ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA HỆ THỐNG THÔNG TIN

BÁO CÁO ĐỒ ÁN CUỐI KỲ

MẠNG XÃ HỘI - IS353.P11.HTCL

PHÂN TÍCH HÀNH VI KHÁCH HÀNG

GVHD: ThS. Nguyễn Thị Kim Phụng

Trương Vĩnh Thuận	21522653
Nguyễn Khánh Văn	21522781

HO CHI MINH CITY, 2024

Mục lục

Danh mục hình ảnh

1. Tổng quan

1.1 Giới thiệu vấn đề

Trong bối cảnh thị trường ngày càng cạnh tranh, việc phân tích xu hướng mua hàng theo khu vực đóng vai trò quan trọng để hiểu hành vi khách hàng và tối ưu hóa chiến lược kinh doanh. Dự án này nhằm xây dựng một mô hình mạng lưới (network) kết nối các khu vực dựa trên việc mua sắm các sản phẩm chung. Bằng cách phân tích mạng lưới này, chúng ta có thể khám phá các cụm khu vực có hành vi mua sắm tương đồng, phát hiện các xu hướng sản phẩm phổ biến và hỗ trợ ra quyết định về phân phối hàng hóa, quảng cáo, hoặc chiến lược tiếp thị.

1.2 Xác định bài toán

Input: Tập dữ liệu về các giao dịch

Output: Các độ đo Centrality, các đặc trưng của các cụm được phân chia

2. Dữ liệu

2.1 Giới thiệu dataset

Link dataset: https://www.kaggle.com/datasets/bhadramohit/customer-shopping-latest-trends-dataset

Giới thiệu: Dữ liệu cung cấp cái nhìn tổng quát về xu hướng mua sắm của người tiêu dùng, nhằm khám phá các mẫu và hành vi trong việc mua sắm bán lẻ. Bao gồm dữ liệu giao dịch chi tiết qua nhiều danh mục sản phẩm khác nhau, thông tin về khách hàng và kênh mua sắm.

Các đặc điểm chính bao gồm:

- Chi tiết giao dịch: Ngày mua, giá trị giao dịch, danh mục sản phẩm và phương thức thanh toán.

- Thông tin khách hàng: Nhóm tuổi, giới tính, địa điểm và tình trạng trung thành.

- Hành vi mua sắm: Tần suất mua hàng, chi tiêu trung bình mỗi giao dịch và xu hướng theo mùa.

2.2 Mô tả dữ liệu

- Tên dữ liệu: Customer Shopping Dataset

- Tác giả: Bhadra Mohit

- Thời gian: Cập nhật cuối từ ngày 23/11/2024

- Bao gồm 19 thuộc tính và 3900 dòng dữ liệu

- Thuộc tính sử dụng:

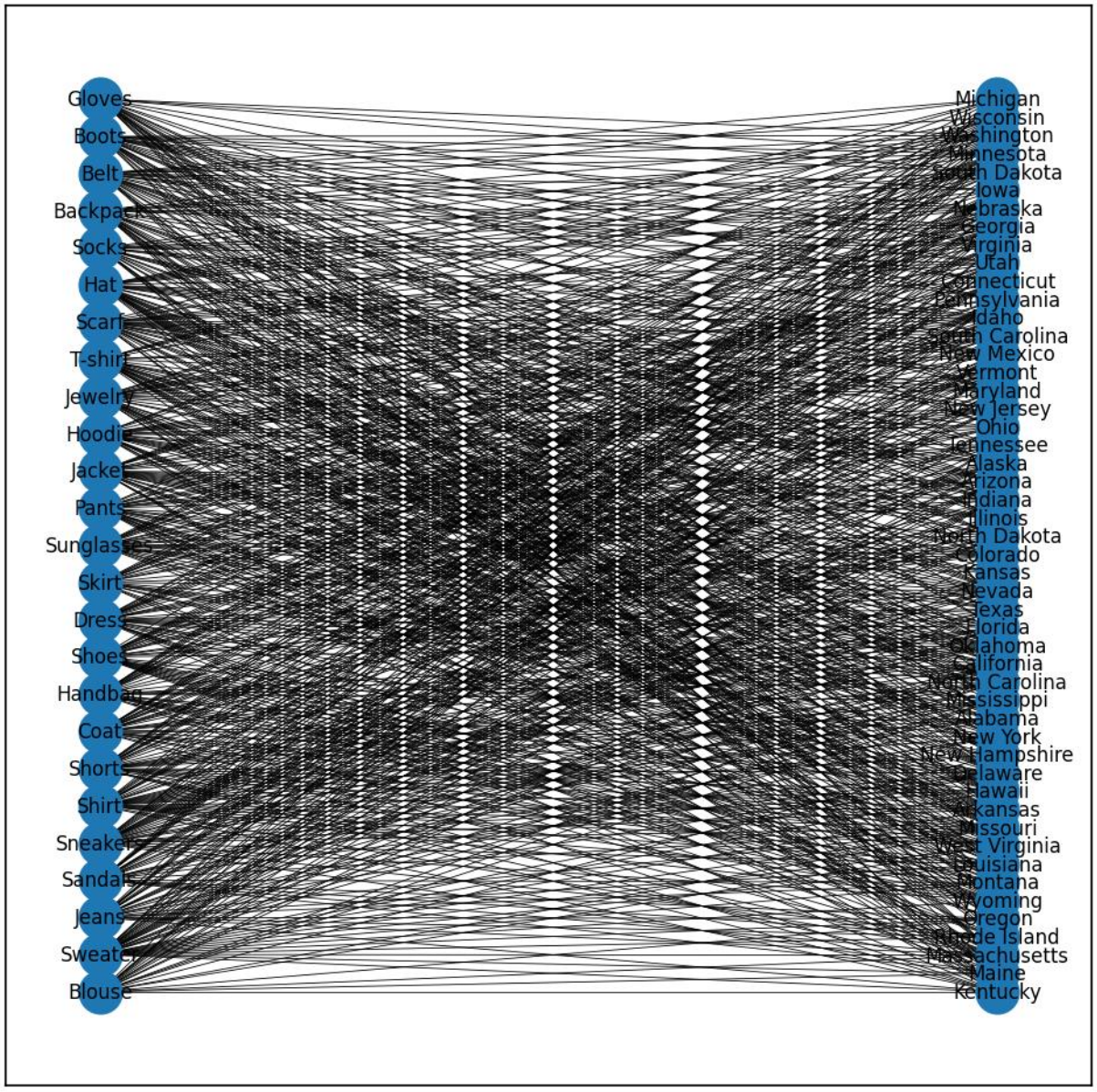
+ Item Purchased: sản phẩm khách hàng mua

+ Location: Nơi khách hàng mua sản phẩm (Thành phố, tiểu bang)

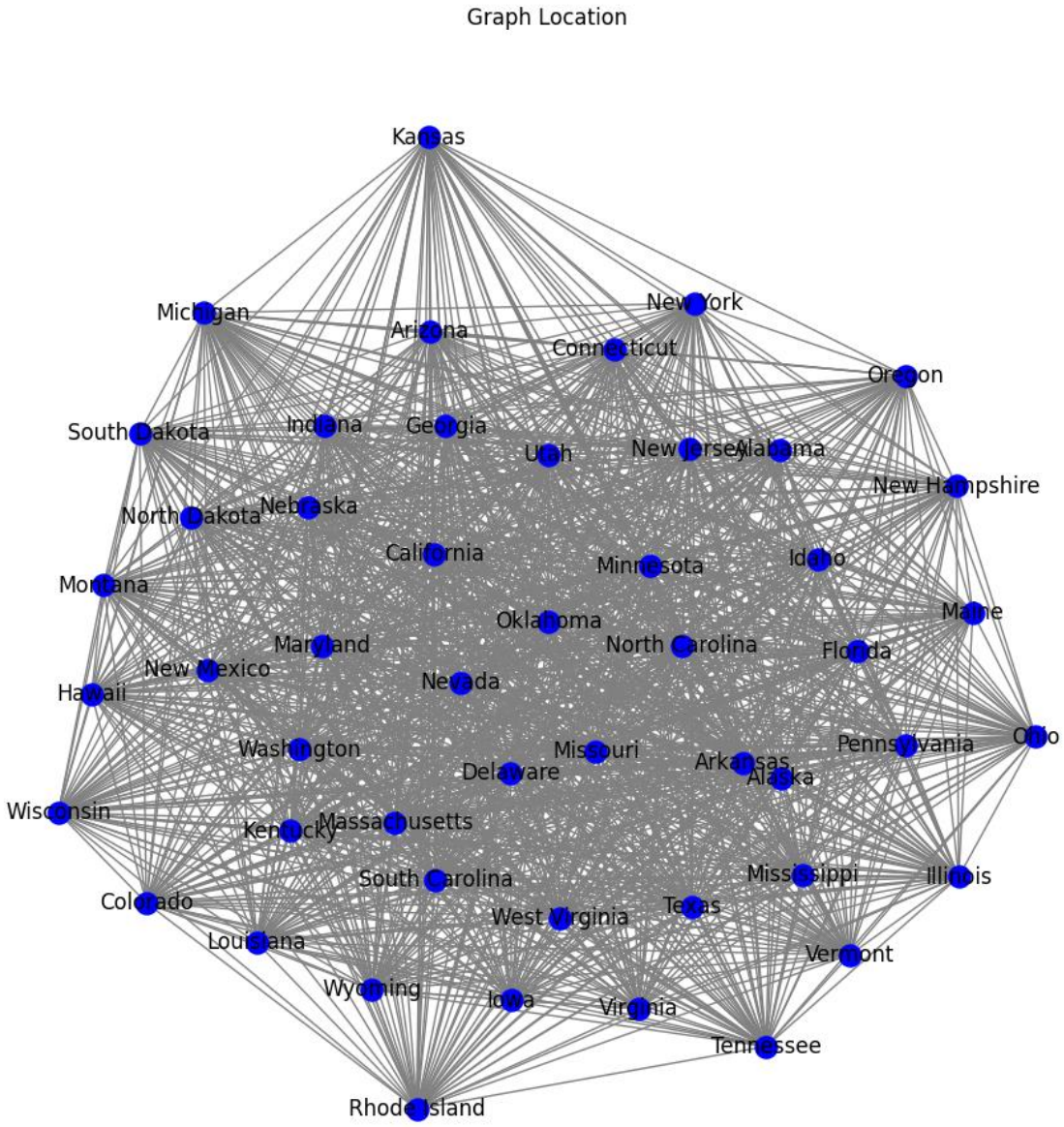
- Dữ liệu cung cấp cái nhìn toàn diện về xu hướng mua sắm của người tiêu dùng tại nước Mỹ.

STT	Thuộc tính	Kiểu dữ liệu	Mô tả
1	Customer ID	int	Mã khách hàng
2	Age	int	Tuổi
3	Gender	string	Giới tính
4	Item Purchased	string	Sản phẩm mua
5	Category	string	Phân loại
6	Purchase Amount (USD)	int	Số tiền phải trả
7	Location	string	Địa điểm mua
8	Size	string	Kích thước
9	Color	string	Màu sắc
10	Season	string	Mùa
11	Review Rating	float	Tỉ lệ đánh giá
12	Subscription Status	string	Trạng thái đăng ký
13	Payment Method	string	Phương thức thanh toán
14	Shipping Type	string	Loại hình vận chuyển
15	Discount Applied	string	Giảm giá đã áp dụng
16	Promo Code Used	string	Mã giảm giá
17	Previous Purchased	int	Số lần mua hàng trước đây
18	Preferred Payment Method	string	Phương thức thanh toán ưa thích
19	Frequency of Purchases	string	Tần xuất mua sắm

2.3 Đồ thị 2 phía



2.4 Đồ thị 1 phía



3. Độ đo Centrality

3.1 Page Rank

3.1.1 Gephi

Id	PageRank
Kentucky	0.02
North Carolina	0.02
Texas	0.02
Rhode Island	0.02
New Jersey	0.02
Utah	0.02
California	0.02
Mississippi	0.02
Alabama	0.02
Washington	0.02

3.1.2 Python

PageRank

```
pagerank = nx.pagerank(G)  # PageRank
print("\nPageRank (Top 10):")
for k, v in sorted(pagerank.items(), key=lambda item: item[1], reverse=True)[:10]:
    print(f"{k}: {v}")
```

```
PageRank (Top 10):
Nevada: 0.02616786283212917
Missouri: 0.026088451869592993
Delaware: 0.026079505410333118
Oklahoma: 0.024963328135245873
North Carolina: 0.024624150871884
Minnesota: 0.02384711190235399
California: 0.023809420211672625
Massachusetts: 0.023510328733406423
Arkansas: 0.022918726973629587
Washington: 0.022390214142341634
```

Hình 1 PageRank Centrality với Python

❖ **Nhận xét:**

Nevada, Missouri và Delaware dẫn đầu: Điều này cho thấy ba bang này có mức độ "trọng yếu" hoặc "liên kết" cao trong mạng lưới phân tích. Có thể hiểu rằng chúng có nhiều kết nối hơn hoặc được kết nối với các nút quan trọng khác trong hệ thống.

