ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KĨ THUẬT THÔNG TIN

ĐỒ ÁN CUỐI KÌ MÔN HỌC-DS102.L21.

Đề tài:

XÂY DỰNG MÔ HÌNH PHÂN KHÚC KHÁCH HÀNG DỰA TRÊN LỊCH SỬ MUA HÀNG, ĐỘ TUỔI, SỞ THÍCH CỦA KHÁCH HÀNG.

Giảng viên hướng dẫn : **Võ Duy Nguyên.**

Sinh viên thực hiện 1: **Trần Nhật Nam - 19521872.**

Lớp **: KHDL 2019**

Khoá **: 2019 – 2023.**

Sinh viên thực hiện 2: **Trần Thành Luân – 19521810.**

Lớp **: KHDL 2019**

Khoá **: 2019 – 2023.**

TP. Hồ Chí Minh, tháng … năm 2021.

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KĨ THUẬT THÔNG TIN

ĐỒ ÁN CUỐI KÌ MÔN HỌC-DS102.L21.

Đề tài:

XÂY DỰNG MÔ HÌNH PHÂN KHÚC KHÁCH HÀNG DỰA TRÊN LỊCH SỬ MUA HÀNG, ĐỘ TUỔI, SỞ THÍCH CỦA KHÁCH HÀNG.

Giảng viên hướng dẫn : **Võ Duy Nguyên.**

Sinh viên thực hiện 1: **Trần Nhật Nam - 19521872.**

Lớp **: KHDL 2019**

Khoá **: 2019 – 2023.**

Sinh viên thực hiện 2: **Trần Thành Luân – 19521810.**

Lớp **: KHDL 2019**

Khoá **: 2019 – 2023.**

TP. Hồ Chí Minh, tháng … năm 2021.

LỜI NÓI ĐẦU

Như chúng ta đã biết, dữ liệu là tất cả mọi thứ trong thế giới công nghệ ngày nay. Hơn nữa, dữ liệu này tiếp tục nhân lên bởi đa tạp mỗi ngày. Trước đó, chúng ta thường xuyên nói về kilobyte và megabyte. Nhưng ngày nay, chúng ta đang nói về terabyte, Dữ liệu là vô nghĩa cho đến khi nó biến thành thông tin và kiến thức hữu ích có thể hỗ trợ an quản lý trong việc đưa ra quyết định.

Và trong quá trình biến những dữ liệu đó thành thông tin thì Học máy (Machine Learning) đóng một nhiệm vụ vô cùng to lớn, nó là lĩnh vực nghiên cứu khoa học về các thuật toán và mô hình thống kê mà các hệ thống máy tính sử dụng để thực hiện một nhiệm vụ cụ thể không cần sử dụng các hướng dẫn rõ ràng, thay vào đó là dựa vào các mẫu và suy luận. Học máy được xem như một tập con của trí tuệ nhân tạo (AI - Artificial Intelligence). Các thuật toán học máy xây dựng một mô hình toán học dựa trên dữ liệu mẫu được gọi là "dữ liệu huấn luyện", để đưa ra dự đoán hoặc quyết định mà không phải lập trình cụ thể để thực hiện nhiệm vụ.

Là một sinh viên chuyên ngành Khoa Học Dữ Liệu – Trường Đại học Công Nghệ Thông tin. Việc được cung cấp những kiến thức khai thác và thu thập dữ liệu từ cơ bản đến nâng cao đã giúp em nâng cao trình độ hiểu biết của minh. Với bộ môn Học Máy Thống Kê chúng em đã có thể vận dụng các kiến thức đã học vào một số vấn đề. Và từ những kiến thức đã học được chúng em xin được đúc kết trong một bản đồ án báo cáo của mình.

LỜI CẢM ƠN

Trong thời gian làm đồ án môn học, chúng em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy cô và bạn bè.

Em xin gửi lời cảm ơn chân thành đến giảng viên Th.s Nguyễn Tấn Trần Minh Khang , trợ giảng Võ Duy Nguyên và Hồ Trần Ngọc Bộ môn Học máy thống kê- Trường Đại học Công Nghệ Thông Tin-Đại học Quốc gia thành phố Hồ Chí Minh người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm khoá luận.

Em cũng xin chân thành cảm ơn các thầy cô giáo trong trường Đại học Công Nghệ Thông Tin nói chung, các thầy cô trong Bộ môn Học máy thống kê nói riêng đã dạy dỗ cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ em trong suốt quá trình học tập.

Cuối cùng, em xin chân thành cảm ơn gia đình và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên em trong suốt quá trình học tập và hoàn thành đồ án.

NHẬN XÉT

(Giảng viên hướng dẫn : **Võ Duy Nguyên.**)

