho vấn đề nhận dạng và điều khiển rô-bốt. Bản chất của SOM là một mạng mơ-ron nhân tạo, được

huấn luyện sử dụng kỹ thuật học không giám sát để biểu diễn dữ liệu với số chiều thấp hơn nhiều, thường là 2 chiều, so với dữ liệu đầu vào nhiều chiều, thường số chiều lớn. Kết quả của SOM gọi là bản đồ (Map). SOM là một ANN, tuy nhiên SOM khác với các ANN là không sử dụng các lớp ẩn (hidden layers) chỉ sử dụng input và output layer. SOM sử dụng khái niệm láng giềng (neighborhood) để giữ lại đặc trưng của các dữ liệu đầu vào trên bản đồ, có nghĩa là các mẫu dữ liệu huấn luyện tương tự nhau thì được đặt gần nhau trên bản đồ. Ưu điểm chính của SOM là biểu diễn trực quan dữ liệu nhiều chiều vào không gian ít chiều hơn, thường là 2 chiều và đặc trưng của dữ liệu đầu vào được giữ lại trên bản đồ.

Trong SOM, các nơ-ron đầu ra được thường được tổ chức thành một bản đồ d-chiều dưới dạng hình chữ nhật hoặc hình lục giác, trong đó mỗi nơ-ron đầu ra được kết nối với tất cả các nơ-ron đầu vào [72]. Cấu trúc mạng SOM của Kohonen được thể hiện ở Hình 3.2.



Hình 3.2. (a) Một ví dụ SOM. (b) Phân bố lục giác và hình chữ nhật của SOM

Trong mạng SOM ví dụ ở Hình 3.2, mỗi một nơ-ron đều có một véc-tơ tham chiếu m_i và một khu vực láng giềng N_c. Những véc-tơ tham chiếu là có cùng kích thước như các véc-tơ đầu vào và được dùng để đánh giá sự gần gũi giữa nơ-ron với các véc-tơ đầu vào. Khu vực láng giềng là một hàm đối xứng và đơn điệu giảm đối với khoảng cách đến các nơ-ron trên bản đồ tính từ nơ-ron chiếm lĩnh. Trong ví dụ ở Hình 3.2b, khu vực láng giềng đơn điệu giảm đối với t1<t2.

SOM đưa ra các nơ-ron chiếm lĩnh đối với các láng giềng của nó trên bản đồ bằng cách thực hiện các thuật toán huấn luyện trên các véc-tơ đầu vào. Kết quả cuối cùng là các nơ-ron được tổ chức trên bản đồ, với các nơ-ron láng giềng có véc-tơ trọng số tương đương nhau.

Kỹ thuật phân cụm SOM có thể được vắn tắt như sau:

Cho một tập dữ liệu ban đầu gồm N đối tượng là những véc-tơ trong không gian d chiều $X_i = \{x_{i1}, x_{i2}, \dots, x_{id}\}$ với $i = 1,2, \dots, N$. Thuật toán huấn luyện mạng SOM với K phân cụm cho trước bao gồm các bước như sau:

Input: *Tập dữ liệu gồm N đối tượng* $X_i, i = 1,2,..,N$;
Số nơ-ron mạng K;

Output: *Mạng SOM với K phân cụm;*

Bước 1. Khởi tạo số bước lặp $t = 1$; Với mỗi véc-tơ X_i, khởi tạo những véc-tơ tham chiếu $\{m_i(t), i = 1,2, \dots K\}$ với những giá trị ngẫu nhiên nhỏ. K là tổng số nơ-ron trong mạng.

Bước 2. Thể hiện một véc-tơ đầu vào $X(t)$. Tính toán khoảng cách Euclid giữa $X(t)$ và tất cả các nơ -ron: $d_i = \sum_{j=1}^{N} \left\| X_j(t) - m_{ij}(t) \right\|, i = 1,2, \dots, K$, với N là mảng các véc-tơ đầu vào, $\| \quad \|$ là khoảng cách Euclid.

Bước 3. Chọn nơ-ron thứ i^* gần gũi với $X(t)$ dựa vào khoảng cách:

$d_i = min_{i=1,2,\dots N}(d_i)$

Bước 4. Cập nhật những véc-tơ tham chiếu của nơ-ron thứ i^* và những láng giềng của nó bằng công thức:

$$m_i(t+1) = m_i(t) + \eta N_{ii^*}(t)\big(X(t) - m_i(t)\big),$$

và những nơ-ron khác bằng công thức: $m_i(t+1) = m_i(t)$.

Trong đó η là tỷ lệ học, và $N_{ii^*} = exp\left(-\frac{\|r_i - r_{i^*}\|^2}{\delta(t)^2}\right)$ là hàm láng giềng Gauss, r_i và r_{i^*} là những véc-tơ vị trí của những nơ-ron láng giềng và nơ-ron trung tâm (nơ-ron chiếm lĩnh) tương ứng.

Bước 5. Lặp lại thực hiện Bước 2 cho đến khi sự thay đổi trong các véc-tơ tham chiếu là ít hơn một giá trị ngưỡng định trước, hoặc số lần lặp đạt đến một giá trị tối đa định trước.

Với khả năng mạnh mẽ trong việc biểu diễn dữ liệu từ không gian nhiều chiều về không gian ít chiều hơn mà vẫn có thể bảo tồn được quan hệ hình trạng của dữ liệu trong không gian đầu vào, nên chức năng chính của SOM là trình diễn cấu trúc của toàn bộ tập dữ liệu và giúp quan sát trực quan cấu trúc cũng như sự phân bố tương quan giữa các mẫu dữ liệu trong không gian của tập dữ liệu. Do đó, SOM được ứng dụng rất nhiều trong các bài toán thực tế, đặc biệt là trong các bài toán nhận dạng tiếng nói, điều khiển tự động, hóa-sinh trắc học, phân tích tài chính và xử lý ngôn ngữ tự nhiên, …

3.4.3. Phân cụm dữ liệu đầu vào bằng SOM

Xét về độ phức tạp thời gian tính toán thì thuật toán phân cụm k-Means và SOM có độ phức tạp gần tương đương nhau là O(*N.K.T*), với *N* là kích thước tập dữ liệu, *K* là số phân cụm hoặc số nơ-ron ban đầu và *T* là số lần lặp điều chỉnh cấu trúc [57]. Trong [41], thông qua thực nghiệm các tác giả đã chứng tỏ kỹ thuật phân cụm SOM hiệu quả hơn k-Means ở cả hiệu quả phân cụm và thời gian thực hiện. Kỹ thuật phân cụm k-Means được đánh giá là đơn giản, dễ cài đặt hơn so với mạng nơ-ron SOM. Tuy nhiên kết quả phân cụm k-Means phụ thuộc rất mạnh vào việc lựa chọn *k* phân cụm ban đầu và k-Means phân cụm kém hiệu quả trong trường hợp dữ liệu bị nhiễu.

Kỹ thuật phân cụm SOM được đánh giá là ít phụ thuộc vào việc chọn số lượng và vị trí các nơ-ron ban đầu hơn so với việc chọn *k* cụm ban đầu trong trường hợp của k-Means. Kỹ thuật phân cụm SOM cũng được đánh giá mang lại hiệu quả phân cụm tốt hơn trong trường hợp dữ liệu bị nhiễu, đồng thời SOM cũng ít bị tối ưu cục

bộ hơn so với k-Means. Gần đây, nhiều nghiên cứu của các tác giả khác đã đề xuất sử dụng SOM như là một giải pháp khá hiệu quả để phân cụm dữ liệu, đặc biệt là đối với dữ liệu chuỗi thời gian tài chính [26], [66]. Kỹ thuật phân cụm SOM được sử dụng để phân dữ liệu đầu vào thành các phân cụm theo sự tương đương về phân bố thống kê của các điểm dữ liệu. Kết quả phân cụm bởi SOM sẽ giúp giải quyết được hai vấn đề:

1) Kích thước dữ liệu trong từng phân cụm sẽ nhỏ hơn làm tăng tốc độ huấn luyện mô hình.

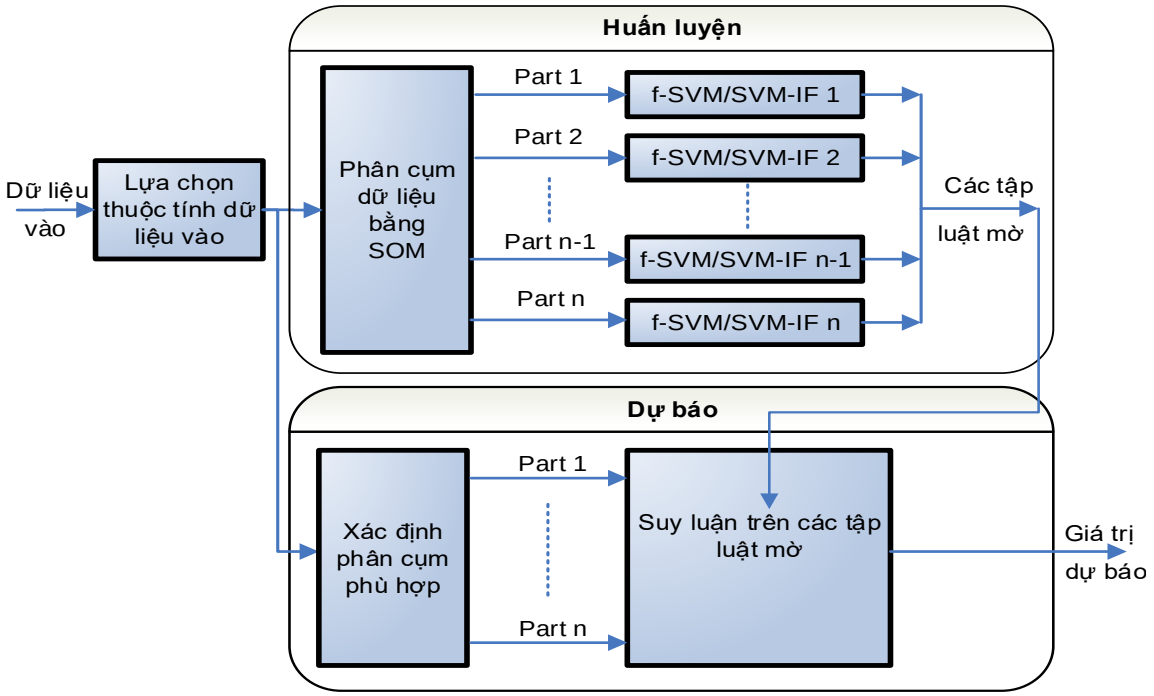
2) Dữ liệu trong các phân cụm có sự tương đương trong phân bố thống kê như vậy sẽ tránh được trường hợp nhiễu.