Khoảng cách nhỏ giữa các giá trị: Các giá trị PageRank không chênh lệch quá lớn, đặc biệt giữa các bang dẫn đầu. Ví dụ, Nevada (0.02616) chỉ cao hơn Missouri (0.02608) một lượng rất nhỏ. Điều này có thể phản ánh sự cạnh tranh ngang bằng giữa các nút trong mạng lưới.

3.2 Closeness Centrality

3.2.1 Gephi

Id	Closeness Centrality
Kentucky	1.0
North Carolina	1.0
Texas	1.0
Rhode Island	1.0
New Jersey	1.0
Utah	1.0
California	1.0
Mississippi	1.0
Alabama	1.0
Washington	1.0

3.2.2 Python

Closeness Centrality

```
closeness = nx.closeness_centrality(G)  # Closeness Centrality
print("\nCloseness Centrality (Top 10):")
for k, v in sorted(closeness.items(), key=lambda item: item[1], reverse=True)[:10]:
    print(f"{k}: {v}")
```

```
Closeness Centrality (Top 10):
Kentucky: 1.0
Maine: 1.0
Massachusetts: 1.0
Rhode Island: 1.0
Oregon: 1.0
Wyoming: 1.0
Montana: 1.0
Louisiana: 1.0
West Virginia: 1.0
Missouri: 1.0
```

Hình 2 Closeness Centrality với Python

❖ **Nhận xét:**

Ý nghĩa: Closeness Centrality đo lường khoảng cách trung bình từ một nút đến tất cả các nút khác trong mạng. Giá trị 1.0 ở đây cho thấy các bang này có thể được kết nối trực tiếp hoặc qua rất ít bước tới tất cả các bang khác trong mạng.

Đặc biệt: Việc tất cả các bang này có giá trị bằng nhau (và đạt mức tối đa) có thể cho thấy mạng lưới rất nhỏ gọn, hoặc cấu trúc mạng có tính đồng nhất cao.

3.3 Betweeness Centrality

3.3.1 Gephi

Id	Betweenness Centrality
Kentucky	0.0
North Carolina	0.0
Texas	0.0
Rhode Island	0.0
New Jersey	0.0
Utah	0.0
California	0.0
Mississippi	0.0
Alabama	0.0
Washington	0.0

3.3.2 Python

Betweenness Centrality

```
betweenness = nx.betweenness_centrality(G, normalized=True)  # Betweenness Centrality
print("Betweenness Centrality (Top 10):")
for k, v in sorted(betweenness.items(), key=lambda item: item[1], reverse=True)[:10]:
    print(f"{k}: {v}")
```

```
Betweenness Centrality (Top 10):
Kentucky: 0.0
Maine: 0.0
Massachusetts: 0.0
Rhode Island: 0.0
Oregon: 0.0
Wyoming: 0.0
Montana: 0.0
Louisiana: 0.0
West Virginia: 0.0
Missouri: 0.0
```

Hình 3 Betweeness Centrality với Python

❖ **Nhận xét:**

Betweenness Centrality đo lường số lượng đường đi ngắn nhất giữa các cặp nút mà một nút nằm trên. Giá trị bằng 0.0 có nghĩa là các bang này không nằm trên bất kỳ đường đi ngắn nhất nào giữa các nút khác trong mạng.

Các bang này không đóng vai trò trung gian trong mạng lưới.

Mạng có cấu trúc dạng sao hoặc các nút trực tiếp kết nối với nhau mà không cần qua trung gian.

3.4 EigenVector Centrality

3.4.1 Gephi

Id	Eigenvector Centrality
Kentucky	1.0
North Carolina	1.0
Texas	1.0
Rhode Island	1.0
New Jersey	1.0
Utah	1.0
California	1.0
Mississippi	1.0
Alabama	1.0
Washington	1.0

3.4.2 Python

Eigenvector Centrality

```
eigenvector = nx.eigenvector_centrality(G, max_iter=1000)  # Eigenvector Centrality
print("\nEigenvector Centrality (Top 10):")
for k, v in sorted(eigenvector.items(), key=lambda item: item[1], reverse=True)[:10]:
    print(f"{k}: {v}")
```

```
Eigenvector Centrality (Top 10):
Kentucky: 0.1414213562373095
Maine: 0.1414213562373095
Massachusetts: 0.1414213562373095
Rhode Island: 0.1414213562373095
Oregon: 0.1414213562373095
Wyoming: 0.1414213562373095
Montana: 0.1414213562373095
Louisiana: 0.1414213562373095
West Virginia: 0.1414213562373095
Missouri: 0.1414213562373095
```

Hình 4 Eigenvector Centrality với Python

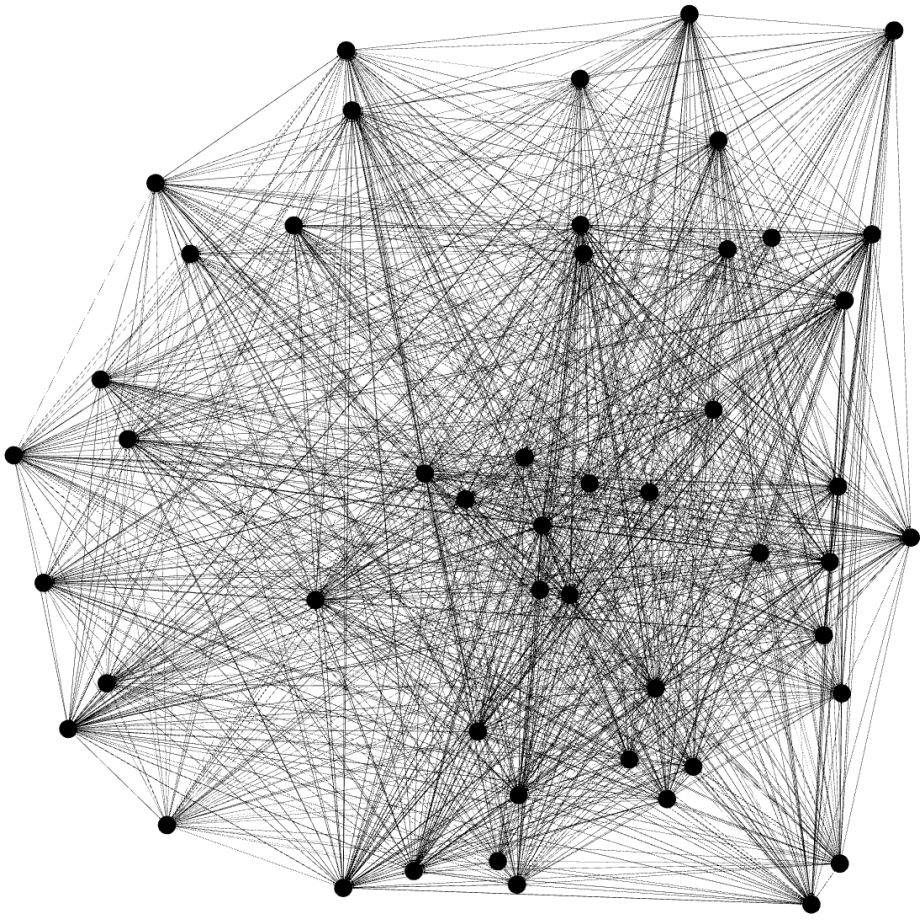
❖ **Nhận xét:**

Eigenvector Centrality đo mức độ ảnh hưởng của một nút, không chỉ dựa trên số lượng liên kết trực tiếp mà còn dựa trên mức độ ảnh hưởng của các nút mà nó liên kết.

Giá trị đồng nhất (0.1414) cho thấy tất cả các bang này có vai trò tương đương trong mạng lưới và các liên kết của chúng (nếu có) dẫn đến các nút có mức độ ảnh hưởng tương tự.

4. Graph Mining

4.1 Thuật toán Louvain [1]



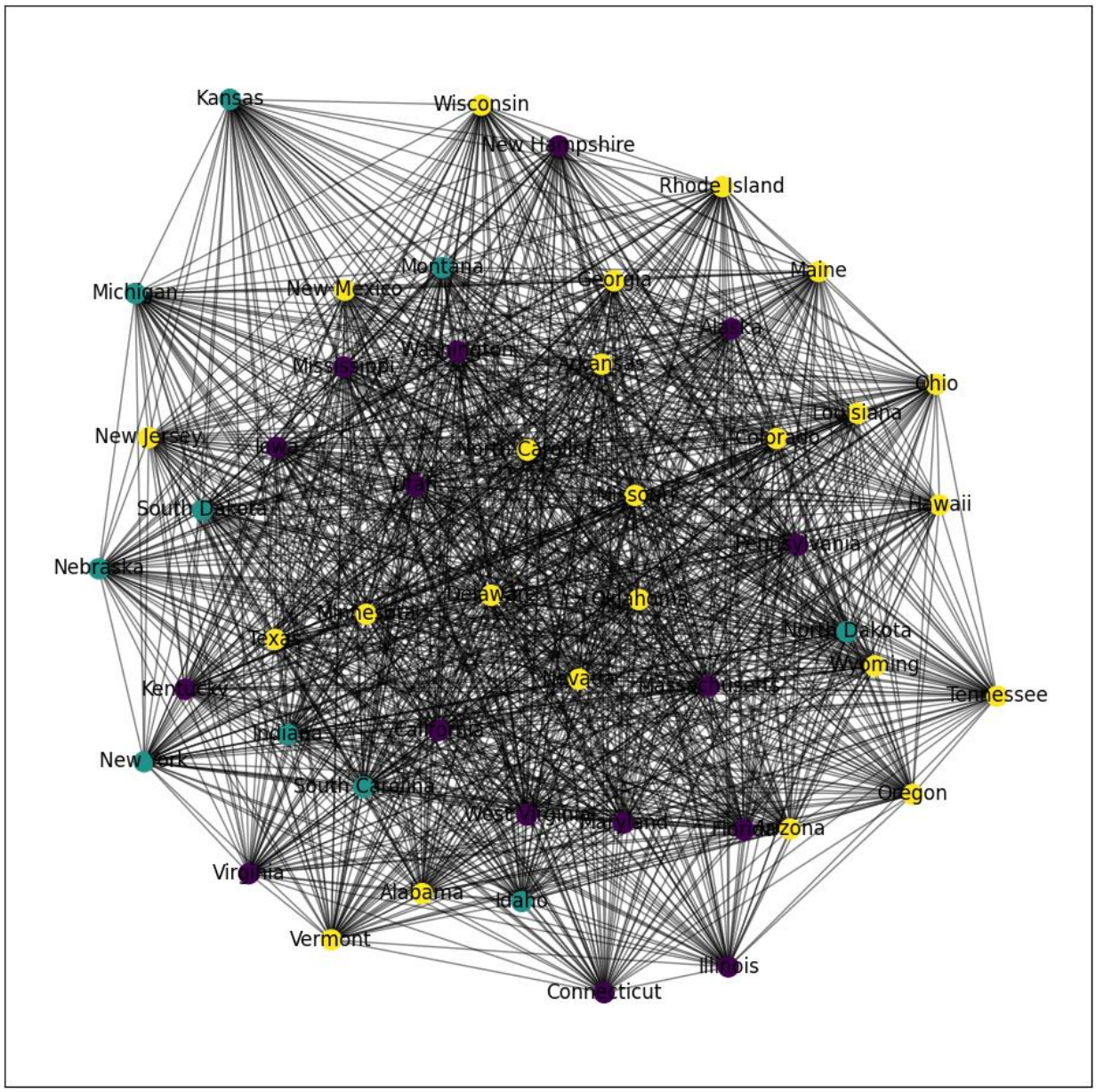
4.1.1 Python

Thuật toán Louvain

```
import matplotlib.cm as cm
import community.community_louvain as community_louvain
```

```
plt.figure(figsize=(12, 12))
partition = community_louvain.best_partition(G)
pos = nx.spring_layout(G)

cmap = cm.get_cmap('viridis', max(partition.values()) + 1)
nx.draw_networkx_nodes(G, pos, partition.keys(), node_color=list(partition.values()), cmap=cmap, node_size=150)
nx.draw_networkx_edges(G, pos, alpha=0.5)
nx.draw_networkx_labels(G, pos)
plt.show()
```