MỤC LỤC

MỞ ĐẦU *6*

DANH MỤC CÁC BẢNG *9*

DANH MỤC CÁC HÌNH, SƠ ĐỒ *10*

CHƯƠNG 1: TỔNG QUÁT VẤN ĐỀ *11*

1.1 Giới thiệu tổng quát về đề tài. **11**

1.2 Mục tiêu đề tài **11**

1.3 Phương pháp nghiên cứu **12**

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT *13*

2.1 Phương pháp đánh giá số cụm **13**

2.1.1 Biểu đồ dendrogram 13

2.1.2 Elbow 14

2.2 Phương pháp phân cụm K-Mean **16**

2.3 Thuật toán huấn luyện mô hình **16**

2.3.1 Naive Bayes Classifier 16

2.3.2 Kernel Support Vector Machine 18

2.3.3 Random forest classification 19

2.4 Phương pháp đánh giá mô hình **20**

2.4.1 Accuracy 20

2.4.2 Confusion matrix 20

CHƯƠNG 3: BỘ DỮ LIỆU *22*

3.1 Codebook **22**

Dữ liệu khách hàng của trung tâm mua sắm (Mall_customers) *22*

3.2 Mô tả bộ dữ liệu **23**

3.3 Kiểm tra bộ dữ liệu **24**

3.4 Trực quan hóa dữ liệu từng thuộc tính **24**

3.4.1 Phân tích thuộc tính Gender 24

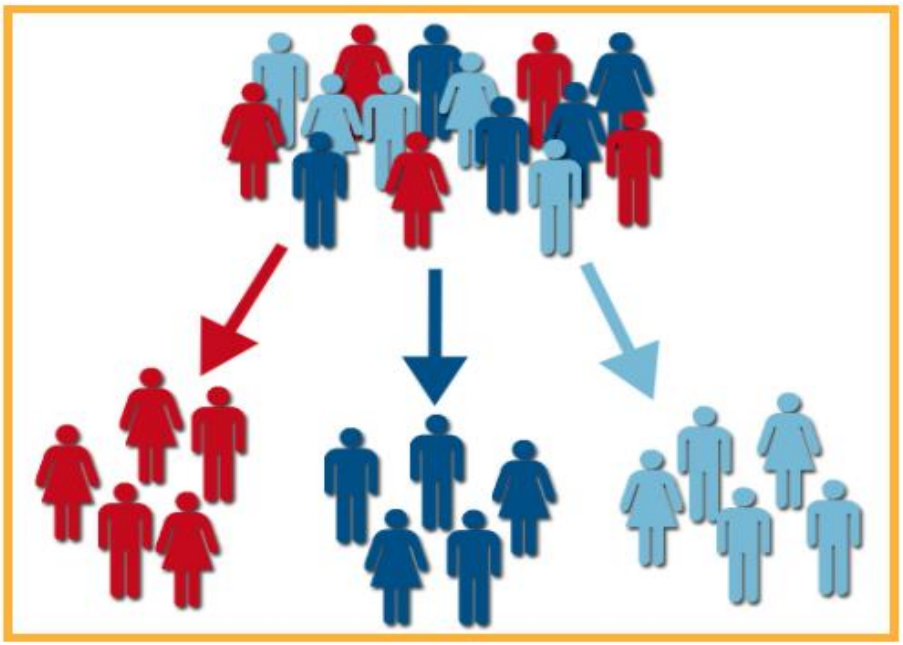
3.4.2 Phân tích thuộc tính Age 25

3.4.3 Phân tích thuộc tính Annual Income Analysis....27
3.4.4 Phân tích thuộc tính Spending Score....28
CHƯƠNG 4: PHÂN KHÚC KHÁCH HÀNG....29
4.1 Xây dựng mô hình phân lớp với 3 thuộc tính Age, Annual Income, Spending Score 29
4.1.1 Xác định số nhóm thích hợp....29
4.1.2 Áp dụng thuật toán Kmean để huấn luyện mô hình....30
4.2 Tìm thuộc tính tiêu biểu....31
4.2.1 Phân tích mối tương quan của 2 thuộc tính Age và Annual Incom với thuộc tính Spending Core....31
4.3 Xây dựng mô hình phân lớp với 2 thuộc tính tiêu biểu....32
4.3.1 Tìm số cụm k tối đa....32
4.3.2 Huấn luyện phân lớp với thuật toán Kmean....34
CHƯƠNG 5: DỰ ĐOÁN PHÂN KHÚC KHÁCH HÀNG....36
5.1 Bộ dữ liệu....36
5.2 Tiền xử lý....37
5.3 Huấn luyện mô hình....37
5.4 Đánh giá mô hình....38
5.4.1 Naive Bayes....38
5.4.2 Random Forest....40
5.4.3 Kernel SVM....42
CHƯƠNG 6: KẾT LUẬN....45
TÀI LIỆU THAM KHẢO....47

MỞ ĐẦU

Hiện nay, các doanh nghiệp không ngừng cạnh tranh nhau để mà phát triển được giữa một rừng các doanh nghiệp cũ và mới như bây giờ. Ngoài việc tập trung vào chất lượng sản phẩm, dịch vụ thì việc phân tích dữ liệu kinh doanh để mà đưa ra các định hướng phát triển trong tương lai là vô cùng quan trọng. Tìm hiểu hành vi, ghi nhận thói quen mua sắm, nắm bắt sở thích khách hàng ... luôn được các doanh nghiệp đầu tư bài bản nhằm tạo ra lợi thế cạnh tranh lâu dài. Những năm gần đây, việc phát triển Trung tâm thương mại (TTTM) đang trở thành xu hướng bởi nó mang đến hàng loạt lợi ích cho cộng đồng, đóng góp tích cực cho sự phát triển ở địa phương cũng như là trung gian không thể thiếu trong kênh bán lẻ hiện đại. Sự phát triển kinh tế khiến các thành phố trở nên chật chội, không gian vui chơi bị bó hẹp, TTTM dần trở thành địa điểm thường xuyên lui tới của nhiều gia đình, bởi sự tiện nghi, sang trọng và mới mẻ mà nó mang lại. Vì vậy, cần phải hiểu được đặc điểm thúc đẩy người tiêu dùng đến thăm quan, mua sắm tại các TTTM và phân khúc của họ dựa trên những đặc điểm để có thể có một cái nhìn sâu sắc hữu ích cho các nhà quản lý TTTM. Như vậy việc phân khúc khách hàng mang lại rất nhiều các lợi ích cho doanh nghiệp. Chính vì thế xuất pháp từ bài toán này sau đi sâu vào nghiên cứu và thực nghiệm bộ dữ liệu chúng tôi xin được chọn đề tài " Xây dựng mô hình phân khúc khách hàng dựa trên độ tuổi, sở thích của khách hàng".

Customer Segmentation



Trong bài báo này chúng tôi sẽ dựa theo bộ dữ liệu có sẵn của một trung tâm thương mại với các dữ liệu cơ bản của khách hàng như ID khách hàng, độ tuổi, giới tính, thu nhập hàng năm và điểm chi tiêu. Điểm chi tiêu được chỉ định cho khách hàng dựa trên một số thông số như hành vi của khách hàng và dữ liệu mua hàng. Sau đó chúng tôi sẽ nghiên cứu từng thuộc tính từ đó đưa ra các thuộc tính tiêu biểu để phân khúc khách hàng, và sử dụng các phương pháp đánh giá như biểu đồ Elbow, Dendrogram để tìm ra số cụm tối ưu. Thuật toán phân lớp cơ bản và tối ưu nhất trong bài toán này là Kmean. Sau khi phân cụm ta được bộ dữ liệu có thêm thuộc tính " Clustering" và ta có bộ dữ liệu mới. Sau đó chúng ta sẽ xây dựng mô hình và huấn luyện dựa vào các thuật toán như : Kernel SVM, Naïve Bayes, Random Forest Classification để dự đoán thuộc tính "Clustering". Cuối cùng thì sẽ sử dụng các phương pháp đánh giá như Confusion matrix và accuracy để đánh giá độ chính xác và cuối cùng đưa ra kết quả chính xác nhất.

Customer Segmentation with Market Basket Analysis

DANH MỤC CÁC BẢNG

Bảng 3.1. CodeBook của bộ dữ liệu 23
Bảng 3.2. Mô tả tổng quan về bộ dữ liệu 23
Bảng 4.2.1.1. Độ tương quan giữa các thuộc tính 32
Bảng 5.4.1. Classification Report- NB trên tập Test 39
Bảng 5.2: Classification Report- RF trên tập Test 41
Bảng 5.4.3. Classification Report- Kernel SVM trên tập Test 43
Bảng 6.1. Bảng so sánh kết quả độ đo các mô hình phân loại 45

DANH MỤC CÁC HÌNH, SƠ ĐỒ

Hình 3.1.1. Đọc dữ liệu từ file csv 22
Hình 3.1.2. Dữ liệu quan sát được từ môi trường colab 22
Hình 3.4.1. Đồ thị phân bố giới tính khách hàng 25
Hình 3.4.2.1. Phân bố của khách hàng theo độ tuổi 26
Hình 3.4.2.2. Phân bố của khách hàng theo phân nhóm độ tuổi 26
Hình 3.4.4. Phân bố của khách hàng theo thu nhập bình quân 27
Hình 3.4.4.1. Biểu đồ số lượng khách hàng của từng điểm mua sắm 28
Hình 3.4.4.2. Biểu đồ số lượng khách hàng theo từng nhóm điểm 28
Hình 4.1.1. Đồ thị khuỷu tay ứng với 3 thuộc tính 30
Hình 4.1.2. Phân lớp khách hàng trong không gian 3 chiều 31
Hình 4.2.1.2. Trực quan hóa các điểm theo các cặp thuộc tính 32
Hình 4.2.1.2.Biểu đồ dendrogram cho 2 thuộc tính 33
Hình 4.2.1.3. Đồ thị khuỷu tay cho 2 thuộc tính 33
Hình 4.2.2.1. Huấn luyện với mô hình Kmean 34
Hình 4.2.2.2. Phân bố của từng lớp khách hàng 34
Hình 4.2.2.3. Tạo file Custer_Customer 35
Hình 5.1.1. Đọc dữ liệu từ file csv 36
Hình 5.1.2. 5 điểm dữ liệu đầu tiên của bộ dữ liệu 36
Hình 5.1.3. Kiểm tra giá trị thiếu của bộ dữ liệu 36
Hình 5.2. Phân chia bộ dữ liệu thành test và train với tỷ lệ 2:8 37
Hình 5.2. Huấn luyện 3 mô hình với tham số mặc định 37
Hình 5.4.1 Biểu đồ giá trị độ đo Classification Report - NB trên tập Test 40
Hình 5.2 Biểu đồ giá trị độ đo Classification Report- RF trên tập Test 42
Hình 5.4.3 Biểu đồ giá trị độ đo Classification Report- Kernel SVM trên tập Test 44

CHƯƠNG 1: TỔNG QUÁT VẤN ĐỀ.

