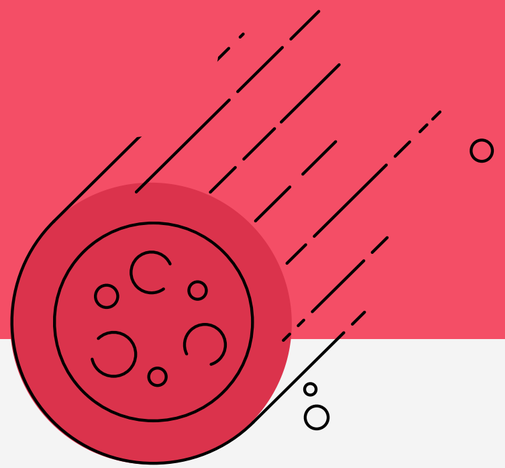
# Phân khúc khách hàng

PITCH DECK v01

# RFM



1 Recency: lần mua hàng gần nhất

2 Frequency: tần suất mua hàng

3 Monetary Value: số tiền, doanh thu mang lại

# Dataset

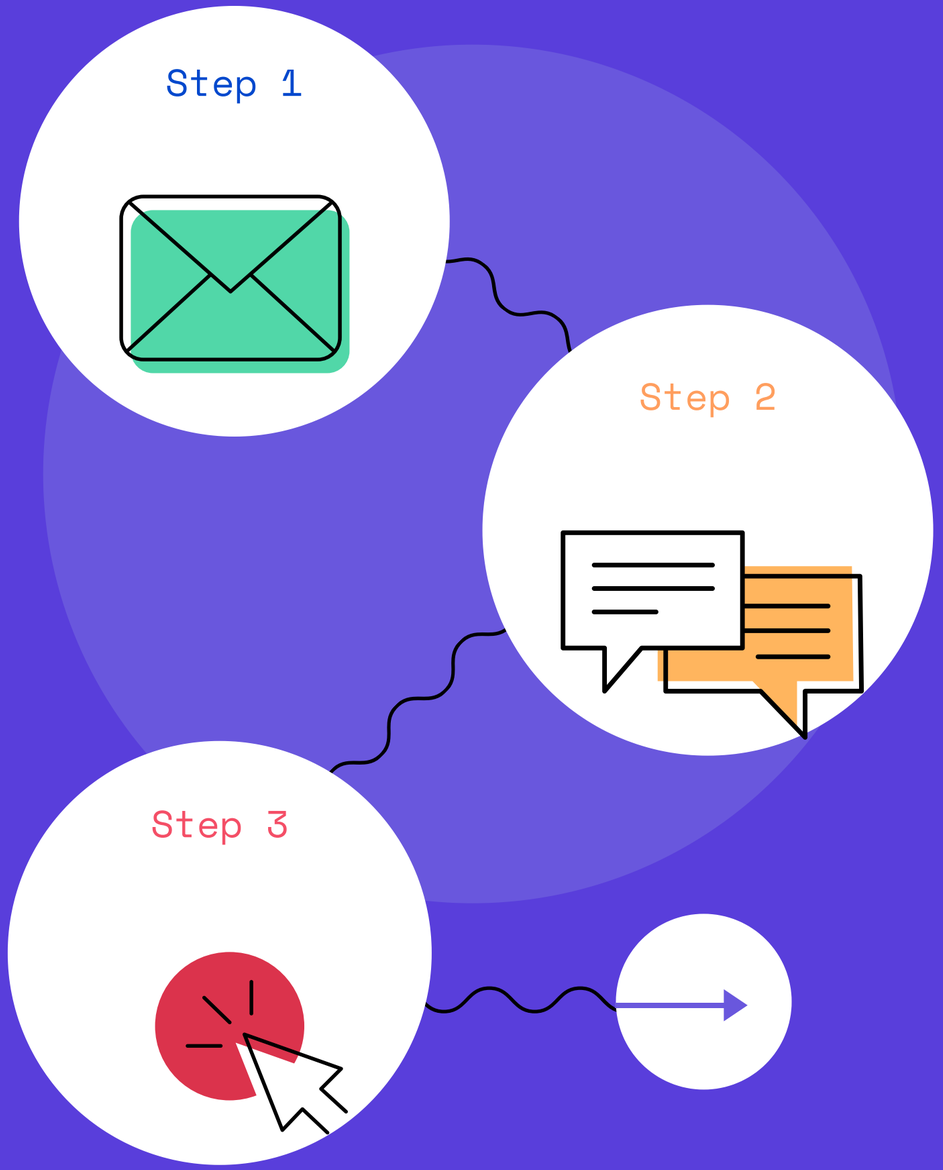
1 Gồm 8 cột, bao gồm các thông tin về đơn hàng và khách hàng, phù hợp với mô hình RFM

