Rút gọn tập luật mờ trích xuất từ máy học Véc-tơ hỗ trợ bằng cách tích hợp thuật toán phân cụm k-Means

Nguyễn Đức Hiển[1]

[1] Trường Cao đẳng Công nghệ thông tin, Đại học Đà Nẵng
ndhien@cit.udn.vn

Tóm tắt. Mô hình mờ TSK và mô hình máy học véc-tơ hỗ trợ hồi quy có sự tương đương nhau trong một số điều kiện nhất định. Bằng cách thỏa mãn các điều kiện để đồng nhất các hàm đầu ra của mô hình mờ TSK và máy học véc-tơ hỗ trợ hồi quy, chúng ta có thể xây dựng một thuật toán cho phép trích xuất mô hình mờ TSK từ máy học véc-tơ hỗ trợ. Bài báo này đề xuất một giải pháp trích xuất tập luật mờ TSK từ máy học véc tơ hỗ trợ, trong đó có kết hợp giải pháp phân cụm K-Means để rút gọn tập luật. Hiệu quả của giải pháp đề xuất được đánh giá thông qua các kết quả thực nghiệm và có sự so sánh với kết quả của một số thực nghệm trên mô hình khác.

Từ khóa: Mô hình mờ TSK, Máy học Véc-tơ hỗ trợ, Mô hình mờ hướng dữ liệu.

1 Đặt vấn đề

Mô hình mờ được biết đến như là một mô hình khá hiệu quả trong việc xử lý những thông tin mơ hồ và không chắc đó chắn; đồng thời nó cũng thể hiện những lợi thế rõ ràng trong việc biểu diễn và xử lý tri thức. Hoạt động của mô hình mờ phụ thuộc nhiều vào hệ thống các luật mờ và quá trình suy diễn trên tập luật mờ đó. Đã có nhiều tác giả nghiên cứu và đề xuất các phương thức để xây dựng các mô hình mờ hướng dữ liệu [3, 4, 5, 6, 7, 9, 10, 11].

Vấn đề trích xuất mô hình mờ từ máy học Véc-tơ hỗ trợ được nhóm tác giả J.-H Chiang và P.-Y Hao nghiên cứu và công bố lần đầu tiên trong [3]. Một trong những vấn đề của máy học Support-vector là tính chính xác của mô hình thu được tỷ lệ thuận với số lượng Support-vector sinh ra; điều này đồng nghĩa với việc số lượng luật mờ của mô hình mờ trích xuất được sẽ tăng lên. Nói cách khác là khi tăng hiệu suất của mô hình thì đồng nghĩa với việc làm giảm tính "sáng sủa" (tính "có thể hiểu được") của mô hình. Như vậy vấn đề đặt ra là làm thế nào có thể trích xuất được hệ thống mờ đảm bảo độ tin cậy của kết quả dự đoán, đồng thời hạn chế được số lượng luật mờ trong mô hình. Trong bài báo này, chúng tôi đề xuất giải pháp tích hợp thuật toán phân cụm k-Means để rút gọn tập luật mờ trích xất được, đồng thời có thể đảm bảo được độ tin cậy của kết quả dự đoán dựa vào mô hình trích xuất được.

Các phần tiếp theo của bài báo bao gồm: phần 2 trình bày sơ lược về sự tương đương của mô hình mờ TSK và máy học véc-tơ hỗ trợ (SVM – Support Vector Machine) làm

co sở cho việc xây dựng thuật toán trích xuất mô hình mờ. Phần 3 giới thiệu thuật toán fm-SVM* cho phép trích xuất tập luật mờ rút gọn từ SVMs, trong đó có tích hợp thuật toán phân cụm K-Mean. Phần 4 trình bày những kết quả thực nghiệm của mô hình đề xuất, trong đó có kết hợp so sánh với một số kết quả của mô hình đề xuất trước đó. Cuối cùng, trong phần 5 chúng tôi nêu lên một số kết luận và định hướng nghiên cứu tiếp theo.