1.1 Giới thiệu tổng quát về đề tài.

Theo nguyên lý Pareto 20% khách hàng sẽ mang lại 80% lợi nhuận cho công ty. Do đó doanh nghiệp cần phải xác định được khách hàng quan trọng chú trọng quan tâm. Bên cạnh đó mỗi phân khúc khách hàng có đặc điểm khác nhau, nhu cầu khác nhau doanh nghiệp cần phải xác định, nhắm tới từng phân khúc để thiết kế sản phẩm, chiến dịch quảng cáo phù hợp. Lấy một vài ví dụ đơn giản:

- Giải sử công ty của bạn kinh doanh xe từ xe đạp đến ô tô. Bạn không thể nói rằng công ty của bạn đã chăm sóc tốt khách hàng nếu như công ty đó đối xử với những khách hàng mua một chiếc xe hơi ngang một chiếc xe đạp.
- Bạn không thể bán một căn chung cư hạng sang cho một người công nhân lương tháng 5 triệu được.
- Bạn cũng khó mà yêu cầu một người trung niên hay lớn tuổi mua một chiếc áo local brand hợp với tuổi teen được.

Qua các ví dụ cụ thể trên chúng ta có thể thấy:

- Chính sách chăm sóc khách hàng cần phải thay đổi để phù hợp với phân khúc.
- Chăm sóc khách hàng cần phải phù hợp với thu nhập.
- Chăm sóc khách hàng đồng thời cũng phải phù hợp với nhu cầu, thị hiếu của khách hàng.

Như vậy việc phân khúc khách hàng mang lại rất nhiều các lợi ích cho doanh nghiệp.

1.2 Mục tiêu đề tài

Mục tiêu đặt ra của bài toán là xây dựng một bộ dữ liệu mới các thông tin của khách hàng từ bộ dữ liệu mà TTTM đã thu thập sẵn. Từ đó, phân tích các thuộc tính để tìm ra số thuộc tính tiêu biểu và tối ưu để tiến hình phân cụm. Trước khi phân cụm ta sẽ dụng các công cụ để tham khảo số cụm ta có thể chia, sau đó tiến hành phân cụm. Khi phân cụm xong sẽ được bộ dữ liệu có nhãn. Áp dụng các mô hình học máy hiện đại để dự báo phân khúc của khách hàng.

1.3 Phương pháp nghiên cứu

Do đây là bài toán thuộc Machine Learning cho nên chúng tôi sẽ sử dụng python. Vì Python là ngôn ngữ được sử dụng phổ biến nhất trong Machine Learning, vậy nên trong bài báo này chúng tôi sẽ dựa vào python để thu thập, quan sát dữ liệu. Đồng thời sử dụng các lệnh trong python để phân tích nghiên cứu bài toán, đánh giá mô hình các thuật toán Deep Learning có sẵn.

Và môi trường được chúng tôi sử dụng là Google Colab, vì Google đã cung cấp Google Colab miễn phí có GPU để chạy code python (Machine learning) cho mục đích nghiên cứu. Ở trên môi trường Colab có cài sẵn các thư viện Machine Learning phổ biến như PyTorch, TensorFlow, Keras,.. Ngoài ra cũng có thể cài thêm thư viện để chạy nếu cần.

Các phương pháp chúng ta sẽ sử dụng là:

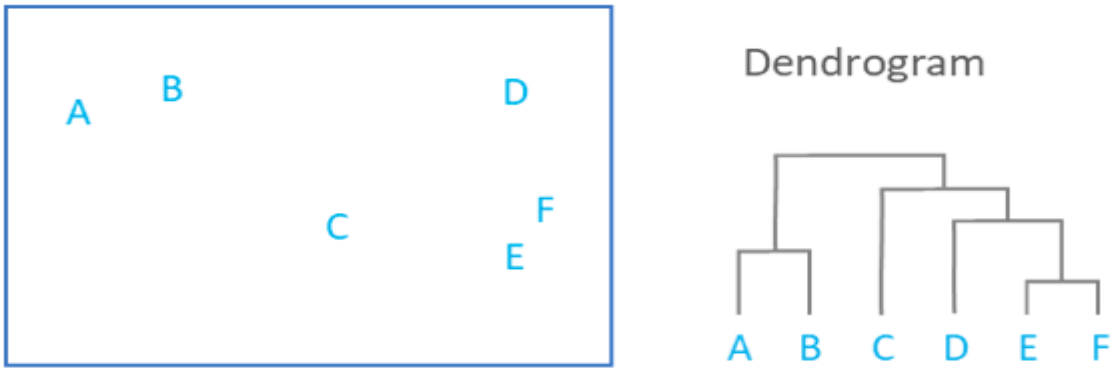
- Phương pháp đánh giá số cụm: **Elbow, Dendrogram.**
- Thuật toán phân cụm**: K- mean Clustering**.
- Thuật toán huấn luyện mô hình : **Kernel SVM, Naïve Bayes , Random Forest Classification**
- Phương pháp đánh giá mô hình : **Confusion matrix và Accuracy**

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Phương pháp đánh giá số cụm

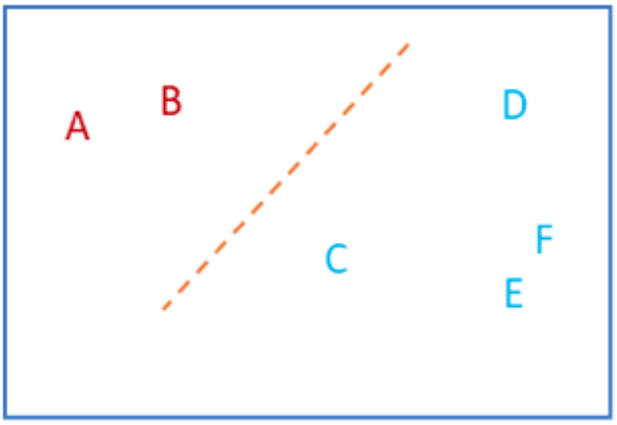
2.1.1 Biểu đồ dendrogram

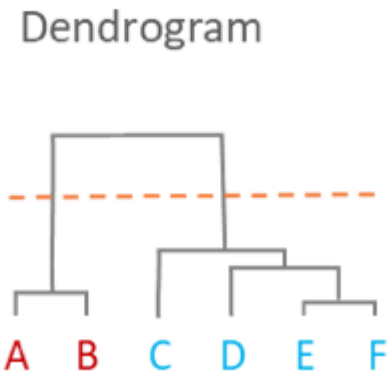
Một dendrogram là một sơ đồ cho thấy mối quan hệ thứ bậc giữa các đối tượng. Nó thường được tạo ra dưới dạng đầu ra từ phân cụm phân cấp . Công dụng chính của dendrogram là tìm ra cách tốt nhất để phân bổ các đối tượng vào các cụm. Hình biểu đồ dưới đây cho thấy sự phân nhóm có thứ bậc của sáu quan sát được hiển thị trên biểu đồ phân tán ở bên trái . (Dendrogram thường bị viết sai thành dedrogram.)



