Phương pháp phân cụm mới bằng thuật toán ABC (Artificial Bee Colony – Bầy ong nhân tạo)

Phạm Thị Hường – Bộ môn CNTT

Tóm tắt: Thuật toán ABC là một trong những thuật toán tối ưu hóa được giới thiệu gần đây nhất, mô phỏng cách thức tìm kiếm thức ăn của một bầy ong. Phân cụm được sử dụng trong nhiều lĩnh vực và ứng dụng, đây là một công cụ quan trọng có nhiệm vụ tìm kiếm để xác định các nhóm đối tượng đồng nhất dựa trên giá trị các thuộc tính của chúng. Trong nghiên cứu này trình bày thuật toán ABC và so sánh với thuật toán PSO (Particle Swarm Optimization – Tối ưu bầy đàn) và 9 kĩ thuật phân lớp khác.13 tập dữ liệu thử nghiệm điển hình từ dữ liệu học máy UCI (The UCI Machine Learning Repository) được sử dụng để chứng minh kết quả của các kĩ thuật. Kết quả mô phỏng cho thấy thuật toán ABC có thể được sử dụng hiệu quả cho phân cụm dữ liệu đa biến.

1. Giới thiệu

Phân cụm là một công cụ quan trọng đối với nhiều ứng dụng trong khai phá dữ liệu, phân tích thống kê dữ liệu, nén dữ liệu và lượng hóa vector… nhằm mục đích thu gom dữ liệu vào các cụm (hay các nhóm), nhờ vậy dữ liệu trong mỗi nhóm được phân chia có sự tương đồng cao, trong khi rất khác với dữ liệu ở các nhóm khác. Mục tiêu của việc phân cụm là để nhóm dữ liệu vào các cụm sao cho sự tương đồng giữa các dữ liệu trong cùng một cụm là tối đa, và sự tương đồng giữa các dữ liệu từ các cụm khác nhau là tối thiểu.

Các thuật toán phân cụm thường được phân loại là thuật toán phân cấp (có thứ bậc) và thuật toán phân hoạch (không có thứ bậc). Phân cụm phân cấp nhóm các đối tượng dữ liệu với một chuỗi các phân vùng, hoặc từ những cụm đơn để một cụm bao gồm tất cả các dữ liệu riêng biệt hoặc ngược lại. Phương pháp phân cấp có thể khiến dữ liệu được gộp nhóm hoặc phân chia.

Phương pháp phân hoạch phân chia tập dữ liệu vào một tập hợp các cụm dữ liệu tách rời nhau mà không có cấu trúc phân cấp. Thuật toán phân hoạch phổ biến nhất là thuật toán phân cụm dựa trên mẫu thử: Xác định đối tượng đại diện của các cụm (là đối tượng ở trung tâm của cụm) *(sau đó mỗi đối tượng dữ liệu sẽ được đưa vào cụm mà khoảng cách từ đối tượng dữ liệu đến đối tượng đại diện cụm là nhỏ nhất, sau mỗi bước đối tượng đại diện mỗi cụm có thể được xác định lại dựa vào các đối tượng dữ liệu thuộc cụm đó) và hàm mục tiêu được dùng (hàm sai số bình phương) là tổng khoảng cách từ đối tượng dữ liệu mẫu đến trung tâm.*

Thuật toán phổ biến nhất trong phân cụm là K-means, dựa vào dữ liệu trung tâm, đơn giản và nhanh. Tuy nhiên, thuật toán K-means phụ thuộc nhiều vào trạng thái đầu vào và luôn tập trung tối ưu địa phương từ vị trí bắt đầu tìm kiếm. Để khắc phục vấn đề tối ưu địa phương, các nhà nghiên cứu từ các lĩnh vực khác nhau đang áp dụng phân cụm phân cấp, phân cụm dựa trên phân hoạch, phân cụm dựa trên mật độ và phương pháp phân cụm dựa trên trí tuệ nhân tạo, dữ liệu thống kê, lý thuyết đồ thị, thuật toán cực đại học kỳ vọng (Expectation-maximization Algorithms), mạng Neural nhân tạo (Artifical Neural Networks), thuật toán tiến hóa, thuật toán bầy đàn thông minh, …..

