

К.Ю. Островська, О.І. Михальов, А.І. Черевко

АЛГОРИТМИ НЕЧІТКОЇ КЛАСТЕРИЗАЦІЇ ДЛЯ АНАЛІЗУ МЕТАЛОГРАФІЧНИХ ЗОБРАЖЕНЬ

Анотація. В роботі було реалізовано два алгоритми нечіткої кластеризації: *s*-середніх та субтрактивний метод нечіткої кластеризації для зразка мікроструктури металографічного зображення $x100$, зразок № 12 (плавка № 22325), полоса мартенсіта.

Ключові слова: кластерний аналіз, кластер, нечітка кластеризація, алгоритм *fst*, субтрактивна кластеризація, центр кластеру, металографічні зображення.

Термін “кластерний аналіз” насправді включає в себе набір різних алгоритмів класифікації. Загальне питання, що ставиться дослідниками у багатьох галузях, полягає в тому, як організувати спостережувані дані в наочні структури.

Концептуальний взаємозв'язок між кластерним аналізом і теорією нечітких множин ґрунтується на тій обставині, що при розв'язанні завдань структуризації складних систем більшість формованих класів об'єктів розмиті за своєю природою. Ця розмитість полягає в тому, що перехід від належності до неналежності елементів до даних класів скоріше поступовий, ніж стрибковий.

У загальному випадку завданням нечіткої кластеризації є знаходження нечіткого розподілу або нечіткого покриття множини елементів досліджуваної сукупності, які утворюють структуру нечітких кластерів, присутніх у розглянутих даних. Ця задача зводиться до знаходження ступенів належності елементів універсуму потрібним нечітким кластерам, які в сукупності і визначають нечіткий розподіл або нечітке покриття вихідної безлічі розглянутих елементів.

Об'єктами дослідження є зразки металографічних зображень колісних сталей - колісних пар, які відносяться до ходових частин і є одним із відповідальних елементів вагона. Вони призначені для спрямування руху вагона по рейковій колії і сприйняття всіх наван-

тажень, що передаються від вагона на рейки при обертанні цих колісних пар. Працюючи в складних умовах завантаження, колісні пари повинні забезпечувати високу надійність, бо від них багато в чому залежить безпека руху поїздів. Конструкція і технічний стан колісних пар впливають на плавність ходу, величину сил, що виникають при взаємодії вагона і колії, і опір руху.

Аналіз одержаних з ВАТ «НТЗ» зразків колісного металу показав наявність у них великих екзогенних включень, здебільшого таких, що являють собою продукти руйнування футеровки сталевоплавильних агрегатів і сталеворозливних пристроїв, а також шихти і шлаків.

У багатьох випадках забрудненість стали неметалевими включеннями комплексна, тому цікаво вивчити сумарну забрудненість колісної сталі неметалевими включеннями, з метою врахування цього фактора у майбутньому при розробці математичних моделей.

Для аналізу металографічних зображень було обрано зразок зображення колісної сталі:

- ліквацийна неоднорідність мікроструктури $\times 100$, зразок № 12 (плавка № 22325), полоса мартенсіта/

У системі MATLAB для розв'язання задачі нечіткої кластеризації реалізовані два алгоритми:

- алгоритм FCM;
- алгоритм субтрактивної нечіткої кластеризації.

За допомогою алгоритмів нечіткої кластеризації (алгоритм FCM; алгоритм субтрактивної нечіткої кластеризації) у пакеті прикладних програм MATLAB буде проведений аналіз над обраним металографічним зображенням. На рисунку 1 представлено зразок мікроструктури колісної сталі - $\times 100$, зразок № 12 (плавка № 22325), полоса мартенсіта.



Рисунок 1 – Зразок № 12 (плавка № 22325), полоса мартенсіта

Після отримання другого зображення ми створюємо цикл за допомогою якого отримуємо зображення з поданням про значення кожного пікселя. Отриманий в результаті файл lom.dat будемо використовувати для подальшого дослідження, тому що дані для кластеризації повинні бути представлені в вигляді матриці чисел.

```
>> i3=I;
>> fid=fopen('lom.dat','w');
>> for i=1:480; fprintf(fid,'%d ',i3(i,:)); fprintf(fid,'\n'); end;
>> fclose(fid);
```