2 Gồm 541909 dòng dữ liệu, cột ID khách hàng và tên hàng có giá trị null

3 Kiểu dữ liệu của cột ngày đặt hàng chưa phù hợp

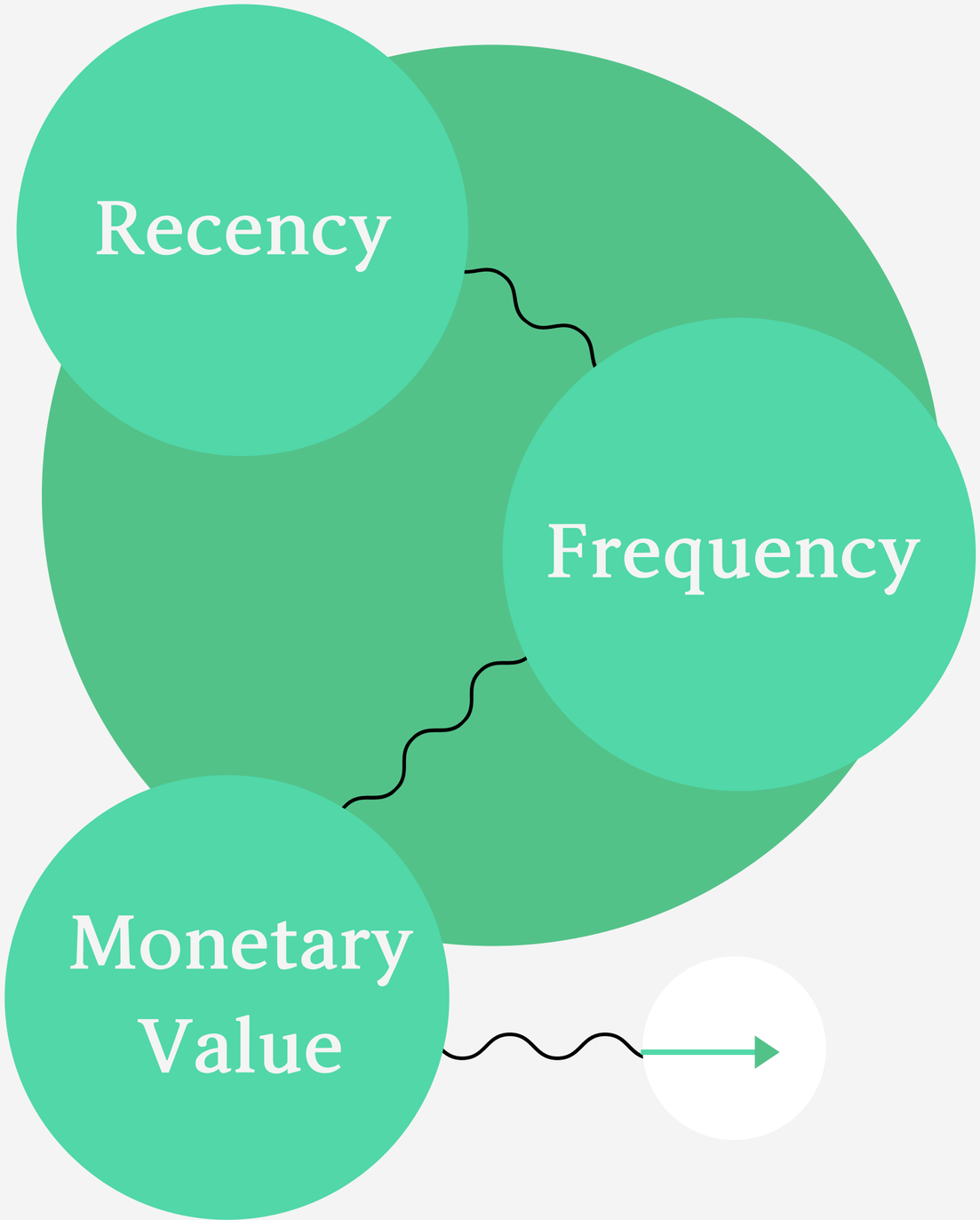
# Data cleaning

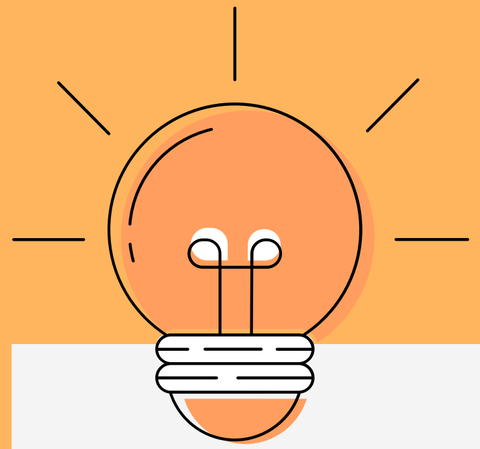
- Xóa các dòng dữ liệu bị thiếu ID khách hàng do đây là dữ liệu quan trọng và không thể điền bằng các giá trị khác
- Xóa các dòng dữ liệu trùng lặp
- Đưa dữ liệu ngày tháng về đúng định dạng



# K-means

Tiến hành phân lớp
theo từng tiêu chí





# Recency

1 Tạo bảng recency gồm CustomerID và ngày đặt hàng gần nhất

2 Tính toán thời gian đặt hàng gần nhất

3 Sử dụng Elbow Test để tìm số phân khúc phù hợp và build model

# Recency

Tạo bảng recency gồm CustomerID và ngày đặt hàng gần nhất

```
recency= df.groupby("CustomerID").InvoiceDate.max().reset_index()
```

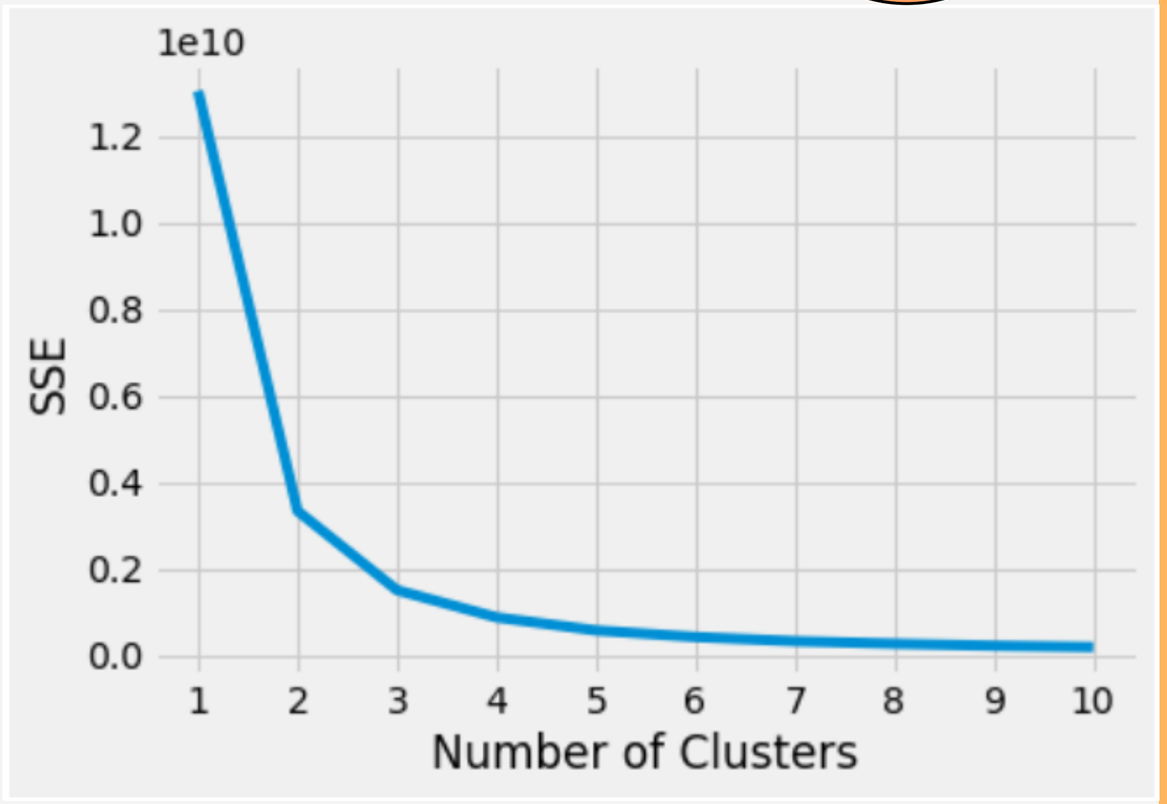
Tính toán thời gian đặt hàng gần nhất

```
recency["Recency"]=(recency["InvoiceDate"].max()-recency["InvoiceDate"]).dt.days
```