Thuật toán ABC là thuật toán tối ưu hóa, được mô tả bởi Karaboga, dựa trên hành vi tìm kiếm thức ăn tối ưu của ong mật, được áp dụng phân lớp vấn đề (13 CSDL thử nghiệm điển hình). Thực nghiệm thuật toán phân cụm ABC được so sánh với kết quả của thuật toán PSO (Tối ưu bầy đàn) trên cùng một tập dữ liệu. Thuật toán ABC & PSO được đặt trong cùng một lớp của thuật toán tối ưu

hóa trí tuệ nhân tạo, dựa trên quần thể và được đề xuất bởi cảm hứng từ trí tuệ bầy đàn. Bên cạnh sự so sánh thuật toán ABC với PSO,thì thuật toán ABC cũng được so sánh với một tập các kĩ thuật phân lớp mở rộng.

2. Vấn đề phân cụm

Phân cụm là một quá trình nhận biết tự nhiên các cụm hoặc các nhóm trong dữ liệu đa chiều dựa trên một vài độ đo tương đồng. Đo khoảng cách thường được sử dụng cho những đánh giá tương đồng giữa các mẫu. Cụ thể, cho N đối tượng, phân bổ từng đối tượng đến một trong những cụm K và tối thiểu tổng bình phương khoảng cách Euclidean giữa mỗi đối tượng và trung tâm của cụm.

Công thức tối thiểu vấn đề phân cụm như sau:

$$J(w,z) = \sum_{i=1}^{N}\sum_{j=1}^{K} w_{ij}\left\|x_i - z_j\right\|^2 (1)$$

Trong đó, K là số lượng các cụm, N là số lượng các mẫu dữ liệu, x_i ($i = 1,\ldots, N$), là vị trí mẫu dữ liệu thứ i và z_j ($j = 1,\ldots, K$) là trung tâm cụm thứ j, được tính bởi công thức:

$$z_j = \frac{1}{N_j}\sum_{i=1}^{N} w_{ij}x_i (2)$$

Trong đó N_j là số lượng các mẫu dữ liệu trong cụm thứ j, w_{ij} là trọng lượng kết hợp của mẫu x_i với cụm j, có thể bằng 1 hoặc 0 (Nếu mẫu dữ liệu x_i thuộc cụm j thì w_{ij} bằng 1, ngược lại bằng 0).

Quá trình phân cụm, phân tách các đối tượng vào các nhóm (các lớp), được thực hiện bởi học có giám sát hoặc không giám sát. Trong phân cụm không giám sát, có thể đặt tên cụm tự động, dữ liệu huấn luyện không cần phải xác định số lượng các lớp. Tuy nhiên, trong phân cụm có giám sát dữ liệu huấn luyện phải xác định những gì được học, số lượng các lớp. Các tập dữ liệu ta giải quyết bao gồm thông tin các lớp, vì vậy mục tiêu tối ưu hóa để tìm ra các trung tâm của các cụm bằng tối thiểu hàm mục tiêu, tổng các khoảng cách từ các mẫu dữ liệu đến các trung tâm của chúng.

Trong nghiên cứu này, sự thích ứng được thực hiện bằng cách tối thiểu (tối ưu hóa) tổng trên tất cả các trường hợp tập huấn luyện của khoảng cách Euclidean trong không gian N-chiều giữa khoảng cách tổng quát x_j và trung tâm của cụm dữ liệu z_j.

Công thức hàm giá trị cho mẫu dữ liệu i như sau:

$$f_i = \frac{1}{D_{Train}}\sum_{j=1}^{D_{Train}} d\left(x_j, p_i^{CL_{known(x_j)}}\right) (3)$$

Trong đó , D_{Train} là số lượng các mẫu dữ liệu huấn luyện được dùng để chuẩn hóa tổng nằm vào khoảng cách bất kì trong [0.0, 1.0] và $p_i^{CL_{known(x_j)}}$ xác định lớp trong trường hợp thuộc vào CSDL.

3. Thuật toán Bầy ong nhân tạo

Thuật toán ABC được đưa ra bởi Karaboga đối với vấn đề tính toán tối ưu. Thuật toán mô phỏng cách thức tìm kiếm thức ăn của một bầy ong mật. Thuật toán rất đơn giản, và tối ưu ngẫu nhiên dựa vào quần thể. Thuật toán ABC được so sánh với những thuật toán phỏng đoán hiện đại nổi tiếng khác như GA (Genetic Algorothm – Thuật toán di truyền), DE (Differential Evolution – Sự tiến hóa khác biệt), PSO (Particle Swarm Optimization – Tối ưu bầy đàn). Thực hiện thuật toán ABC trên mạng nơ-ron được kiểm tra bằng cách thử nghiệm trên XOR, mã hóa – giải mã (Decoder – Encoder), thử nghiệm trên mô hình phân lớp đối với các thuật toán tối ưu dựa trên quần thể và dựa trên độ dốc (gradient-based).

