TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐÀO TẠO CHẤT LƯỢNG CAO

NGÀNH CÔNG NGHỆ THÔNG TIN

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



BÁO CÁO ĐỒ ÁN 1

Viết chương trình phân nhóm người theo thu nhập và chi tiêu hàng tháng của họ. Xuất ra khoảng trị thu nhập và chi tiêu hàng tháng của mỗi nhóm

GVHD : ThS. Từ Tuyết Hồng

SVTH : Trần Ngọc Khoa 15110064

Đào Văn Toản 15110142

Tp. Hồ Chí Minh, tháng 07 năm 2017

###### Mục Lục

[1 . Mô tả đồ án 4](#_Toc498496920)

[1.1 Phần mềm dùng để : 4](#_Toc498496921)

[1.2 Use case diagram 4](#_Toc498496922)

[1.3 Ý nghĩa , mục đích sử dụng 5](#_Toc498496923)

[2 . Mô tả quá trình làm 6](#_Toc498496924)

[2.1 Cách thức xây dựng phần mềm 6](#_Toc498496925)

[2.2 Thiết kế giao diện 7](#_Toc498496926)

[2.3 Thiết kế lớp 8](#_Toc498496927)

[2.3.1 Class diagram 8](#_Toc498496928)

[2.3.2 Sequence diagram 8](#_Toc498496929)

[2.4 Thiết kế cơ sở dữ liệu 9](#_Toc498496930)

[2.5 Cài đặt và kiểm thử 10](#_Toc498496931)

[2.5.1 Các hàm sử dụng: 10](#_Toc498496932)

[2.5.2 Chạy giải thuật 11](#_Toc498496933)

[2.5.3 Nhận xét kết quả 15](#_Toc498496934)

[3 . Mô tả phân công công việc 15](#_Toc498496935)

[4 . Kết luận 16](#_Toc498496936)

[4.1 Mức độ hoàn thành 16](#_Toc498496937)

[4.2 Khó khăn gặp phải 16](#_Toc498496938)

[4.3 Nhận xét 16](#_Toc498496939)

[4.3.1 Khuyết điểm : 16](#_Toc498496940)

###### Danh mục các hình

[1.1 Use case diagram 6](#_Toc498590266)

[2.2‑1 Giao diện chính 7](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590248)

[2.2‑2 Biểu Đồ 7](#_Toc498590249)

[2.3‑1 Class diagram 8](#_Toc498590250)

[2.5‑1 k=1 chạy lần 1 11](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590251)

[2.5‑2 k=1 chạy lần 2 11](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590252)

[2.5‑3 k=2 chạy lần 1 12](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590253)

[2.5‑4 k=2 chạy lần 3 12](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590254)

[2.5‑5 k=3 chạy lần 4 12](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590255)

[2.5‑6 k=2 chạy lần 5 13](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590256)

[2.5‑7 k=3 chạy lần 1 13](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590257)

[2.5‑8 k=3 chạy lần 2 13](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590258)

[2.5‑9 k=4 chạy lần 1 14](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590259)

[2.5‑10 k=4 chạy lần 2 14](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590260)

[2.5‑11 k=5 chạy lần 1 14](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590261)

[2.5‑12 k=5 chạy lần 2 15](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590262)

[2.5‑13 k=6 chạy lần 1 15](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590263)

[2.5‑14 k=6 chạy lần 2 16](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590264)

[2.5‑15 k=6 chạy lần 3 16](file:///C:\Users\Toandao\Downloads\Báo-cáo-đồ-án-1.docx#_Toc498590265)

###### Danh mục các bảng

[Bảng 1 Thiết kế cơ sở dữ liệu 9](#_Toc498584608)

[Bảng 2 Mô tả phân công công việc 17](#_Toc498584609)

NỘI DUNG BÁO CÁO ĐỒ ÁN 1

# 1. Mô tả đồ án

Bài toán phân loại là một bài toán phổ biến trong công việc thu thập và xử lý dữ liệu. Nếu việc phân loại dữ liệu tốt thì sẽ giúp công việc xử lý dữ liệu nhanh và hiệu quả hơn. Nhờ việc phân loại dữ liệu cũng giúp chúng ta trích xuất thông tin từ tập dữ liệu thô thu được qua quá trình thu thập.