Trong phạm vi nghiên cứu của luận án, với bài toán đặt ra là dự báo dữ liệu chuỗi thời gian có đặc điểm là kích thước dữ liệu lớn, tính ngẫu nhiên và khả năng bị nhiễu của dữ liệu khá cao, mục tiêu hướng đến là giảm kích thước, giảm nhiễu dữ liệu, từ đó giảm số lượng, đơn giản hóa tập luật mờ học được từ dữ liệu. Bên cạnh đó để có cơ sở đối chiếu, so sách hiệu quả của mô hình đề xuất với các mô hình được đề xuất trước đó bởi các tác giả khác trong [26][66], kỹ thuật phân SOM được lựa chọn để phân cụm dữ liệu chuỗi thời gian đầu vào, sau đó áp dụng thuật toán trích xuất mô hình mờ TSK dựa vào máy học véc-tơ hỗ trợ để trích xuất các mô hình mờ tương ứng với từng phân cụm. Với các cụm luật mờ có số lượng hạn chế sẽ tạo điều kiện thuận lợi cho các chuyên gia có thể hiểu, phân tích, đánh giá được, và qua đó có thể tối ưu hóa để nâng cao hiệu quả sử dụng các luật mờ học được từ dữ liệu này.

3.5. Mô hình thực nghiệm cho bài toán dự báo giá giá cổ phiếu

Để giải quyết bài toán dự báo giá cổ phiếu, luận án đề xuất mô hình lai ghép giữa kỹ thuật phân cụm SOM và thuật toán trích xuất mô hình mờ từ máy học véc-tơ hỗ trợ hồi quy. Theo đó, tập dữ liệu đầu vào được phân chia thành các cụm tách rời bằng kỹ thuật phân cụm SOM trước khi ứng dụng thuật toán trích xuất mô hình mờ dựa trên máy học véc-tơ hỗ trợ để trích xuất ra các mô hình mờ.

Mô hình thực nghiệm trong hai trường hợp sử dụng thuật toán f-SVM và SVM-IF được thể hiện ở Hình 3.3.



Hình 3.3. Mô hình dự báo giá cổ phiếu lai ghép giữa SOM và f-SVM hoặc SVM-IF

Quá trình thực hiện thực nghiệm dự báo giá cổ phiếu theo mô hình đề xuất được thể hiện qua hai đoạn như sau:

- **Giai đoạn 1: Huấn luyện mô hình bằng tập dữ liệu huấn luyện**

Bước 1. Lựa chọn thuộc tính dữ liệu đầu vào và đầu ra

Bước 2. Phân cụm tập dữ liệu huấn luyện bằng SOM (n phân cụm)

Bước 3. Sử dụng thuật toán f-SVM hoặc SVM-IF để trích xuất ra các mô hình mờ TSK cho mỗi phân cụm dữ liệu

Bước 4. Thực nghiệm dự báo trên tập dữ liệu xác thực để chọn giá trị tối ưu cho các tham số ε , số phân cụm n

Bước 5. Trích xuất ra các mô hình mờ cho các phân cụm

- **Giai đoạn 2: Thực hiện dự báo trên tập dữ liệu thử nghiệm**

Bước 1. Xác định phân cụm tương ứng với từng mẫu dữ liệu của tập thử nghiệm

Bước 2. Thực hiện dự báo trên tập dữ liệu thử nghiệm

Bước 3. Tính toán các sai số trên kết quả dự báo để đánh giá mô hình

Để triển khai các thực nghiệm, luận án xây dựng một hệ thống công cụ trên Matlab. Thuật toán học SVM của thư viện LIBSVM được phát triển bởi nhóm của Chih-Chung Chang [20], được sử dụng để sản sinh ra các SV từ dữ liệu huấn luyện, làm cơ sở để xây dựng thuật toán trích xuất các luật mờ f-SVM và SVM-IF. Trong thực nghiệm xây dựng thuật toán f-SVM và SVM-IF, luận án có sử dụng hàm *SVMgenfis()* và hàm *anfis()*. Trong đó, hàm *SVMgenfis()* được xây dựng để sinh ra mô hình mờ TSK ban đầu dựa vào những véc-tơ hỗ trợ nhận được từ kết quả huấn luyện SVM, theo đúng cấu trúc của hệ thống mờ ANFIS trong thư viện Matlab. Hàm *anfis()* của thư viện Fuzzy Toolbox của phần mềm Matlab được sử dụng để tối ưu hóa các tham số hàm thành viên bằng phương pháp Gradient descent và trích xuất ra mô hình mờ theo chuẩn ANFIS đã được tối ưu hóa các tham số. Việc phân cụm dữ liệu đầu vào được thực hiện dựa trên bộ công cụ SOM Toolbox 2.0 được phát triển bởi Juha Vesanto, Esa Alhoniemi và các đồng sự [39]. Sau cùng, hàm *evalfis()* trong thư viện công cụ Fuzzy Toolbox của phần mềm Matlab được sử dụng để suy luận dự báo giá cổ phiếu dựa trên mô hình mờ TSK trích xuất được.

3.5.1. Lựa chọn dữ liệu đầu vào

Việc lựa chọn thuộc tính đầu vào cho bài toán dự báo phụ thuộc vào từng lĩnh vực chuyên môn của bài toán. Những kinh nghiệm của chuyên gia trong lĩnh vực tương ứng và những kết quả phân tích, thống kê sẽ cho ta những gợi ý về việc lựa chọn các thuộc tính đầu vào của mô hình. Như đã đề cập trong Chương 2, việc lựa chọn các thuộc tính đầu vào trước khi học mô hình được xem như là kỹ thuật tích hợp tri thức tiên nghiệm vào quá trình học mô hình mờ từ dữ liệu theo kịch bản EBL. Việc lựa chọn thuộc tính đầu vào với giá trị và số lượng hợp lý sẽ đảm bảo hiệu quả dự báo của mô hình đồng thời không làm tăng tính phức tạp của mô hình.

Đối với bài toán dự báo thị trường chứng khoán, nhiều nghiên cứu của các nhóm tác giả khác nhau đã có nhiều cách khác nhau để lựa chọn thuộc tính đầu vào, ví dụ

như: sử dụng các chỉ số kinh tế vi mô [22], [28], [31], [54], sử dụng các chỉ số giá cổ phiếu hàng ngày <opening, high, low, closing price> [6], [26], [31], [45], [54], hoặc sử dụng kết hợp cả giá ngày và các chỉ số kinh tế vi mô,…[28], [31], [54]. Ở nghiên cứu này, chúng tôi lựa chọn chỉ số giá cổ phiếu hàng ngày làm dữ liệu vào. Tuy nhiên, tập dữ liệu vào sẽ được tiền xử lý trước khi đưa vào huấn luyện cho mô hình.

Bảng 3.1. Thể hiện các thuộc tính lựa chọn và công thức tính của chúng

Ký hiệu	Thuộc tính	Công thức tính
x_1	EMA100	$P_i - \overline{EMA_{100}(\imath)}$
x_2	RDP-5	$\big(P(i) - P(i-5)\big)/P(i-5) * 100$
x_3	RDP-10	$\big(P(i) - P(i-10)\big)/P(i-10) * 100$
x_4	RDP-15	$\big(P(i) - P(i-15)\big)/P(i-15) * 100$
x_5	RDP-20	$\big(P(i) - P(i-20)\big)/P(i-20) * 100$
y	RDP+5	$(\overline{P(\imath+5)} - \overline{P(\imath)})/\overline{P(\imath)} * 100$ $với\ \overline{P(\imath)} = \overline{EMA_3(\imath)}$

Trong đó, $P(i)$ là chỉ số giá đóng phiên của ngày thứ *i*, và $EMA_m(i)$ là *m-day exponential moving average* của giá đóng phiên ngày thứ *i*.

Theo sự phân tích và đánh giá của L.J. Cao và Francis E.H. Tay trong [26], việc chuyển đổi chỉ số giá ngày thành tỷ lệ sai biệt trung bình 5 ngày (5-day relative difference in percentage of price – RDP) sẽ mang lại một số hiệu quả nhất định, đặc biệt là cải thiện được hiệu quả dự báo. Trong mô mình này, trên cơ sở những đánh giá trong [26], đồng thời để thuận tiện cho việc so sánh đánh giá hiệu quả của mô hình, luận án lựa chọn các biến đầu vào và đầu ra theo đề xuất và tính toán của L.J. Cao và Francis E.H. Tay trong [26].