| | CustomerID | Recency |
|---|---|---|
| **0** | 12346.0 | 326 |
| **1** | 12347.0 | 40 |
| **2** | 12348.0 | 76 |
| **3** | 12349.0 | 19 |
| **4** | 12350.0 | 311 |

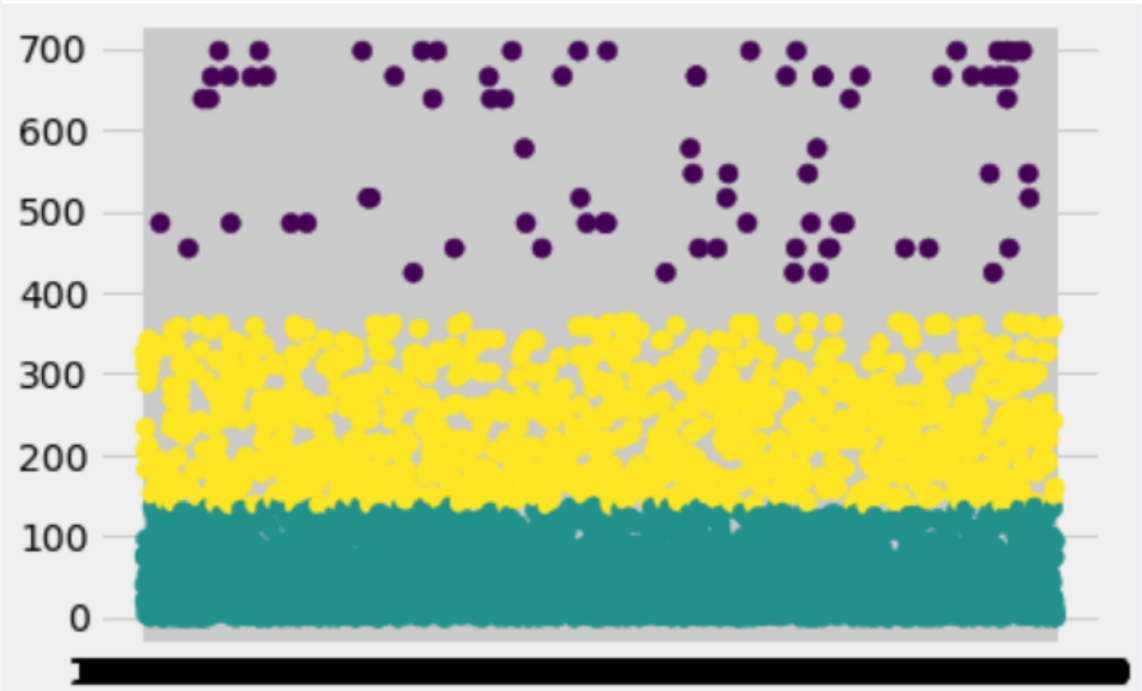
# Recency

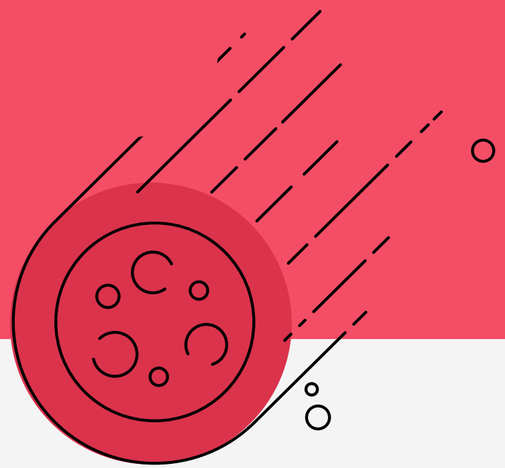
Sử dụng Elbow Test.
Kết quả cho thấy 3 là
số phân khúc phù hợp.



# Recency

Kết quả phân lớp theo Recency với màu xanh là nhóm khách hàng tiềm năng, có thời gian mua gần nhất sớm; tiếp đến là màu vàng; cuối cùng là màu tím.





# Frequency

1 Tạo bảng frequency gồm CustomerID và số lần đặt hàng

2 Sử dụng Elbow Test để tìm số phân khúc phù hợp

3 Build model

# Frequency

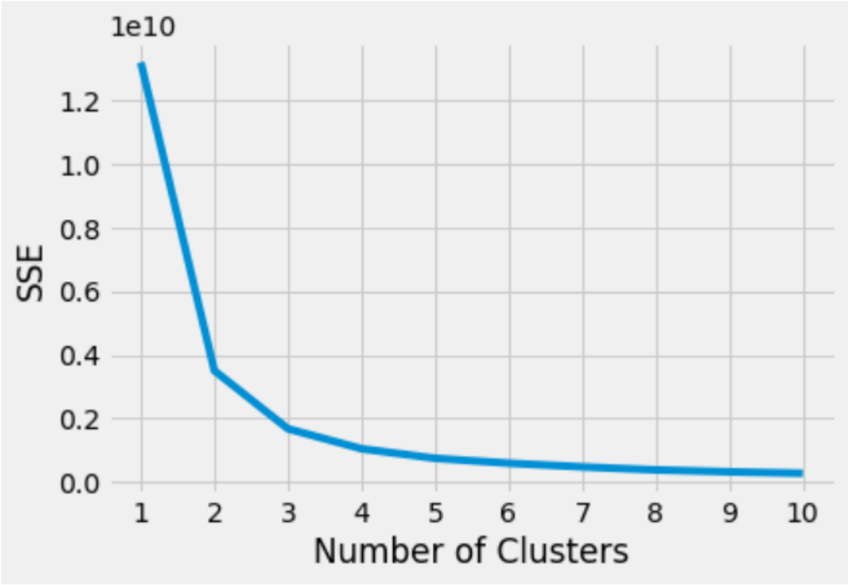
Tạo bảng frequency gồm CustomerID và số lần đặt hàng

```
frequency= df.groupby('CustomerID').InvoiceDate.count().reset_index()
frequency.columns = ['CustomerID','Freq']
```

| | CustomerID | Freq |
|---|---|---|
| **0** | 12346.0 | 2 |
| **1** | 12347.0 | 182 |
| **2** | 12348.0 | 31 |
| **3** | 12349.0 | 73 |
| **4** | 12350.0 | 17 |