Chìa khóa để giải thích một biểu đồ dendrogram là tập trung vào độ cao mà tại đó hai đối tượng bất kỳ được nối với nhau. Trong ví dụ trên, chúng ta có thể thấy rằng E và F giống nhau nhất, vì độ cao của liên kết nối chúng với nhau là nhỏ nhất. Hai vật giống nhau nhất tiếp theo là A và B.

Các quan sát được phân bổ cho các cụm bằng cách vẽ một đường ngang qua biểu đồ hình ảnh. Các quan sát được nối với nhau bên dưới dòng này thành từng cụm. Trong ví dụ dưới đây, chúng ta có hai cụm, một cụm kết hợp A và B, và cụm thứ hai kết hợp C, D, E và F.





Dendrogram không thể cho bạn biết bạn nên có bao nhiêu cụm

Một sai lầm phổ biến mà mọi người thường mắc phải khi đọc dendrogram là cho rằng hình dạng của dendrogram cho biết có bao nhiêu cụm tồn tại. Trong ví dụ trên, cách giải thích là dendrogram cho thấy có hai cụm, vì khoảng cách giữa các cụm là cao nhất giữa hai và ba cụm.

Cách giải thích như vậy chỉ hợp lý khi tồn tại bất đẳng thức cây siêu vi lượng, như đã đề cập ở trên, là rất hiếm. Nói chung, thật sai lầm khi sử dụng dendrograms như một công cụ để xác định số lượng cụm trong dữ liệu. Ở những nơi có số lượng cụm rõ ràng là "chính xác", điều này thường được hiển thị rõ ràng trong một biểu đồ hình ảnh. Tuy nhiên, dendrograms thường đề xuất một số cụm chính xác khi không có bằng chứng thực tế để hỗ trợ kết luận.

2.1.2 Elbow

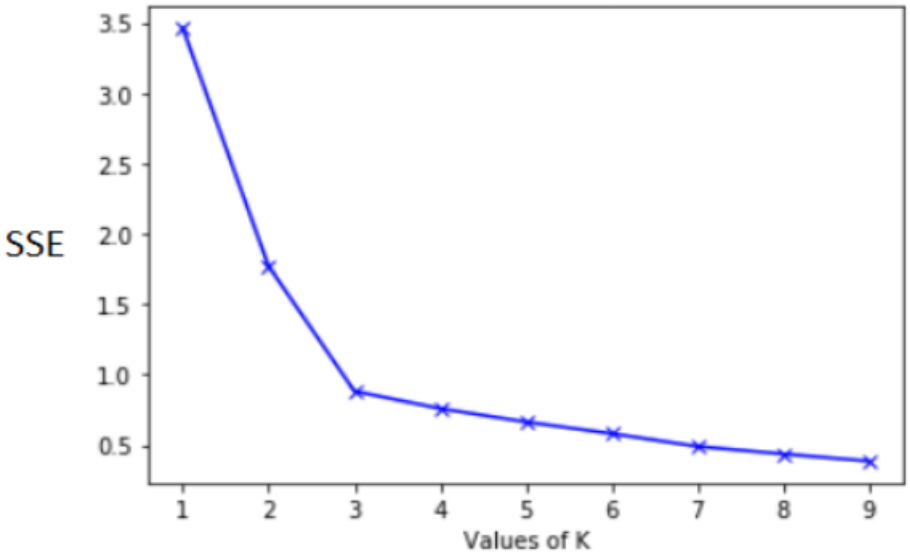
Elbow method là phương pháp xác định số k cho k-means clustering được coi là phổ biến nhất, bên cạnh các phương pháp khác như Pseudo F-statistic hay Silhouette index mà chúng tôi sắp trình bày dưới đây.

Elbow method được minh họa dưới dạng đồ thị đường cong với trục hoành là số k các cluster, trục tung sẽ là tiêu chí đánh giá bao gồm SSE, Silhouette. Ở phần này chúng ta tìm hiểu trước về SSE - Sum of Errors – đo lường sự khác biệt giữa các điểm trong cluster. Trong k-means clustering, SSE được tính là tổng các khoảng cách tính từ các điểm trong cluster đến điểm trung tâm Centroid của cluster, tính tất cả các cluster, dựa theo công thức Euclidean. Khi các điểm dữ liệu hay các object, các quan sát càng gần

nhau thì sẽ có đặc điểm gần giống nhau, được phân trong một cụm, thì cụm đó chứng tỏ "chất lượng" và ngược lại.

$$\text{SSE} = \sum_{i=1}^{k} \sum_{j=1}^{n_j} \text{Distance}^2(x_{ij}, m_i)$$

Sẽ có k cluster cần tính giá trị SSE thường k sẽ chạy từ 1 đến 10 hay 20. Như vậy với mỗi k chúng ta sẽ có 1 SSE. Minh họa các cặp k và SSE lên đồ thị. Số k tối ưu chính là điểm mà ở đó SSE bắt đầu giảm đều, nhìn trên đồ thị nó là điểm "turning point", nói với điểm nằm ở vị trí "Cùi chỏ" sẽ là số k cần tìm.



Nhìn hình minh họa giống khuỷu tay, k = 3 sẽ thích hợp là số cụm ban đầu cần phân của k-means clustering.

Giải thích: Elbow method dựa trên giả định khi càng có nhiều cluster, thì có nghĩa các điểm dữ liệu giống nhau đã được gom cụm, mỗi cụm sẽ chỉ có ít điểm dữ liệu bên trong, và những điểm này sẽ ko nằm xa nhau, do đó SSE sẽ giảm, khi k tăng. Tuy nhiên khi k càng tăng chúng ta sẽ càng có nhiều cluster cần phân tích, dẫn đến không hiệu quả. Do đó chúng ta nên chọn số k mà ở đó SSE bắt đầu giảm đều. Đây được xem là quy tắc chung khi sử dụng elbow method.

2.2 Phương pháp phân cụm K-Means

Phân cụm k-means là 1 phương pháp lượng tử hóa vector dùng để phân các điểm dữ liệu cho trước vào các cụm khác nhau. Phân cụm k-means có nhiều ứng dụng, nhưng được sử dụng nhiều nhất trong Trí tuệ nhân tạo và Học máy (cụ thể là Học không có giám sát).

Thuật toán k-means sử dụng phương pháp tạo và cập nhật trung tâm để phân nhóm các điểm dữ liệu cho trước vào các nhóm khác nhau. Đầu tiên chúng sẽ tạo ra các điểm trung tâm ngẫu nhiên. Sau đó gán mỗi điểm trong tập dữ liệu vào trung tâm gần nó nhất. Sau đó chúng sẽ cập nhật lại trung tâm và tiếp tục lặp lại các bước đã kể trên. Điều kiện dừng của thuật toán: Khi các trung tâm không thay đổi trong 2 vòng lặp kế tiếp nhau. Tuy nhiên, việc đạt được 1 kết quả hoàn hảo là rất khó và rất tốn thời gian, vậy nên thường người ta sẽ cho dừng thuật toán khi đạt được 1 kết quả gần đúng và chấp nhận được.

2.3 Thuật toán huấn luyện mô hình

2.3.1 Naive Bayes Classifier

Xét bài toán classification với CC classes 1,2,…,C1,2,…,C. Giả sử có một điểm dữ liệu $x \in R^d$. Hãy tính xác suất để điểm dữ liệu này rơi vào class cc. Nói cách khác, hãy tính: $p(y=c|x)$ (1)

Tức tính xác suất để đầu ra là class cc biết rằng đầu vào là vector xx.