Giải thuật của thuật toán ABC:

1. Tập ví dụ huấn luyện
2. Phần tử khởi tạo $z_i^{c,}$ i = 1….SN
3. Đánh giá sự thích hợp của phần tử (f_i)
4. Khởi tạo vòng lặp thứ 1
5. Lặp:
6. For mỗi ong thợ {

- Tạo ra giải pháp mới v_i bằng cách sử dụng (6)

- Tính toán giá trị f_i
- Áp dụng tiến trình chọn lựa tham lam}

7. Tính toán giá trị xác suất p_i cho giải pháp (z_i) tại (5)
8. For mỗi ong quan sát {
   - Lựa chọn zi phụ thuộc vào pi
   - Tạo ra giải pháp mới vi
   - Tính toán giá trị fi
   - Áp dụng tiến trình chọn lựa tham lam}
9. Nếu có một giải pháp bị loại bỏ thì thay nó bằng một giải pháp mới, được tạo ngẫu nhiên bởi (7)
10. Ghi nhận giải pháp tốt nhất
11. Bước lặp = Bước lặp + 1
12. Cho đến khi Bước lặp = MCN

Trong thuật toán ABC, bầy ong nhân tạo gồm 3 nhóm: ong thợ, quan sát và trinh sát. Sẽ có một con ong đợi để đưa ra quyết định chọn lựa nguồn thức ăn, gọi là ong quan sát, và một con ong đi đến nguồn thức ăn nó khám phá trước đó, gọi là ong thợ. Một loại khác là ong trinh sát có nhiệm vụ thực hiện tìm kiếm ngẫu nhiên để khám phá nguồn thức ăn mới. Vị trí của một nguồn thức ăn đại diện cho một giải pháp có thể cho vấn đề tối ưu hóa và số lượng mật hoa của nguồn thực phẩm tương ứng với chất lượng (sự thích hợp) của các giải pháp liên quan, được tính toán bằng:

$$fit_i = \frac{1}{1+f_i} \qquad (4)$$

Trong thuật toán, một nửa đầu của bầy ong gồm những con ong thợ nhân tạo, một nửa thứ hai tạo thành từ những ong quan sát. Số lượng ong thợ hoặc ong quan sát bằng số giải pháp (những trung tâm cụm) trong một quần thể. Ở bước đầu, thuật toán ABC tạo ra một quần thể ban đầu được phân phối ngẫu nhiên P(C = 0) của SN các giải pháp (Các vị trí nguồn thức ăn), SN biểu thị kích thước của quần thể. Mỗi giải pháp z_i, i = 1, 2, …, SN là một véc tơ D – chiều. D là số lượng sản phẩm của kích thước đầu vào và kích thước cụm cho mỗi bộ dữ liệu, tức là số lượng các thông số tối ưu. Sau khi khởi tạo, vị trí của các quần thể (các giải pháp) là các đối tượng của vòng lặp, C = 1, 2, …, MCN, trong quá trình tìm kiếm của các ong thợ, ong quan sát và ong trinh sát. Một ong thợ tạo ra sự thay đổi vị trí (giải pháp) trong bộ nhớ của nó phụ thuộc vào thông tin cục bộ (thông tin thị giác) và kiểm tra số lượng mật hoa (giá trị sự thích hợp) của nguồn thức ăn mới (giải pháp mới). Với điều kiện là số lượng mật hoa của nguồn mới này cao hơn nguồn trước đó, những con ong sẽ ghi nhớ vị trí mới và quên đi vị trí cũ. Nếu không, nó giữ lại vị trí của nguồn trước đó trong trí nhớ. Sau khi tất cả ong thợ hoàn tất quá trình tìm kiếm, nó chia sẻ những thông tin mật hoa của nguồn thức ăn và thông tin vị trí với ong quan sát trên khu vực quản lý. Một ong quan sát sẽ đánh giá thông tin mật hoa lấy từ tất cả các ong thợ và chọn một nguồn thức ăn với xác suất liên quan đến số lượng mật hoa. Như trong trường hợp của ong thợ, nó tạo ra sự thay đổi vị trí trong trí nhớ và kiểm tra số lượng mật hoa của nguồn ứng cử. Với điều kiện là số lượng mật hoa của nguồn mới này cao hơn nguồn trước đó, những con ong sẽ ghi nhớ vị trí mới và quên đi vị trí cũ. Một con ong quan sát nhân tạo chọn một nguồn thức ăn phụ thuộc vào giá trị xác suất liên kết với nguồn thức ăn p_i, được tính như sau:

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \qquad (5)$$

Trong đó, SN là số lượng nguồn thức ăn, bằng số ong thợ, và fit_i là sự thích hợp của giải pháp đưa ra trong công thức (4), fit_i tỉ lệ nghịch với f_i được đưa ra trong công thức (3), trong đó f_i là hàm chi phí của vấn đề phân cụm. Để tạo ra một vị trí nguồn thức ăn ứng cử từ một nguồn cũ trong bộ nhớ, thuật toán ABC sử dụng công thức sau:

$$v_{ij} = z_{ij} + \emptyset_{ij}(z_{ij} - z_{kj}) \qquad (6)$$

Trong đó, k ∈ {1, 2, …, SN} và j ∈ {1, 2, …, D} là những chỉ số chọn lựa ngẫu nhiên. Mặc dù k được xác định ngẫu nhiên, nó phải khác i. $\emptyset_{i,j}$ là một số ngẫu nhiên trong khoảng [-1, 1]. Nó kiểm soát việc sản xuất ra các nguồn thức ăn lân cận quanh $z_{i,j}$ đại diện cho so sánh hai vị trí thức ăn được tìm thấy bởi một con ong. Có thể thấy từ công thức (6), sự khác biệt giữa thông số $z_{i,j}$ và $z_{k,j}$ giảm, sự nhiễu loạn trên vị trí $z_{i,j}$ cũng được giảm.

Nguồn thức ăn có mật hoa bị những con ong bỏ đi sẽ được thay thế bằng nguồn thức ăn mới của các ong trinh sát. Trong thuật toán ABC, điều này được mô phỏng bằng cách tạo ra một vị trí ngẫu nhiên và thay thế nó vào vị trị bị bỏ. Trong ABC, cung cấp một vị trí không thể được cải thiện hơn thông qua một số xác định số vòng lặp, sau đó nguồn thức ăn được giả định là bị bỏ. Giá trị số xác định vòng lặp là một thông số kiểm soát quan trọng của thuật toán ABC, được gọi là giới hạn để bỏ đi. Giả sử rằng, nguồn thức ăn bị bỏ là z_i và j ∈ {1, 2, …, D}, sau đó ong trinh sát sẽ phát hiện một nguồn thức ăn mới được thay thế bằng z_i. Hoạt động này có thể được định nghĩa như sau:

$$z_i^j = z_{min}^j + ran(0,1)(z_{max}^j - z_{min}^j) \quad (7)$$

Sau mỗi vị trí nguồn thức ăn ứng cử v_{ij} được tạo ra và sau đó được đánh giá bởi những con ong nhân tạo, hiệu quả của nó được so sánh với nguồn thức ăn cũ của nó. Nếu nguồn mới có mật hoa bằng hoặc tốt hơn nguồn cũ, nó được thay thế bằng một nguồn cũ trong bộ nhớ. Ngược lại, một trong những nguồn cũ được giữ lại trong bộ nhớ. Nói cách khác, một cơ chế lựa chọn tham lam được sử dụng như hoạt động chọn lựa giữa nguồn cũ và một nguồn ứng cử. Có ba thông số điều khiển trong ABC: số lượng nguồn thức ăn bằng số ong thợ hoặc số ong quan sát (SN), giá trị giới hạn, số vòng lặp tối đa (MCN).

Trong quá trình tìm kiếm, thăm dò, khai thác được thực hiện cùng nhau. Trong thuật toán ABC, trong khi những con ong quan sát và ong thợ thực hiện quá trình khai thác trong không gian tìm kiếm thì những con ong trinh sát điều khiển quá trình thăm dò. Việc thực hiện tìm kiếm cục bộ của thuật toán ABC phụ thuộc vào tìm kiếm lân cận và cơ chế chọn lựa tham lam thực hiện bởi những con ong thợ và ong quan sát. Việc thực hiện tìm kiếm toàn bộ của thuật toán phụ thuộc vào quá trình tìm kiếm ngẫu nhiên của ong trinh sát và cơ chế sản xuất giải pháp lân cận thực hiện bởi những con ong thợ và ong quan sát.

4. Nghiên cứu thực nghiệm

Trong bài báo này, 13 vấn đề phân lớp từ kho lưu trữ cơ sở dữ liệu nổi tiếng UCI được sử dụng để đánh giá hiệu suất của thuật toán ABC. Các tập dữ liệu và thuộc tính của dữ liệu như: các mẫu, giá trị đầu vào và các lớp được trình bày trong Bảng 1. 13 vấn đề được chọn để so sánh.