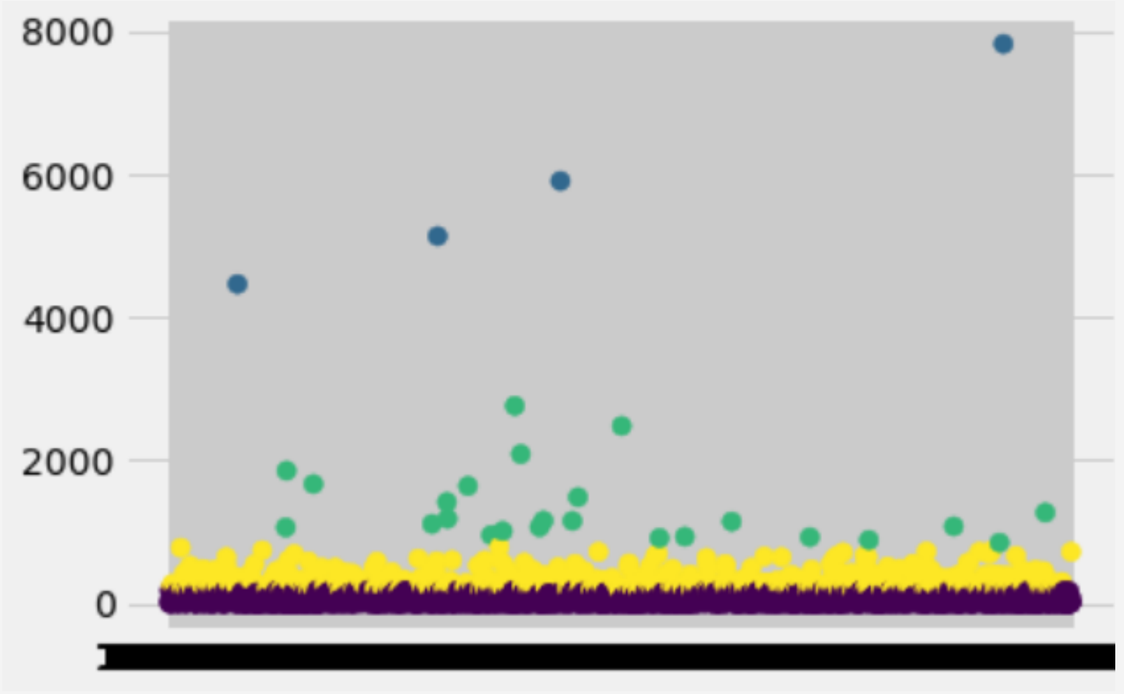
# Frequency

Sử dụng Elbow test. Kết quả cho thấy 4 phân khúc là phù hợp



# Frequency

Kết quả phân lớp theo Frequency với màu lam là nhóm khách hàng có tần suất mua hàng sớm nhất; tiếp đến là màu lục, màu vàng; cuối cùng là màu tím.



# Revenue

1 Tạo bảng revenue gồm CustomerID và tổng giá trị các đơn hàng của họ

2 Sử dụng Elbow Test để tìm số phân khúc phù hợp

3 Build model

# Revenue

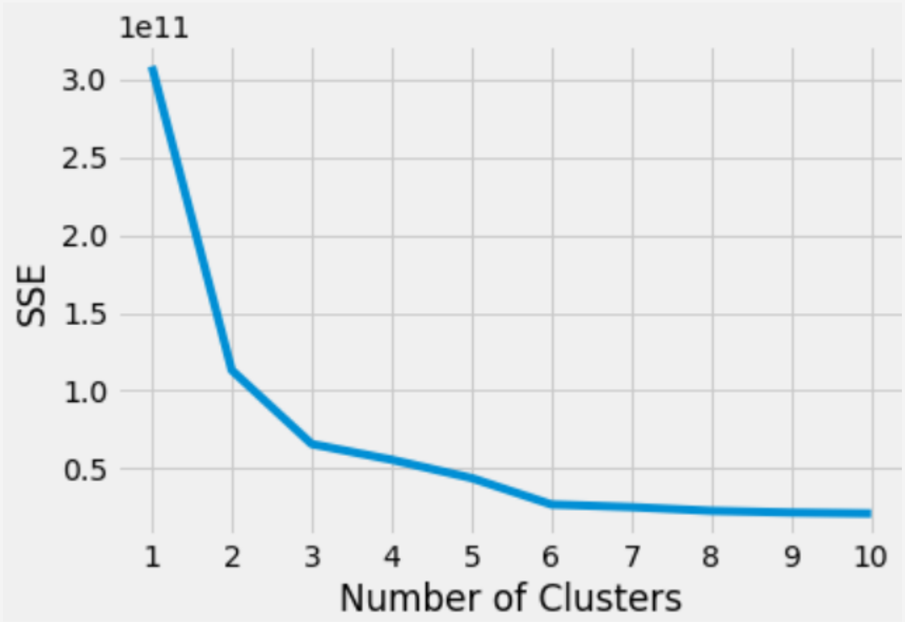
Tạo bảng revenue gồm CustomerID và tổng giá trị các đơn hàng

```
df['Revenue']=df['Quantity']*df['UnitPrice']
revenue=df.groupby('CustomerID').Revenue.sum().reset_index()
revenue.columns = ['CustomerID','Rev']
```

| | CustomerID | Rev |
|---|---|---|
| 0 | 12346.0 | 0.00 |
| 1 | 12347.0 | 4310.00 |
| 2 | 12348.0 | 1797.24 |
| 3 | 12349.0 | 1757.55 |
| 4 | 12350.0 | 334.40 |