3.5.2. Lựa chọn các thông số đánh giá hiệu quả mô hình

Để đánh giá hiệu quả của các mô hình dự đoán, cụ thể là trong bài toán dự báo giá cổ phiếu, nhiều tác giả khác nhau đã lựa chọn các thông số khác nhau, phổ biến gồm NMSE, MAE, và DS (Directional Symmetry) [28], [31], [54]. Luận án lựa chọn

3 thông số này để đánh giá kết quả dự báo của mô hình còn có mục đích so sánh với các mô hình đề xuất của các tác giả trong [26] và [45]. Các thông số MAE và NMSE được tính toán theo các công thức (3.3) và (3.6) tương ứng đã được nêu ở mục 3.2.2. Theo đó, độ chính xác của mô hình dự báo càng cao nếu giá trị sai số NMSE và MAE càng nhỏ. Riêng thông số DS nhằm mục tiêu đo lường tỷ lệ dự báo đúng xu hướng (giữ chiều hay đảo chiều) của giá trị cần dự báo RDP+5. Giá trị của DS lớn chứng tỏ tỷ lệ dự báo đúng xu hướng của giá cổ phiếu cao, điều này chứng tỏ mô hình dự báo tốt. Công thức tính giá trị DS như sau:

$$DS = \frac{100}{k}\sum_{t=1}^{k} d_t \tag{3.7}$$

$$\text{với } d_t = \begin{cases} 1 & nếu\ \left((y_t - y_{t-1})(\hat{y}_t - \hat{y}_{t-1})\right) \geq 0 \\ 0 & nếu\ ngược\ lại \end{cases}$$

3.6. Triển khai thực nghiệm

3.6.1. Dữ liệu thực nghiệm

Nguồn dữ liệu thực nghiệm được chọn từ bốn mã cổ phiếu của các tập đoàn và tổ chức tài chính lớn của Mỹ, bao gồm: IBM Corporation stock (IBM), The Apple inc. stock (APPL), The Standard & Poor's stock index (S&P500) và The Down Jones Industrial Average index (DJI) (xem Bảng 3.2). Tất cả các dữ liệu trên được thu thập trực tiếp từ kho dữ liệu lịch sử của sàn chứng khoán Yahoo Finance (http://finance.yahoo.com/). Dữ liệu được thu thập và sử dụng là giá đóng phiên của các mã cổ phiếu lựa chọn trong khoảng thời gian 10 năm. Sau khi thu thập, tất cả dữ liệu được tiền xử lý bằng công cụ Excel qua các bước sau:

1) Loại bỏ những dữ liệu trong các khoảng thời gian mã cổ phiếu bị khóa giao dịch
2) Tính toán các giá trị thuộc tính dữ liệu vào - ra theo Bảng 3.1
3) Scale toàn bộ các giá trị dữ liệu vào - ra trong phạm vi [-0.9, 0.9] như đề xuất trong [26] và [45]. Việc scale dữ liệu này cũng nhằm mục đích đáp ứng

tốt điều kiện áp dụng thuật toán huấn luyện máy học véc-tơ hỗ trợ của thư viện LIBSVM [20].

4) Trích lập dữ liệu thành 3 tập dữ liệu riêng biệt, gồm: Tập dữ liệu huấn luyện, Tập dữ liệu xác thực và Tập dữ liệu thử nghiệm (Bảng 3.2).

3.6.2. Phân tích kết quả thực nghiệm

Thực nghiệm được tiến hành trên từng mã cổ phiếu riêng biệt. Dữ liệu huấn luyện của mỗi mã cổ phiếu sẽ được sử dụng để huấn luyện và trích xuất ra các mô hình mờ riêng biệt, sau đó tập dữ liệu xác thực sẽ được dùng để chạy thử nghiệm và chọn ra các giá trị tối ưu của tham số epsilon và số phân cụm *k*. Cuối cùng tập dữ liệu thử nghiệm tương ứng của từng mã cổ phiếu được dùng để thử nghiệm dự báo và tính toán các giá trị thông số đánh giá mô hình.

Bảng 3.2. Nguồn dữ liệu thực nghiệm

Mã cổ phiếu	Thời gian	Tập dữ liệu huấn luyện	Tập dữ liệu xác thực	Tập dữ liệu thử nghiệm
IBM Corporation stock (IBM)	03/01/2000 - 30/06/2010	2209	200	200
Apple inc. stock (APPL)	03/01/2000 - 30/06/2010	2209	200	200
Standard & Poor's stock index (S&P500)	03/01/2000 - 23/12/2008	2016	200	200
Down Jones Industrial Average index (DJI)	02/01/1991 - 28/03/2002	2152	200	200

Bên cạnh việc thực nghiệm dự đoán dựa trên tập luật mờ sản xuất được từ mô hình SOM+f-SVM và SOM+SVM-IF, các thử nghiệm trên cùng bộ dữ liệu cũng được thực hiện trên các mô hình được đề xuất bởi các tác giả khác, bao gồm mô hình RBN,

mô hình SVM nguyên thủy, mô hình kết hợp SOM+SVM và mô hình kết hợp SOM+ANFIS. Trong đó, mô hình RBN được xây dựng dựa trên mạng nơ-ron hồi qui Generalized là một kiểu của Radial Basis Network (RBN). Mạng nơ-ron hồi qui Generalized được nhiều tác giả nghiên cứu, đề xuất giải quyết bài toán dự đoán [31], [32], [83]. Mô hình SOM+SVM là mô hình dựa trên sự kết hợp của SOM và SVM, được đề xuất để cải tiến hiệu quả vấn đề dự báo dữ liệu chuỗi thời gian mà cụ thể là dự báo giá cổ phiếu [26], [66].

Bảng 3.3. Kết quả thử nghiệm trên mô hình SVM nguyên thủy

Mã cổ phiếu	SVM		
	NMSE	MAE	DS
IBM	1.1215	0.0585	43.01
APPL	1.3230	0.0468	45.84
SP500	1.2308	0.1233	51.23
DJI	1.0785	0.1212	50.05

Bảng 3.4. Kết quả thử nghiệm trên mô hình RBN

Mã cổ phiếu	RBN		
	NMSE	MAE	DS
IBM	1.1510	0.0577	43.72
APPL	1.3180	0.0475	45.73
SP500	1.2578	0.1322	51.76
DJI	1.0725	0.1191	50.05

Bảng 3.3 và Bảng 3.4 thể hiện giá trị các thông số đánh giá kết quả dự báo trên 200 mẫu dữ liệu thử nghiệm với mô hình mạng nơ-ron RBN và mô hình SVM nguyên thủy ứng với cả 4 bộ dữ liệu thực nghiệm. Bảng 3.5 thể hiện giá trị các thông số đánh giá kết quả dự báo ứng với mô hình SOM+SVM cùng trên 4 bộ dữ liệu thực nghiệm đó. So sánh kết quả giá trị các thông số NMSE, MAE và DS trong các thực nghiệm có kết hợp kỹ thuật phân cụm SOM (Bảng 3.5) với các thực nghiệm không có kết hợp kỹ thuật phân cụm SOM (Bảng 3.3 và Bảng 3.4), ta thấy các trường hợp thực nghiệm có kết hợp kỹ thuật phân cụm SOM cho kết quả NMSE và MAE bé hơn, trong khi đó giá trị DS thì lớn hơn. Điều đó có nghĩa là kết quả dự báo của mô hình có kết hợp kỹ thuật phân cụm SOM tốt hơn so với trường hợp không phân cụm.

Bảng 3.5. Kết quả thử nghiệm trên mô hình SOM+SVM

Mã cổ phiếu	Số phân cụm	SOM + SVM			
		Số SV	NMSE	MAE	DS
IBM	6	1355	1.1028	0.0577	44.22
APPL	55	1287	1.1100	0.0445	52.76
SP500	6	965	1.1081	0.1217	52.76
DJI	35	1025	1.0676	0.1186	50.25

Trong quá trình thực nghiệm, luận án cũng triển khai thực nghiệm với mô hình ANFIS đã chuẩn hóa trong thư viện Matlab, tuy nhiên với các tập dữ liệu huấn luyện chưa phân cụm thì quá trình huấn luyện cho mô hình ANFIS quá chậm, vì vậy luận án đã bỏ qua không thực nghiệm mô hình này. Bảng 3.6 thể kết quả thực nghiệm trên cùng tập dữ liệu với các thực nghiệm trên đối với mô hình kết hợp kỹ thuật phân cụm SOM với mô hình ANFIS chuẩn hóa trong thư viện Matlab. Với cùng số phân cụm như nhau, giá trị của thông số NMSE, MAE trong Bảng 3.6 nhỏ hơn so với giá trị của cùng thông số đó trong Bảng 3.5, đồng thời giá trị tương ứng của DS trong Bảng 3.5

thì lớn hơn trong Bảng 3.6. Điều này chứng tỏ mô hình kết hợp SOM-SVM cho kết quả dự báo tốt hơn so với mô hình SOM+ANFIS.