Hình 5 Thuật toán Louvain với Python



Hình 6 Phân cụm với Louvain Python

Số lượng cụm:

```
values = list(partition.values())
print("Số lượng community: ", len(set(values)))
```

```
Số lượng community:  3
```

Community 0:

Kentucky, Massachusetts, West Virginia, New Hampshire, Mississippi, California, Florida, Illinois, Alaska, Maryland, Pennsylvania, Connecticut, Utah, Virginia, Iowa, Washington

Community 1:

Montana, New York, Kansas, North Dakota, Indiana, South Carolina, Idaho, Nebraska, South Dakota, Michigan

Community 2:

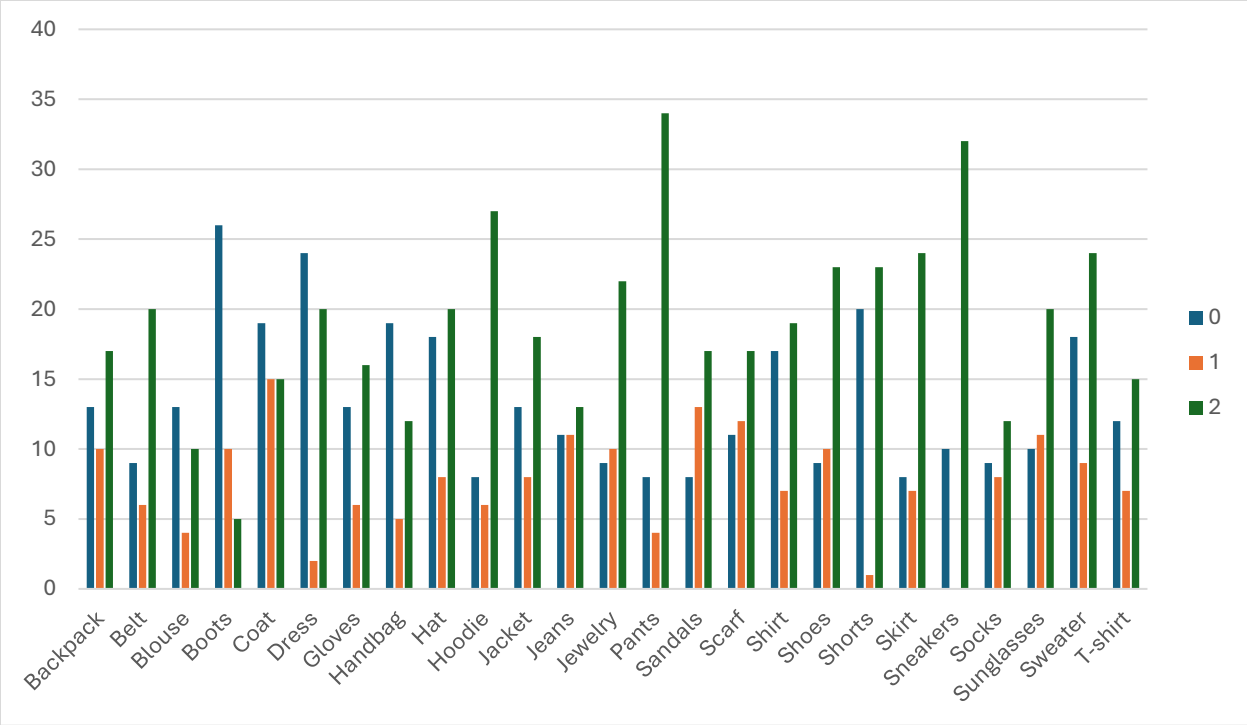
Maine, Rhode Island, Oregon, Wyoming, Louisiana, Missouri, Arkansas, Hawaii, Delaware, Alabama, North Carolina, Oklahoma, Texas, Nevada, Colorado, Arizona, Tennessee, Ohio, New Jersey, Vermont, New Mexico, Georgia, Minnesota, Wisconsin

4.1.2 Pivot Excel

4.1.2.1 Với thuộc tính Item Purchased

Count of Item Purchased	Cluster		
Row Labels	0	1	2
Backpack	13	10	17
Belt	9	6	20
Blouse	13	4	10
Boots	26	10	5
Coat	19	15	15
Dress	24	2	20
Gloves	13	6	16
Handbag	19	5	12
Hat	18	8	20
Hoodie	8	6	27
Jacket	13	8	18
Jeans	11	11	13
Jewelry	9	10	22
Pants	8	4	34
Sandals	8	13	17
Scarf	11	12	17
Shirt	17	7	19
Shoes	9	10	23
Shorts	20	1	23
Skirt	8	7	24
Sneakers	10		32
Socks	9	8	12
Sunglasses	10	11	20
Sweater	18	9	24
T-shirt	12	7	15
Grand Total	**335**	**190**	**475**

Hình 7 Số lượng đơn hàng của các cụm theo Item Purchased - Louvain



Hình 8 Biểu đồ số lượng đơn hàng của các cụm theo Item Purchased - Louvain

❖ **Nhận xét:**

Cluster 0:

Số lượng đơn hàng: 335 -> 33.5%

Sản phẩm bán chạy: Boots (26), Dress (24)

Sản phẩm bán chậm: Hoodie, Pants, Sandals, Sneakers (8); Belt, Jewelry, Shoes, Shocks (9)

Cluster 1:

Số lượng đơn hàng: 190 -> 19%

Sản phẩm bán chạy: Coat (15)

Sản phẩm bán chậm: Sneakers (0), Shorts (1), Dress (2)

Cluster 2:

Số lượng đơn hàng: 475 -> 47.5%

Sản phẩm bán chạy: Pants (34), Sneakers (32), Hoodie (27)

Sản phẩm bán chậm: Boots (5)

4.1.2.2 Với thuộc tính Category

Count of Category	Column Labels		
Row Labels	**0**	**1**	**2**
Accessories	102	68	144
Clothing	148	66	221
Footwear	53	33	77
Outerwear	32	23	33
Grand Total	**335**	**190**	**475**

Hình 9 Số lượng sản phẩm đã mua của các cụm theo Category - Louvain

Hình 10 Biểu đồ số lượng sản phẩm đã mua của các cụm theo Category - Louvain

❖ **Nhận xét:**

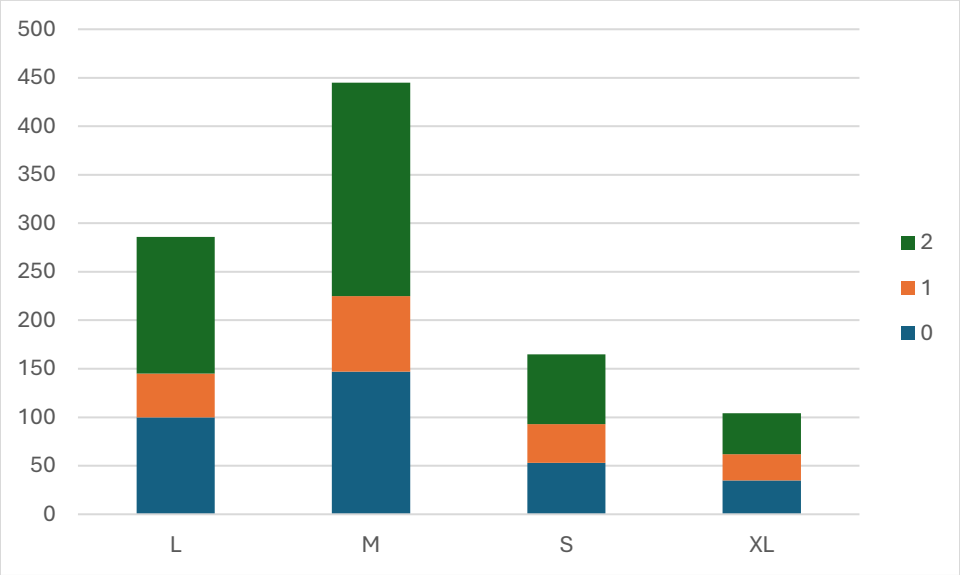
Xu hướng mua hàng chính của cả 3 cụm chủ yếu là Accessories và Clothing với số lượng đơn hàng gấp đôi so với Outerwear và Footwear.

Cụm 0 và cụm 2 có xu hướng mua tương đối nhiều sản phẩm Footwear với lần lượt là 53 và 77

4.1.2.3 Với thuộc tính Size

Count of Size	Column Labels		
Row Labels	0	1	2
L	100	45	141
M	147	78	220
S	53	40	72
XL	35	27	42
Grand Total	**335**	**190**	**475**

Hình 11 Số lượng đơn hàng của các cụm dựa trên Size - Louvain



Hình 12 Biểu đồ số lượng đơn hàng của các cụm theo từng Size - Louvain

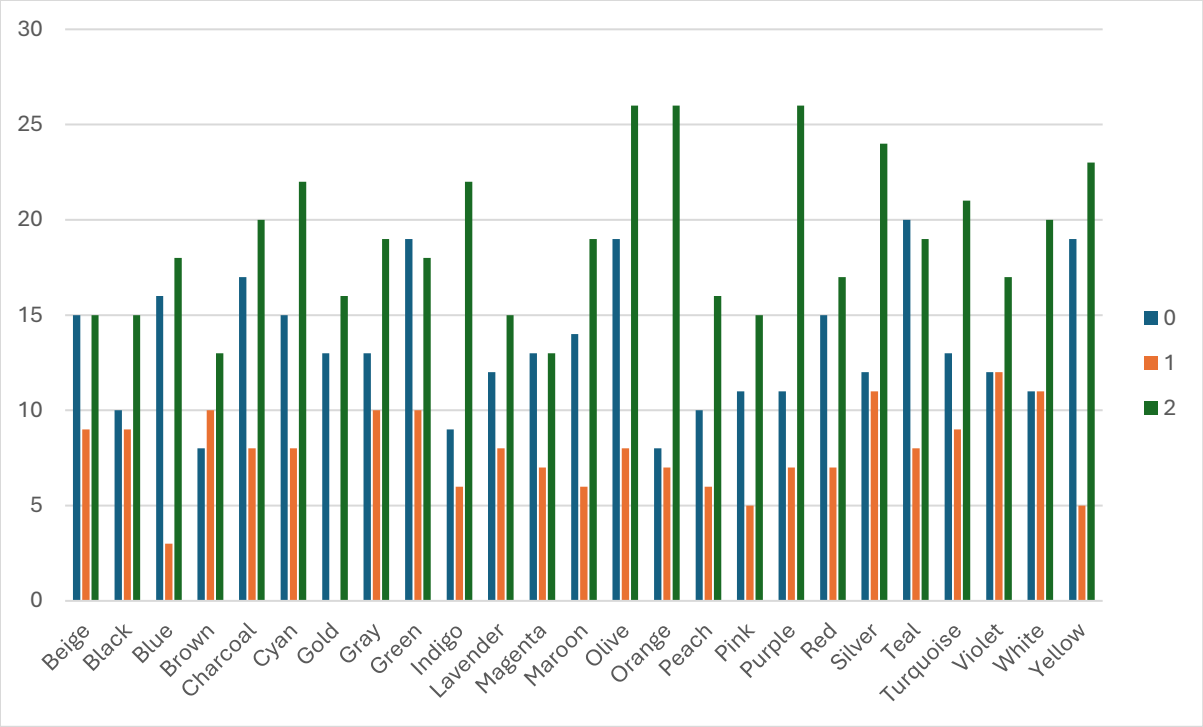
❖ **Nhận xét:**

Cả 3 cụm đều có xu hướng mua tăng dần từ Size XL, S, L, M

4.1.2.4 Với thuộc tính Color

Count of Color	Column Labels		
Row Labels	0	1	2 G
Beige	15	9	15
Black	10	9	15
Blue	16	3	18
Brown	8	10	13
Charcoal	17	8	20
Cyan	15	8	22
Gold	13		16
Gray	13	10	19
Green	19	10	18
Indigo	9	6	22
Lavender	12	8	15
Magenta	13	7	13
Maroon	14	6	19
Olive	19	8	26
Orange	8	7	26
Peach	10	6	16
Pink	11	5	15
Purple	11	7	26
Red	15	7	17
Silver	12	11	24
Teal	20	8	19
Turquoise	13	9	21
Violet	12	12	17
White	11	11	20
Yellow	19	5	23
Grand Total	**335**	**190**	**475**