Từ cơ sở dữ liệu, đầu tiên 75% dữ liệu được sử dụng đểhuấn luyện và 25% dữ liệu còn lại được sử dụng trong quá trình thử nghiệm. Mặc dù, một số lớp của bộ dữ liệu (như glass, thyroid, wine) được cho dưới dạng danh sách tuần tự, nhưng nó được đổi chỗ để đại diện cho mỗi lớp trong cả huấn luyện và kiểm thử. Kích thước của tập huấn luyện và kiểm thử được trình bày trong bảng 1.

4.1. Các vấn đề kiểm thử

Các vấn đề được xem xét trong bài báo này được mô tả ngắn gọn như sau:

Bảng 1. Thuộc tính của các vấn đề

	Data	Train	Test	Input	Class
Balance	625	469	156	4	3
Cancer	569	427	142	30	2
Cancer-Int	699	524	175	9	2
Credit	690	518	172	51	2
Dermatology	366	274	92	34	6
Diabetes	768	576	192	8	2
E. coli	327	245	82	7	5
Glass	214	161	53	9	6
Heart	303	227	76	35	2
Horse	364	273	91	58	3
Iris	150	112	38	4	3
Thyroid	215	162	53	5	3
Wine	178	133	45	13	3

Tập dữ liệu **Balance (**cân bằng) được tạo ra để mô hình hóa cho các kết quả thực nghiệm tâm lý. Bộ dữ liệu bao gồm 4 yếu tố đầu vào, 3 lớp và có 625 ví dụ, được chia thành 469 ví dụ cho huấn luyện và 156 ví dụ cho kiểm thử.

Các tập dữ liệu **Cancer** and **Cancer-Int** dựa trên tập dữ liệu "ung thư vú Wisconsin - Chẩn đoán" và " ung thư vú Wisconsin – Nguyên thủy " tương ứng. Đó là các chẩn đoán ung thư vú, với 2 đầu ra (phân loại một khối u lành tính hoặc ác tính). Thử nghiệm lần đầu tiên gồm 569 mẫu trong đó gồm 30 mẫu đầu vào và thử nghiệm lần sau gồm 699 mẫu, 9 đầu vào.

Tập dữ liệu **Credit** (thẻ tín dụng tại Úc) để đánh giá mức độ ứng dụng của thẻ tín dụng dựa trên một số thuộc tính. Có tổng số 690 ứng viên và đầu ra gồm có hai lớp. Với 14 thuộc tính, bao gồm 6 giá trị số và 8 giá trị rời rạc, có 2-14 giá trị có khả năng được thực hiện trong 51 giá trị đầu vào.

Tập hợp dữ liệu **Dermatology** (da liễu) chứa số lượng lớn các lớp; 6 lớp là bệnh vẩy nến, viêm da, Liken phẳng, bệnh vảy phấn hồng, viêm da mãn tính, và vảy phấn đỏ nang lông. Có 366 mẫu, trong đó có 34 đầu vào.

Các tập dữ liệu **Diabetes** (bệnh tiểu đường), hai lớp vấn đề chẩn đoán bệnh tiểu đường đó là xác định một người có bệnh tiểu đường hay không, gồm 768 mẫu. Đầu tiên sử dụng 576 mẫu huấn luyện và còn lại 192 mẫu như là tập kiểm thử. Có 8 đầu vào cho mỗi mẫu.

Đối với các vấn đề của **Escherichia coli** (nhiễm khuẩn e.coli), tập dữ liệu ban đầu có 336 ví dụ tạo thành 8 lớp, nhưng 3 lớp được biểu diễn với chỉ 2, 2, 5 ví dụ. Vì vậy, 9 ví dụ được bỏ qua và 327 ví dụ của tổng số, đầu tiên 245 ví dụ dùng cho huấn luyện và còn lại 82 ví dụ dùng kiểm thử. Các tập dữ liệu chứa 327 ví dụ với 7 đầu vào và 5 lớp.

Tập hợp dữ liệu **Glass** chiếm số lượng lớp lớn nhất (6 lớp) trong số các vấn đề chúng ta giải quyết, nó được sử dụng để phân loại các loại kính.

Nó được sử dụng để phân loại các loại kính như xử lí nổi các cửa sổ tòa nhà, xử lý không nổi các cửa sổ tòa nhà, cửa sổ xe, thùng chứa, bộ đồ ăn, hoặc đèn. 9 bộ đầu vào này dựa trên 9 độ đo hóa học với một trong 6 loại kính, tiếp tục với 70, 76, 17, 13, 9, và 29 trường hợp của mỗi lớp tương ứng. Tổng số 214 trường hợp được phân chia gồm 161 trường hợp cho huấn luyện và 53 để kiểm thử.