Bảng 3.6. Kết quả thử nghiệm trên mô hình SOM+ANFIS

Mã cổ phiếu	Số phân cụm	SOM + ANFIS		
		NMSE	MAE	DS
IBM	6	1.2203	0.0617	47.74
APPL	55	2.8274	0.0650	49.75
SP500	6	1.7836	0.1421	48.24
DJI	35	1.7602	0.1614	49.75

Bảng 3.7. Kết quả thử nghiệm trên mô hình SOM+f-SVM

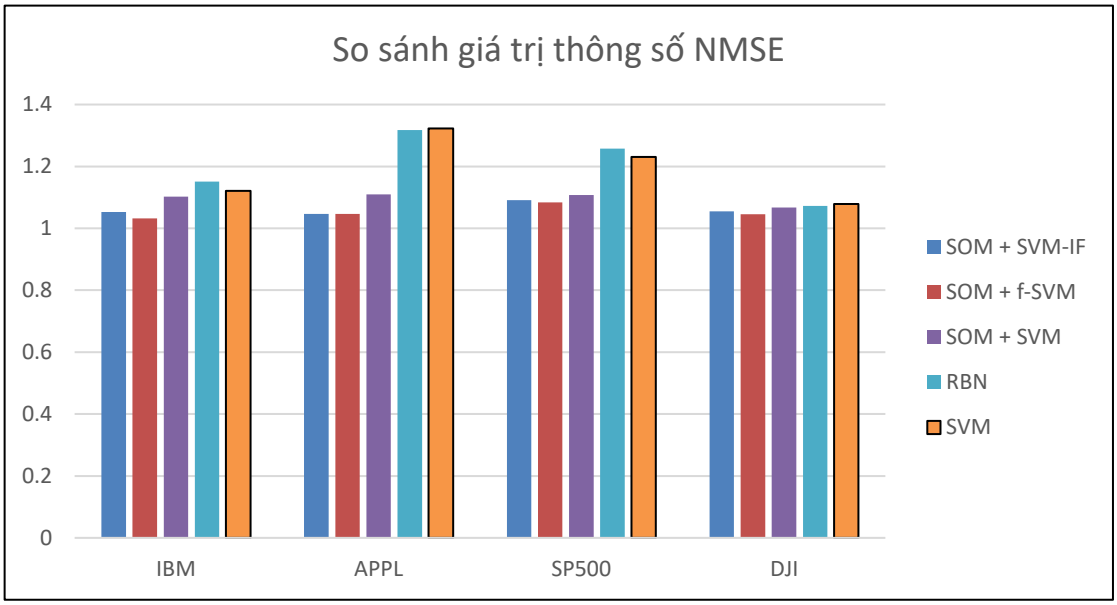
Mã cổ phiếu	Số phân cụm	SOM + f-SVM			
		Số luật	NMSE	MAE	DS
IBM	6	1355	1.0324	0.0554	50.75
APPL	55	1287	1.0467	0.0435	53.27
SP500	6	965	1.0836	0.1207	53.27
DJI	35	1025	1.0459	0.1181	51.76

Bảng 3.7 thể hiện kết quả thử nghiệm dự đoán theo mô hình SOM+f-SVM, đây là mô hình kết hợp kỹ thuật phân cụm SOM với thuật toán f-SVM mà luận án đã đề xuất ở Chương 1 (mô hình ở Hình 3.3). Theo mô hình này, dữ liệu đầu vào sẽ được phân cụm bằng kỹ thuật phân cụm SOM, sau đó mỗi phân cụm dữ liệu sẽ được dùng để huấn luyện cho máy học véc-tơ hỗ trợ để trích xuất ra mô hình mờ theo thuật toán

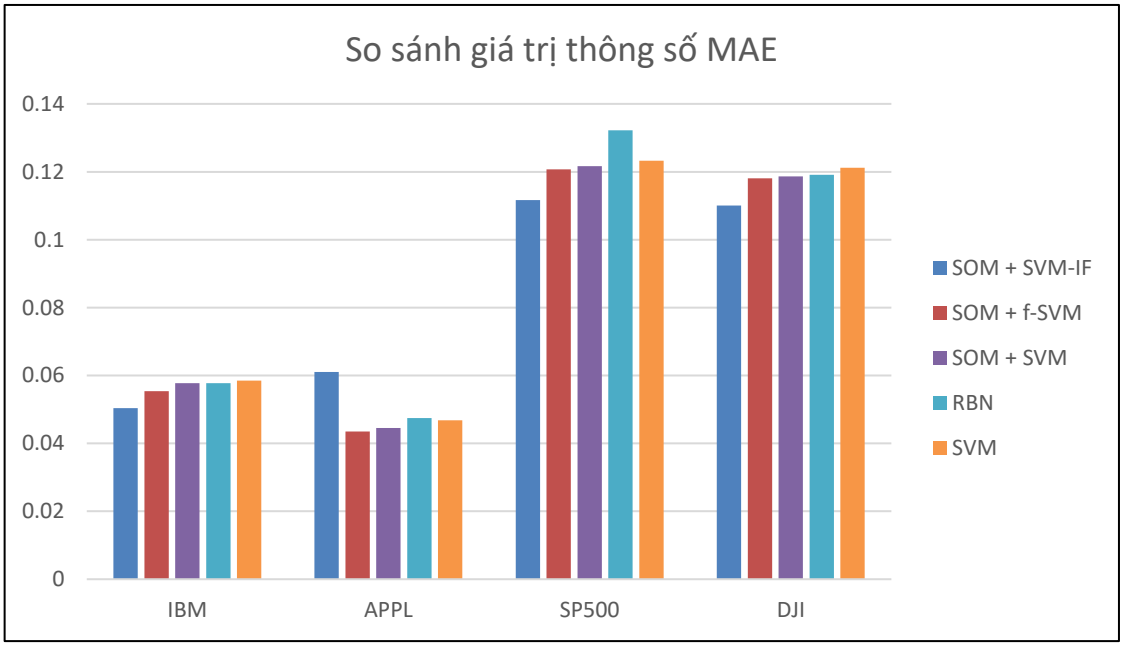
f-SVM. Mô hình SOM+f-SVM mà luận án đề xuất cùng với những kết quả thực nghiệm trên 4 bộ dữ liệu thực tế đã được công bố tại công trình [A2]. Bên cạnh thực nghiệm này, trong quá trình nghiên cứu luận án, một mô hình thực nghiệm khác cũng được triển khai trên cơ sở kết hợp kỹ thuật phân cụm k-Means với thuật toán f-SVM để dự báo cho một số mã cổ phiếu của trị trường chứng khoán Việt Nam. Mô hình hai giai đoạn đề xuất kết hợp k-Means với thuật toán f-SVM cùng với những kết quả thực nghiệm đã được công bố ở công trình [A5]. Tuy nhiên mô hình ứng dụng dự báo một số mã cổ phiếu của thị trường chứng khoán Việt Nam cho độ chính xác của kết quả dự báo không cao, đặc biệt là tỷ lệ dự báo đúng xu hướng của giá cổ phiếu dưới 45%. Một thực nghiệm khác cũng thực hiện trên các mã cố phiểu của thị trường chứng khoán Việt Nam với mô hình kết hợp kỹ thuật phân cụm SOM với thuật toán f-SVM, có điều chỉnh giá trị tham số epsilon để giảm số luật mờ trích xuất được. Kết quả dự đoán cũng gần tương đương với mô hình kết hợp k-Means với f-SVM. Kết quả thực nghiệm này được công bố ở công trình [A7].

Mô hình dự báo đề xuất kết hợp SOM+f-SVM cho kết quả dự báo tốt hơn so với mô hình kết hợp SOM và SVM nguyên thủy. Điều này thể hiện thông qua giá trị các thông số đánh giá mô hình, cụ thể giá trị của các sai số NMSE và MAE trong Bảng 3.7 là nhỏ hơn so với các giá trị các sai số tương ứng trong Bảng 3.5, xét trên cùng mã cổ phiểu (xem biểu đồ sa sánh trong Hình 3.4 và 3.5), trong khi giá trị thông số DS thể hiện cho tỷ lệ dự đoán đúng xu hướng giá cổ phiếu thì lớn hơn (xem biểu đồ so sánh trong Hình 3.6). Qua các thông số đo lường hiệu quả dự báo của mô hình trong Bảng 3.5 và Bảng 3.7 cho thấy mức độ cải thiện của kết quả dự báo theo mô hình SOM+f-SVM so với mô hình SOM+SVM là không nhiều. Tuy nhiên một hiệu quả khác của mô hình SOM+f-SVM đề xuất mang lại chính là tập luật mờ của các mô hình mờ trích xuất được. Các chuyên gia trong lĩnh vực chứng khoán, thậm chí có thể là người sử dụng mô hình dự báo có thể hiểu và giải nghĩa được các luật mờ này, và qua đó có thể hiểu được cơ chế dự báo của mô hình. Điều này hoàn toàn không thể có khi áp dụng mô hình dự báo dựa trên SVM nguyên thủy. Tuy vậy với số lượng luật mờ trong mỗi mô hình lên đến hàng nghìn, tương đương với ½ kích