Đề tài của nhóm được đưa ra dựa trên tình huống này - “Viết chương trình phân nhóm người theo thu nhập và chi tiêu hàng tháng của họ. Xuất ra khoảng trị thu nhập và chi tiêu hàng tháng của mỗi nhóm”. Với dữ liệu về chi tiêu và thu nhập thô cần được xử lý ban đầu, nhóm cần phải tìm cách phân loại những dữ liệu này theo từng nhóm.

Dữ liệu đầu vào là dữ liệu thô và người xử lý không có một tiêu chí nào cụ thể để phân loại dữ liệu. Vì vậy, phương pháp tiếp cận khả thi và đơn giản nhất là sử dụng giải thuật kmeans clustering.

## Phần mềm dùng để :

Phân loại nhóm người dựa theo số liệu chi tiêu và thu nhập của họ.

Mục tiêu: Trích xuất được các thông tin hữu ích từ dữ liệu ban đầu.

## 

# . Mô tả quá trình thực hiện

## Cách thức xây dựng phần mềm

Các bước xây dựng:

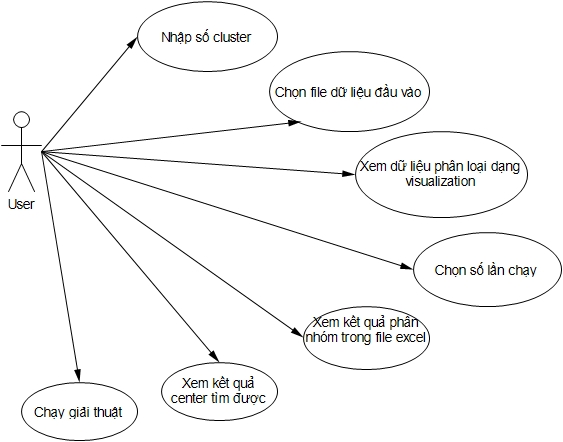
B1: Tìm hiểu và hiện thực giải thuật kmean

B2: Xây dựng giao diện phần mềm

B3: Kết hợp giao diện và giải thuật thông qua event handler.

B4: Kiểm tra và sửa lỗi

## Use case diagram



1.1 Use case diagram

Bảng mô tả các use case

|  |  |  |
| --- | --- | --- |
| TT | Tên use case | Ý nghĩa |
| 01 | Nhập số cluster | Cho phép user chọn số cluster từ 1 đến 9 |
| 02 | Chọn file dữ liệu đầu vào | Cho phép user chọn file dữ liệu txt đầu vào |
| 03 | Xem dữ liệu phân loại dạng visualization | Cho phép user xem dữ liệu trên hệ trục toạ độ. Áp dụng cho bảng dữ liệu có 3 cột. |
| 04 | Chọn số lần chạy | Cho phép user chọn số lần chạy giải thuật để tìm cách phân nhóm tối ưu. 1 lần hoặc 100 lần. |
| 05 | Xem kết quả phân nhóm trên file excel | Cho phép user xem kết quả phân nhóm trong file excel đã được tạo sau khi phân nhóm. |
| 06 | Xem kết quả center tìm được | Cho phép user xem kết quả center tìm được. Tương ứng với khoảng trị thu nhập của từng nhóm |
| 07 | Chạy giải thuật | Cho phép user bắt đầu chạy giải thuật phân nhóm. |