2 Sự tương đồng của mô hình mờ TSK và máy học véc-tơ hỗ trợ

Hệ thống mờ TSK bao gồm một tập các luật mờ "IF – THEN", với phần kết luận của mỗi luật này là một hàm (không mờ) ánh xạ từ các tham số đầu vào tới tham số đầu ra của mô hình [3,4,5,6,11].

Giả sử có một hệ thống mờ TSK với m luật mờ được biểu diễn như sau:

$$R_j\text{: IF } x_1 \text{ is } A_1^j \text{ and } x_2 \text{ is } A_2^j \text{ and } \dots \text{ and } x_n \text{ is } A_n^j$$
$$\text{THEN } z = g_j(x_1, x_2, \dots, x_n), \text{với } j = 1, 2, \dots, m$$

Trong đó $x_i (i = 1, 2, \dots . n)$ là các biến điều kiện; z là các biến quyết định của hệ thống mờ được xác định bởi hàm không mờ $g_j(.)$; A_i^j là những thuật ngữ ngôn ngữ xác định bởi hàm thành viên tương ứng $\mu_{A_i^j}(x_i)$. Lưu ý, $\mu_{A_i^j}(x_i)$ được định nghĩa như sau:

$$\mu_{A^j}(x_i) = \prod_{i=1}^{n} \mu_{A_i^j}(x_i) \qquad (1)$$

Quá trình suy luận được thực hiện như sau:

1) Kích hoạt các giá trị thành viên.

$$\prod_{i=1}^{n} \mu_{A_i^j}(x_i) \qquad (2)$$

2) Kết quả đầu ra của suy luận được tính như sau:

$$f(x) = \frac{\sum_{j=1}^{m} \overline{z}^j \left(\prod_{i=1}^{n} \mu_{A_i^j}(x_i) \right)}{\sum_{j=1}^{m} \prod_{i=1}^{n} \mu_{A_i^j}(x_i)} \qquad (3)$$

Trong đó, $\overline{z}^j$ là giá trị đầu ra của hàm $g_j(.)$.

Bên cạnh đó, 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 [2,3,6,9]:

Cho một tập dữ liệu huấn luyện $\{(x_1, y_1), \dots, (x_l, y_l)\} \subset \mathcal{X} \times \mathbb{R}$, trong đó $\mathcal{X}$ xác định miền dữ liệu đầu vào. Với ε-Support Vector Regression, bài toán tối ưu hóa ràng buộc cần giải quyết là:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2} w^T w + C \sum_{i=1}^{l} (\xi_i + \xi_i^*) \qquad (4)$$

Sao cho: $(w^T . \Phi(x_i) + b) - y_i \leq \varepsilon + \xi_i,$

$$y_i - (w^T . \Phi(x_i) + b) \leq \varepsilon + \xi_i^*,$$
$$\xi_i, \xi_i^* \geq 0, \text{và } i = 1, 2, \dots, l$$

Và đưa đến bài toán Quadratic Programming:

$$\max_{\alpha,\alpha^*} -\frac{1}{2}\sum_{i,j}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(\Phi(x_i)^T.\Phi(x_j)) - \varepsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) - \sum_{i=1}^{l} y_i(\alpha_i + \alpha_i^*) \quad (5)$$

Sao cho:

$$\sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0, \text{and } C \geq \alpha_i, \alpha_i^* \geq 0, \qquad i = 1, 2, \dots, l$$

Trong đó, C là tham số chuẩn tắc, ε là sai số cho phép, ξ_i, ξ_i^* là những biến lỏng, và α_i, α_i^* là những nhân tử Lagrange.