Biểu thức này, nếu tính được, sẽ giúp chúng ta xác định được xác suất để điểm dữ liệu rơi vào mỗi class. Từ đó có thể giúp xác định class của điểm dữ liệu đó bằng cách chọn ra class có xác suất cao nhất:

$$c = \arg \max_{c \in \{1,\ldots,C\}} p(c|\mathbf{x}) \quad (2)$$

Biểu thức (2) thường khó được tính trực tiếp. Thay vào đó, quy tắc Bayes thường được sử dụng:

$$c = \arg\max_c p(c|\mathbf{x}) \ (3) \quad = \arg\max_c \frac{p(\mathbf{x}|c)p(c)}{p(\mathbf{x})} \quad 4) \quad = \arg\max_c p(\mathbf{x}|c)p(c) \ (5)$$

Từ (3) sang (4) là vì quy tắc Bayes. Từ (4) sang (5) là vì mẫu số p(x) không phụ thuộc vào c.

Tiếp tục xét biểu thức (5), p(c) có thể được hiểu là xác suất để một điểm rơi vào class cc. Giá trị này có thể được tính bằng MLE, tức tỉ lệ số điểm dữ liệu trong tập training rơi vào class này chia cho tổng số lượng dữ liệu trong tập training; hoặc cũng có thể được đánh giá bằng MAP estimation. Trường hợp thứ nhất thường được sử dụng nhiều hơn.

Thành phần còn lại p(x|c), tức phân phối của các điểm dữ liệu trong class cc, thường rất khó tính toán vì xx là một biến ngẫu nhiên nhiều chiều, cần rất rất nhiều dữ liệu training để có thể xây dựng được phân phối đó. Để giúp cho việc tính toán được đơn giản, người ta thường giả sử một cách đơn giản nhất rằng các thành phần của biến ngẫu nhiên xx là độc lập với nhau, nếu biết cc (given c). Tức là:

$$p(\mathbf{x}|c) = p(x_1, x_2, \ldots, x_d|c) = \prod_{i=1}^{d} p(x_i|c) \quad (6)$$

NBC, nhờ vào tính đơn giản, có tốc độ training và test rất nhanh. Việc này giúp nó mang lại hiệu quả cao trong các bài toán large-scale.

Ở bước training, các phân phối p(c)p(c) và p(xi|c),i=1,…,dp(xi|c),i=1,…,d sẽ được xác định dựa vào training data. Việc xác định các giá trị này có thể dựa vào Maximum Likelihood Estimation hoặc Maximum A Posteriori.

Ở bước test, với một điểm dữ liệu mới x, class của nó sẽ được xác định bởi:

$$c = \arg \max_{c \in \{1,\ldots,C\}} p(c) \prod_{i=1}^{d} p(x_i|c) \quad (7)$$

Khi d lớn và các xác suất nhỏ, biểu thức ở vế phải của (7) sẽ là một số rất nhỏ, khi tính toán có thể gặp sai số. Để giải quyết việc này, (7) thường được viết lại dưới dạng tương đương bằng cách lấy log của vế phải:

$$c = \arg \max_{c \in \{1,\ldots,C\}} = \log(p(c)) + \sum_{i=1}^{d} \log(p(x_i|c)) \quad (7.1)$$

Việc này không ảnh hưởng tới kết quả vì log là một hàm đồng biến trên tập các số dương.

Mỗi giá trị p(c),c=1,2,…,Có thể được xác định như là tần suất xuất hiện của class c trong training data.

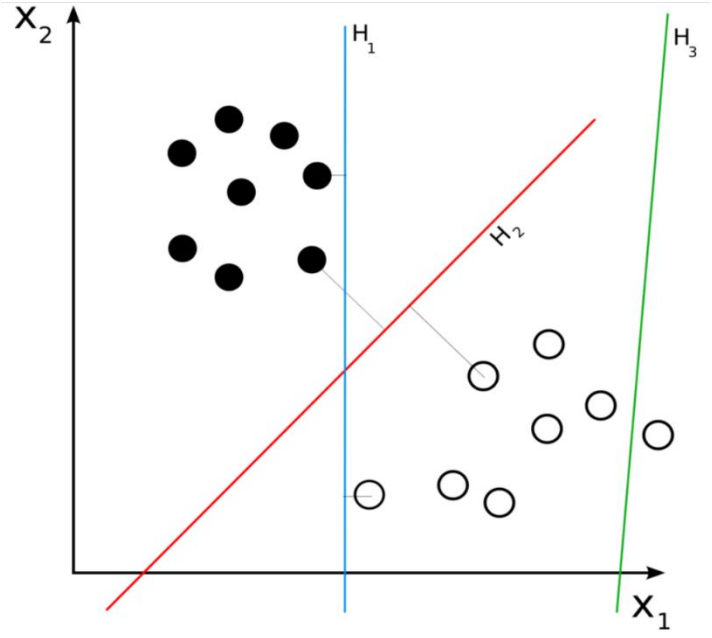
2.3.2 Kernel Support Vector Machine

SVM (Support Vector Machine) là 1 thuật toán học máy thuộc nhóm Supervised Learning (học có giám sát) được sử dụng trong các bài toán phân lớp dữ liệu (classification) hay hồi quy (Regression).

SVM là 1 thuật toán phân loại nhị phân, SVM nhận dữ liệu vào và phân loại chúng vào hai lớp khác nhau. Với 1 bộ các ví dụ luyện tập thuộc hai thể loại cho trước, thuật toán luyện tập SVM xây dựng 1 mô hình SVM để phân loại các ví dụ khác vào hai thể loại đó.

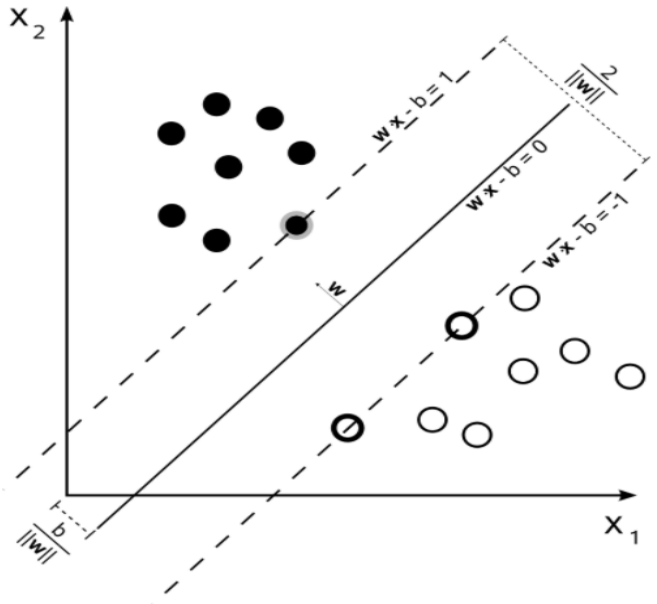
Ví dụ về SVM tuyến tính

Có 1 không gian có nhiều điểm và các kí hiệu như sau:



- yi: là các lớp (bản lề) chứa các điểm dữ liệu xi, ví dụ này mang giá trị 1 và -1.
- xi: là 1 vector thực nhiều chiều (p chiều).
- Nhiệm vụ là cần phải tìm 1 siêu phẳng (Optimal hyperplane) có lề lớn nhất chia tách các điểm dữ liệu có ban đầu để huấn luyện và các điểm sau này. Mỗi siêu

phẳng (Optimal hyperplane) đều có thể được viết dưới dạng 1 tập các điểm thỏa mãn w.x-b = 0



- w: là 1 vector pháp tuyến của siêu phẳng (optimal hyperplane).
- b/||w||: xác định khoảng cách giữa gốc tọa độ và siêu phẳng theo hướng vectơ pháp tuyến w.
- Giả sử có tới 3 siêu phẳng (Optimal hyperplane) là H1 (Xanh dương), H2 (Đỏ), H3 (Xanh lá). H3 sẽ bị loại đầu tiên vì không thể phân loại các điểm huấn luyện cho trước. H1 bị loại vì khoảng cách từ các điểm Support Vector đến siêu phẳng (Optimal hyperplane) chưa phải là cực đại. H2 là siêu phẳng cần tìm. Lúc này các siêu phẳng đó được xác định: w.x - b = 1 và w.x - b = -1.