Bảng mô tả các actor

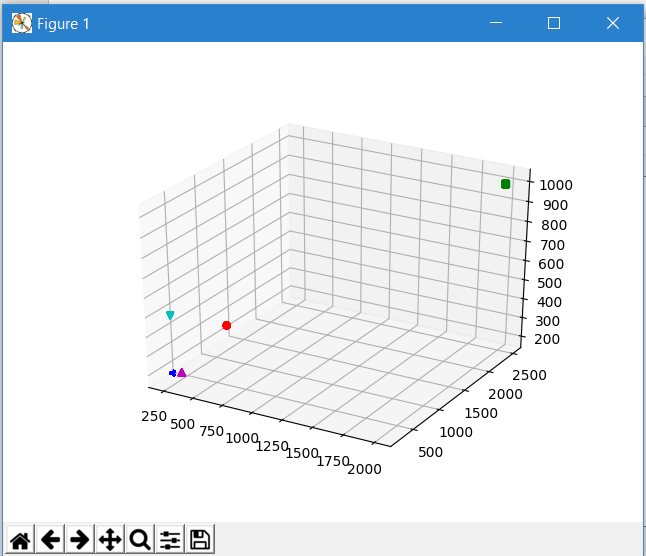
|  |  |  |
| --- | --- | --- |
| TT | Tên Actor | Ý nghĩa |
| 01 | User | Người tương tác với chương trình. |

## Ý nghĩa, mục đích sử dụng

Mục đích của phân cụm là tìm ra bản chất bên trong các nhóm của dữ liệu.

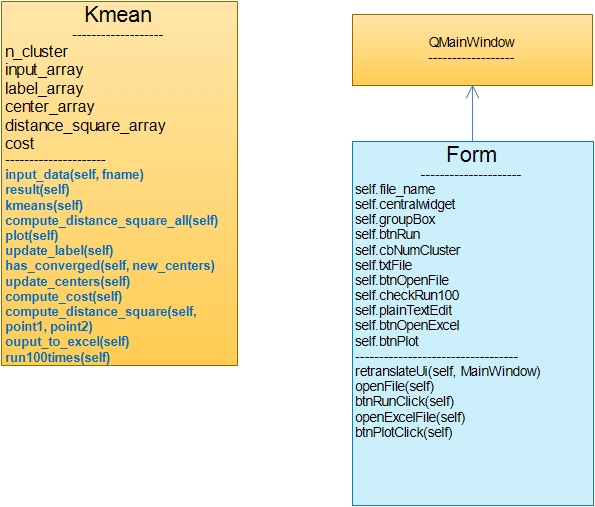
Ý nghĩa: Xác định các nhóm khách hàng (khách hàng tiềm năng, khách hàng giá trị, phân loại và dự đoán hành vi khách hàng,…) sử dụng sản phẩm hay dịch vụ của công ty để giúp công ty có tầm nhìn chiến lược kinh doanh hiệu quả hơn.

## Thiết kế giao diện



## Thiết kế lớp

### **Class diagram**



2.3‑1 Class diagram

Bảng danh mục các lớp được sử dụng trong chương trình

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên lớp | Tên các SV phụ trách viết  (Nếu là code lấy từ ngồn khác thì phải chỉ rõ lấy từ ngồn nào). | Mục đích chính của lớp trong chương trình |
| 01 | Kmean | Trần Ngọc Khoa | Chứa các dữ liệu và phương thức cần thiết cho giải thuật Kmean |
| 02 | Form | Trần Ngọc Khoa, Đào Văn Toản | Chứa các thành phần của cửa sổ giao diện. Các thành phần để user tương tác với chương trình. Gọi các hàm của lớp Kmean. |