Cơ sở dữ liệu **Heart** là một chẩn đoán bệnh tim quyết định ít nhất một trong bốn buồng tim giảm đường kính hơn 50% hay không. Nó chứa 76 thuộc tính cho từng mẫu, 35 mẫu được sử dụng cho các giá trị đầu vào. Các dữ liệu được dựa trên dữ liệu Cleveland Heart từ kho lưu trữ với 303 mẫu.

Tập dữ liệu **Horse** được sử dụng để dự đoán số phận của một con ngựa với một cơn đau bụng và để phân loại con ngựa sẽ sống, sẽ chết, hoặc sẽ bị tiêm chết. Các tập dữ liệu được tạo ra

dựa trên dữ liệu **Horse Colic** với 364 mẫu, trong đó có 58 đầu vào từ 27 thuộc tính và 3 kết quả đầu ra.

Tập hợp dữ liệu **Iris** bao gồm 150 đối tượng hoa từ các loài Iris: Setosa, Versicolor, Virginica. Một trong số 50 đối tượng trong ba lớp có 4 biến số: chiều dài đài hoa, chiều rộng đài hoa, chiều dài cánh hoa và chiều rộng cánh hoa.

Thyroid (tuyến giáp) là chẩn đoán tuyến giáp cho dù đó là super hay hypo-chức năng. 5 yếu tố đầu vào được sử dụng để phân loại 3 nhóm chức năng tuyến giáp như là overfunction, normal function, or underfunction. Tập dữ liệu được dựa trên dữ liệu tuyến giáp mới và chứa 215 mẫu.

Dữ liệu **Wine** được lấy từ một phân tích hóa học rượu vang được lấy từ 3 người sản xuất khác nhau. Do đó, phân tích dữ liệu xác định ba loại rượu vang. Có 178 trường hợp của mẫu rượu với 13 đầu vào.

4. Giải thuật và cài đặt

Thuật toán PSO (Tối ưu bầy đàn) là một thuật toán tiến hóa dựa trên quần thể và trên trí thông minh bầy đàn để giải quyết vấn đề. Trong thuật toán PSO mô phỏng các hành vi xã hội của một nhóm các loài chim bay về nguồn tài nguyên, lặp đi lặp lại đánh giá sự thích hợp của những giải pháp ứng cử và ghi nhớ vị trí đó là tốt nhất.

Các tham số của thuật toán PSO là (như trong [26]): n = 50, T_{max} = 1000, v_{max} = 0.05, v_{min} = -0.05, c_1 = 2.0, c_2 = 2.0, w_{max} = 0.9, w_{min} = 0.4. Để thực hiện một so sánh bằng, các giá trị của kích thước quần thể và số vòng lặp tối đa của thuật toán ABC được chọn bằng hoặc ít hơn so với các giá trị kích thước bầy đàn và số lần lặp tối đa được sử dụng trong trường hợp PSO tương ứng. Chẳng hạn như chúng ta chọn kích thước quần thể là 20, số vòng lặp/sự tạo thành tối đa (MCN) là 1000, và giá trị giới hạn 1000. Như vậy, tổng số đánh giá giá trị của thuật toán ABC là 20.000, thuật toán PSO là 50.000. Chúng ta thấy rằng trong tất cả các bước chạy của các thuật toán thì kết quả không khác nhau nhiều, vì vậy mà các thí nghiệm được giảm sau 5 lần khi cho cùng một kết quả.

Kỹ thuật phân lớp khác được đưa vào nhóm của Bayes, dựa trên chức năng, kỹ thuật meta, cây, kỹ thuật dựa trên các luật. Đối với mỗi nhóm, các kỹ thuật được lựa chọn là: Bayes Net từ Bayesian; mạng Perceptron đa tầng (MLP) và mạng dựa trên chức năng - Radial Basis Function Artificial Neural Network (RBF); KStar; Bagging và MultiBoostAB từ công nghệ meta; Naive Bayes Tree (NBTree) dựa trên cây; Ripple Down Rule (Ridor) dựa trên luật và Voting Feature Interval (VFI).