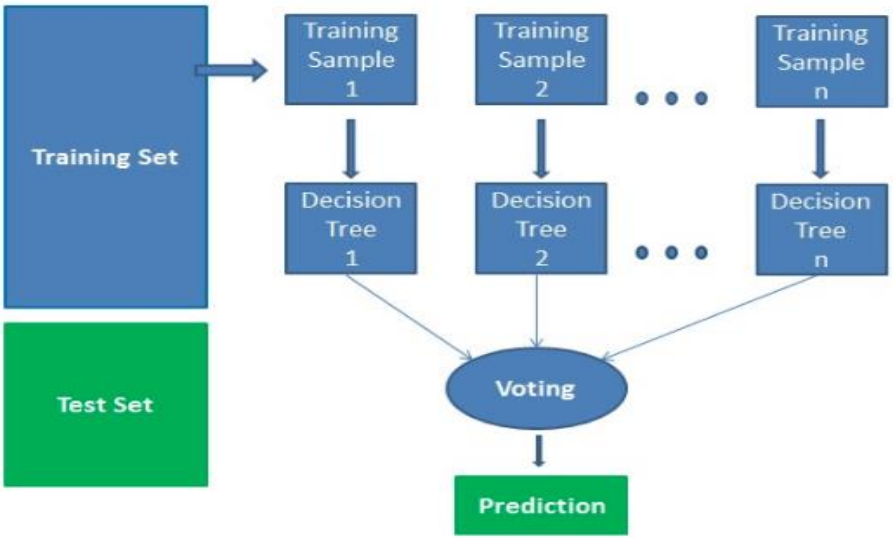
2.3.3 Random forest classification

Random Forests là thuật toán học có giám sát (supervised learning). Nó có thể được sử dụng cho cả phân lớp và hồi quy. Nó cũng là thuật toán linh hoạt và dễ sử dụng nhất. Thuật toán Random Forests

Nó hoạt động theo bốn bước:

- Chọn các mẫu ngẫu nhiên từ tập dữ liệu đã cho.
- Thiết lập cây quyết định cho từng mẫu và nhận kết quả dự đoán từ mỗi quyết định cây.
- Hãy bỏ phiếu cho mỗi kết quả dự đoán.

- Chọn kết quả được dự đoán nhiều nhất là dự đoán cuối cùng.



2.4 Phương pháp đánh giá mô hình

Khi xây dựng một mô hình Machine Learning, chúng ta cần một phép đánh giá để xem mô hình sử dụng có hiệu quả không và để so sánh khả năng của các mô hình. Trong bài viết này, tôi sẽ giới thiệu các phương pháp đánh giá các mô hình classification.

2.4.1 Accuracy

Cách đơn giản và hay được sử dụng nhất là accuracy (độ chính xác). Cách đánh giá này đơn giản tính tỉ lệ giữa số điểm được dự đoán đúng và tổng số điểm trong tập dữ liệu kiểm thử.

2.4.2 Confusion matrix

Cách tính sử dụng accuracy như ở trên chỉ cho chúng ta biết được bao nhiêu phần trăm lượng dữ liệu được phân loại đúng mà không chỉ ra được cụ thể mỗi loại được phân loại như thế nào, lớp nào được phân loại đúng nhiều nhất, và dữ liệu thuộc lớp nào thường bị phân loại nhầm vào lớp khác. Để có thể đánh giá được các giá trị này, chúng ta sử dụng một ma trận được gọi là confusion matrix.

Confusion matrix với Precision, Recall và F1.

Chúng ta đi đến một ví dụ để có thể hiểu về chúng.

Một ma trận nhầm lẫn để hình dung một hệ thống phân loại nhị phân đối với các nhãn tiêu chuẩn vàng .

Tiêu chuẩn nhãn vàng

Nhãn đầu ra của hệ thống	**Nhãn vàng tích cực**	**Nhãn vàng tiêu cực**
Nhãn hệ thống tích cực	Đúng tích cực	Sai tích cực
Nhãn hệ thống tiêu cực	Sai tiêu cực	Đúng tiêu cực

Chúng tôi thường chuyển sang hai chỉ số khác được hiển thị trong bảng trên: precision và recall .

Precision đo tỷ lệ phần trăm của các mục có độ chính xác hệ thống đã phát hiện (tức là hệ thống được gắn nhãn là tích cực) trên thực tế là tích cực (tức là là tích cực theo nhãn vàng của con người). Precision được định nghĩa là:

Precision = Đúng tích cực/(Đúng tích cực +Sai tích cực)

Recall được định nghĩa là :

Recall = Đúng tích cực/ (Đúng tích cực +Sai tiêu cực)

Có nhiều cách để xác định một số liệu duy nhất kết hợp các khía cạnh của cả hai precision and recall. Sự kết hợp đơn giản nhất trong số những cách kết hợp này là thước đo F1, được định nghĩa là:

F1= 2*Precision *Recall/ (Precision +Recall)

CHƯƠNG 3: BỘ DỮ LIỆU

Trong phần này, chúng tôi trình bày các thông tin cơ bản về bộ dữ liệu và quy trình tiền xử lý trên bộ dữ liệu bằng python trên môi trường colab. Chúng ta sẽ trực quan và phân tích rõ từng thuộc tính của bộ dữ liệu.

```
#Loading Dataset
dataset = pd.read_csv("Mall_Customers.csv")
```

Hình 3.1.1. Đọc dữ liệu từ file csv

5 điểm dữ liệu đầu tiên:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

Hình 3.1.2. Dữ liệu quan sát được từ môi trường colab

3.1 Codebook

Bộ dữ liệu ban đầu là một tập hợp thông tin khách hàng được thu thập từ một trung tâm thương mại với các dữ liệu cơ bản của khách hàng như ID khách hàng, độ tuổi, giới tính, thu nhập hàng năm và điểm chi tiêu. Điểm chi tiêu được chỉ định cho khách hàng dựa trên một số thông số như hành vi của khách hàng và dữ liệu mua hàng.

STT	Thông tin	Nội dung
1	Tên bộ dữ liệu	**Dữ liệu khách hàng của trung tâm mua sắm (Mall_customers)**
2	Nguồn dữ liệu	customer-segmentation-dataset.zip - Google Drive
4	Kích thước bộ dữ liệu	Bộ dữ liệu gồm 200 điểm dữ liệu.
5	Số thuộc tính	Có 5 thuộc tính.

7	Ý nghĩa các nhãn của thuộc tính	- CustomerID: Mã ID của khách hàng được đánh theo số thứ tự từ 1 đến 200. - Gender: Giới tính của khách hàng: + Male: Khách hàng giới tính nam. + Female: Khách hàng giới tinh nữ. - Age: Độ tuổi của khách hàng. - Annual Income: Thu nhập bình quân theo tháng của khách hàng (Đơn vị k$) - Spending Score: Điểm chi tiêu được chỉ định cho khách hàng dựa trên một số thông số như hành vi của khách hàng và dữ liệu mua hàng(Được đánh giá từ 1 đến 100)
8	Tác giả và các thức thu thập.	**Vijay Choudhary** - Trợ lý Giám đốc tại Vodafone Bengaluru, Karnataka, Ấn Độ - Nguồn dữ liệu được thu thập tại một trung tâm thương mại.