Hình 13 Số lượng đơn hàng của các cụm theo Color - Louvain



Hình 14 Biểu đồ số lượng đơn hàng của các cụm theo Color - Louvain

❖ **Nhận xét:**

Cụm 0 có xu hướng mua nhiều sản phẩm có màu Teal (20); mua ít sản phảm có màu Brown, Orange (8)

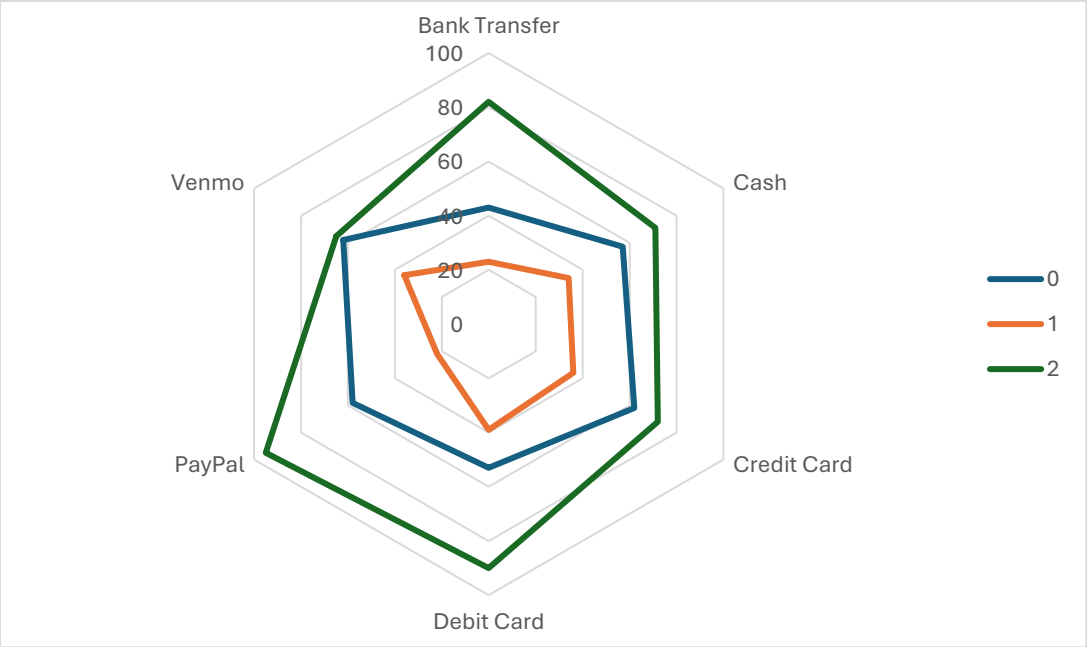
Cụm 1 có xu hướng mua nhiều sản phẩm có màu Violet (12); có xu hướng mua ít sản phẩm có màu Gold (0)

Cụm 2 có xu hướng mua nhiều sản phẩm có màu: Olive, Orange, Purple (26); có xu hướng mua ít sản phẩm có màu Brown, Magenta (13)

4.1.2.5 Với thuộc tính Payment

Count of Preferred Payment Method	Column Labels			
Row Labels	0	1	2	Grand Total
Bank Transfer	43	23	82	148
Cash	57	34	71	162
Credit Card	62	36	72	170
Debit Card	53	39	90	182
PayPal	58	22	95	175
Venmo	62	36	65	163
Grand Total	**335**	**190**	**475**	**1000**

Hình 15 Số lượng đơn hàng của từng cụm theo Payment - Louvain



Hình 16 Biểu đồ số lượng đơn hàng của từng cụm theo Payment - Louvain

❖ **Nhận xét:**

Cụm 0 có xu hướng thanh toán nhiều với Venmo, Credit Card (62); thanh toán ít với Bank Transfer (43)

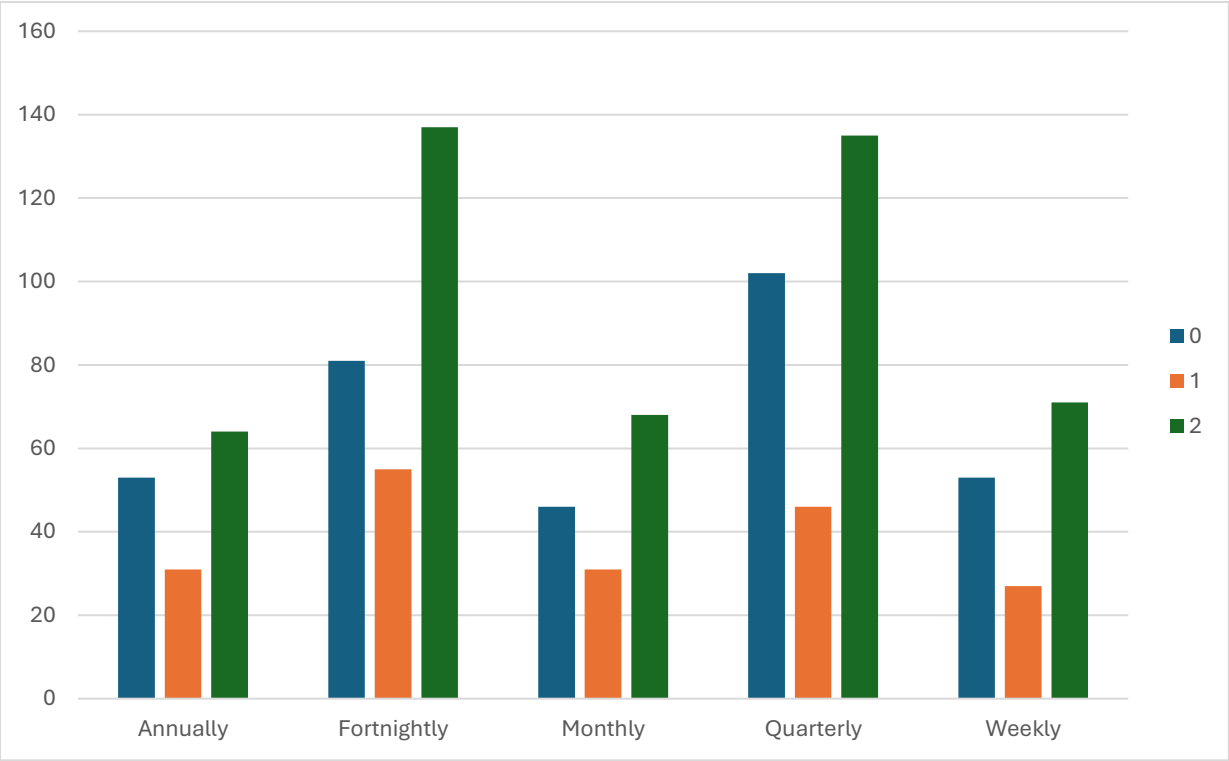
Cụm 1 có xu hướng thanh toán nhiều với Debit Card (39); thanh toán ít với PayPal (22)

Cụm 2 có xu hướng thanh toán nhiều với PayPal (95), Debit Card (90); thanh toán ít với Venmo (65)

4.1.2.6 Với thuộc tính Frequency

Count of Frequency of Purchases	Column Labels			
Row Labels	**0**	**1**	**2**	**Grand Total**
Annually	53	31	64	148
Fortnightly	81	55	137	273
Monthly	46	31	68	145
Quarterly	102	46	135	283
Weekly	53	27	71	151
Grand Total	**335**	**190**	**475**	**1000**

Hình 17 Số lượng đơn hàng của từng cụm theo Frequency - Louvain



Hình 18 Biểu đồ số lượng đơn hàng của từng cụm theo Frequency - Louvain

❖ **Nhận xét:**

Cụm 0 có xu hướng mua nhiều từng quý (Quarterly); mua ít từng tháng (Monthly)

Cụm 1 có xu hướng mua nhiều mỗi 2 tuần (Fornightly); mua ít từng tuần (Weekly)

Cụm 2 có xu hướng mua nhiều ở mỗi 2 tuần (Fortnightly), mỗi quý (Quarterly); mua ít ở hằng năm (Annually)

4.1.3 Gephi

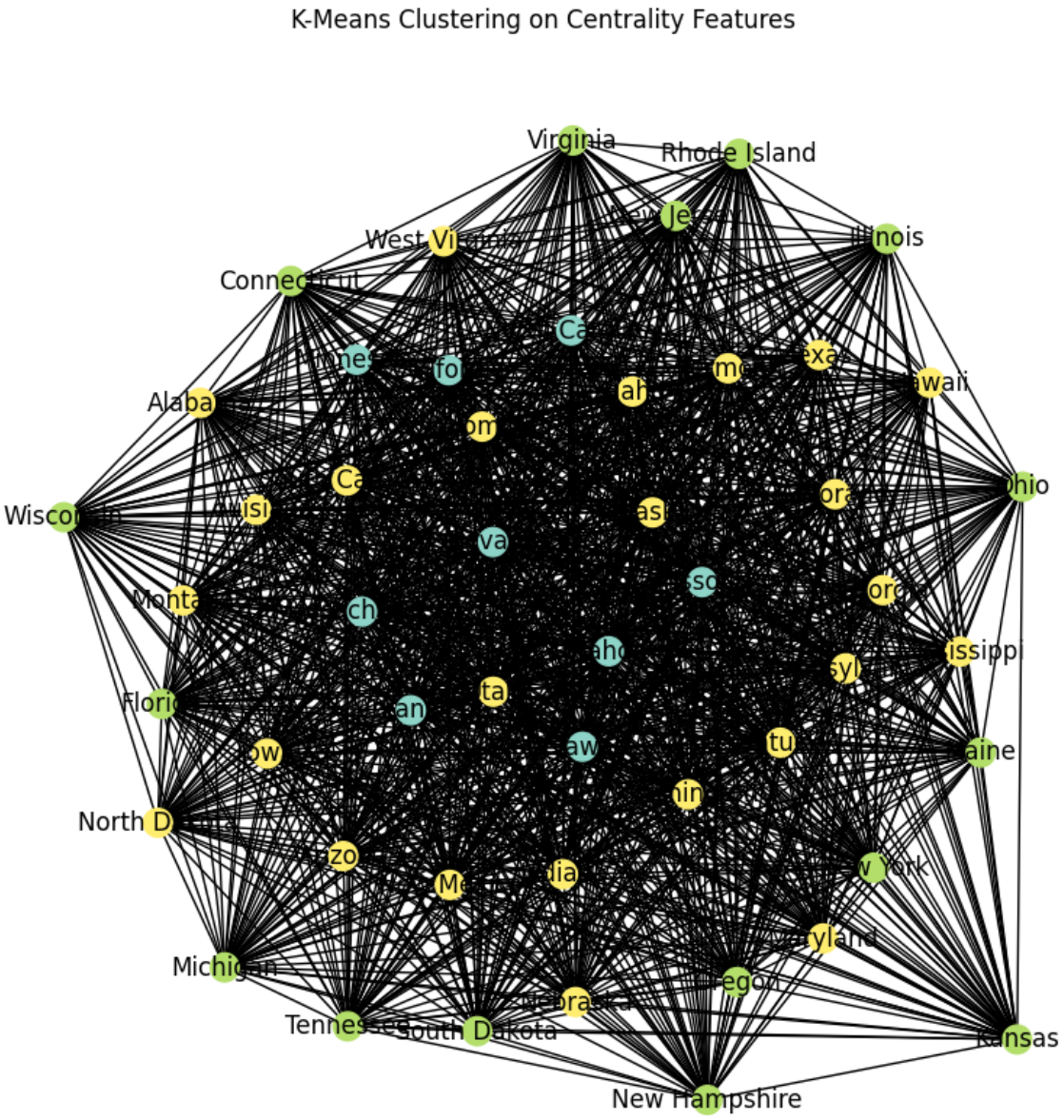
4.2 Thuật toán Kmeans [2]