4.3. Kết quả và thảo luận

Đối với mỗi vấn đề, chúng ta cần thống kê tỷ lệ lỗi phân loại (CEP) là tỷ lệ mẫu phân lớp không chính xác của bộ dữ liệu thử nghiệm. Chúng ta phân lớp mỗi mẫu bằng cách gán nó vào lớp mà gần giá trị trung tâm nhất, sử dụng khoảng cách Euclide tính đến trung tâm của các cụm. Giá trị đầu ra này được so sánh với đầu ra mong muốn và nếu chúng không giống nhau, mẫu phân loại không chính xác. Nó được tính toán cho tất cả các dữ liệu thử nghiệm và tổng các sai số mẫu phân loại không chính xác là tỷ lệ kích thước của bộ thử nghiệm, được đưa ra bởi công thức:

$$CEP = 100 \times \frac{ví\ dụ\ phân\ loại\ sai}{kích\ thước\ tập\ dữ\ liệu\ kiểm\ thử} \quad (8)$$

Như đã mô tả ở trên, dữ liệu được đưa ra trong hai phần: tập huấn luyện (75%) và tập kiểm thử (25%). Các kết quả của thuật toán ABC và PSO và tỷ lệ lỗi phân loại (giá trị CEP) được thể hiện trong Bảng 2. Thuật toán ABC thực hiện nhanh hơn thuật toán PSO 12 vấn đề, trong khi PSO có kết quả tốt hơn so với các thuật toán ABC chỉ một vấn đề (vấn đề Glass) trong các mẫu phân loại lỗi. Thêm vào đó, tỷ lệ phân loại lỗi trung

bình cho tất cả các vấn đề là 13,13% cho ABC và 15,99% cho PSO.

Bảng 2: Tỷ lệ phân lớp lỗi trên các tập dữ liệu kiểm thử

	ABC	PSO
Balance	15.38	25.47
Cancer	2.81	5.80
Cancer-Int	0	2.87
Credit	13.37	22.96
Dermatology	5.43	5.76
Diabetes	22.39	22.50
E. coli	13.41	14.63
Glass	41.50	39.05
Heart	14.47	17.46
Horse	38.26	40.98
Iris	0	2.63
Thyroid	3.77	5.55
Wine	0	2.22

Bảng 3: Tỷ lệ phân lớp lỗi trung bình và xếp hạng các kĩ thuật được đưa ra và thuật toán ABC trong từng mẫu

	ABC	PSO	BayesNet	MlpAnn	RBF	KStar	Bagging	MultiBoost	NBTree	Ridor	VFI
Balance	15.38(4)	25.47(9)	19.74(5)	9.29(1)	33.61(10)	10.25(2)	14.77(3)	24.20(8)	19.74(5)	20.63(7)	38.85(11)
Cancer	2.81(2)	5.80 (6)	4.19 (4)	2.93(3)	20.27(11)	2.44(1)	4.47(5)	5.59(6)	7.69(10)	6.36(8)	7.34(9)
Credit	13.37(5)	22.96(10)	12.13(2)	13.81(6)	43.29(11)	19.18(9)	10.68(1)	12.71(4)	16.18(7)	12.65(3)	16.47(8)
Cancer-Int	0.00(1)	2.87 (2)	3.42(3)	5.25(7)	8.17(11)	4.57(5)	3.93(4)	5.14(6)	5.71(9)	5.48(8)	5.71 (9)
Dermatology	5.43(6)	5.76(7)	1.08(1)	3.26(3)	34.66(10)	4.66(5)	3.47(4)	53.26(11)	1.08(1)	7.92(9)	7.60(8)
Diabetes	22.39(1)	22.50(2)	25.52(3)	29.16(7)	39.16(11)	34.05(9)	26.87(5)	27.08(6)	25.52(3)	29.31(8)	34.37(10)
E. coli	13.41(1)	14.63(3)	17.07(5)	13.53(2)	24.38(10)	18.29(8)	15.36(4)	31.70(11)	20.73(9)	17.07(5)	17.07(5)
Glass	41.50(9)	39.05(7)	29.62(5)	28.51(4)	44.44(10)	17.58(1)	25.36(3)	53.70(11)	24.07(2)	31.66(6)	41.11(8)
Heart	14.47(1)	17.46(2)	18.42(3)	19.46(6)	45.25(11)	26.70(10)	20.25(7)	18.42(3)	22.36(8)	22.89(9)	18.42(3)
Horse	38.26(7)	40.98(10)	30.76(2)	32.19(5)	38.46(8)	35.71(6)	30.32(1)	38.46(8)	31.86(3)	31.86(3)	41.75(11)
Iris	0.00(1)	2.63(7)	2.63(7)	0.00(1)	9.99(11)	0.52(5)	0.26(4)	2.63(7)	2.63(7)	0.52(5)	0.00(1)
Thyroid	3.77(2)	5.55(3)	6.66(5)	1.85(1)	5.55(3)	13.32(10)	14.62(11)	7.40(6)	11.11(8)	8.51(7)	11.11(8)
Wine	0.00(1)	2.22(4)	0.00(1)	1.33(3)	2.88(7)	3.99(8)	2.66(6)	17.77(11)	2.22(4)	5.10(9)	5.77(10)