Bảng mô tả các phương thức trong lớp Kmean

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TT | Tên phương thức | Tên các SV phụ trách viết | Mục đích chính của phương thức trong chương trình | Input | Output | Mã giả (lưu ý viết dạng mã giả thuật toán dùng trong phương thức) |
| 1 | input\_data | Trần Ngọc Khoa | Hàm đọc dữ liệu từ file | fname | list | input\_data(self, fname):  X = []  with open(fname) as fileobject:  for line in fileobject:  Xi = [float(n) for n in line.strip()]  X.append(Xi)  return X |
| 2 | result | Trần Ngọc Khoa | Hàm trả về kết quả |  | string | result(self):  str\_ = ''  str\_ += "Center was found by algorithm:\n\n"  for center in self.center\_array:  str\_ += str(center) + " "  str\_ += "\n"  str\_ += "\n"  str\_ += "Cost for this run:\n"  str\_ += str(sum(self.cost))  return str\_ |
| 3 | compute\_distance\_square\_all | Trần Ngọc Khoa | Hàm tính khoảng cách giữa các điểm trong input\_array với center\_array |  |  | compute\_distance\_square\_all(self):  i = 0  for point in self.input\_array:  j = 0  for center in self.center\_array:  self.distance\_square\_array[i][j] = self.compute\_distance\_square(point, center)  j += 1  i += 1 |
| 3 | compute\_distance\_square | Trần Ngọc Khoa | Hàm tính khoảng cách giữa 2 điểm | Point1, point2 | float | compute\_distance\_square(self, point1, point2):  distance\_square = 0  for i in range(len(point1)):  distance\_square += (point1[i] - point2[i]) \*\* 2  return distance\_square |
| 4 | Kmean | Trần Ngọc Khoa | Hàm kmean |  |  | def kmeans(self):  while True:  self.compute\_distance\_square\_all()  self.update\_label()  new\_centers = self.update\_centers()  if self.has\_converged(new\_centers):  break  self.center\_array = new\_centers  self.cost = self.compute\_cost() |

## Thiết kế cơ sở dữ liệu

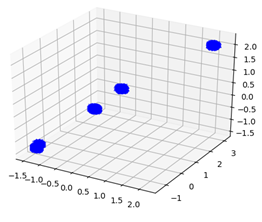
|  |  |  |
| --- | --- | --- |
| **Tiền lương** | **Tổng thu nhập** | **Chi tiêu** |
| 200🡪 2100 | 200 🡪 2500 | 200 🡪 2100 |

Bảng 1 Thiết kế cơ sở dữ liệu

Dữ liệu lúc đầu được thiết kế trên excel sau đó chuyển xang file .txt để dẽ dàng sử dụng.

## Cài đặt và kiểm thử

### **Chạy giải thuật**

Cho k chạy các giá trị từ 1 đến 6:

k=1 :

*Lần 1*

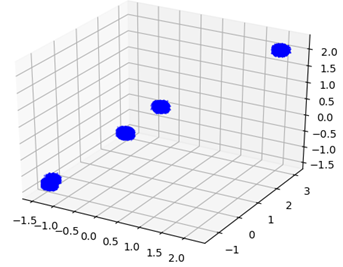
*Center was found by algorithm:*

*747.351466026 1147.36926357 834.427384634*

*Cost for this run:*

*779821.174126*

***2.5‑1 k=1 chạy lần 1***

*Lần 2*

*Center was found by algorithm:*

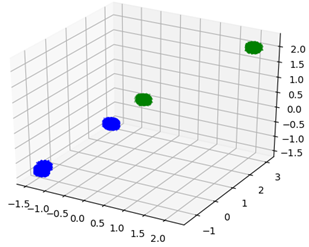
*747.351731683 1147.36952066 834.42746176*

*Cost for this run:*

*779822.272729*

K=2 :