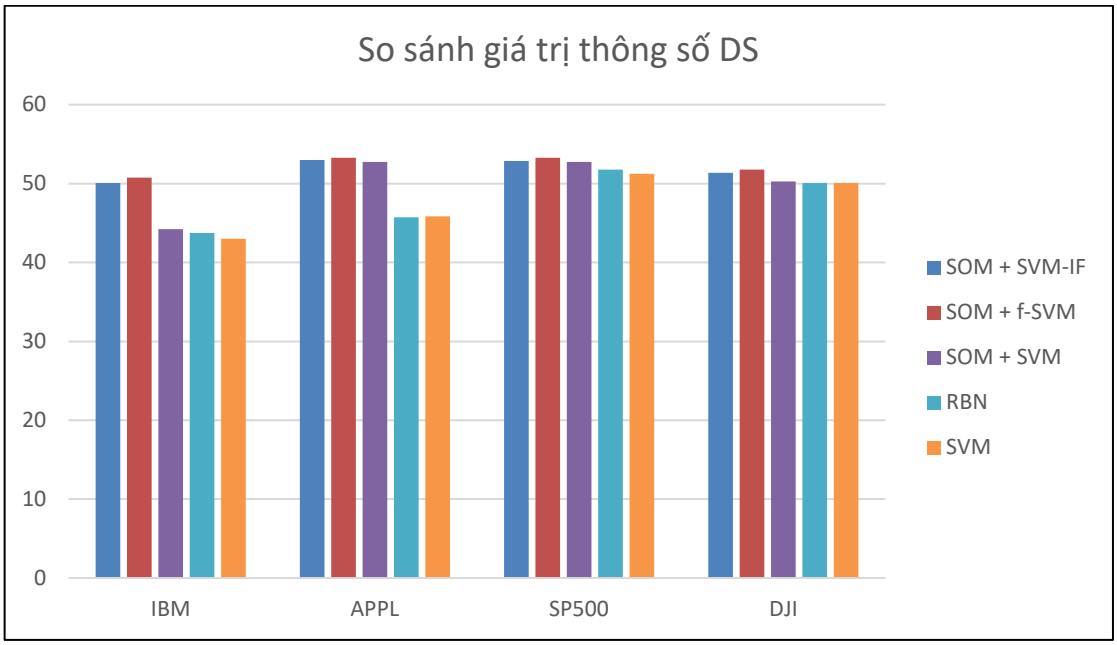
thước dữ liệu huấn luyện (xem Bảng 3.7), thì việc diễn dịch ý nghĩa của tập luật này vẫn là một thách thức rất lớn đối với con người, chưa kể đến sự nhập nhằng của các luật mờ khi chưa được tối ưu hóa vị trí của các hàm thành viên. Mô hình tiếp theo được đề xuất kết hợp kỹ thuật phân cụm SOM với thuật toán SVM-IF sẽ nhằm mục tiêu vượt qua khách thức này. Mô hình đề xuất SOM+SVM-IF cùng với những kết quả thực nghiệm trên 4 mã cổ phiếu đã được công bố trong các công trình [A3], [A6].



Hình 3.4. Biểu đồ so sánh giá trị thông số NMSE



Hình 3.5. Biểu đồ so sánh giá trị thông số MAE



Hình 3.6. Biểu đồ so sánh giá trị thông số DS

Bảng 3.8. Kết quả thử nghiệm trên mô hình SOM+SVM-IF

Mã cổ phiếu	Số phân cụm	SOM + SVM-IF			
		Số luật	NMSE	MAE	DS
IBM	6	30	1.0530	0.0504	50.05
APPL	55	270	1.0466	0.0610	53.00
SP500	6	30	1.0906	0.1117	52.86
DJI	35	175	1.0550	0.1101	51.35

Bảng 3.8 thể hiện kết quả dự báo trên 200 mẫu dữ liệu thử nghiệm theo mô hình kết hợp SOM+SVM-IF. Giá trị các thông số NMSE, MAE và DS của mô hình đề xuất SOM+SVM-IF cho thấy, với cùng số phân cụm được chọn, các kết quả dự báo trên tập dữ liệu thử nghiệm của mô hình SOM+SVM-IF có cải thiện nhiều so với mô hình SOM+ANFIS, mô hình SOM+SVM truyền thống và các mô hình RBN, SVM (xem biểu đồ so sách trong Hình 3.4, 3.5 và 3.6). Riêng so sánh kết quả của mô hình

SOM+SVM-IF với mô hình SOM+f-SVM thì giá trị các thông số đánh giá chỉ tương đương. Tuy nhiên, điểm vượt trội của mô hình SOM+SVM-IF so với mô hình SOM+f-SVM chính là số luật mờ của mô hình mờ trích xuất được đã giảm đi đáng kể trong khi vẫn đảm bảo được hiệu quả khi dự báo. Xét một trường hợp cụ thể, với dữ liệu của mã cổ phiếu S&P500, số luật mờ trích xuất được trong trường hợp áp dụng mô hình SOM+SVM-IF cho cả 6 phân cụm là 5*6=30 luật (Bảng 3.8), trong khi số luật mờ trong trường hợp tương tự ứng với mô hình SOM+f-SVM (Bảng 3.7) và số lượng véc-tơ hỗ trợ trích xuất được theo mô hình SOM+SVM truyền thống (Bảng 3.5) là 965.

Bảng 3.9. Tập 5 luật trong 1 phân cụm trích xuất từ dữ liệu huấn luyện của mã cổ phiếu S&P500

Thứ tự	Luật
R1	IF x1=Gaussmf(0.10,-0.02) and x2=Gaussmf(0.10,-0.08) and x3=Gaussmf(0.10,0.02) and x4=Gaussmf(0.10,0.04) and x5=Gaussmf(0.10,0.02) THEN y=-0.02
R2	IF x1=Gaussmf(0.10,0.02) and x2=Gaussmf(0.09,-0.00) and x3=Gaussmf(0.10,0.06) and x4=Gaussmf(0.10,0.05) and x5=Gaussmf(0.09,0.00) THEN y=0.04
R3	IF x1=Gaussmf(0.09,-0.04) and x2=Gaussmf(0.10,0.07) and x3=Gaussmf(0.09,-0.16) and x4=Gaussmf(0.09,-0.14) and x5=Gaussmf(0.11,-0.05) THEN y=0.16
R4	IF x1=Gaussmf(0.09,0.01) and x2=Gaussmf(0.10,0.08) and x3=Gaussmf(0.09,-0.06) and x4=Gaussmf(0.09,-0.09) and x5=Gaussmf(0.09,-0.04) THEN y=0.01
R5	IF x1=Gaussmf(0.09,-0.05) and x2=Gaussmf(0.09,0.04) and x3=Gaussmf(0.10,-0.13) and x4=Gaussmf(0.10,-0.08) and x5=Gaussmf(0.08,-0.04) THEN y=-0.18

Việc giảm số luật mờ trong các mô hình mờ nhờ sử dụng thuật toán SVM-IF sẽ làm giảm độ phức tạp của mô hình mờ, cải thiện được tốc độ suy diễn, dự báo. Ngoài

ra, ý nghĩa quan trọng hơn của việc áp dụng thuật toán SVM-IF đó là đảm bảo tính có thể diễn dịch được của mô hình mờ trích xuất được. Với việc kết hợp kỹ thuật phân cụm SOM và thuật toán trích xuất mô hình mờ có tích hợp tri thức tiên nghiệm SVM-IF, kết quả mô hình trích xuất được cho mỗi phân cụm sẽ có số luật mờ hạn chế và đã được tối ưu hóa phân bố các hàm thành viên, đảm bảo tính có thể diễn dịch được. Bảng 3.9 thể hiện tất cả 5 luật của một mô hình mờ, tương ứng với một trong 6 phân cụm, trích xuất được từ tập dữ liệu huấn luyện của mã cổ phiểu S&P500 trong trường hợp áp dụng mô hình lai ghép SOM+SVM-IF.

Một trong những mục tiêu hướng đến của việc đề xuất mô hình lai ghép SOM+SVM-IF là có thể phối hợp với những chuyên gia trong lĩnh vực chứng khoản để diễn dịch ngữ nghĩa cho những tập luật rút gọn được trích xuất từ dữ liệu. Việc áp dụng thuật toán SVM-IF có tích hợp tri thức tiên nghiệm để trích xuất ra các mô hình mờ thì tập luật mờ sẽ được tối ưu hóa về số lượng và vị trí các hàm thành viên, đảm bảo tính diễn dịch được. Đồng thời việc kết hợp kỹ thuật phân cụm SOM đã giúp tạo ra các mô hình mờ theo từng phân cụm có số lượng luật mờ hạn chế. Như vậy, rõ ràng mô hình SOM+SVM-IF đề xuất ngoài khả năng tăng hiệu quả dự báo còn hướng đến mục tiêu tiếp tục cải thiện hiệu quả dự báo bằng cách kết hợp với tri thức của chuyên gia trong lĩnh vực dự báo. Với mỗi tập luật rút gọn và đảm bảo tính diễn dịch của mỗi mô hình mờ tích xuất được từ dữ liệu theo từng phân cụm, các chuyên gia có thể diễn dịch ngữ nghĩa và trên cơ sở đó có thể điều chỉnh, bổ sung các luật tinh túy của chuyên gia vào mô hình mờ, qua đó tăng hiệu quả sử dụng mô hình.

3.7. Tiểu kết Chương 3

Bài toán dự báo dữ liệu chuỗi thời gian đã được nhiều tác giả nghiên cứu và đề xuất nhiều mô hình dự báo khác nhau. Máy học véc-tơ hỗ trợ hồi quy đã được nhiều nghiên cứu áp dụng để giải quyết bài toán dự báo chuỗi thời gian và chứng tỏ mang lại hiệu quả. Tuy nhiên mô hình dự đoán dựa trên SVM hồi quy cũng giống như các mô hình máy học thống kê khác, quá trình suy luận hoàn toàn là "hộp đen" đối với con người. Việc xây dựng các mô hình mờ đề giải quyết bài toán dự báo chuỗi thời gian là một trong những hướng nghiên cứu mới thu hút sự quan tâm của nhiều tác giả

và nhà phát triển ứng dụng. Các thuật toán f-SVM và SVM-IF được luận án đề xuất cho phép trích xuất các mô hình mờ dự báo dữ liệu chuỗi thời gian từ dữ liệu thu thập được. Tập các luật mờ "IF…THEN" kết hợp với quá trình suy luận dựa trên tập mờ đã phần nào giúp con người giải được tính "hộp đen" của mô hình máy học thống kê.

