**Vietnamese**

1. ****∈[0, 1] còn được sử dụng để thể hiện mối liên hệ giữa một data point **** và đại diện cụm **C***j*. Trong GFKM, **** phải đáp ứng ràng buộc sau:

= 1, cho *i* = 1 tới *N*.

1. *NNTi* bao gồm *M* trọng tâm cụm gần nhất cho data point****.
2.  là khoảng cách Ơ-le bình phương giữa các data point **** và đại diện cụm gần nhất thứ *j* của nó.

= 

1.  = , cho *r* = 1 tới *N* và *s* = 1 tới *M* (11)

Nếu **C***j*∈*NNTi* là hàng xóm gần nhất thứ *l* của **X***i*, gán =**;** ngược lại gán = 0, với  là các thành viên giữa các điểm **X***i* và các trung tâm cụm **C***j*.

1. Cho S *j* = {**X** *i*: **X** *i*∈ *NNTj*, *i* = 1 tới *N*}, chúng ta có:

**C***j* =  (16)

1. Thuật toán phân cụm GFKM
2. Đưa vào một tập hợp khởi tạo các trung tâm cụm *SC*0 = {**C***j*(0)} và các giá trị ε và *M*. Gán *p* = 0.
3. Nhận một tập hợp các trung tâm cụm *SCp*, tính  cho *i* = 1 tới *N* và *j* = 1 tới *M*. Cập nhật thành viên **** sử dụng biểu thức (11). Nếu **C***j*∈*NNTi* hàng xóm gần nhất thứ *l* của **X***i*, gán = ****; ngược lại cho  = 0.
4. Tính trung tâm của mỗi cụm sử dụng biểu thức (16) để xác định một tập hợp mới của các đại diện cụm *SCp+*1 = {**C***j*(*p*+1)}.
5. Cập nhật *NNTi* cho *i* = 1 tới *N*.
6. Nếu < ε cho *j* = 1 tới *k*, sau đó dừng, với ε > 0 là một số nguyên dương rất nhỏ. Ngược lại gán *p = p + 1* và quay lại bước (2).
7. Điều kiện tới hạn cụm này là hàm Xie-Beni V*XB* [12], nó được định nghĩa như sau:

V*XB* =  (17)

với *N* là số lượng các data point; *k* là số lượng các cụm; *SCD*minlà khoảng cách bình phương Ơ-clit nhỏ nhất giữa các trọng tâm cụm; và **X***i* là data point thứ *i*; và **C***j* là trọng tâm cụm thứ *j.* Chú ý ở đây một kết quả phân cụm tốt hơn với giá trị V*XB* nhỏ hơn.

**English**

1. ****∈[0, 1] is also used to represent the relationship between a data point **** and cluster representative **C***j*. For GFKM, **** should satisfy the following constraint:

= 1, for *i* = 1 to *N*.

1. *NNTi* consist of *M* nearest cluster centers for the data point****.
2.  is the squared Euclidean distance between data points **** and its *j*th nearest cluster representative
3. = , for *r* = 1 to *N* and *s* = 1 to *M* (11)

If **C***j*∈*NNTi* is the *l*th nearest neighbor of **X***i*, set =**;** otherwise let = 0, where are the memberships between points **X***i* and cluster centers **C***j*

1. Let S *j* = {**X** *i*: **X** *i*∈ *NNTj*, *i* = 1 to *N*}, we have:

**C***j* =  (16)

1. GFKM Clustering Algorithm
2. Input an initial set of cluster centers *SC*0 = {**C***j*(0)} and the values of ε and *M*. Set *p* = 0.
3. Given the set of cluster centers *SCp*, compute  for *i* = 1 to *N* and *j* = 1 to *M*. Update membership **** using equation (11). If **C***j*∈*NNTi* is the *l*th nearest neighbor of **X***i*, set  = ****; otherwise let  = 0.
4. Compute the center for each cluster using equation (16) to obtain a new set of cluster representatives *SCp+*1 = {**C***j*(*p*+1)}.
5. Update *NNTi* for *i* = 1 to *N*.
6. If < ε for *j* = 1 to *k*, then stop, where ε > 0 is a very small positive number. Otherwise set *p = p + 1* and go to step (2).
7. This cluster validity criterion is the Xie-Beni function V*XB* [12], which is defined as follows:

V*XB* =  (17)

where *N* is the number of data points; *k* is the number of clusters; *SCD*minis the minimum squared Euclidean distance between cluster centers; and **X***i* is the *i*th data point; and **C***j* is the *j*th cluster center*.* Note here that a better clustering result is with the smaller V*XB* value.