2.5‑2 k=1 chạy lần 2

*Lần 1*

*Center was found by algorithm:*

*554.919413325 922.121636886 626.979771709*

*1335.42970508 1835.73290046 1468.39363502*

*Cost for this run:*

*450548.206299*

*Lần 2*

2.5‑3 k=2 chạy lần 1

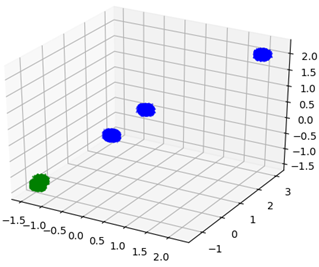
*Center was found by algorithm:*

*900.936127717 1401.12865107 1011.41619082*

*204.174548448 249.908277614 208.477936726*

*Cost for this run:*

*230278.916305*

*Lần 3*

*Center was found by algorithm:*

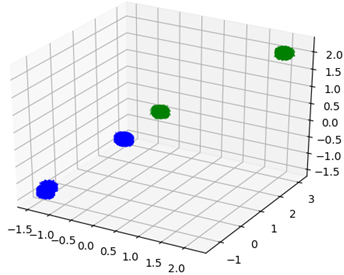
*900.936468488 1401.12898085 1011.41628976*

*204.174548448 249.908277614 208.477936726*

*Cost for this run:*

*230279.46739*

***2.5‑4 k=2 chạy lần 3***

*Lần 4*

*Center was found by algorithm:*

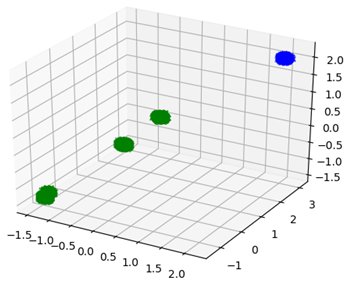
*554.919413325 922.121636886 626.979771709*

*1335.43078259 1835.73394321 1468.39394785*

*Cost for this run:*

*450548.884506*

***2.5‑5 k=3 chạy lần 4***

*Lần 5*

***2.5‑6 k=2 chạy lần 5***

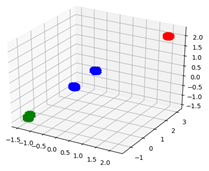
*Center was found by algorithm:*

*2000.09867428 2500.20584643 2000.02763175*

*634.414678974 1025.40925699 729.346823877*

*Cost for this run:*

*191742.350859*

K=3 :

***2.5‑7 k=3 chạy lần 1***

*Lần 1*

*Center was found by algorithm:*

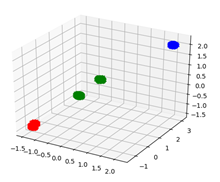
*770.502192961 1270.70483239 894.100752987*

*204.174548448 249.908277614 208.477936726*

*2000.09867428 2500.20584643 2000.02763175*

*Cost for this run:*

*21813.3876415*

*Lần 2*

2.5‑8 k=3 chạy lần 2

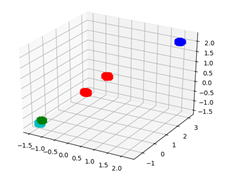
*2000.09867428 2500.20584643 2000.02763175*

*770.502192961 1270.70483239 894.100752987*

*204.174548448 249.908277614 208.477936726*

*Cost for this run:*

*21813.3876415*

K=4 :

*Lần 1*

Center was found by algorithm:

*2000.09867428 2500.20584643 2000.02763175*

*250.04668149 249.701636207 299.821210431*

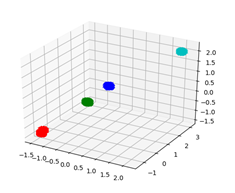
*770.502192961 1270.70483239 894.100752987*

*199.800843816 249.927979957 199.768760308*

2.5‑9 k=4 chạy lần 1

*Cost for this run:*

*16388.4085831*

*Lần 2*

2.5‑10 k=4 chạy lần 2

*Center was found by algorithm:*

*999.968169896 1500.37024729 1200.07464609*

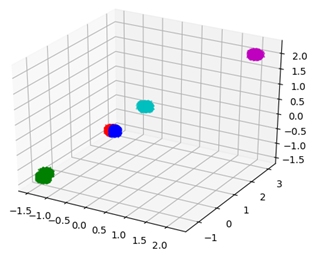
*699.965384617 1200.10671771 800.045755622*