Với một bài toán dự báo dữ liệu chuỗi thời gian thực tế, thách thức lớn nhất đặt ra đó là tập dữ liệu huấn luyện có kích thước lớn, mức độ nhiễu của tập dữ liệu huấn luyện cao. Nhằm vượt qua thách thức đó, luận án đã đề xuất mô hình tích hợp nhiều giai đoạn: lựa chọn thuộc tính dữ liệu vào, phân cụm dữ liệu, trích xuất mô hình mờ và áp dụng dự báo. Giải pháp gom cụm dữ liệu theo các thuật toán K-Means hoặc SOM trong giai đoạn tiền xử lý dữ liệu đầu vào là một trong những giải pháp để khắc phục vấn đề gây ra bởi kích thước dữ liệu lớn. Đặc biệt đối với bài toán dự báo dữ liệu chuỗi thời gian tài chính thì việc gom cụm dữ liệu bằng SOM không những khắc phục được vấn đề kích thước dữ liệu lớn, mà còn có thể gom cụm các dữ liệu có sự tương đương nhau về phân bố thống kê. Chính vì vậy độ chính xác của kết quả dự đoán khi áp dụng mô hình lai ghép với kỹ thuật phân cụm sẽ cao hơn. Với việc áp dụng thuật toán SVM-IF để trích xuất mô hình mờ từ dữ liệu huấn luyện, kết hợp với việc sử dụng tập dữ liệu xác thực, mô hình mờ trích xuất được đảm bảo tính diễn dịch được đồng thời đảm bảo được hiệu quả dự báo (trong giới hạn sai số dự báo cho phép).

Những kết quả thực nghiệm trên bài toán dự báo dữ liệu chuỗi thời gian tài chính (cụ thể là 4 mã cổ phiếu thực nghiệm) đã chứng tỏ hiệu quả của mô hình dự báo đề xuất. Cụ thể, mô hình kết hợp SOM+SVM-IF cho kết quả dự báo có độ chính xác cao hơn so với một số mô hình dự báo được đề xuất bởi các tác giả khác. Ngoài ra, với mô hình đề xuất, tập luật mờ rút gọn của mỗi mô hình trích xuất được có thể diễn dịch ngữ nghĩa bởi các chuyên gia trong lĩnh vực dự báo. Qua đó có thể mở ra một hướng phát triển mới cho mô hình dự báo mờ, đó là phối hợp với các chuyên gia trong lĩnh vực dự báo để tối ưu hóa tập luật bằng cách phân tích tập luật học được từ dữ liệu, điều chỉnh các luật hoặc bổ sung thêm luật từ chuyên gia.

KẾT LUẬN

Với mục tiêu là xây dựng mô hình hướng dữ liệu lai ghép dựa trên việc tích hợp tri thức tiên nghiệm với mô hình mờ hướng dữ liệu cho bài toán dự báo hồi quy. Luận án đã đạt được một số kết quả chính như sau:

1) Nghiên cứu các phương pháp xây dựng mô hình mờ, đặc biệt là mô hình mờ hướng dữ liệu, từ đó xây dựng thuật toán trích xuất tập luật mờ TSK từ dữ liệu dựa vào máy học véc-tơ hỗ trợ hồi quy. Thuật toán f-SVM đề xuất cho phép tối ưu hóa các tham số của hàm thành viên mờ và lựa chọn giá trị tham số epsilon để điều chỉnh số lượng luật mờ trích xuất được. Luận án cũng đề xuất sử dụng tập dữ liệu xác thực để thực nghiệm chọn giá trị tham số epsilon tối ưu cho từng mô hình mờ tương ứng với từng bài toán cụ thể. Những thực nghiệm trên các ví dụ cụ thể cho thấy thuật toán f-SVM kết hợp với giải pháp chọn lựa giá trị tham số tối ưu cho phép trích xuất được tập luật mờ từ dữ liệu huấn luyện với số luật mờ được rút gọn nhưng vẫn đảm bảo được hiệu quả dự báo.

2) Nghiên cứu các kịch bản tích hợp tri thức tiên nghiệm vào quá trình học mô hình mờ; đồng thời phân tích điều kiện đảm bảo tính "có thể diễn dịch được" của một mô hình mờ để qua đó lựa chọn, xác định các tri thức tiên nghiệm cụ thể để tích hợp vào quá trình học mô hình mờ TSK dựa vào máy học véc-tơ hỗ trợ. Thuật toán SVM-IF đề xuất có tích hợp tri thức tiên nghiệm về cấu trúc mô hình cho phép trích xuất được tập luật mờ đảm bảo tính "có thể diễn dịch được". Tập luật mờ trích xuất được từ dữ liệu huấn luyện bằng cách sử dụng thuật toán SVM-IF có số luật được rút gọn và đồng thời phân bố của các hàm thành viên mờ được điều chỉnh đều, ít nhập nhằng hơn so với trường hợp sử dụng thuật toán f-SVM.

3) Đề xuất mô hình lai ghép kỹ thuật phân cụm SOM với mô hình mờ trích xuất được từ máy học véc-tơ hỗ trợ để giải quyết bài toán dự báo dữ liệu chuỗi thời gian. Mô hình đề xuất cho phép giải quyết được vấn đề dữ liệu có kích thước lớn và độ nhiễu cao của các bài toán dự báo dữ liệu chuỗi thời gian tài chính nói riêng và các

bài toán dự báo dữ liệu chuỗi thời gian trong thực tế nói chung. Việc tích hợp kỹ thuật phân cụm dữ liệu đầu vào đã làm giảm nhiễu cục bộ trong từng phân cụm và đồng thời giảm kích thước dữ liệu, từ đó làm tăng hiệu quả, giảm độ phức tạp về thời gian của thuật toán huấn luyện mô hình. Số luật mờ trong từng phân cụm tất nhiên là nhỏ hơn so với khi không thực phân cụm, và do vậy tốc độ dự báo dựa vào mô hình cũng sẽ được cải thiện. Mô hình lai ghép giữa kỹ thuật phân cụm SOM và f-SVM do Luận án đề xuất đã được công bố lần đầu ở công trình [A2], đã được trích dẫn ít nhất trong 7 công bố quốc tế của các tác giả ngoài nước, đặc biệt có những trích dẫn mới trong năm 2018 và 2019.

Bên cạnh đó với từng cụm luật mờ có số lượng hạn chế và đã được cải thiện tính "có thể diễn dịch được" bằng thuật toán SVM-IF, những chuyên gia trong từng lĩnh vực cụ thể có thể diễn dịch ngữ nghĩa các tập luật, hiểu được các tập luật, từ đó có thể quyết định lựa chọn bổ sung những luật cần thiết hoặc loại bỏ những luật không phù hợp để tối ưu tập luật. Ở đây, một điểm tồn tại cần được tiếp tục nghiên cứu giải quyết, đó là phân tích ngôn ngữ tập luật mờ trích xuất được từ các tập dữ liệu chuỗi thời gian. Một trong những định hướng nghiên cứu tiếp theo của đề tài luận án là phối hợp với những chuyên gia trong lĩnh vực dự báo để phân tích ngôn ngữ các tập luật mờ trích xuất được và đồng thời tối ưu hóa tập luật bằng tri thức của các chuyên gia.

Điểm tồn tại thứ hai trong vấn đề nghiên cứu của luận án đó là trong các thuật toán f-SVM và SVM-IF đề xuất, việc thay đổi và xác định giá trị tối ưu cho các tham số thông qua thực nghiệm trên tập dữ liệu xác thực không được thực hiện tự động trong thuật toán. Giá trị của các tham số được xác định tùy thuộc vào các tập dữ liệu của từng bài toán dự báo cụ thể. Một định hướng nghiên cứu tiếp theo của đề tài luận án đó là tiến hành nhiều thực nghiệm trên các bài toán xác định, qua đó có sự tổng hợp, thống kê các giá trị tham số được chọn để đề xuất các ngưỡng giá trị tham số phù hợp cho từng bài toán.

Ngoài ra, việc nghiên cứu xác định và lựa chọn những tri thức tiên nghiệm cần thiết để tích hợp vào quá trình huấn luyện mô hình mờ cũng là một hướng nghiên cứu tiếp theo để cải tiến hiệu quả của mô hình.

Những công trình của tác giả liên quan đến luận án

[A1] Duc-Hien Nguyen, Manh-Thanh Le (2013), *Improving the Interpretability of Support Vector Machines-based Fuzzy Rules*, Advances in Smart Systems Research, Future Technology Publications, PO Box 2115, United Kingdom, ISSN: 2050-8662, Vol. 3, No. 1, 7-14.

[A2] Duc-Hien Nguyen, Manh-Thanh Le (2014), *A two-stage architecture for stock price forecasting by combining SOM and fuzzy-SVM*, International Journal of Computer Science and Information Security (IJCSIS), USA, ISSN: 1947-5500, Vol. 12, No. 8, 20-25.

[A3] D.H Nguyen, V.M Le (2018), *Hybrid Model of Self-Organized Map and Integrated Fuzzy Rules with Support Vector Machine: Application to Stock Price Analysis*, Proceedings of Fourth International Conference on Information system Design and Intelligent Applications (INDIA 2017), Advances in Intelligent Systems and Computing, Springer, Singapore, vol 672, 314-322.

[A4] Ngyễn Đức Hiển (2013), *Ứng dụng mô hình máy học véc-tơ tựa (SVM) trong việc phân tích dữ liệu điểm sinh viên*, Tạp chí Khoa học và Công nghệ Đại học Đà Nẵng. Số 12(73), Quyển 2, 33-37.

[A5] Nguyễn Đức Hiển (2014), *Mô hình hai giai đoạn dự báo giá cổ phiếu với K-mean và Fuzzy-SVM*, Tạp chí Khoa học và Công nghệ Đại học Đà Nẵng, Số 12(85), Quyển 2, 20-24.

[A6] Nguyễn Đức Hiển, Lê Mạnh Thạnh (2015), *Mô hình tích hợp f-SVM và tri thức tiên nghiệm cho bài toán dự báo hồi quy*, Tạp chí Khoa học Đai học Huế, Số T. 106, S. 7, 1-14.