4.2.1 Python

Thuật toán Kmeans

```python
# Tính lại centrality (nếu chưa có)
betweenness = nx.betweenness_centrality(G)
closeness = nx.closeness_centrality(G)
pagerank = nx.pagerank(G)
eigenvector = nx.eigenvector_centrality(G)

# Tạo ma trận đặc trưng cho K-Means
nodes = list(G.nodes())
X = []
for n in nodes:
    X.append([
        betweenness[n],
        closeness[n],
        pagerank[n],
        eigenvector[n]
    ])
X = np.array(X)

# Áp dụng K-Means, giả sử phân thành 3 cụm
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans_labels = kmeans.fit_predict(X)

print("K-Means Clusters:")
for i in range(3):
    cluster_nodes = [nodes[j] for j in range(len(nodes)) if kmeans_labels[j] == i]
    print(f"Cluster {i}:", cluster_nodes)

# Vẽ đồ thị, tô màu theo kết quả K-Means
color_map = kmeans_labels
plt.figure(figsize=(8,8))
nx.draw(G, pos, node_color=color_map, with_labels=True, cmap=plt.cm.Set3, node_size=200)
plt.title("K-Means Clustering on Centrality Features")
plt.show()
```

Hình 19 Thuật toán Kmeans với Python



Hình 20 Phân cụm với Kmeans

Số lượng cụm: 3

Cluster 0: ['Massachusetts', 'Missouri', 'Arkansas', 'Delaware', 'North Carolina', 'California', 'Oklahoma', 'Nevada', 'Minnesota']

Cluster 1: ['Maine', 'Rhode Island', 'Oregon', 'New Hampshire', 'New York', 'Florida', 'Kansas', 'Illinois', 'Tennessee', 'Ohio', 'New Jersey', 'Connecticut', 'Virginia', 'South Dakota', 'Wisconsin', 'Michigan']

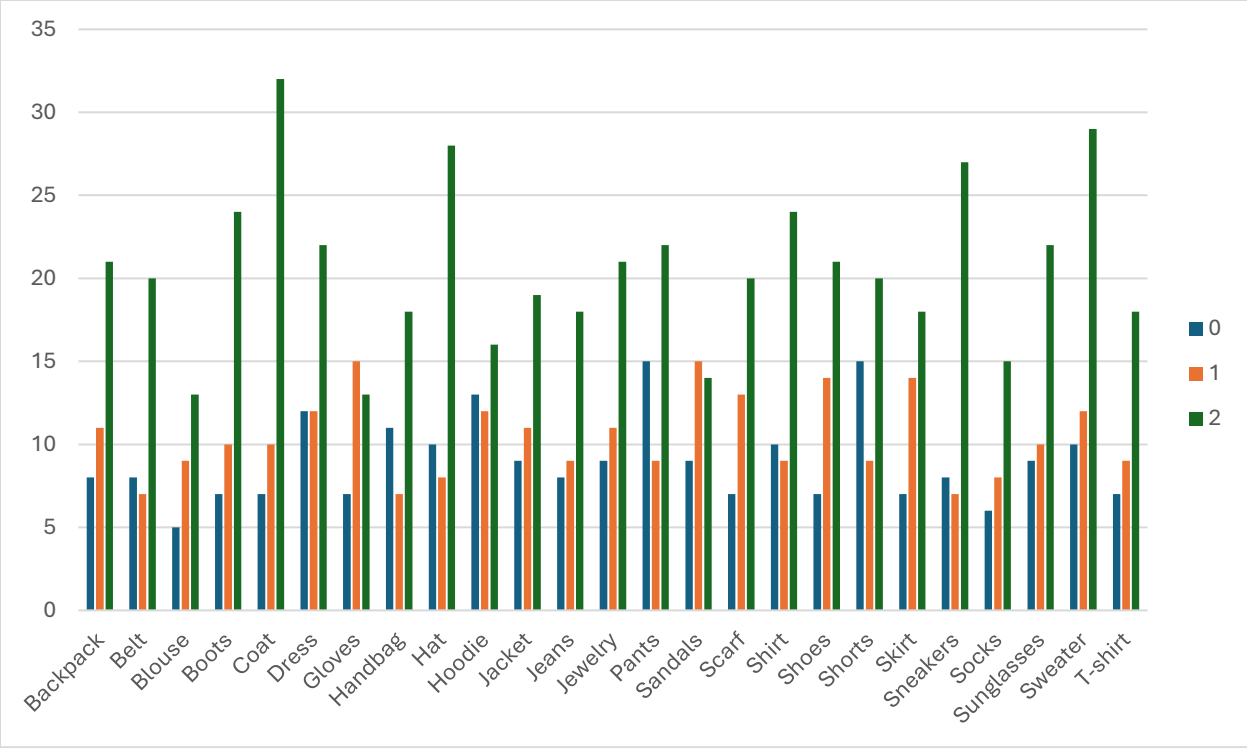
Cluster 2: ['Kentucky', 'Wyoming', 'Montana', 'Louisiana', 'West Virginia', 'Hawaii', 'Alabama', 'Mississippi', 'Texas', 'Colorado', 'North Dakota', 'Indiana', 'Arizona', 'Alaska', 'Maryland', 'Vermont', 'New Mexico', 'South Carolina', 'Idaho', 'Pennsylvania', 'Utah', 'Georgia', 'Nebraska', 'Iowa', 'Washington']

4.2.2 Pivot Excel

4.2.2.1 Với thuộc tính Item Purchased

Count of Item Purchased	Column Labels			
Row Labels	0	1	2	Grand Total
Backpack	8	11	21	40
Belt	8	7	20	35
Blouse	5	9	13	27
Boots	7	10	24	41
Coat	7	10	32	49
Dress	12	12	22	46
Gloves	7	15	13	35
Handbag	11	7	18	36
Hat	10	8	28	46
Hoodie	13	12	16	41
Jacket	9	11	19	39
Jeans	8	9	18	35
Jewelry	9	11	21	41
Pants	15	9	22	46
Sandals	9	15	14	38
Scarf	7	13	20	40
Shirt	10	9	24	43
Shoes	7	14	21	42
Shorts	15	9	20	44
Skirt	7	14	18	39
Sneakers	8	7	27	42
Socks	6	8	15	29
Sunglasses	9	10	22	41
Sweater	10	12	29	51
T-shirt	7	9	18	34
Grand Total	**224**	**261**	**515**	**1000**

Hình 21 Số lượng đơn hàng của các cụm theo Item Purchased - Kmeans



Hình 22 Biểu đồ số lượng đơn hàng của các cụm theo Item Purchased - Kmeans

❖ **Nhận xét:**

Cụm 0 mua nhiều ở sản phẩm Pants, Short (15); mua ít ở sản phẩm Blouse (5)

Cụm 1 mua nhiều ở sản phẩm Gloves, Sandals (15); mua ít ở sản phẩm Belt, Handbag (7)

Cụm 2 mua nhiều ở sản phẩm Coat (32); mua ít ở sản phẩm Blouse, Gloves (13)

4.2.2.2 Với thuộc tính Category

Count of Category	Column Labels			
Row Labels	0	1	2	Grand Total
Accessories	69	82	163	314
Clothing	108	112	215	435
Footwear	31	46	86	163
Outerwear	16	21	51	88
Grand Total	**224**	**261**	**515**	**1000**

Hình 23 Số lượng đơn hàng của các cụm theo Category - Kmeans

Hình 24 Biểu đồ số lương đơn hàng của các cụm theo Category - Kmeans

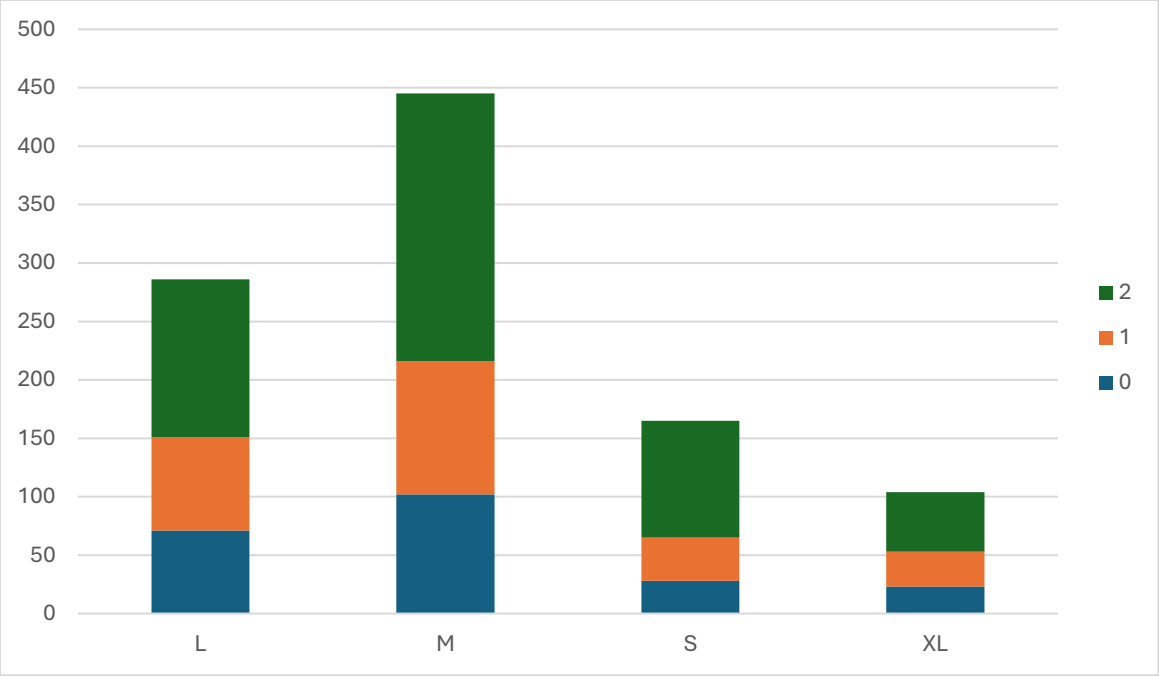
❖ **Nhận xét:**

Xu hướng của 3 cụm với số lượng tăng dần từ Outerwear, Footwear, Accessories, Clothing

4.2.2.3 Với thuộc tính Size

Count of Size	Column Labels			
Row Labels	**0**	**1**	**2**	**Grand Total**
L	71	80	135	286
M	102	114	229	445
S	28	37	100	165
XL	23	30	51	104
Grand Total	**224**	**261**	**515**	**1000**