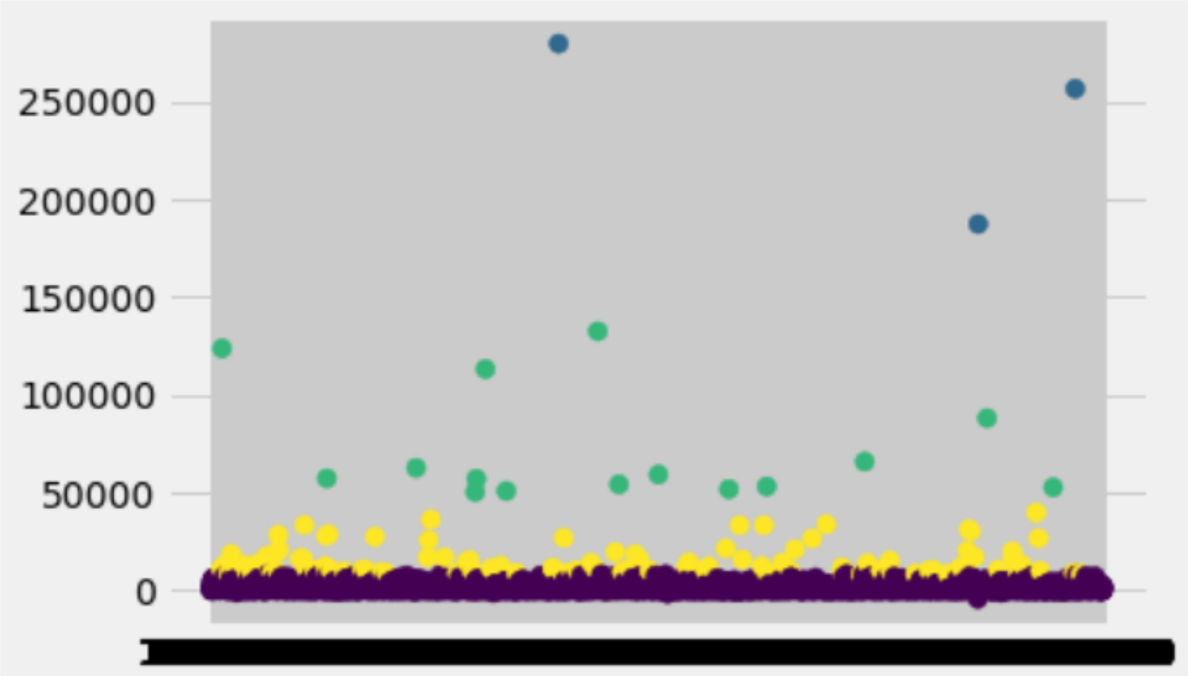
# Revenue

Sử dụng Elbow test. Kết quả cho thấy 4 phân khúc là phù hợp



# Revenue

Kết quả phân lớp theo Revenue với màu lam là nhóm khách hàng mang lại doanh thu cao nhất; tiếp đến là màu lục, màu vàng; thấp nhất là màu tím.



# Overall

1 Tạo bảng overall tổng hợp các kết quả trên, và cột Overall bằng tổng các giá trị phân lớp qua 3 tiêu chí.

2 Tiếp tục phân chia kết quả cột Overall thành 3 phân khúc High-Mid-Low

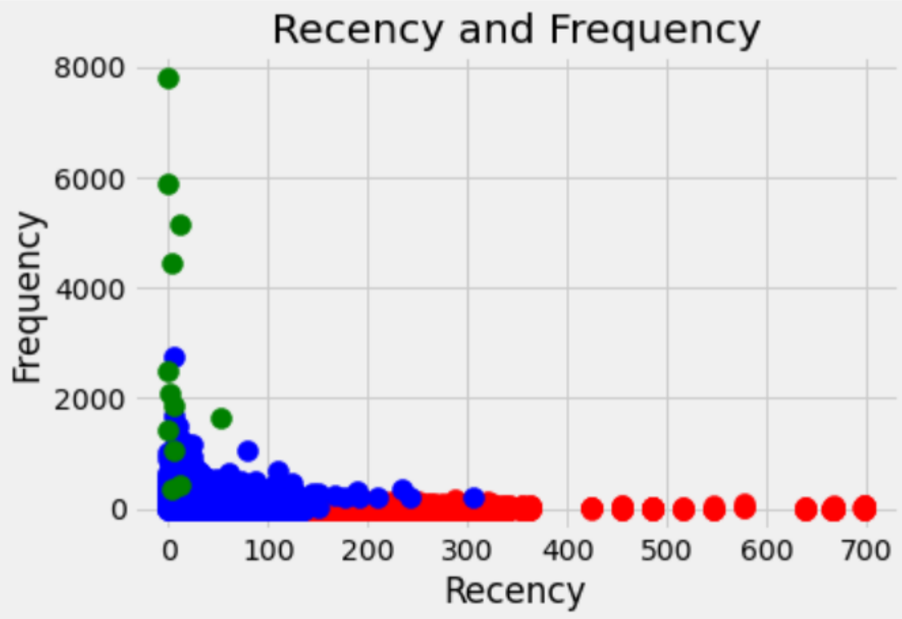
3 Plot kết quả phân chia để đánh giá lại hiệu quả.