*204.174548448 249.908277614 208.477936726*

*2000.09867428 2500.20584643 2000.02763175*

*Cost for this run:*

*1338.34686145*

K=5 :

2.5‑11 k=5 chạy lần 1

*Lần 1*

*Center was found by algorithm:*

*683.261760562 1200.4052491 799.688949025*

*204.174548448 249.908277614 208.477936726*

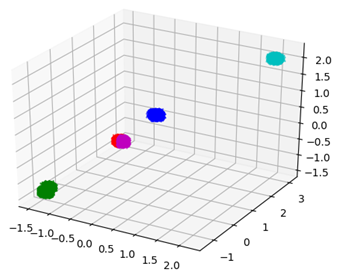
*999.968169896 1500.37024729 1200.07464609*

*2000.09867428 2500.20584643 2000.02763175*

*716.627167809 1199.8089341 800.401668456*

*Cost for this run:*

*1177.47313654*

*Lần 2*

2.5‑12 k=5 chạy lần 2

*Center was found by algorithm:*

*999.968169896 1500.37024729 1200.07464609*

*204.174548448 249.908277614 208.477936726*

*683.261216375 1200.40459646 799.688714859*

*2000.09867428 2500.20584643 2000.02763175*

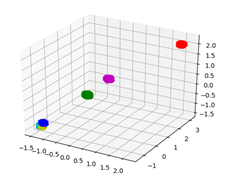
*716.626639133 1199.80960422 800.401879132*

*Cost for this run:*

*1177.47318608*

K=6 :

*Lần 1*

*Center was found by algorithm:*

2.5‑13 k=6 chạy lần 1

*250.091151646 249.703133668 300.014407479*

*699.965384617 1200.10671771 800.045755622*

*2000.09867428 2500.20584643 2000.02763175*

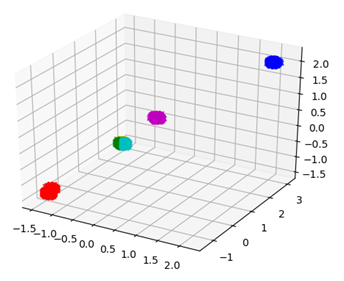
*182.954848699 250.081388973 199.729396675*

*999.968169896 1500.37024729 1200.07464609*

*216.531111952 249.775421801 199.839111647*

*Cost for this run:*

*997.33280591*

*Lần 2*

2.5‑14 k=6 chạy lần 2

*Center was found by algorithm:*

*2000.09867428 2500.20584643 2000.02763175*

*684.790064578 1186.17834296 799.780319549*

*204.174548448 249.908277614 208.477936726*

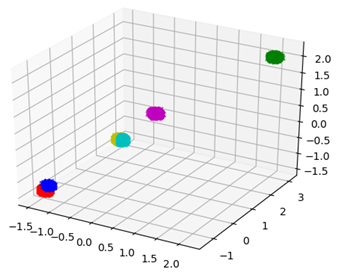
*720.849794072 1196.66084605 800.518242362*

*999.968169896 1500.37024729 1200.07464609*

*692.480254957 1218.70830707 799.794095423*

*Cost for this run:*

*1042.73565118*

*Lần 3*

2.5‑15 k=6 chạy lần 3

*Center was found by algorithm:*

*250.04668149 249.701636207 299.821210431*

*2000.09867428 2500.20584643 2000.02763175*

*199.800843816 249.927979957 199.768760308*

*716.627167809 1199.8089341 800.401668456*

*999.968169896 1500.37024729 1200.07464609*

*683.261760562 1200.4052491 799.688949025*

*Cost for this run:*

*1000.13951513*

### **Nhận xét kết quả**

Việc phân loại phụ thuộc khá lớn vào các center được chọn ban đầu. Vì các center được chọn ngẫu nhiên cho nên nếu các center được chọn gần nhau thì các nhóm được phân ra sẽ có thể bị chồng lấn lên nhau.