[A7] Nguyễn Đức Hiển, Lê Mạnh Thạnh (2015), *Mô hình mờ TSK dự đoán giá cổ phiếu dựa trên máy học véc-tơ hỗ trợ hồi quy*, Tạp chí khoa học Trường Đai học Cần Thơ, Số chuyên đề Công nghệ thông tin, 144-151.

[A8] Nguyễn Đức Hiển, Lê Mạnh Thạnh (2015), *Tối ưu hóa mô hình mờ TSK trích xuất từ máy học véc-tơ hỗ trợ hồi qui với tham số epsilon*, Tạp chí Khoa học và Công nghệ Đại học Đà Nẵng, Số 12(97), Quyển 2, 15-19.

[A9] Nguyễn Đức Hiển, Lê Mạnh Thạnh (2018), *Cải thiện mô hình mờ hướng dữ liệu với tri thức tiên nghiệm.* Tạp chí KH&CN Trường Đại học khoa học – Đại học Huế, Volume 12, 39-49.

[A10] Nguyễn Đức Hiển, Lê Mạnh Thạnh (2018), *Một số giải pháp tối ưu tập luật mờ TSK trích xuất từ máy học véc-tơ hỗ trợ hồi quy*. Kỷ yếu Hội nghị FAIR’2018.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Trần Quang Duy, Nguyễn Công Điều, Vũ Như Lân (2015), *Dự báo chuỗi thời gian mờ dựa trên đại số gia tử*, Kỷ yếu công trình khoa học 2015 - Phần I, Trường Đại học Thăng Long, 30-46.

[2] Nguyễn Cát Hồ, Nguyễn Công Điều, Vũ Như Lân (2016), *Ứng dụng của đại số gia tử trong dự báo chuỗi thời gian mờ*, Journal of Science and Technology, 54(2), 161.

[3] Đào xuân Kỳ (2017), *Ứng dụng mô hình xích Markov và chuỗi thời gian mờ trong dự báo*, Luận án Tiến sỹ Toán học.

[4] Dương Thăng Long (2010), *Phương pháp xây dựng hệ mờ dạng luật với ngữ nghĩa dựa trên đại số gia tử và ứng dụng trong bài toán phân lớp*, Luận án tiến sĩ Toán học, Viện Công nghệ Thông tin - Viện Khoa học và Công nghệ Việt Nam.

[5] Nguyễn Thiện Luận (2015), *Lý thuyết mờ ứng dụng trong tin học*, Nhà xuất bản thống kê.

[6] Vạn Duy Thanh Long, Lê Minh Duy, Nguyễn Hoàng Tú Anh (2011), *Phương pháp dự đoán xu hướng cổ phiếu dựa trên việc kết hợp K-means và SVM với ước lượng xác suất lớp*, Đại học quốc gia – Tp HCM.

[7] Đỗ Thanh Nghị, Nguyễn Minh Trung, Phạm Nguyên Khang (2014), *Phân lớp dữ liệu với giải thuật Newton-SVM*, Tạp chí khoa học Trường Đại học Cần Thơ, 32, 35-41.

[8] Nguyễn Đình Thuận, Hồ Công Hoài (2018), *Kết hợp mô hình arima và support vector machine (SVM) để dự báo tại công ty dịch vụ trực tuyến cộng đồng việt*, Kỷ yếu Hội nghị Fair'2018.

[9] Hoàng Trọng, Chu Nguyễn Mộng Ngọc (2007), *Thống kê ứng dụng trong kinh tế xã hội*, Nhà xuất bản Thống kê.

[10] Chu Văn Tuấn (2008), *Giáo trình Lý thuyết thống kê và Phân tích dự báo*, Nhà xuất bản Tài chính.

Tiếng Anh

[11] Abhishek Verma, Prashant Shukla, Abhishek, Shekhar Verma (2018), *An Interpretable SVM Based Model for Cancer Prediction in Mammograms*, First International Conference -CNC 2018.

[12] Abonyi, J., Babuska, R., Szeifert, F. (2001), *Fuzzy modeling with multivariate membership fuctions: Gray-box identification and control design*, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 31(5), 755-767.

[13] A.J. Smola and B. Scholkopf (1998), *A Tutorial on Support Vector Regression*, NEUROCOLT2 echnical Report Series, NC2-TR- 1998-030.

[14] Andri Riid, Ennu Rüstern (2014), *Adaptability, interpretability and rule weights in fuzzy rule-based systems*, Information Sciences 257, 301–312.

[15] Anuchin Chatchinarat, K. W. Wong, Chun Che Fung (2017), *Rule extraction from electroencephalogram signals using support vector machine*, 9th International Conference on Knowledge and Smart Technology (KST).

[16] B. Scholkopf, P. Bartlett, A. Smola and R. Williamson (1998), *Shrinking the Tube: A New Support Vector Regression Algorithm*, NIPS Conference, Denver, Colorado, USA, November 30 - December 5.

[17] C. F. F. Carraro, M. Vellasco, R. Tanscheit (2013), *A Fuzzy-Genetic System for Rule Extraction from Support Vector Machines*, IEEE.

[18] Chen G. and Pham T.T. (2001), *Introduction to Fuzzy Sets, Fuzzy Logic and Fuzzy Control Systems*, CRC Press, USA.

[19] Chia-Feng Juang, Cheng-Da Hsieh (2012), *A Fuzzy System Constructed by Rule Generation and Iterative Linear SVR for Antecedent and Consequent Parameter Optimization*, IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 20, NO. 2.

[20] Chih-Chung Chang and Chih-Jen Lin (2011), *LIBSVM : a library for support vector machines*, ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm (access time: 18/11/2018).

[21] Chuanhou Gao, Qinghuan Ge, and Ling Jian (2014), *Rule Extraction From Fuzzy-Based Blast Furnace SVM Multiclassifier for Decision-Making*, IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 22, NO. 3.

[22] Christan Pierdzioch, Jorg Dopke, Daniel Hartmann (2008), *Forecasting stock market volatility with macroeconomic variables in real time*. Journal of Economics and Business 60, 256-276.

[23] Corinna Cortes and Vladimir Vapnik (1995), *Support-Vector Networks*. Machine Learning, 20, 273-297.

[24] D. Martens et al. (2008), *Rule Extraction from Support Vector Machines: An Overview of Issues and Application in Credit Scoring*, Studies in Computational Intelligence (SCI) 80, 33–63.

[25] Erol Egrioglu, Yaprak Aslan, Cagdas Hakan Aladag (2014), *A New Fuzzy Time Series Method Based On Artificial Bee Colony Algorithm*, An Official Journal of Turkish Fuzzy Systems Association, Vol.5, No.1, pp. 59-77.

[26] Francis Eng Hock Tay, Li Yuan Cao (2001), *Improved financial time series forecasting by combining Support Vector Machines with self-organizing feature map*. Intelligent Data Analysis 5, IOS press, 339-354.

[27] George Bojadjev, Maria Bojadjev (2007), *Fuzzy logic for Business, Finance, and Management*, World Scientific Publishing Co. Pte. Ltd.

[28] Hajizadeh E., Ardakani H. D., Shahrabi J. (2010), *Application Of Data Mining Techniques In Stock Markets: A Survey*. Journal of Economics and International Finance Vol. 2(7), 109-118.

[29] Hexiang Bai, Yong Ge, Jinfeng Wang, Deyu Li, Yilan Liao, Xiaoying Zheng (2014), *A method for extracting rules from spatial data based on rough fuzzy sets*, Knowledge-Based Systems, 57, 28–40.

[30] H.P. Oak, and Shrikant J. Honade (2015), *ANFIS Based Short Term Load Forecasting*, International Journal of Current Engineering and Technology, Vol.5, No.3.

[31] H. P. Oak, S. J. Honade (2015), *A Survey on Short Term Load Forecasting, Special Issue of International Journal of Electronics*, Communication & Soft Computing Science and Engineering, National Conference on Advanced Technologies in Computing and Networking - ATCON.

[32] Isaac Ibidapo, Ayodele Adebiyi & Olatunji Okesola (2017), *Soft Computing Techniques for Stock Market Prediction: A Literature Survey*, Covenant Journal of Informatics & Communication Technology. Vol. 5 No. 2.

[33] Jang, J.-S. R. (1993), *Anfis: adaptive-network-based fuzzy inference system*, IEEE Tracsctions on Systems, Man and Cybernetic, 23(3), 665-685.

[34] Jin Gou, Feng Hou, Wenyu Chen, Cheng Wang, Wei Luo (2015), *Improving Wang–Mendel method performance in fuzzy rules generation using the fuzzy C-means clustering algorithm*, Neurocomputing 151, 1293–1304.

[35] J.-H Chiang and P.-Y Hao (2004), *Support vector learning mechanism for fuzzy rule-based modeling: a new approach*. IEEE Trans. On Fuzzy Systems, vol. 12, 1-12.

[36] J.L. Castro, L.D. Flores-Hidalgo, C.J. Mantas and J.M. Puche (2007), *Extraction of fuzzy rules from support vector machines*, Elsevier. Fuzzy Sets and Systems, 158, 2057 – 2077.

[37] John Yen, Rezza Langari, Fuzzy logic (1999): *Intelligence, Control, and Information*, Prentice hall, Upper saddle river, New jersy 07458.

[38] J.-S. R. Jang and C.-T. Sun (1993), *Functional equivalence between radial basis function networks and fuzzy inference systems*, IEEE Transactions on Neural Networks, vol. 4, no. 1, 156-159.

[39] Juha Vesanto, Johan Himberg, Esa Alhoniemi, and Jaha Parhankangas (1999), *Self-organizing map in Matlab: the SOM Toolbox*, Proceedings of the Matlab DSP Conference 1999, 35-40.