Bảng 3.1. CodeBook của bộ dữ liệu

3.2 Mô tả bộ dữ liệu

Do dữ liệu của Gender là dạng chuỗi cho nên chúng ta sẽ mô tả nó cụ thể ở phần sau:

	Customer ID	Age	Annual Income	Annual Income
Độ lệch chuẩn	57.879185	13.969007	26.264721	25.823522
Trung Bình	100.5	38.85	60.56	50.20
Min	1	18	15	1
25%	50.76	28.75	41.5	43.75
50%	100.5	36	61.5	50
75%	150.25	49	78	73
Max	200	70	137	99

Bảng 3.2. Mô tả tổng quan về bộ dữ liệu.

Qua bảng trên cho ta thấy tổng quan nhất về bộ dữ liệu. Ví dụ như khách hàng vào mua sắm ở trung tâm có độ tuổi từ 18 tuổi đến 70 tuổi và độ tuổi trung là 38 đến 39.

3.3 Kiểm tra bộ dữ liệu

Dữ liệu đã ở dạng số, khá hoàn hảo cho quá trình xây dựng mô hình. Việc duy nhất ở đây chúng ta cần là kiểm tra xem có giá trị null không bộ dữ liệu hay không.

```
1 # checking if there is any NULL data
2 dataset.isnull().any()
```

```
CustomerID              False
Gender                  False
Age                     False
Annual Income (k$)      False
Spending Score (1-100)  False
dtype: bool
```

Có thể thấy dữ liệu không bị mất hay mang giá trị null.

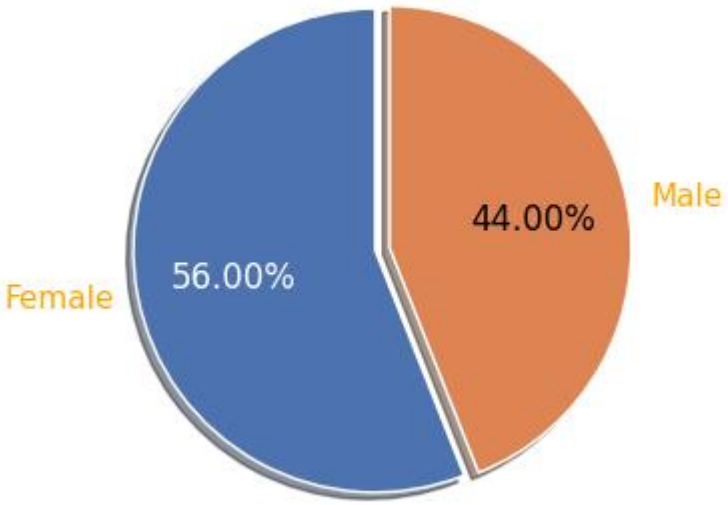
3.4 Trực quan hóa dữ liệu từng thuộc tính

Trong phần này chúng ta sẽ trực quan hóa, phân tích rõ cụ thể về từng thuộc tính để đưa ra các nhận xét, từ đó tìm ra các thuộc tính quan trọng ảnh hưởng trực tiếp đến mô hình phân lớp cũng như nó có tác động như thế nào đến việc mua hàng hay không.

Lưu ý: Do thuộc tính CustomerID chỉ là số thứ tự của khách hàng nó không ảnh hưởng đến doanh thu cũng như quá trình phân lớp khách hàng, chính vì chúng tôi không phân tích thuộc tính này.

3.4.1 Phân tích thuộc tính Gender

Bằng cách sử dụng công cụ vẽ đồ thị tròn chúng ta thu được:



Figure

Hình 3.4.1. Đồ thị phân bố giới tính khách hàng

Điều thú vị là nữ giới chiếm tỷ lệ lớn hơn trong khi nam giới chỉ chiếm 44%, điều đó khác đặc biệt khi dân số của nam giới tương đối cao hơn nữ giới. Nhưng trong bài toán phân khúc này tỷ lệ này không ảnh hưởng đến việc phân lớp cho lắm. Vì nó gần tiến tới tỷ lệ 5:5.

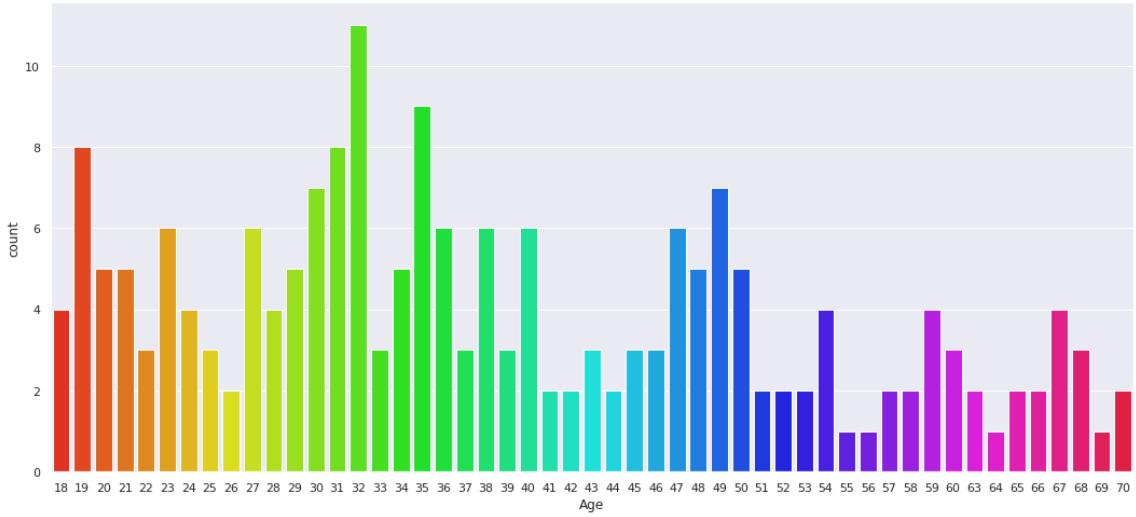
➔ Trong bài toán này chúng ta sẽ không sử dụng thuộc tính này để thực hiện mô hình phân lớp.

3.4.2 Phân tích thuộc tính Age

Theo tâm tâm lý học hành vi yếu tộ độ tuổi ảnh hướng đến hành vi mua hàng của họ:

Tuổi tác và giai đoạn trong đời sống gia đình: Nhu cầu về các loại hàng hoá, dịch vụ cũng như khả năng mua của người tiêu dùng gắn liền với tuổi tác và giai đoạn trong đời sống gia đình của họ:

Chúng ta sẽ trực quan hóa thuộc tính này bằng biểu đồ.