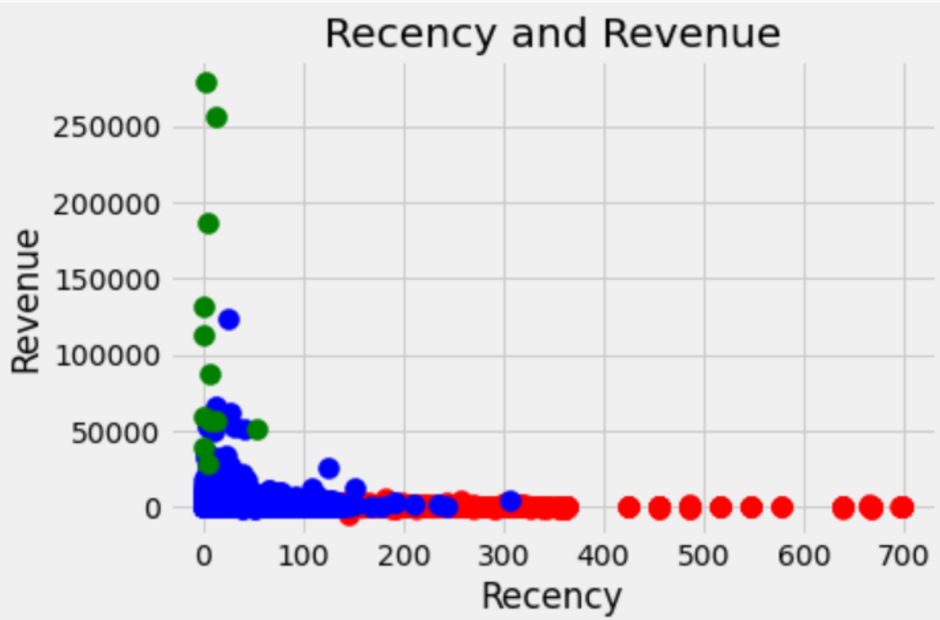
# Overall

Tạo bảng overall tổng hợp các kết quả ở trên. Cột overall là tổng điểm các phân khúc, với điểm 7 là các khách hàng có tiềm năng nhất, 0 là các khách hàng ít tiềm năng nhất. Các khách hàng có điểm 6,7 được chia vào phân khúc High, 2-5 chia vào phân khúc Mid, 0 và 1 chia vào phân khúc Low.

| Overall | Recency | Freq | Rev |
|---|---|---|---|
| 0 | 578.753086 | 19.135802 | 218.129630 |
| 1 | 230.368421 | 29.608893 | 455.510018 |
| 2 | 48.195865 | 58.354887 | 1100.699595 |
| 3 | 25.516432 | 296.401408 | 4364.829930 |
| 4 | 15.137500 | 511.025000 | 14165.706000 |
| 5 | 19.909091 | 947.727273 | 44402.842727 |
| 6 | 9.333333 | 2390.666667 | 97871.555556 |
| 7 | 4.666667 | 4370.333333 | 156356.220000 |