Hình 25 Số lượng đơn hàng của các cụm theo Size - Kmeans



Hình 26 Đồ thị số lượng đơn hàng của các cụm theo Size - Kmeans

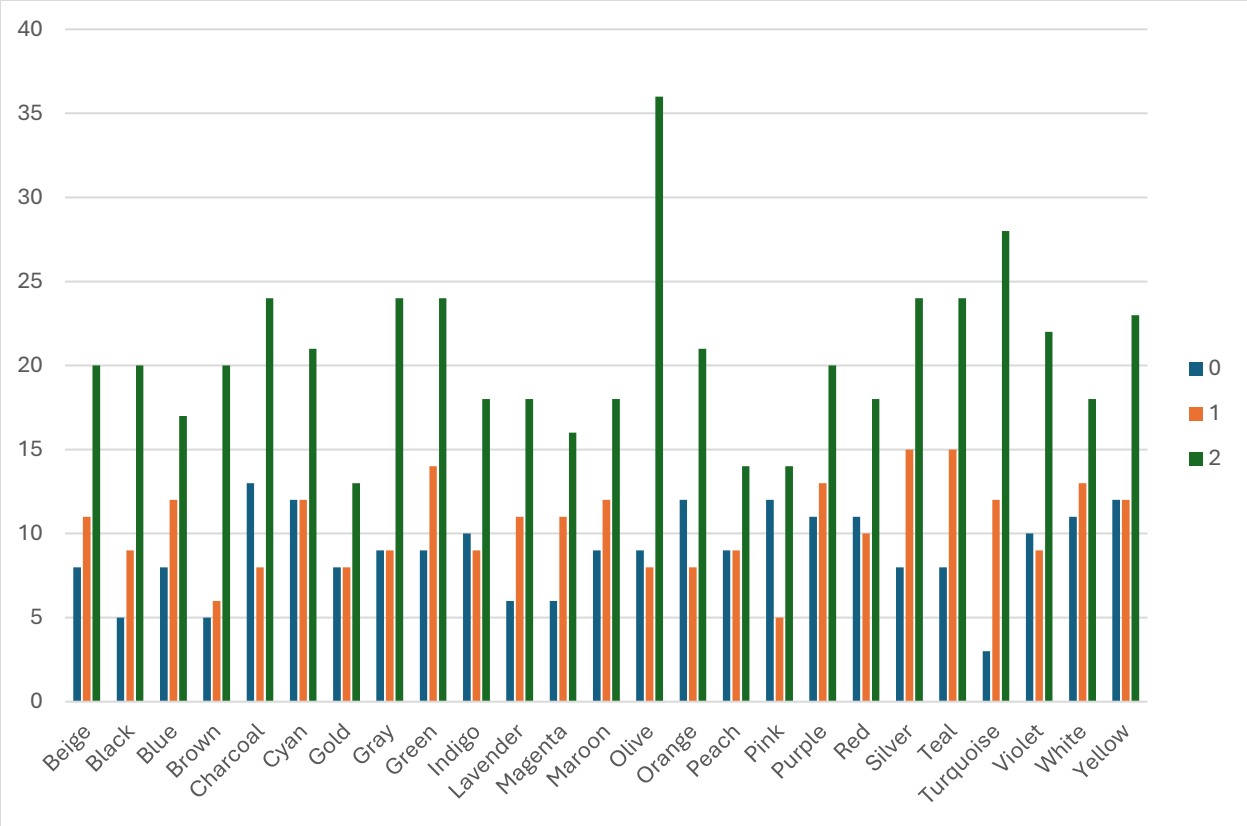
❖ **Nhận xét:**

Xu hướng mua của cả 3 cụm tăng dần từ Size XL, S, L, M

4.2.2.4 Với thuộc tính Color

Count of Color	Column Labels			
Row Labels	0	1	2	Grand Total
Beige	8	11	20	39
Black	5	9	20	34
Blue	8	12	17	37
Brown	5	6	20	31
Charcoal	13	8	24	45
Cyan	12	12	21	45
Gold	8	8	13	29
Gray	9	9	24	42
Green	9	14	24	47
Indigo	10	9	18	37
Lavender	6	11	18	35
Magenta	6	11	16	33
Maroon	9	12	18	39
Olive	9	8	36	53
Orange	12	8	21	41
Peach	9	9	14	32
Pink	12	5	14	31
Purple	11	13	20	44
Red	11	10	18	39
Silver	8	15	24	47
Teal	8	15	24	47
Turquoise	3	12	28	43
Violet	10	9	22	41
White	11	13	18	42
Yellow	12	12	23	47
Grand Total	**224**	**261**	**515**	**1000**

Hình 27 Số lượng đơn hàng của các cụm theo Color - Kmeans



Hình 28 Biểu đồ số lượng đơn hàng của các cụm theo Color - Kmeans

❖ **Nhận xét:**

Cụm 0 có xu hướng mua nhiều ở màu Charcoal (13); mua ít ở màu Gurquoise (3)

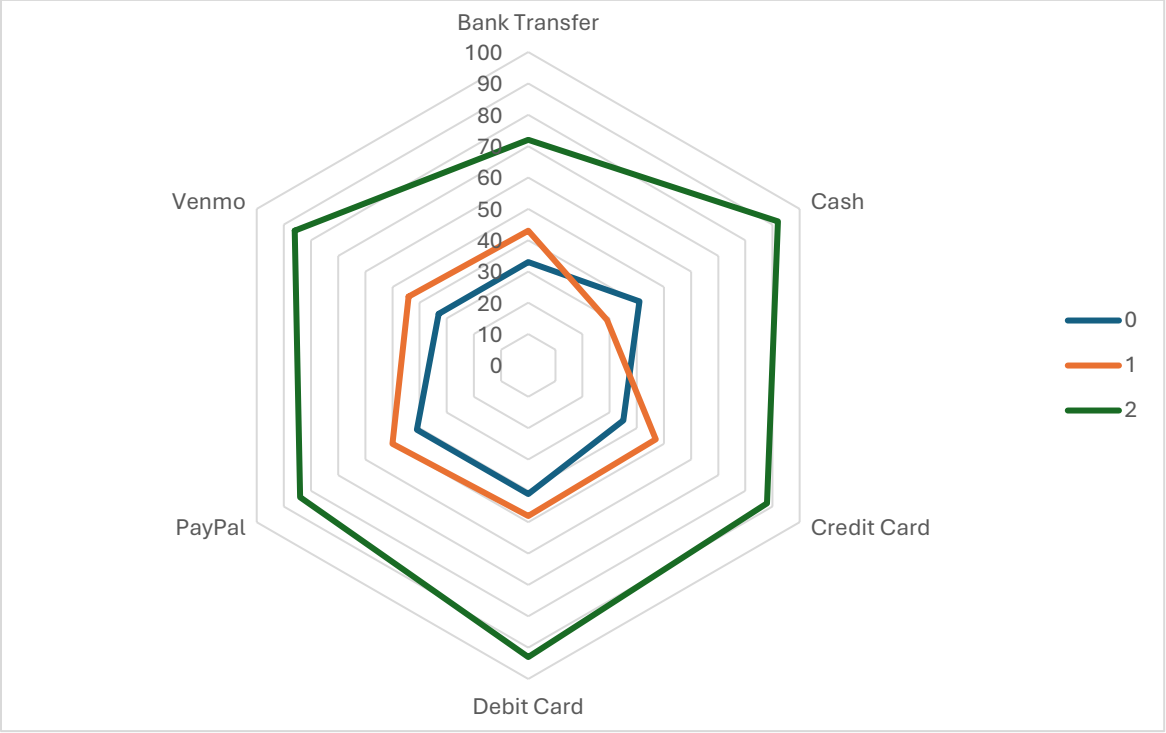
Cụm 1 có xu hướng mua nhiều ở màu Silver, Teal (15); mua ít ở màu Pink (5)

Cụm 2 có xu hướng mua nhiều ở màu Olive (36); mua ít ở màu Gold (13)

4.2.2.5 Với thuộc tính Payment

Count of Preferred Payment Method	Column Labels			
Row Labels	**0**	**1**	**2**	**Grand Total**
Bank Transfer	33	43	72	148
Cash	41	29	92	162
Credit Card	35	47	88	170
Debit Card	41	48	93	182
PayPal	41	50	84	175
Venmo	33	44	86	163
Grand Total	**224**	**261**	**515**	**1000**

Hình 29 Số lượng đơn hàng của các cụm theo Payment - Kmeans



Hình 30 Đồ thị số lượng đơn hàng của các cụm theo Payment - Kmeans

❖ **Nhận xét:**

Cụm 0 có xu hướng thanh toán nhiều bằng Cash, Debit Card, PayPal (41); thanh toán ít bằng Bank Transfer, Venmo (33)

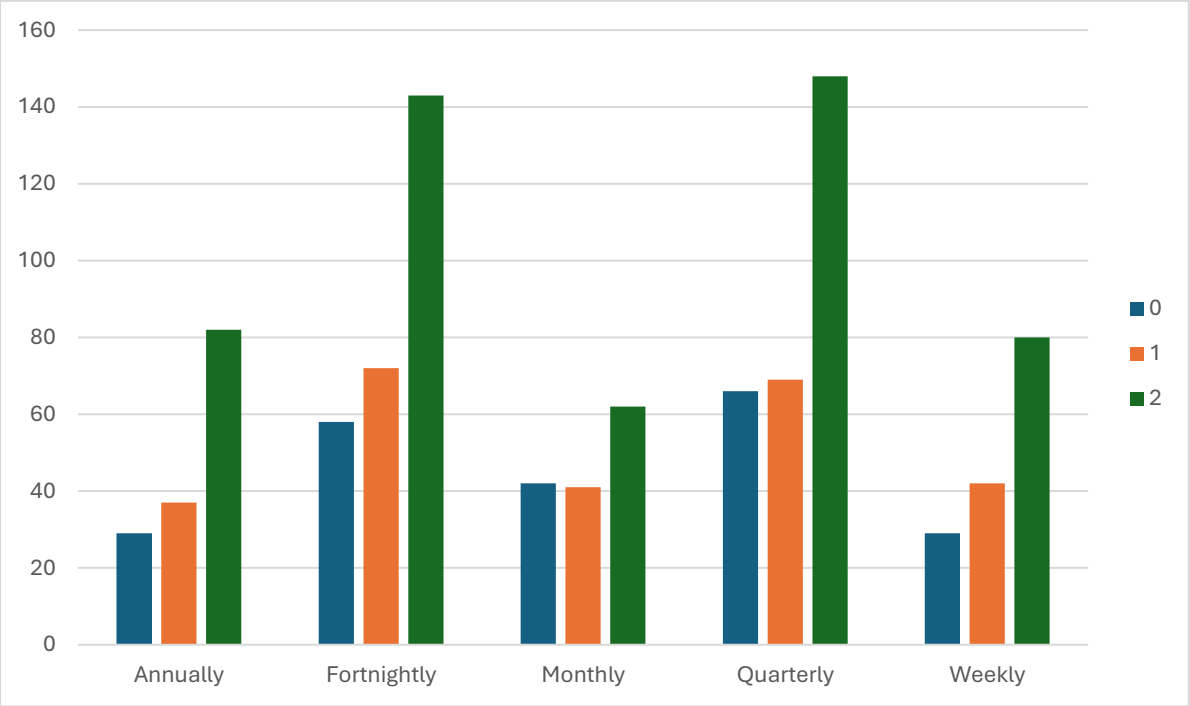
Cụm 1 có xu hướng thanh toán nhiều bằng PayPal (50); thanh toán ít bằng Cash (29)

Cụm 2 có xu hướng thanh toán nhiều bằng Debit Card (93); thanh toán ít bằng Bank Transfer (72)

4.2.2.6 Với thuộc tính Frequency

Count of Frequency of Purchases	Column Labels			
Row Labels	0	1	2	Grand Total
Annually	29	37	82	148
Fortnightly	58	72	143	273
Monthly	42	41	62	145
Quarterly	66	69	148	283
Weekly	29	42	80	151
Grand Total	**224**	**261**	**515**	**1000**

Hình 31 Số lượng đơn hàng của các cụm theo Frequency - Kmeans



Hình 32 Biểu đồ số lượng đơn hàng của các cụm theo Frequency - Kmeans

❖ **Nhận xét:**

Cụm 0 có xu hướng mua hàng theo từng quý (Quarterly)

Cụm 1 có xu hướng mua hàng mỗi 2 tuần (Fortnightly)

Cụm 2 có xu hướng mua hàng theo quý (Quarterly) và mỗi 2 tuần (Fortnightly)

4.2.3 Gephi

4.3 Thuật toán KNN [3]

4.3.1 Python

Thuật toán KNN

```python
from sklearn.neighbors import NearestNeighbors
from sklearn.cluster import SpectralClustering

# X đã có ở trên (từ centrality)
# Xây dựng k-NN graph
k = 5
nbrs = NearestNeighbors(n_neighbors=k, algorithm='ball_tree').fit(X)
distances, indices = nbrs.kneighbors(X)

# Tạo một đồ thị k-NN (chỉ giữ cạnh giữa nút và k láng giềng gần nhất)
knn_graph = nx.Graph()
knn_graph.add_nodes_from(nodes)
for i, nbr_list in enumerate(indices):
    n1 = nodes[i]
    for idx in nbr_list[1:]: # Bỏ qua phần tử đầu tiên vì chính nó
        n2 = nodes[idx]
        # thêm cạnh vào knn_graph
        knn_graph.add_edge(n1, n2)

# Bây giờ phân cụm trên knn_graph sử dụng Spectral Clustering
adj_mat = nx.to_numpy_array(knn_graph, nodelist=nodes)
sc = SpectralClustering(n_clusters=3, affinity='precomputed', random_state=42)
sc_labels = sc.fit_predict(adj_mat)

print("Spectral Clustering on k-NN Graph:")
for i in range(3):
    cluster_nodes = [nodes[j] for j in range(len(nodes)) if sc_labels[j] == i]
    print(f"Cluster {i}:", cluster_nodes)

# Vẽ đồ thị k-NN, màu theo cụm
color_map = sc_labels
pos_knn = nx.spring_layout(knn_graph)
plt.figure(figsize=(8,8))
nx.draw(knn_graph, pos_knn, node_color=color_map, with_labels=True, cmap=plt.cm.Set3, node_size=200)
plt.title("Spectral Clustering on k-NN Graph")
plt.show()
```