Bảng 4: Tỷ lệ phân lớp lỗi trung bình và xếp hạng chung của các kĩ thuật trên tất cả các mẫu

	ABC	PSO	BayesNet	MlpAnn	RBF	KStar	Bagging	MultiBoost	NBTree	Ridor	VFI
Average	13.13	15.99	13.17	12.35	26.93	14.71	13.30	22.92	14.68	15.38	18.89
Rank	2	8	3	1	11	5	4	10	6	7	9

Bảng 5: Tổng xếp hạng các kĩ thuật và xếp hạng chung dựa trên tổng xếp hạng

	ABC	PSO	BayesNet	MlpAnn	RBF	KStar	Bagging	MultiBoost	NBTree	Ridor	VFI
Total	41	72	46	49	124	79	58	98	76	87	101
Rank	1	5	2	3	11	7	4	9	6	8	10

Trong bảng 3, tỷ lệ phân lớp lỗi của thuật toán ABC và 10 kỹ thuật được đưa ra, và bảng xếp hạng của các kỹ thuật trên mỗi vấn đề được đề cập trong các ngoặc đơn. Người ta có thể dễ dàng thấy rằng, thuật toán ABC thu được giải pháp tốt nhất trong 6 mẫu và thu được giải pháp tốt thứ hai trong 2 mẫu. Có thể so sánh tất cả các thuật toán trong Bảng 4 và 5. Bảng 4 cho biết các phân lớp lỗi trung bình của tất cả các mẫu và xếp hạng chung dựa trên các giá trị trung bình. Bảng 5 là tổng của bảng xếp hạng của các thuật toán của mỗi mẫu và sắp xếp những số từ giá trị nhỏ đến giá trị lớn. Thời gian thực hiện thì các kỹ thuật không được xem xét đến khi mà thời gian thực hiện ít hơn 1 phút trên máy tính với 2.6GHz Core 2 Duo và 2.0GB-RAM.

Kỹ thuật mạng nơ ron nhân tạo MLP là tốt nhất, ABC là tốt thứ hai, và

BayesNet là thứ ba khi xem xét đến giá trị CEP từ bảng 4. Tuy nhiên, ngay cả khi kết quả trong bảng có thể so sánh, chúng ta thấy rằng có một số điểm quan trọng có thể bỏ qua khi sự phân bố của các lỗi không tỷ lệ thuận nhau. Hơn nữa, trong khi tỷ lệ lỗi khác nhau khoảng 5% trong một số mẫu, nhiều hơn 30% so với một số các trường hợp khác. Vì vậy, việc xếp hạng chung của các kỹ thuật trong Bảng 5 được thực hiện bằng cách tính toán tổng các xếp hạng mỗi mẫu ở bảng 3. Từ bảng xếp hạng này, thuật toán ABC xếp đầu tiên, BayesNet xếp thứ hai, và MLP xếp thứ ba. Tỷ lệ kiểm tra lỗi và thứ hạng trong các bảng cho phân cụm với thuật toán ABC có khả năng khái quát cao.

5. Kết luận

Thuật toán ABC là một kỹ thuật mới, đơn giản và tối ưu mạnh mẽ, được sử dụng trong kỹ thuật phân cụm. Phân cụm là một kỹ thuật phân lớp quan trọng, tập hợp dữ liệu được gom vào lớp (hoặc cụm) mà dữ liệu trong mỗi cụm có sự tương tự cao và có sự khác biệt lớn về dữ liệu so với cụm khác. Việc thực hiện thuật toán ABC so sánh với thuật toán PSO và kỹ thuật khác. Các kết quả của thí nghiệm cho thấy, thuật toán ABC được áp dụng thành công để phân cụm với mục đích phân lớp.

6. Tài liệu tham khảo

[1]https://econ.ubbcluj.ro/rodica.lung/taco/ArtificialBeeColony.pdf
[2] http://mf.erciyes.edu.tr/abc/
[3] A Simple and Efficient Artificial Bee Colony Algorithm - Yunfeng Xu, Ping Fan, and Ling Yuan.
[4] Overview of Artificial Bee Colony (ABC) algorithm and its applications.