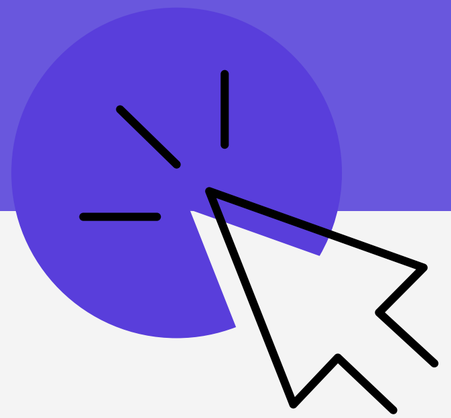
# Overall



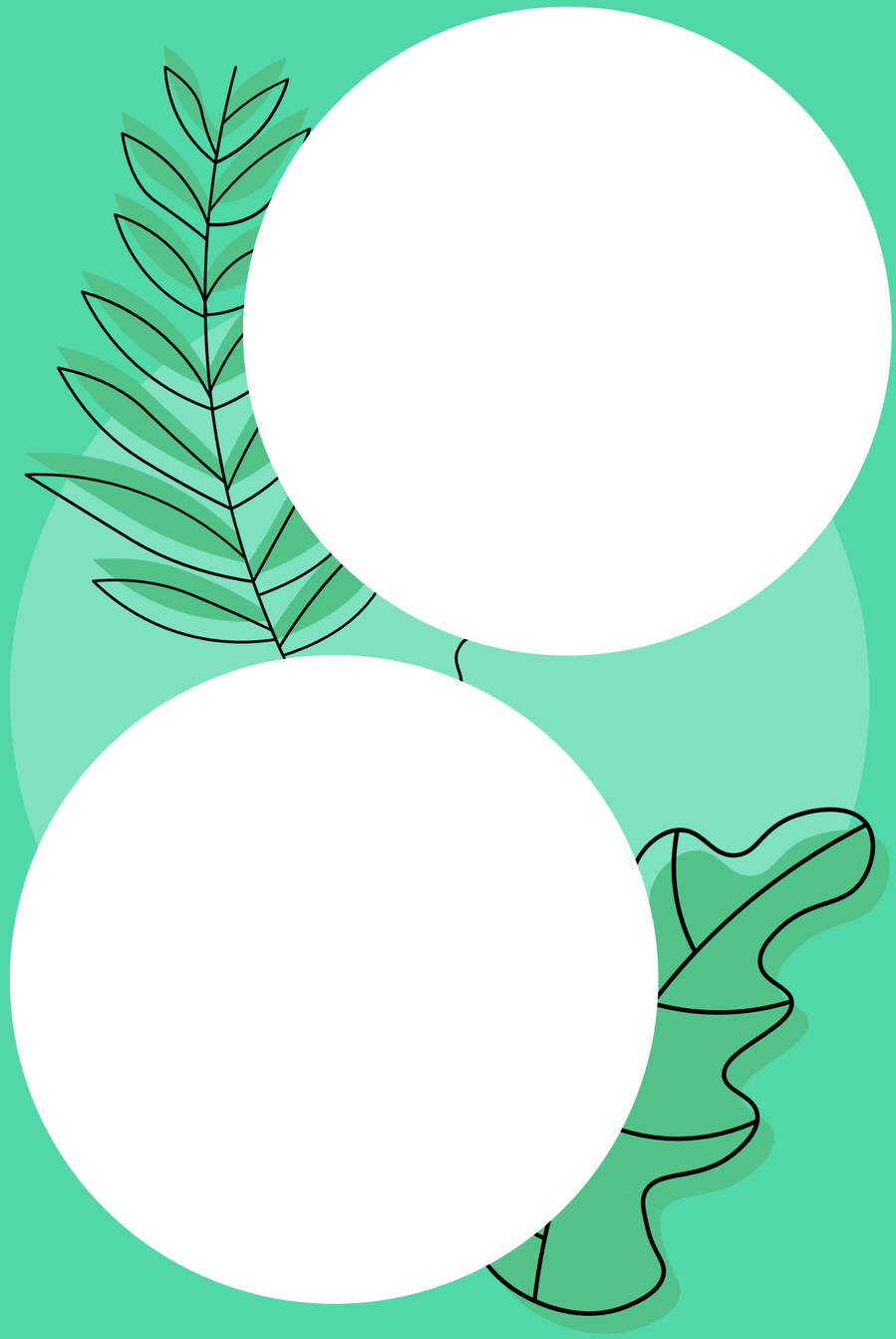

- Low
- Mid
- High

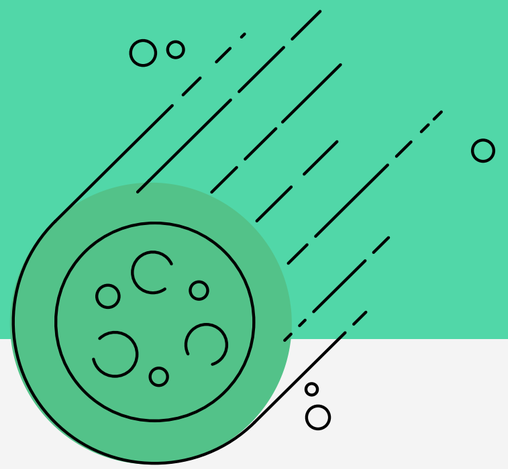
# Overall



Kết quả phân lớp tổng thể khá phù hợp và hiệu quả. Nhóm khách hàng High được phân chia có tần suất mua lớn nhất, thời gian mua hàng gần đây nhất và đem lại nhiều doanh thu nhất. Các yếu tố trên giảm dần với nhóm Mid và Low.

# Hierarchical Clustering





# Hierarchical Clustering

1 Tạo bảng tổng hợp recency, frequency, revenue

2 Vẽ dendrogram, xác định số phân khúc

3 Build model

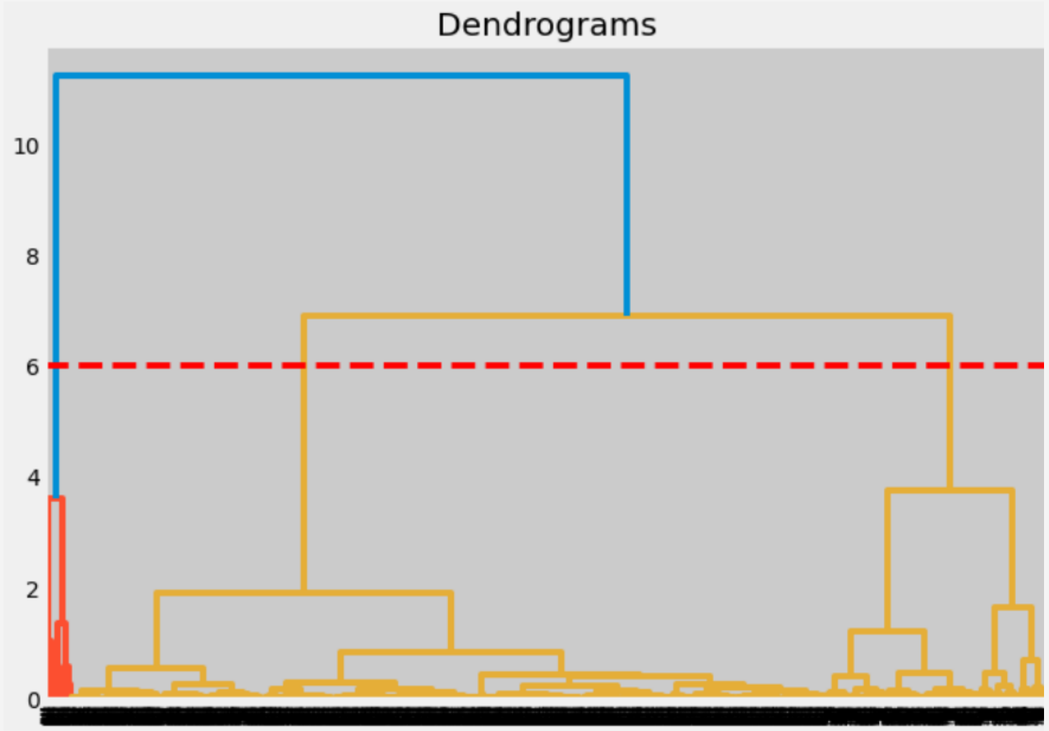
# Hierarchical Clustering

Tạo bảng tổng hợp recency, frequency, revenue. Normalize dữ liệu.

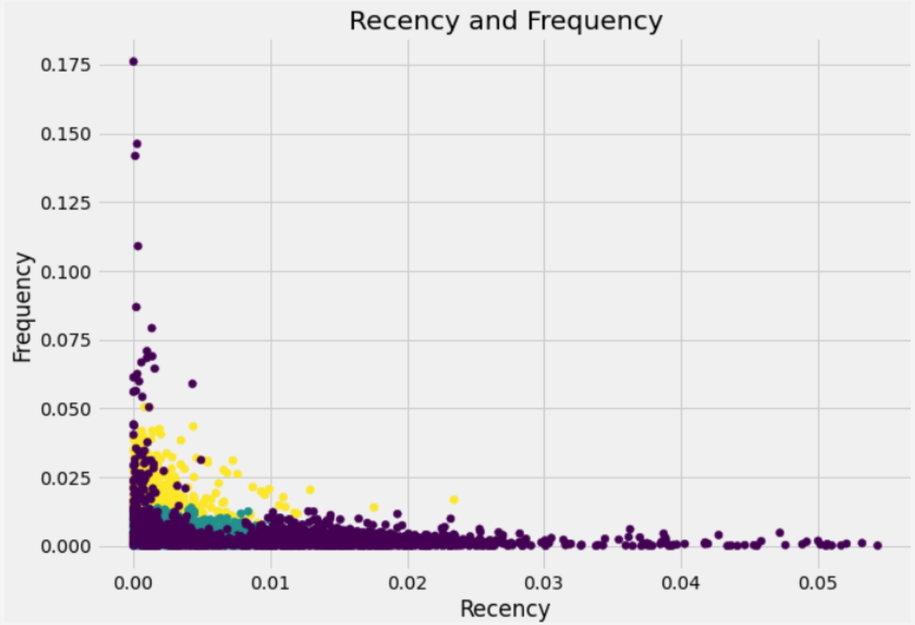
| | CustomerID | Recency | Freq | Rev |
|---|---|---|---|---|
| **0** | 0.958446 | 0.003705 | 0.016592 | 0.284767 |
| **1** | 0.973154 | 0.003506 | 0.014619 | 0.229665 |
| **2** | 0.868200 | 0.000000 | 0.017318 | 0.495911 |
| **3** | 0.945307 | 0.000000 | 0.022912 | 0.325375 |
| **4** | 0.890442 | 0.000143 | 0.017120 | 0.454776 |