Hình 33 Thuật toán KNN với Python

Hình 34 Phân cụm với KNN Python

Số lượng cụm: 3

Cluster 0: ['Maine', 'Rhode Island', 'Oregon', 'New Hampshire', 'New York', 'Florida', 'Kansas', 'Illinois', 'Tennessee', 'Ohio', 'New Jersey', 'Connecticut', 'Virginia', 'South Dakota', 'Wisconsin', 'Michigan']

Cluster 1: ['Kentucky', 'Massachusetts', 'Wyoming', 'Montana', 'West Virginia', 'Missouri', 'Arkansas', 'Delaware', 'North Carolina', 'California', 'Oklahoma', 'Nevada', 'Indiana', 'New Mexico', 'South Carolina', 'Idaho', 'Utah', 'Georgia', 'Minnesota', 'Washington']

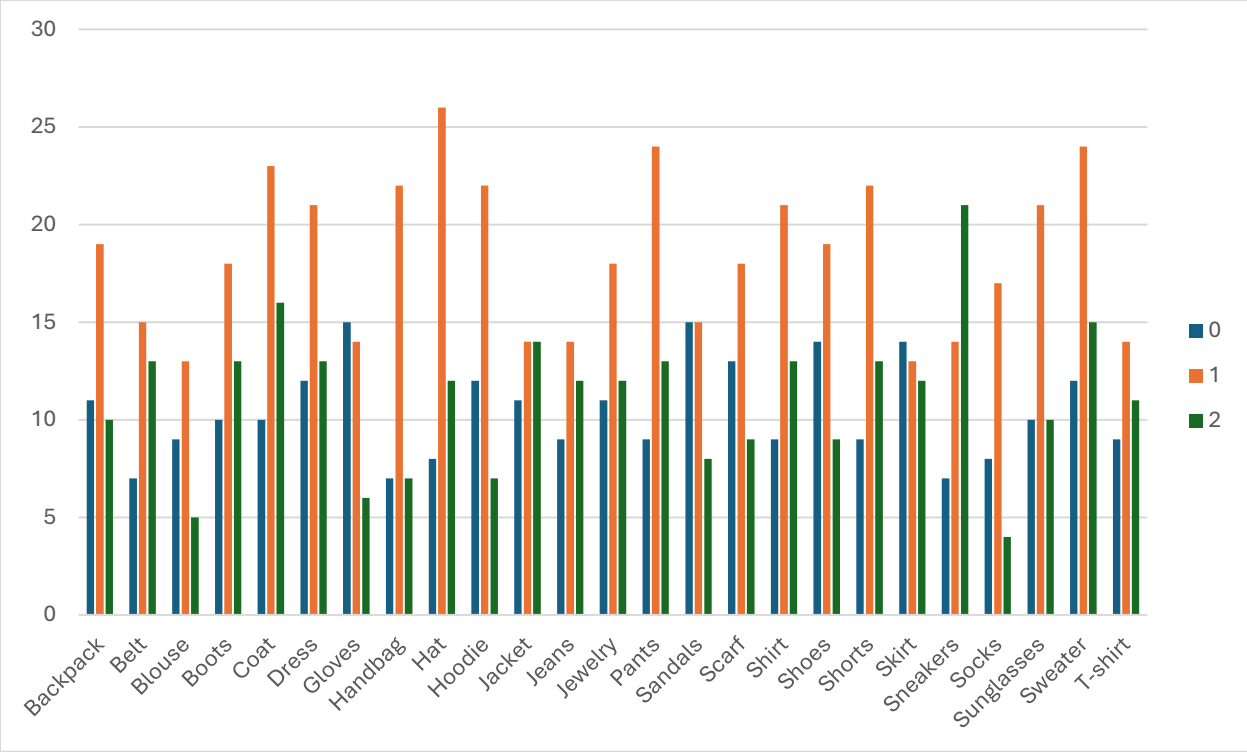
Cluster 2: ['Louisiana', 'Hawaii', 'Alabama', 'Mississippi', 'Texas', 'Colorado', 'North Dakota', 'Arizona', 'Alaska', 'Maryland', 'Vermont', 'Pennsylvania', 'Nebraska', 'Iowa']

4.3.2 Pivot Excel

4.3.2.1 Với thuộc tính Item Purchased

Count of Item Purchased	Column Labels			
Row Labels	0	1	2	Grand Total
Backpack	11	19	10	40
Belt	7	15	13	35
Blouse	9	13	5	27
Boots	10	18	13	41
Coat	10	23	16	49
Dress	12	21	13	46
Gloves	15	14	6	35
Handbag	7	22	7	36
Hat	8	26	12	46
Hoodie	12	22	7	41
Jacket	11	14	14	39
Jeans	9	14	12	35
Jewelry	11	18	12	41
Pants	9	24	13	46
Sandals	15	15	8	38
Scarf	13	18	9	40
Shirt	9	21	13	43
Shoes	14	19	9	42
Shorts	9	22	13	44
Skirt	14	13	12	39
Sneakers	7	14	21	42
Socks	8	17	4	29
Sunglasses	10	21	10	41
Sweater	12	24	15	51
T-shirt	9	14	11	34
Grand Total	**261**	**461**	**278**	**1000**

Hình 35 Số lượng đơn hàng của các cụm theo Item Purchased – KNN



Hình 36 Biểu đồ số lượng đơn hàng của các cụm theo Item Purchased – KNN

❖ **Nhận xét:**

Cụm 0 có xu hướng mua nhiều sản phẩm Gloves, Sandals (15); mua ít sản phẩm Belt, Handbag, Sneakers (7)

Cụm 1 có xu hướng mua nhiều sản phẩm Hat (26); mua ít sản phẩm Blouse, Skirt (13)

Cụm 2 có xu hướng mua nhiều sản phẩm Sneakers (21); mua ít sản phẩm Socks (4)

4.3.2.2 Với thuộc tính Category

Count of Category	Column Labels			
Row Labels	0	1	2	Grand Total
Accessories	82	153	79	314
Clothing	112	205	118	435
Footwear	46	66	51	163
Outerwear	21	37	30	88
Grand Total	**261**	**461**	**278**	**1000**

Hình 37 Số lượng đơn hàng của các cụm theo Category – KNN

Hình 38 Biểu đồ số lượng đơn hàng của các cụm theo Category – KNN

❖ **Nhận xét:**

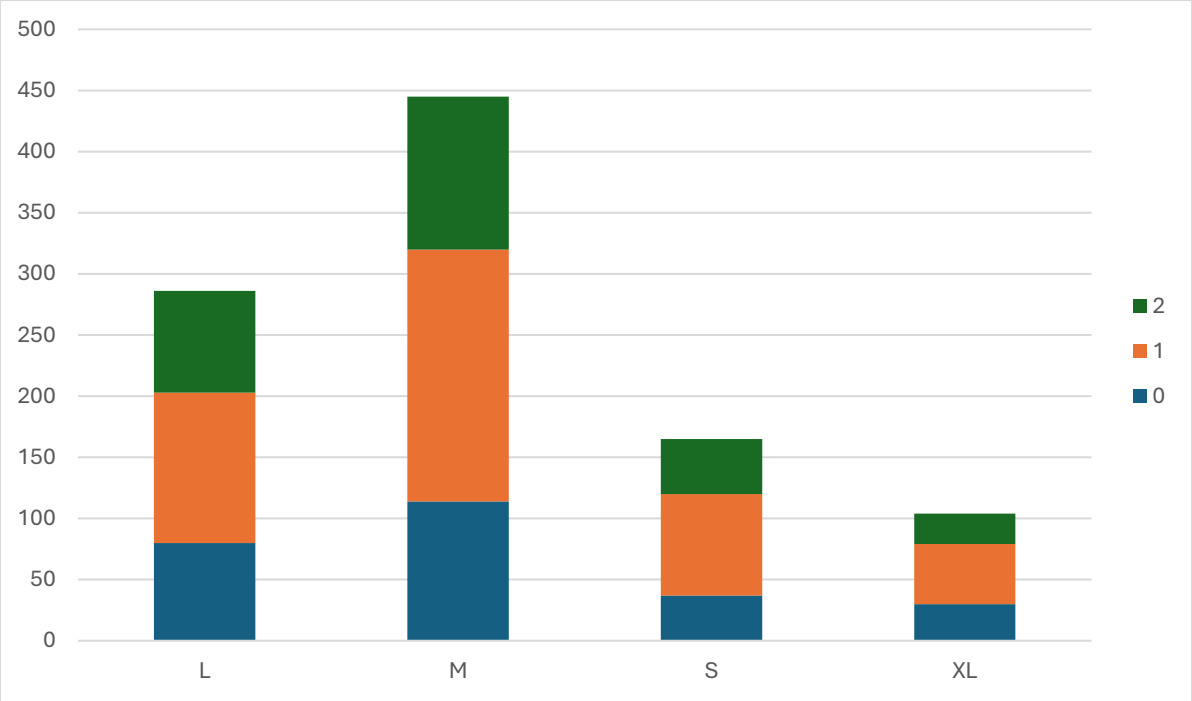
Cả 3 cụm có xu hướng đơn hàng tăng dần lần lượt: Outerwear, Footwear, Accessories, Clothing.

Cụm 1 có xu hướng mua Accessories với tỉ trọng tương đối cao so với 2 cụm còn lại

4.3.2.3 Với thuộc tính Size

Count of Size	Column Labels			
Row Labels	0	1	2	Grand Total
L	80	123	83	286
M	114	206	125	445
S	37	83	45	165
XL	30	49	25	104
Grand Total	**261**	**461**	**278**	**1000**

Hình 39 Số lượng đơn hàng của các cụm theo Size



Hình 40 Biểu đồ số lượng đơn hàng của các cụm theo Size

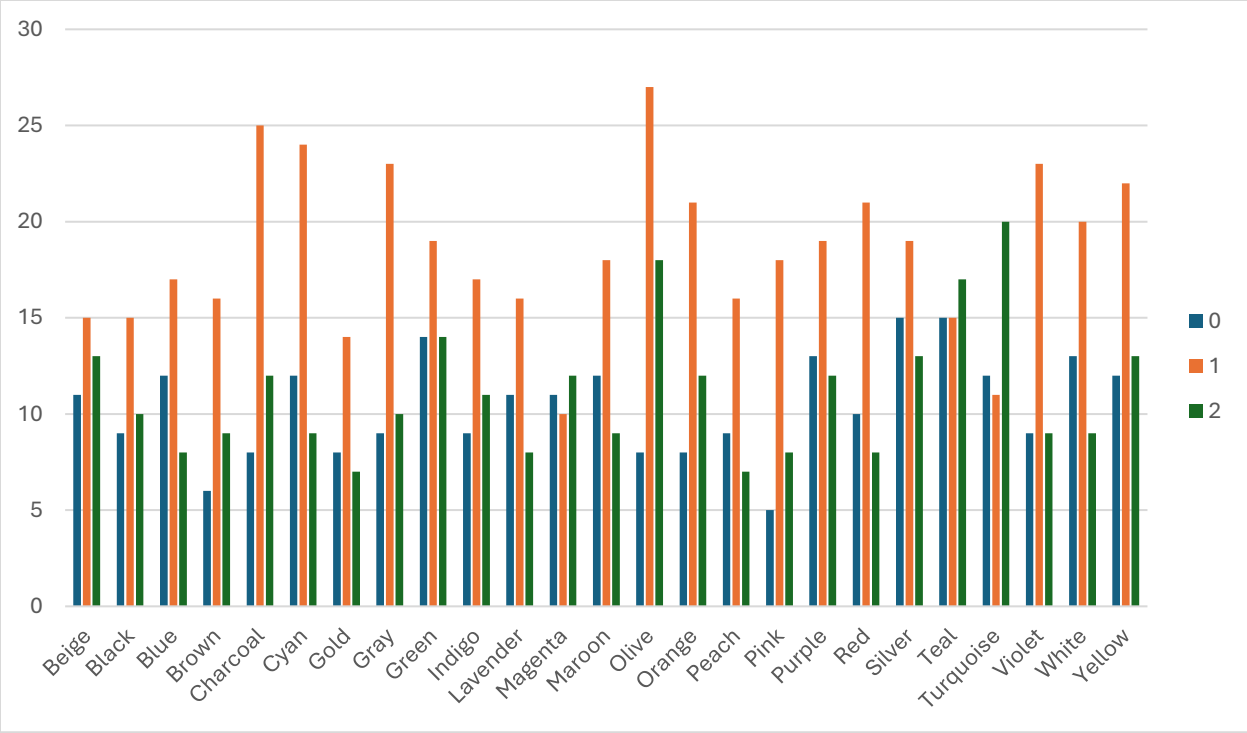
❖ **Nhận xét:**

Xu hướng mua hàng theo Size của 3 cụm tăng dần lần lượt: XL, S, L, M

4.3.2.4 Với thuộc tính Color

Count of Color	Column Labels			
Row Labels	0	1	2	Grand Total
Beige	11	15	13	39
Black	9	15	10	34
Blue	12	17	8	37
Brown	6	16	9	31
Charcoal	8	25	12	45
Cyan	12	24	9	45
Gold	8	14	7	29
Gray	9	23	10	42
Green	14	19	14	47
Indigo	9	17	11	37
Lavender	11	16	8	35
Magenta	11	10	12	33
Maroon	12	18	9	39
Olive	8	27	18	53
Orange	8	21	12	41
Peach	9	16	7	32
Pink	5	18	8	31
Purple	13	19	12	44
Red	10	21	8	39
Silver	15	19	13	47
Teal	15	15	17	47
Turquoise	12	11	20	43
Violet	9	23	9	41
White	13	20	9	42
Yellow	12	22	13	47
Grand Total	**261**	**461**	**278**	**1000**