Véc-tơ w có dạng:

$$w = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*).x_i \quad (6)$$

Và hàm quyết định là:

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\,(\Phi(x_i)^T.\Phi(x_j)) + b \quad (7)$$

Gọi $K(x_i, x_j) = \Phi(x_i)^T.\Phi(x_j)$ là hàm nhân của không gian đầu vào; và hàm quyết định (7) được viết lại như sau:

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(x_i, x) + b \quad (8)$$

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

Xét hàm đầu ra của mô hình mờ TSK (3) và hàm quyết định của mô hình máy học Véc-tơ hồi quy (8). Để (3) và (8) đồng nhất với nhau, trước tiên chúng ta phải đồng nhất giữa hàm nhân trong (8) và hàm thành viên trong (3). Ở đây, để thỏa mãn điều kiện Mercer [1] hàm thành viên Gauss được chọn làm hàm nhân; đồng thời giá trị của b trong (8) phải bằng 0.

Khi hàm Gauss được chọn làm hàm thành viên và hàm nhân, đồng thời số luật mờ bằng với số véc-tơ hỗ trợ (m = l) thì (3) và (8) trở thành:

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\exp\left(-\frac{1}{2}\left(\frac{x_i - x}{\sigma_i}\right)^2\right) \quad (9)$$

và

$$f(x) = \frac{\sum_{j=1}^{l}\overline{z}^j\exp\left(-\frac{1}{2}\left(\frac{x_j - x}{\sigma_j}\right)^2\right)}{\sum_{j=1}^{l}\exp\left(-\frac{1}{2}\left(\frac{x_j - x}{\sigma_j}\right)^2\right)} \quad (10)$$

Như cách biến đổi trong [8], hàm suy luận mờ (10) có thể viết lại như sau:

$$f(x) = \sum_{j=1}^{l}\overline{z}^j\exp\left(-\frac{1}{2}\left(\frac{x_j - x}{\sigma_j}\right)^2\right) \quad (11)$$

Và chúng ta chọn: $\overline{z}^j = (\alpha_i - \alpha_i^*)$ (12)

Như vậy, trên cơ sở thỏa mãn các điều kiện để đồng nhất hàm đầu ra của SVMs và hệ thống mờ TSK, chúng ta có thể trích xuất được mô hình mờ TSK từ máy học Véc-tơ hỗ trợ.

3 Thuật toán trích xuất tập luật mờ rút gọn từ SVM bằng cách tích hợp thuật toán phân cụm K-Means

Từ những phân tích trên, chúng tôi đã đề xuất thuật toán fm-SVM* cho phép trích xuất mô hình mờ TSK từ máy học véc-tơ hỗ trợ gồm các bước nhưu sau:

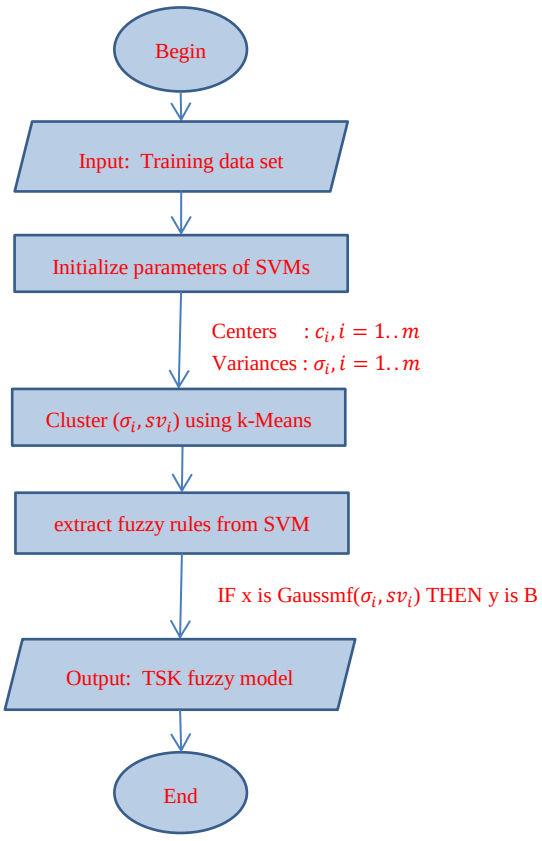
Bước 1: huấn luyện SVMs với tập dữ liệu huấn luyện để trích xuất ra các véc-tơ hỗ trợ tương ứng với các giá trị $x_i = sv_i$, phương sai σ_i, và $(\alpha_i - \alpha_i^*) = B$