Hướng tiếp cận: Thay vì chọn center ban đầu một cách ngẫu nhiên, ta chọn các center sao cho chúng cách xa nhau. Như vậy, xác suất chồng lấn của các nhóm sẽ được giảm đi. Việc này giúp cho phân nhóm có kết quả tốt hơn (đối với tập dữ liệu mà nhóm đã chạy thử) và thời gian chạy nhanh hơn do hội tụ nhanh hơn.

# . Mô tả phân công công việc

|  |  |  |
| --- | --- | --- |
| Tên sinh viên | Phần trăm đóng góp | Mảng sinh viên thực hiện |
| Trần Ngọc Khoa | 50% | * Tìm hiểu thuật toán (k-mean) xây dựng code * Viết code, giao diện * Mô phỏng dữ liệu đầu vào * Viết báo cáo |
| Đào Văn Toản | 50% | * Tìm công thức, tìm hiểu thuật toán. * Chạy, thống kê, kiểm tra chương trình * Viết báo cáo, tìm hiểu định dạng word |

Bảng 2 Mô tả phân công công việc

# . Kết luận

## Mức độ hoàn thành

Nhìn chung, nhóm đánh giá mức độ hoàn thành gần như hoàn toàn. Phần mềm có thể xử lý các loại dữ liệu số với số cột tuỳ ý. Với khuyết điểm về việc chọn center ban đầu, nhóm đã khắc phục được bằng cách khởi tạo center theo hướng tiếp cận khác, giúp cho việc phân nhóm có kết quả tốt hơn và thời gian hội tụ nhanh hơn.

## Khó khăn gặp phải

Vì số liệu đầu vào liên quan đến lĩnh vực kinh tế và thông tin cần thu thập là thông tin cá nhân khá nhạy cảm (tiền lương), cho nên nhóm đã gặp phải khó khăn trong việc tìm kiếm dữ liệu thực tế. Những dữ liệu tìm kiếm được từ internet đều là dữ liệu đã qua xử lý, không phù hợp với dữ liệu đầu vào của chương trình. Nhóm đã mạo hiểm khi thử một ngôn ngữ lập trình mới là Python.

## Nhận xét

Trong thực tế rất khó để xác định số cụm cần gom là bao nhiêu thì phù hợp. Để có số cụm phù hợp một chiến thuật hay được sử dụng là gom cụm với nhiều số cụm khác nhau, sau khi gom xong thì xem xét các cụm bằng cách quan sát cho đến khi nào thấy phù hợp thì dừng lại. Tuy nhiên việc xem xét, quan sát đó thường mang tính chủ quan nên cũng khó có kết quả tốt, nhất là trong trường hợp số cụm và số phần tử cần gom cụm là lớn.

### **Khuyết điểm:**

* Cần cho tham số k (số lượng clustes)
* Kết quả phụ thuộc vào việc khởi động ngẫu nhiên K tâm (center) của K clusters: có thể khắc phục bằng cách khởi động lại nhiều lần hoặc chọn center theo tiêu chí thay vì chọn ngẫu nhiên.
* Thời gian thực hiện chương trình phụ thuộc vào độ lớn của cơ sở dữ liệu và số cụm cần phân cụm.

### **Ưu điểm:**

* Chương trình cho kết quả dễ hiểu.
* Chương trình sử dụng thuật toán đơn giản (thuật toán k-mean).
* Thuật toán chạy tốt trên các mẫu dữ liệu mà mỗi cụm kỳ vọng có dạng hình tròn và tách rời nhau.
* Giải thuật được dùng để tạo cơ sở cho việc phân tích dữ liệu ở mức sâu hơn.