Функція fcm виконується ітераційно до тих пір, поки зміни цільової функції перевищують деякий заданий поріг. На кожному кроці в командному вікні Matlab виводяться порядковий номер ітерації і відповідне поточне значення цільової функції. Якщо після запису функції fcm у другому рядку не ставити крапку з комою (;), то у вікні команд будуть показані значення координат центрів нечітких кластерів, значення функцій приналежності об'єктів нечітким кластерам і значення цільової функції на кожній з ітерацій роботи алгоритму FCM.

```
// завантаження даних, що підлягають кластеризації з файлу lom.dat
>> load lom.dat
// визначення центру кластеризації (4 кластери)
>> [center,U,obj_fcn]=fcm(lom,4);
Iteration count = 1, obj. fcn = 506505969.155656
Iteration count = 2, obj. fcn = 391847719.089683
Iteration count = 3, obj. fcn = 391811913.210103
Iteration count = 4, obj. fcn = 391811481.300471
Iteration count = 5, obj. fcn = 391811467.524598
Iteration count = 6, obj. fcn = 391811466.982057
Iteration count = 7, obj. fcn = 391811466.959814
Iteration count = 8, obj. fcn = 391811466.958894
Iteration count = 9, obj. fcn = 391811466.958856
Iteration count = 10, obj. fcn = 391811466.958854
>> plot(obj_fcn);
```

Для оцінки динаміки зміни значень цільової функції використовується команда побудови графіка plot(obj_fcn). Результати показані на рисунку 2, з рисунку видно, що чим вище значення цільової функції fcn тим краще кількість ітерацій нечіткої кластеризації.

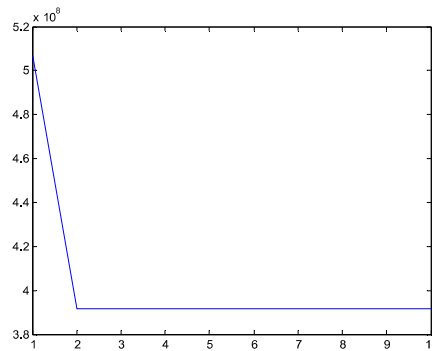


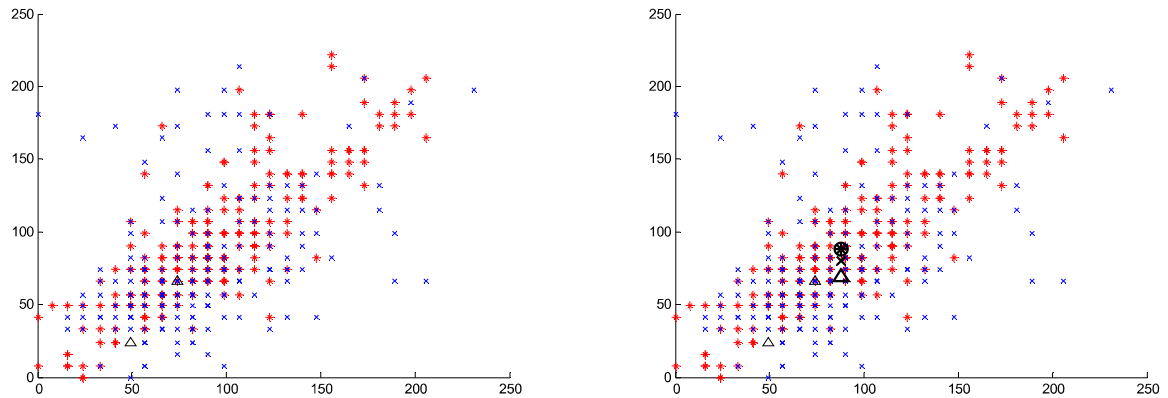
Рисунок 2 – Графік зміни значень цільової функції

Далі визначаємо максимальну степінь приналежності окремого елемента даних кластеру.

// визначення максимальної степені приналежності окремого елемента даних кластеру

```
>> maxU=max(U);
// розподіл рядків матриці даних між відповідними кластерами
>> index1=find(U(1,:)==maxU);
>> index2=find(U(2,:)==maxU);
>> index3=find(U(3,:)==maxU);
>> index4=find(U(4,:)==maxU);
// побудова даних, що відповідають першому кластеру
>>line(lom(index1,1),lom(index1,2),'linestyle','none','marker','*','color','red');
// побудова даних, що відповідають другому кластеру
>>line(lom(index2,1),lom(index2,2),'linestyle','none','marker','o','color','green');
// побудова даних, що відповідають третьому кластеру
>>line(lom(index3,1),lom(index3,3),'linestyle','none','marker','x','color','blue');
// побудова даних, що відповідають четвертому кластеру
>>line(lom(index4,1),lom(index4,4),'linestyle','none','marker','^','color','black');
>> hold on
// побудова кластерних центрів
>> plot(center(1,1),center(1,2),'k*','markersize',10,'LineWidth',2);
// побудова кластерних центрів
>> plot(center(2,1),center(2,2),'ko','markersize',10,'LineWidth',2);
// побудова кластерних центрів
>> plot(center(3,1),center(3,3),'kx','markersize',10,'LineWidth',2);
// побудова кластерних центрів
>> plot(center(4,1),center(4,4),'k^','markersize',10,'LineWidth',2);
```