Hình 41 Số lượng đơn hàng của các cụm theo Color – KNN



Hình 42 Biểu đồ số lượng đơn hàng của các cụm theo Color – KNN

❖ **Nhận xét:**

Cụm 0 có xu hướng mua nhiều ở màu Silver, Teal (15); mua ít ở màu Pink (5)

Cụm 1 có xu hướng mua nhiều ở màu Olive (27); mua ít ở màu Magenta (10)

Cụm 2 có xu hướng mua nhiều ở màu Turquoise (20); mua ít ở màu Gold, Peach (7)

4.3.2.5 Với thuộc tính Payment

Count of Preferred Payment Method	Column Labels			
Row Labels	0	1	2	Grand Total
Bank Transfer	43	66	39	148
Cash	29	97	36	162
Credit Card	47	74	49	170
Debit Card	48	79	55	182
PayPal	50	75	50	175
Venmo	44	70	49	163
Grand Total	**261**	**461**	**278**	**1000**

Hình 43 Số lượng đơn hàng của các cụm theo Payment - KNN

Hình 44 Biểu đồ số lượng đơn hàng của các cụm theo Payment – KNN

❖ **Nhận xét:**

Cụm 0 có xu hướng thanh toán nhiều bằng PayPal (50); thanh toán ít bằng Cash (29)

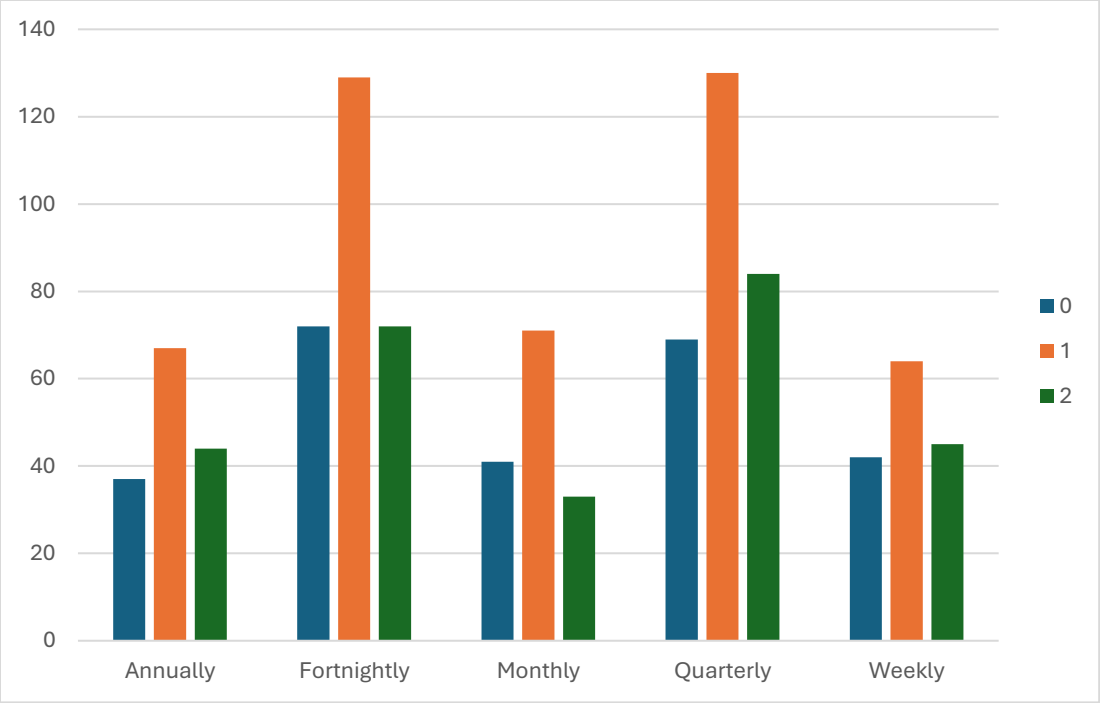
Cụm 1 có xu hướng thanh toán nhiều bằng Cash (94); thanh toán ít bằng Bank Transfer (66)

Cụm 2 có xu hướng thanh toán nhiều bằng Debit Cash (55), thanh toán ít bằng Cash (36)

4.3.2.6 Với thuộc tính Frequency

Count of Frequency of Purchases	Column Labels			
Row Labels	0	1	2	Grand Total
Annually	37	67	44	148
Fortnightly	72	129	72	273
Monthly	41	71	33	145
Quarterly	69	130	84	283
Weekly	42	64	45	151
Grand Total	**261**	**461**	**278**	**1000**

Hình 45 Số lượng đơn hàng của các cụm theo Frequency – KNN



Hình 46 Biểu đồ số lượng đơn hàng của các cụm theo Frequency – KNN

❖ **Nhận xét:**

Các cụm có xu hướng mua mỗi 2 tuần (Fortnightly) và mỗi quý (Quarterly)

4.3.3 Gephi

4.4 Thuật toán Girvan Newman [4]

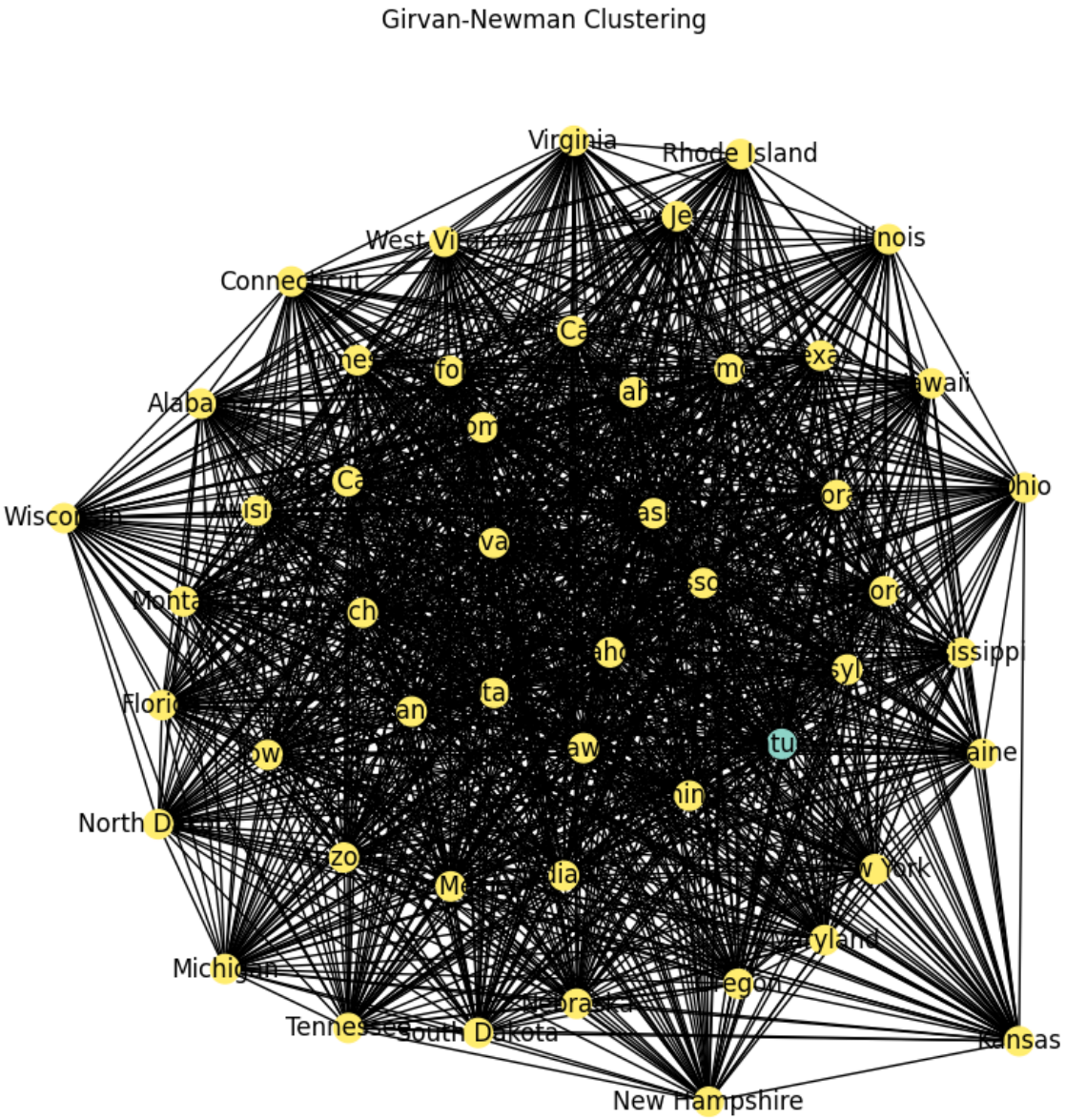
4.4.1 Python

Thuật toán Girvan-Newman

```
from sklearn.cluster import KMeans
from networkx.algorithms import community
```

```
comp = community.girvan_newman(G) # comp là một generator, mỗi bước yield ra một phân hoạch
# Lấy ra phân hoạch đầu tiên (tách G thành 2 cộng đồng)
gn_communities = tuple(sorted(c) for c in next(comp))

print("Girvan-Newman communities:")
for i, comm in enumerate(gn_communities):
    print(f"Community {i}:", comm)

# Vẽ đồ thị với màu sắc theo cộng đồng Girvan-Newman
color_map = {}
for i, comm in enumerate(gn_communities):
    for node in comm:
        color_map[node] = i

pos = nx.spring_layout(G)
colors = [color_map[node] for node in G.nodes()]

plt.figure(figsize=(8,8))
nx.draw(G, pos, node_color=colors, with_labels=True, cmap=plt.cm.Set3, node_size=200)
plt.title("Girvan-Newman Clustering")
plt.show()
```

Hình 47 Thuật toán Girvan-Newman Python



Hình 48 Phân cụm với Girvan-Newman Python

Số lượng cụm: 2

Community 0: ['Kentucky']

Community 1: ['Alabama', 'Alaska', 'Arizona', 'Arkansas', 'California', 'Colorado', 'Connecticut', 'Delaware', 'Florida', 'Georgia', 'Hawaii', 'Idaho', 'Illinois', 'Indiana', 'Iowa', 'Kansas', 'Louisiana', 'Maine', 'Maryland', 'Massachusetts', 'Michigan', 'Minnesota', 'Mississippi', 'Missouri', 'Montana', 'Nebraska', 'Nevada', 'New Hampshire', 'New Jersey', 'New Mexico', 'New York', 'North Carolina', 'North Dakota', 'Ohio', 'Oklahoma', 'Oregon',

'Pennsylvania', 'Rhode Island', 'South Carolina', 'South Dakota', 'Tennessee', 'Texas', 'Utah', 'Vermont', 'Virginia', 'Washington', 'West Virginia', 'Wisconsin', 'Wyoming']

4.4.2 Gephi

5. Tham khảo

[1] "networkx," [Online]. Available: https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.louvain.louvain_communities.html.

[2] "machinelearningcoban," [Online]. Available: https://machinelearningcoban.com/2017/01/01/kmeans/.

[3] "machinelearningcoban," [Online]. Available: https://machinelearningcoban.com/2017/01/08/knn/.

[4] "networkx," [Online]. Available: https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.community.centrality.girvan_newman.html.