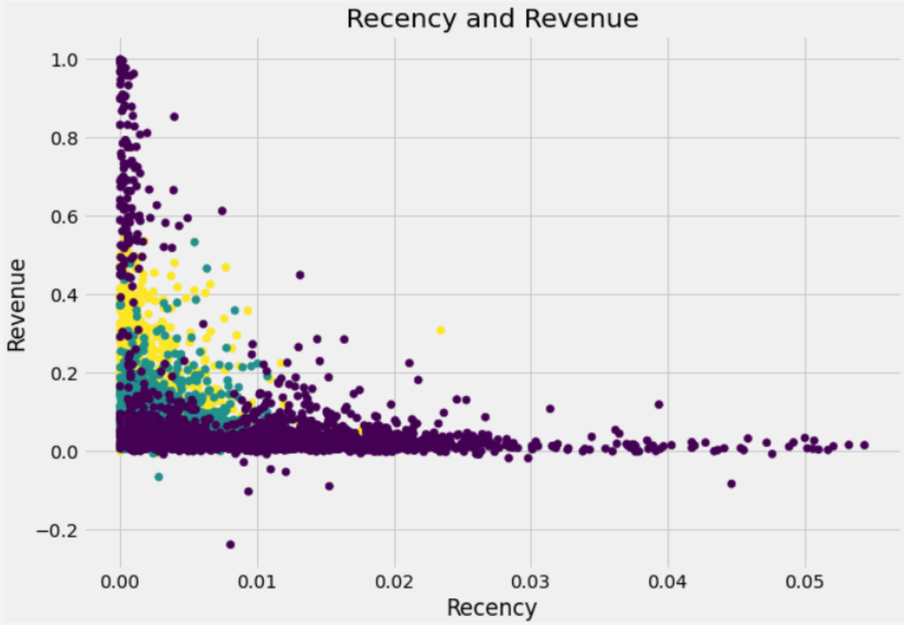
# Hierarchical Clustering

Vẽ dendrogram, xác định được 3 phân khúc khách hàng



# Hierarchical Clustering

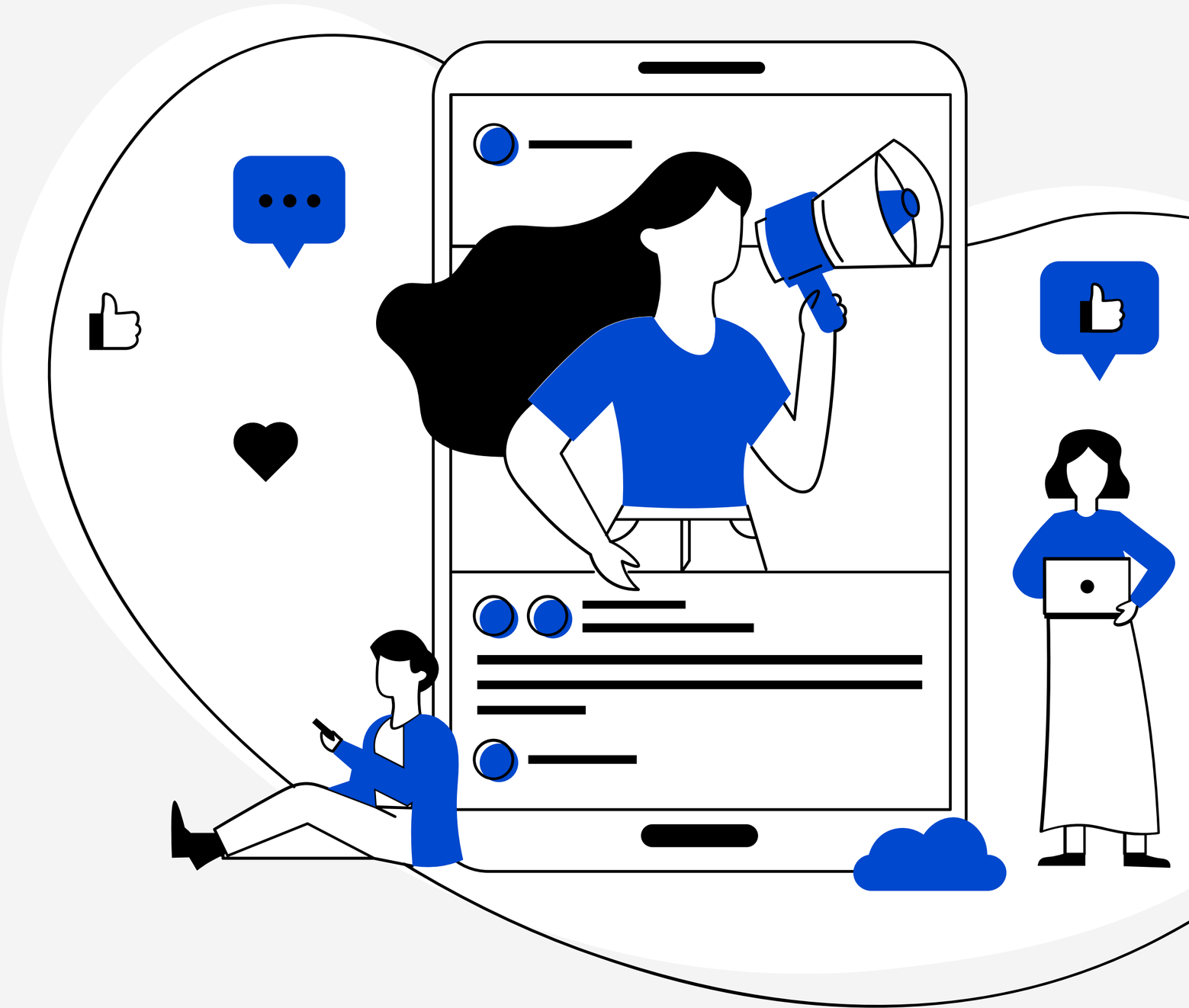


# Hierarchical Clustering

Không có sự phân biệt rõ ràng giữa các nhóm khách hàng. Kết quả thu được kém hiệu quả hơn so với mô hình K-means

# Thank you!



PITCH DECK v01