Toolbox available at http://www.cis.hut.fi/projects/somtoolbox/ .

[40] Juan C. Figueroa-García, Cynthia M. Ochoa-Rey, José A. Avellaneda-González (2015), *Rule generation of fuzzy logic systems using a self-organized fuzzy neural network*, Neurocomputing– ELSEVIER, 151, 955–962.

[41] Kamalpreet Kaur Jassar, Kanwalvir Singh Dhindsa (2016), *Comparative Study and Performance Analysis of Clustering Algorithms*, IJCA - Proceedings on International Conference on ICT for Healthcare ICTHC 2015(1), 1-6.

[42] Kreesuradej W., Wunsch D., Lane M. (1994), *Time-delay Neural Network for Small Time Series Data Sets*, in World Congress Neural Networks, San Diego, CA, vol 2, II-248-II-253.

[43] L. Martin, E Herrera-Viedma, F Herrera, M Delgado (1996), *Combining Numerical and Linguistic Information in Group Decision making*, Journal of Information Sciences, no. 107, 177-194.

[44] Lee C.S. George and Lin C.T. (1995), *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*, Prentice-Hall International, Inc.

[45] L.J. Cao and Francis E.H. Tay (2003), *Support vector machine with adaptive parameters in Financial time series forecasting*, IEEE trans. on neural network, vol. 14, no. 6.

[46] Lorenz E. N. (1963), *Deterministic nonperiodic flow*, Journal of the Atmospheric Sciences, vol. 20, 130–141.

[47] Lua W, Chen X, Pedryczc W, Liu X, Yang J (2015), *Using interval information granules to improve forecasting in fuzzy time series*. International Journal of Approximate Reasoning, 57, 1–18.

[48] MacQueen J. B. (1967), *Some Methods for classification and Analysis of Multivariate Observations*, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297.

[49] Mark Steyvers, Padhraic Smyth, and Chaitanya Chemuduganta (2011), *Combining Background Knowledge and Learned Topics*, Topics in Cognitive Science, Volume 3, Issue 1, 18–47.

[50] Mamdani, E., Asilan, S. (1999), *Experiment in linguistic synthesis with a fuzzy locgic controller*, International Journal of Human Cumputer Studies, 51(2), 135-147.

[51] Mamdani, E.H. (1974), *Application of fuzzy algorithms for control of single dymanic plat*, Proceedings of the Intitutio of Electrical Engineers, 121(12), 1585-1588.

[52] M.C. MacKey and L. Glass (1997), *Oscillation and chaos in physiological control systems*, Science, vol. 197, 287–289.

[53] Md. Rafiul Hassan, Baikunth Nath, Michael Kirley (2007), *A fusion model of HMM, ANN and GA for stock market forecasting*, Expert Systems with Applications 33, 171–18.

[54] Meizhen Liu, Chunmei Duan (2018), *A Review of Using Support Vector Machine Theory to Do Stock Forecasting*, 2018 International Conference on Network, Communication, Computer Engineering.

[55] Muhammad Saleheen Aftab, Muhammad Bilal Kadri (2013), *Parameter Identification of Takagi-Sugeno Fuzzy Model of Surge Tank System*, IEEE.

[56] Nahla Barakat, Andrew P. Bradley (2010), *Rule extraction from support vector machines: A review*, Neurocomputing – ELSEVIER, 74, 178–190.

[57] O. Maimon, L. Rokach (2010), *Chapter 14 & 56*, Data mining and knowledge discovery handbook, 2nd edition, Springer, New York.

[58] Ouahib Guenounoua, Boutaib Dahhoub, Ferhat Chabour (2015), *TSK fuzzy model with minimal parameters*, Applied Soft Computing, 30, 748–757.

[59] Platt J. C. (1999), *Fast Training Of Support Vector Machines Using Sequential Minimal Optimization*, MIT Press, Cambridge, MA, USA.

[60] Prashant Shukla, Abhishek, Shekhar Verma (2017), *A compact fuzzy rule interpretation of SVM classifier for medical whole slide images*, IEEE Region 10 Conference.

[61] R. Courant, D. Hilbert (1953), *Methods of Mathematical Physics*. Wiley, New York.

[62] R. Sindelar and R. Babuska (2004), *Input selection for nonlinear regression models,* IEEE Trans. on Fuzzy Systems, vol. 12, no. 5, 688-696.

[63] S. Chen, J. Wang and D. Wang (2008), *Extraction of fuzzy rules by using support vector machines*. IEEE, Computer society, 438-441.

[64] S. Guillaume (2001), *Designing fuzzy inference systems from data: an interpretability-oriented review,* EEE Transactions on Fuzzy Systems, Institute of Electrical and Electronics Engineers, 9 (3), 426-443.

[65] Serge Guillaume, Luis Magdalena (2006), *Expert guided integration of induced knowledge into a fuzzy knowledge base*, Soft Comput, Springer-Verlag, 10, 773–784.

[66] Sheng-Hsun Hsu, JJ Po-An Hsieh, Ting-Chih CHih, Kuei-Chu Hsu (2009), *A two-stage architecture for stock price forecasting by integrating self-organizing map and support vector regression*, Expert system with applications 36, 7947-7951.

[67] Shri Bharathi, Angelina Geetha (2017), *Sentiment Analysis for Effective Stock Market Prediction*, International Journal of Intelligent Engineering and Systems, Vol.10, No.3, 146-154.

[68] Shumeet Baluja (2002), *Using a priori knowledge to create probabilistic models for optimization*, International Journal of Approximate Reasoning, Volume 31, Issue 3, 193-220.

[69] Sugeno, M. (1985), *Industrial Applications of Fuzzy Control*, North Holland.

[70] Sugeno, M., Yasukawa, T. (1993), *Fuzzy-logic-based approach to qualitative modeling*, IEEE Transactions on Fuzzy Systems, 1(1), 7-31.

[71] Stuart Rusell, Perter Norvig (1989), *Artificial Intelligence: A Modern Approach*, Second Editor, Prentice Hall - Series in Artificial Intelligence.

[72] Teuvo Kohonen (1998), *The self-organizing map*, Elsevier, Neurocomputing 21, 1-6.

[73] T.G. Dietterich (1997), *Machine learning research: Four current directions*, AI Magazine, 18(4), 97-136.

[74] Tulleken, H. (1993), *Grey-box modelling and indentification using physical knowledge and bayesian techniques*, Automatica, 29(2), 285-308.

[75] V.A. Parasich, A.V. Parasich, I.V. Parasich (2017), *methods and principles of using a priori knowledge in recognition tasks*, Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника, Т. 17, № 3. С. 15–23.

[76] Vladimir Cherkassky and Yunqian Ma (2004), *Practical Selection of SVM Parameters and Noise Estimation for SVM Regression*, Neural Networks, Vol 17, Issue 1, Elsevier, 113-126.

[77] Volkan Uslan, Huseyin Seker (2013), *Support Vector-Based Takagi-Sugeno Fuzzy System for the Prediction of Binding Affinity of Peptide*, 35th Annual International Conference of the IEEE.

[78] Xianchang Wang, Xiaodong Liu, Witold Pedrycz, Lishi Zhang (2015), *Fuzzy rule based decision trees*, Pattern Recognition– ELSEVIER, 48, 50–59.

[79] Ying H. (1998), *General Tagaki-Sugeno fuzzy systems with simplifier linear rule consequent are universal controllers, models and filters*, Journal of Information Sciences, no. 108, 91-107.

[80] Y. Jin and B. Sendhoff (2003), *Extracting interpretable fuzzy rules from RBF networks*, Neural Processing Letters, vol. 17, no. 2, 149-164.

[81] Y. Jin, W.V. Seelen, and B. Sendhoff (1998), *An Approach to Rule-Based Knowledge Extraction*, IEEE International Conference on Fuzzy Systems, vol. 2, 1188-1193.

[82] Yolcu, Ufuk Cagcag, Ozge Aladag, Cagdas Hakan Egrioglu, Erol (2014), *An enhanced fuzzy time series forecasting method based on artificial bee colony*, Journal of Intelligent & Fuzzy Systems, vol. 26, no. 6, 2627-2637.

[83] Younes Chtioui, Suranjan Panigrahi, Leonard Francl (1999), *A generalized regression neural network and its application for leaf wetness prediction to forecast plant disease*, Chemometrics and Intelligent Laboratory System 48, 47-58.

[84] Zadeh L.A. (1965), *Fuzzy sets*, Information and Control 8, 338-358.

[85] Zadeh L.A. (1997), *Towward a theory of fuzzy information granulation and its centraliy in human reasoning and fuzzy logic*, Fuzzy sets and Systems, 90(2), 111-127.

[86] Zadeh L.A. (2000), *Fuzzy sets and fuzzy information granulation theory – key selected papers*, Beijing Normal University Press, China

[87] Zhe Gao, and Jianjun Yang (2014), *Financial Time Series Forecasting with Grouped Predictors using Hierarchical Clustering and Support Vector Regression*, International Journal of Grid Distribution Computing Vol.7, No.5, 53-64.

[88] Wang and Mendel (1992), *Fuzzy basis functions, universal approximation, and orthogonal least-squares learning*, IEEE Transactions on Neural Networks, vol. 3, no. 5, 807-814.

[89] Weibei Dou, Ruan, S., Chen, Y., Bloyet, D., and Constans, J.-M (2007), *A framework of fuzzy information fusion for the segmentation of brain tumor tissues on RM images*, Image and Vision Computing, vol. 25, no. 2, 164-171.

[90] Wen Fenghuaa, Xiao Jihongb, He Zhifanga, Gong Xua (2014), *Stock Price Prediction Based on SSA and SVM*, ScienceDirect, Procedia Computer Science 31, 625 – 631.