Результат розв'язання задачі нечіткої кластеризації для 4-х нечітких кластерів із використанням зазначеної послідовності команд може бути візуалізовано на рисунку 3, де представлено множини даних, що підлягають кластеризації з знайденими центрами кластерів.



а)

б)

а) множина аналізованих даних

б) множина аналізованих даних та центри кластерів

Рисунок 3 – Результати роботи алгоритму нечіткої кластеризації fcm

На рисунку 4 відображені результати роботи нечіткої кластеризації з розбивкою на нечіткі кластера (функція приналежності – розбивка по яскравості, вид - трапецевидний). Візуалізація спрощує задачу експерта з аналізу зображень та зменшує ймовірність помилки.

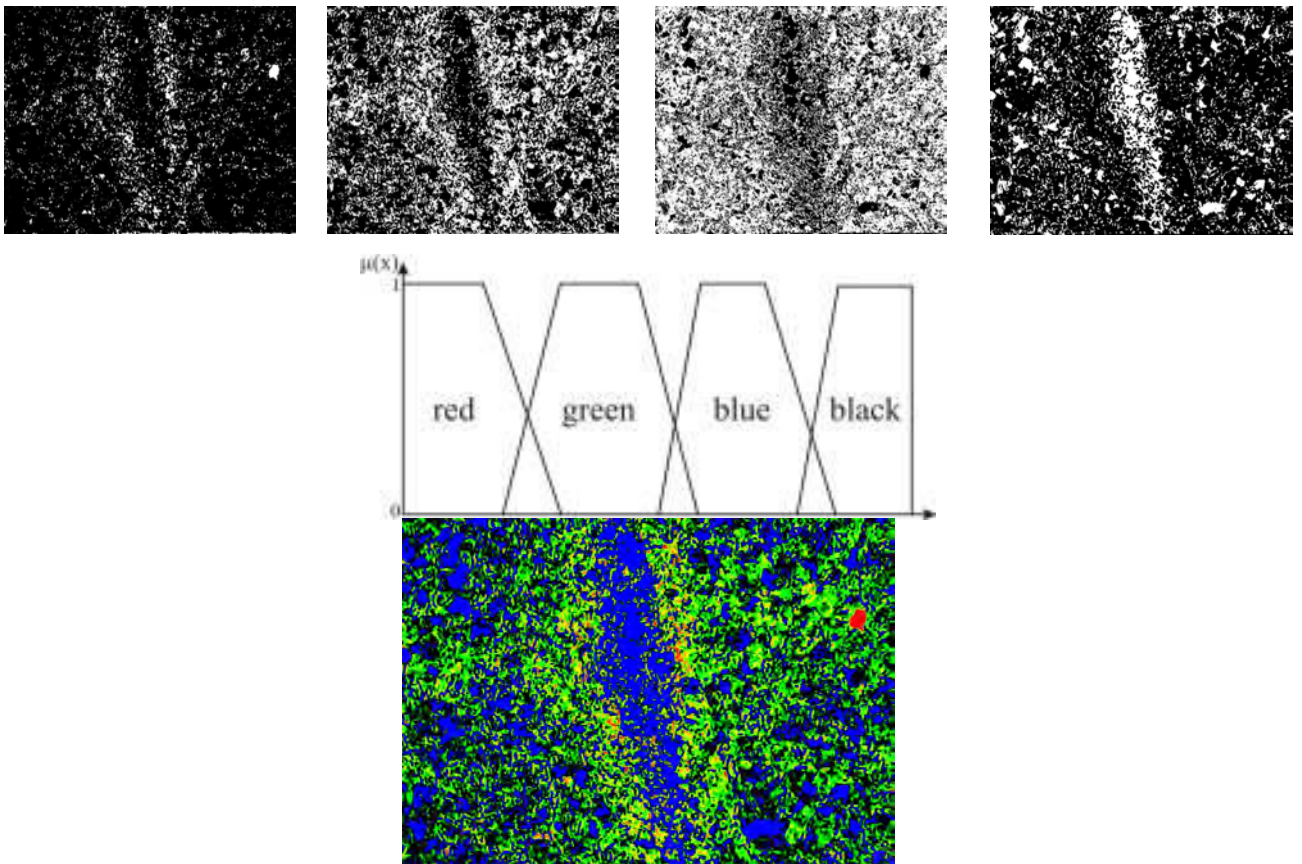


Рисунок 4 - Результати роботи нечіткої кластеризації

Проведемо дослідження алгоритму fcm для аналізу чорно – білого зображення зразка №12 (пл. №22325), смуга мартенсіта.

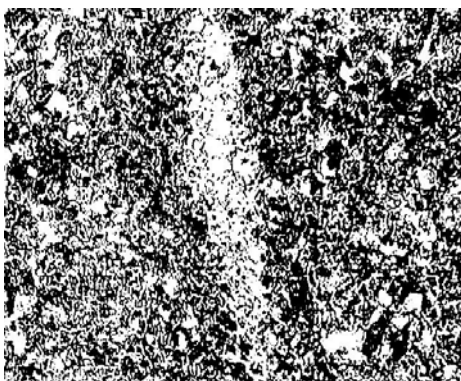


Рисунок 5 – Чорно-біле зображення зразка №12 (пл. №22325), смуга мартенсіта

Після отримання чорно-білого зображення ми створюємо цикл за допомогою якого отримуємо зображення з поданням про значення кожного пікселя. Отриманий в результаті файл fdv.dat будемо використовувати для подальшого дослідження, тому що дані для кластеризації повинні бути представлені в вигляді матриці чисел.

```
>> i4=I;
>> fid=fopen('fdv.dat', 'w');
>> for i=1:480; fprintf(fid, '%d ',i4(i,:)); fprintf(fid, '\n'); end;
>> fclose(fid);
```

Наступним кроком дослідження є визначення центрів кластеризації, для цього в Matlab застосовується функція fcm, яка виконується ітераційно до тих пір, поки зміни цільової функції перевищують деякий заданий поріг. На кожному кроці в командному вікні Matlab виводяться порядковий номер ітерації і відповідне поточне значення цільової функції. Якщо після запису функції fcm у другому рядку не ставити крапку з комою (;), то у вікні команд будуть показані значення координат центрів нечітких кластерів, значення функцій приналежності об'єктів нечітким кластерам і значення цільової функції на кожній з ітерацій роботи алгоритму FCM. У цьому прикладі використовується перший формат запису функції fcm.