Bước 2: dùng thuật toán K-Means để phân cụm các (σ_i, sv_i) thành k cụm

Bước 3: chọn các (σ_i, sv_i) là trung tâm của các cụm để thiết lập các hàm thành viên Gauss cho các luật mờ TSK; trong đó sv_i là trung tâm của hàm thành viên và σ_i là biên độ mờ. Tập các luật mờ TSK trích xuất được sẽ có dạng:

$$IF\ x_i\ is\ Gaussmf(\sigma_i, sv_i)\ THEN\ y_i\ is\ B$$

Sơ đồ thuật toán fm-SVM* được thể hiện ở hình 1.



Hình. 1. Mô hình thuật toán trích xuất tập luật mờ TSK.

4 Kết quả thực nghiệm

Để đánh giá thuật toán fm-SVM* đã đề xuất, chúng tôi xây dựng một hệ thống thử nghiệm dựa trên bộ công cụ Matlab. Trong thuật toán fm-SVM* chúng tôi có sử dụng:
1) Thuật toán học SVM của thư viện LIBSVM được phát triển bởi nhóm của Chih-Chung Chang [12], được sử dụng để sản xuất ra các SVs, làm cơ sở để trích xuất các luật mờ,
2) Hàm công cụ kmeans trong Matlab [14] được sử dụng để phân cụm các SVs.
Ngoài ra, chúng tôi sử dụng hàm AVALFIS trong thư viện công cụ Matlab Fuzzy Logic để thực hiện suy luận dựa trên các luật mờ trích xuất được. Bên cạnh đó, trong thực nghiệm chúng tôi có kết hợp sử dụng thuật toán trích xuất luật mờ f-SVM* đã nghiên cứu đề xuất trong [13] để so sánh và đánh giá kết quả.

4.1 Ví dụ hồi quy phi tuyến tính

Với trường hợp thực nghệm này, chúng tôi tiến hành xây dựng bộ dữ liệu dựa trên bài toán hồi quy phi tuyến sau:

$$\mathrm{Sinc}(x) = \begin{cases} \dfrac{\sin(x)}{x} & \text{if } x \neq o \\ 1 & \text{if } x = 0 \end{cases} \qquad (13)$$

Tập dữ liệu huấn luyện được xác định trong phạm vi từ -3π đến $+3\pi$. 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.

Kết quả dự đoán với 200 mẫu dữ liệu, dựa trên tập luật trích xuất được với số lượng luật khác nhau, đánh giá qua thông số RMSE (Root Mean Squared Error), thể hiện ở bảng 1. Tập luật mờ trích xuất được trong trường hợp gom thành cụm, tương ứng với 6 luật thể hiện ở bảng 2.

Bảng 1. So sánh kết quả các mô hình qua thông số RMSE – 4.1

Số luật mờ	Mô hình áp dụng	
	f-SVM*	fm-SVM*
50	$<10^{-10}$	$<10^{-10}$
10	0.0011	0.0015
8	0.0010	0.0014
6	0.0183	0.0122
4	0.0553	0.0586

Bảng 2. Tập luật trích xuất được trong trường hợp gom thành 6 cụm - 4.1

Thứ tự	Luật
R1	if x is Gaussmf(0.69, -2.502) then y = 0.425

R2	if x is Gaussmf(0.73, -1.820) then y = -1.732
R3	if x is Gaussmf(0.70, -0.582) then y = 1.25
R4	if x is Gaussmf(0.75, 0,580) then y = 1.27
R5	if x is Gaussmf(0.72, 1.815) then y = -1.721
R6	if x is Gaussmf(0.74, 2.861) then y = 0.412

4.2 Ví dụ mô hình dự đoán chuỗi hồi qui hỗn loạn Mackey-Glass

Với trường hợp thực nghiệm này, chúng tôi lựa chọn thử nghiệm trên dữ liệu chuỗi thời gian Mackey-Glass (Mackey-Glass time series). Dữ liệu chuỗi thời gian Mackey-Glass được sinh theo công thức sau:

$$\dot{x} = \frac{ax(t-\tau)}{1+x^b(t-\tau)} - cx(t) \qquad (14)$$

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 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 02 đầu vào và 01 đầu ra.

Bảng 3 thể hiện kết quả dự đoán trên 200 mẫu dữ liệu thử nghiệm, đánh giá thông qua thông số RMSE. Kết quả 9 luật học được từ 800 mẫu dữ liệu huấn luyện trong trường hợp gom thành 9 cụm, tương ứng với 9 luật, thể hiện trong bảng 4.

Bảng 3. So sánh kết quả các mô hình qua thông số RMSE - 4.2

Số luật mờ	Mô hình áp dụng	
	f-SVM*	fm-SVM*
170	$<10^{-10}$	$<10^{-10}$
9	0.0073	0.0078
6	0.0080	0.0082
5	0.0090	0.0082
4	0.0091	0.0095
3	0.0091	0.0103
2	0.0188	0.0205

Bảng 4. Tập luật trích xuất được trong trường hợp gom thành 9 cụm – 4.2

Thứ tự	Luật
R1	if x(t-2) is Gaussmf(0.54,0.40) and x(t-1) is Gaussmf(0.50, 1.11) then x(t) is 1.08
R2	if x(t-2) is Gaussmf(0.52,0.44) and x(t-1) is Gaussmf(0.56, 0.53) then x(t) is 1.12
R3	if x(t-2) is Gaussmf(0.55,0.50) and x(t-1) is Gaussmf(0.55, 1.50) then x(t) is 0.95
R4	if x(t-2) is Gaussmf(0.56,1.02) and x(t-1) is Gaussmf(0.55, 0.51) then x(t) is 1.22
R5	if x(t-2) is Gaussmf(0.60,1.05) and x(t-1) is Gaussmf(0.58, 1.14) then x(t) is 0.98
R6	if x(t-2) is Gaussmf(0.63,1.10) and x(t-1) is Gaussmf(0.54, 1.43) then x(t) is 1.05
R7	if x(t-2) is Gaussmf(0.51,1.27) and x(t-1) is Gaussmf(0.55, 0.53) then x(t) is 1.14

R8	if x(t-2) is Gaussmf(0.56,1.34) and x(t-1) is Gaussmf(0.62, 0.99) then x(t) is 0.98
R9	if x(t-2) is Gaussmf(0.55,1.38) and x(t-1) is Gaussmf(0.57, 1.36) then x(t) is 1.02

5 Kết luận

Bài báo này chúng tôi đề xuất một hướng tiếp cận tích hợp thuật toán phân cụm k-Means vào quá trình trích xuất hệ thống mờ từ máy học Véc-tơ hỗ trợ hồi quy. Máy học Support-vector cung cấp một *framework* để trích xuất ra các Véc-tơ hỗ trợ làm cơ sở để hình thành các luật mờ. Trong bài báo, chúng tôi đã phân tích mối quan hệ tương đương giữa mô hình máy học Véc-tơ hỗ trợ hồi quy và mô hình mờ TSK, để làm cơ sở cho việc xây dựng thuật toán trích xuất mô hình mờ TSK từ máy học Véc-tơ hỗ trợ hồi quy. Bên cạnh đó, để có thể trích xuất được tập uật rút gọn, chúng tôi đã đề xuất thuật toán fm-SVM* dựa trên giải pháp tích hợp thuật toán phân cụm k-Means với quá trình trích xuất tập luật mờ từ tập Véc-tơ hỗ trợ. Kết quả thực nghiệm trên dữ liệu thử nghiệm cho thấy được hiệu quả của mô hình đề xuất. Mặt khác, với tập luật rút gọn trích xuất được, sẽ góp phần làm giảm độ phức tạp của quá trình dự đoán dua trên tập luật học được. Ngoài ra, tập luật rút gọn cũng làm tăng tính "sáng sủa" mô bình mờ, các chuyên gia con người có thể đọc hiểu và phân tích được tập luật này, và từ đó có thể đánh giá tập luật mờ và có thể có giải pháp để tối ưu hóa tập luật.