```
// завантаження даних, що підлягають кластеризації з файлу fdv.dat
>> load fdv.dat
// визначення центру кластеризації (2 кластери)
>> [center,U,obj_fcn]=fcm(fdv,2);
Iteration count = 1, obj. fcn = 44787.959585
```

```

Iteration count = 2, obj. fcn = 37126.147586
Iteration count = 3, obj. fcn = 37123.721925
Iteration count = 4, obj. fcn = 37123.704452
Iteration count = 5, obj. fcn = 37123.704172
Iteration count = 6, obj. fcn = 37123.704167
>> plot(obj_fcn);

```

Для оцінки динаміки зміни значень цільової функції використовується команда побудови графіка `plot(obj_fcn)`. Результати показані на рисунку 6, з рисунку видно, що чим вище значення цільової функції `fcn` тим краще кількість ітерацій нечіткої кластеризації.

Далі визначаємо максимальну степінь приналежності окремого елемента даних кластеру.

```

// визначення максимальної степені приналежності окремого елемента даних
кластеру
>> maxU=max(U);
// розподіл рядків матриці даних між відповідними кластерами
>> index1=find(U(1,:)==maxU);
>> index2=find(U(2,:)==maxU);
// побудова даних, що відповідають першому кластеру
>>line(fdv(index1,1),fdv(index1,2),'linestyle','none','marker','*','color','black');
// побудова даних, що відповідають другому кластеру
>>line(fdv(index2,1),fdv(index2,2),'linestyle','none','marker','o','color','red');
>> hold on
// побудова кластерних центрів
>> plot(center(1,1),center(1,2),'k*','markersize',10,'LineWidth',2);
// побудова кластерних центрів
>> plot(center(2,1),center(2,2),'ko','markersize',10,'LineWidth',2);

```

Результат розв'язання задачі нечіткої кластеризації для 2-х нечітких кластерів із використанням зазначеної послідовності команд може бути візуалізовано на рисунку 7, де представлено множини даних, що підлягають кластеризації з знайденими центрами кластерів.