Trong định hướng nghiên cứu tiếp theo, chúng tôi sẽ ứng dụng thuật toán trích xuất luật mờ fm-SVM* để xây dựng các mô hình mờ tích hợp giải quyết một số bài toán dự đoán, dự báo. Các mô hình mờ tích hợp này có thể được xây dựng dựa trên cơ sở phân tích ngữ nghĩa các luật mờ "có thể hiểu được" học được từ dữ liệu để từ đó tối ưu hóa tập luật và lựa chọn bổ sung một số luật thu thập được từ chuyên gia con người.

Tài liệu tham khảo

1. R. Courant, D. Hilbert: Methods of Mathematical Physics. Wiley, New York (1953).
2. Corinna Cortes and Vladimir Vapnik: Support-Vector Networks, Machine Learning, 20, 273-297 (1995).
3. J.-H Chiang, P.-Y Hao: Support vector learning mechanism for fuzzy rule-based modeling: a new approach, IEEE Trans. On Fuzzy Systems, vol. 12, pp. 1-12 (2004).
4. J.L. Castro, L.D. Flores-Hidalgo, C.J. Mantas, J.M. Puche: Extraction of fuzzy rules from support vector machines, Elsevier. Fuzzy Sets and Systems, 158, 2057–2077 (2007).
5. J.-S. R. Jang: Anfis: adaptive-network-based fuzzy inference system, IEEE Transactions on Systems, Man and Cybernetics, vol. 23, no. 3, pp. 665-685 (1993).
6. David Martens, Johan Huysmans, Rudy Setiono, Jan Vanthienen, Bart Baesens: Rule Extraction from Support Vector Machines - An Overview of issues and application in credit scoring, Studies in Computational Intelligence (SCI) 80: 33–63 (2008).
7. M. Azeem, M. Hanmandlu, N. Ahmad: Generalization of adaptive neuro-fuzzy inference systems, IEEE Transactions on Neural Networks, vol. 11, no. 6.
8. R. Sindelar, R. Babuska: Input selection for nonlinear regression models, IEEE Trans. on Fuzzy Systems, vol. 12, no. 5, 688-696 (2004).
9. S. Chen, J. Wang, D. Wang: Extraction of fuzzy rules by using support vector machines, IEEE, Computer society, pp. 438-441 (2008).

10. Serge Guillaume, Luis Magdalena: Expert guided integration of induced knowledge into a fuzzy knowledge base, Soft Comput, Springer-Verlag 2006, 10:733-784 (2006).
11. Y. Jin, B. Sendhoff: Extracting interpretable fuzzy rules from RBF networks, Neural Processing Letters, vol. 17, no. 2, pp. 149-164 (2003).
12. Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen lin: A practical Guide to Support Vector Classification. http://www.csie.ntu.edu.tw/~cjlin/libsvm/ (2010)
13. Nguyễn Đức Hiển, Lê Mạnh Thạnh: Cải thiện mô hình mờ hướng dữ liệu với tri thưc tiên nghiệm. Kỷ yếu Hội thảo CITA'2016 (2016).
14. k-means clustering, https://www.mathworks.com/help/stats/kmeans.html, last accessed 2017/11/05.