В процесі виконання дослідження було виявлено, що для аналізу кольорових зображень неможливо застосувати алгоритм субтрактивної кластеризації, тому що вхідна матриця даних повинна мати розмір $n \times 2$, далі дослідження проводились вже для чорно-білого зображення.

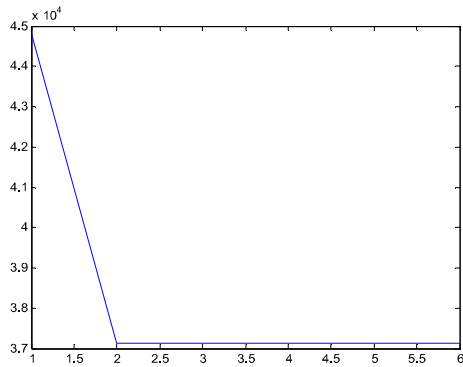


Рисунок 6 – Графік зміни значень цільової функції

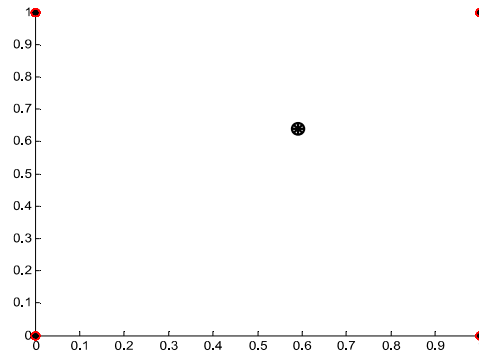


Рисунок 7 – Результат розв'язання задачі нечіткої кластеризації для 2-х нечітких кластерів

Для того, щоб зображення представити в потрібному форматі для подальшого дослідження, використовуючи алгоритм субтрактивної кластеризації було написано М-файл `convert.m`, який присвоює чорному кольору – нуль, а білому – одиницю:

```
function im1 = convert(img1)
img1 = 1 - img1;
row = numel(find(img1));
[x y] = size(img1);
im1 = zeros( row, 2);
i_row = 1;
for i = 1:x
    for j = 1:y
        if (img1(i,j) == 1)
            im1(i_row, 1) = i;
            im1(i_row, 2) = j;
            i_row = i_row + 1;
        end
    end
end
end
end
>> im1=convert(F);
```

У разі відсутності будь-яких апріорних припущень щодо кількості нечітких кластерів у системі MATLAB можна використовувати функцію командного рядка `subclust`: В результаті виконання цього фрагмента команд будуть одержані значення матриці центрів кластерів і вектора σ - значень.

```
>> [c,s]=subclust(im1,[0.5 0.5],[],[1.25 0.5 0.15 1])
Normalizing data...
```


Computing potential for each data point...

Found cluster 1, potential = 1

Found cluster 2, potential = 0.784906

Found cluster 3, potential = 0.616624

Found cluster 4, potential = 0.577341

Found cluster 5, potential = 0.289288

c =

186 453

343 193

115 138

398 503

31 547

s = 84.6760 112.9603

Як можна помітити, для наведених значень аргументів розглянута функція `subcluster` знаходить п'ять нечітких кластерів та відображає координати їхніх центрів у командному вікні системи MATLAB.

Алгоритми нечіткої кластеризації дозволяють розбити досліджувані об'єкти на нечіткі кластери, а нечіткі кластери в свою чергу можуть бути візуалізовані шляхом управління рівнем яскравості відповідних кольорів, аналогічним функції приналежності. Візуалізація спрощує завдання експерта з аналізу зображень і зменшує ймовірність помилки.

ЛІТЕРАТУРА

1. Классификация и кластер/Под ред. Дж.Вэн Райзина. – М.: Мир, 1980. – 392 с.
2. Мандель Н.Д. Кластерный анализ. – М.: Финансы и статистика, 1988. – 176с.
3. Bezdek J.C. Some recent applications of fuzzy c-means in pattern recognition and image processing. – IEEE Workshop Lang. Autom, 1983, pp. 247 – 252.
4. Windham M.P. Cluster validity for the fuzzy c-means clustering algorithm. – IEEE Transactions on Pattern Analysis and Mashine Intelligence, Vol. 4, no. 4, 1982, pp. 357 – 363.
5. Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH. – СПб.: БХВ – Петербург, 2003. – 736с.: ил.