Hình 3.4.2.1. Phân bố của khách hàng theo độ tuổi

Do độ tuổi phân bố rải rác từ 18 đến 70 rất khó để đưa ra kết luận chung vì thế chúng tôi sẽ chia độ tuổi thành 5 lớp dựa theo 5 giai đoạn cuộc sống gia đình:

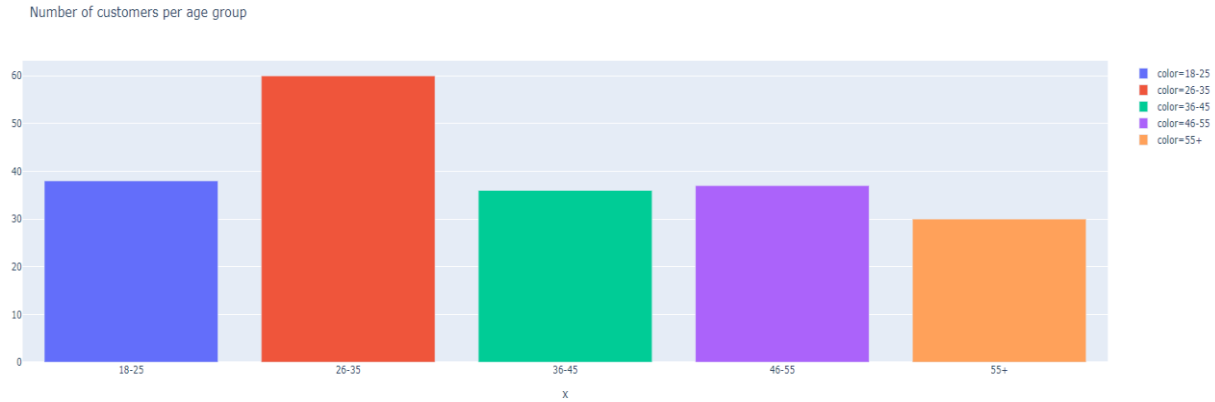
+18-25:

+26-35:

+35-45:

+46-55:

+55+ :



Hình 3.4.2.2. Phân bố của khách hàng theo phân nhóm độ tuổi

Từ 2 biểu đồ này hiển thị rõ hơn về sự tương tác giữa các độ tuổi khách hàng đến việc mua sắm ở trung tâm thương mại.
Bằng cách nhìn vào biểu đồ trên, có thể thấy rằng độ tuổi từ 26 đến 35 là rất thường xuyên nhưng không có mô hình rõ ràng. Những người ở độ tuổi 55,56,69,64 rất ít ghé

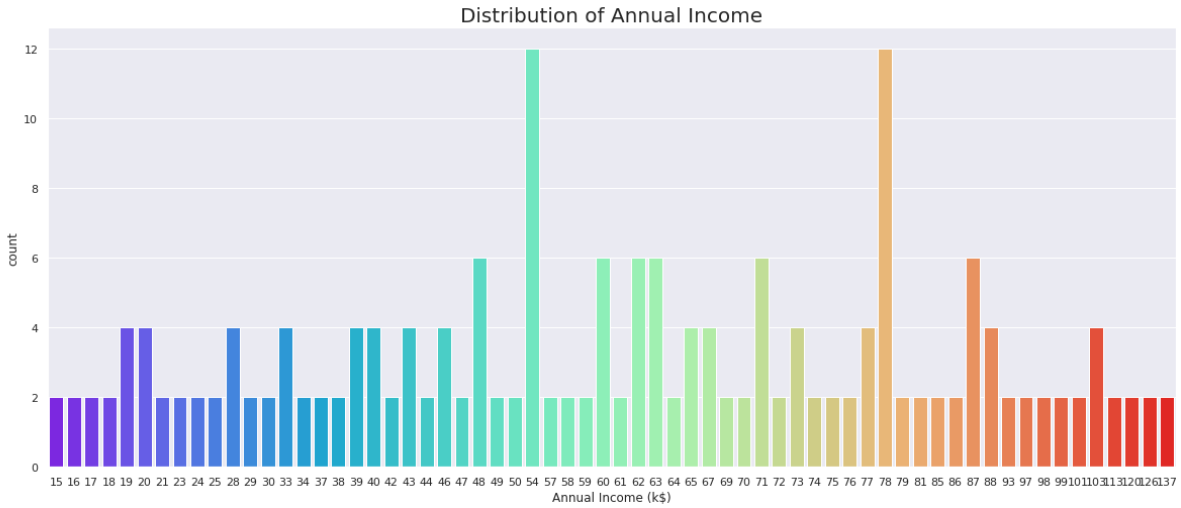
đến trung tâm mua sắm, Những người ở tuổi 32 là những người hay ghé thăm trung tâm mua sắm nhất.

➔ Ta có thể thấy độ phân hóa mô hình của các lớp này là không cao chênh lệch rất ít, nhưng trong bài toán chúng ta vẫn sẽ thử sử dụng thuộc tính này để phân khúc khách hàng.

Các yếu tố ảnh hưởng đến hành vi của người tiêu dùng: (uef.edu.vn)

3.4.3 Phân tích thuộc tính Annual Income Analysis

Theo tâm lý học hành vi để phân khúc được khách hàng điều kiện kinh tế là yếu tố đặc biệt quan trọng, nó quyết định khách hàng có thể mua được hàng hóa, dịch vụ. Khi ngân sách tiêu dùng cao thì tỷ lệ các hàng hóa xa xỉ càng tăng lên, tỷ lệ chi tiêu cho các mặt hàng thiết yếu càng giảm xuống.

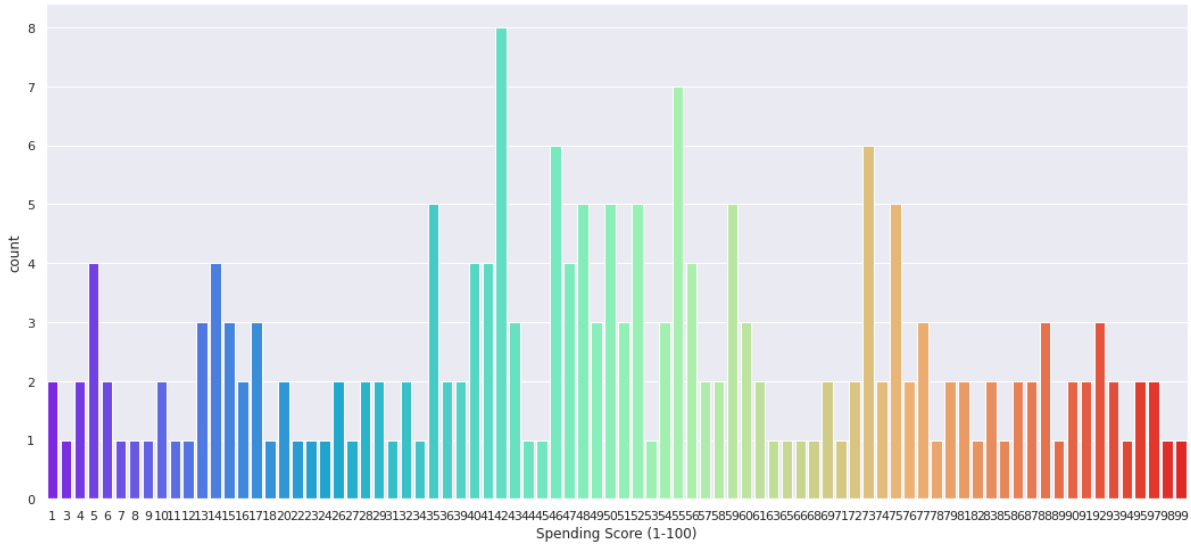


Hình 3.4.4. Phân bố của khách hàng theo thu nhập bình quân

➔ Đây cũng là một biểu đồ để giải thích rõ hơn về Phân bổ của từng mức Thu nhập, Điều thú vị là có những khách hàng trong trung tâm mua sắm có tần suất hoạt động tương đương rất nhiều với Thu nhập hàng năm của họ từ 15 đô la Mỹ đến 137 nghìn đô la Mỹ. Có nhiều Khách hàng hơn trong Trung tâm mua sắm có Thu nhập hàng năm là 54 nghìn đô la Mỹ hoặc 78 đô la Mỹ.

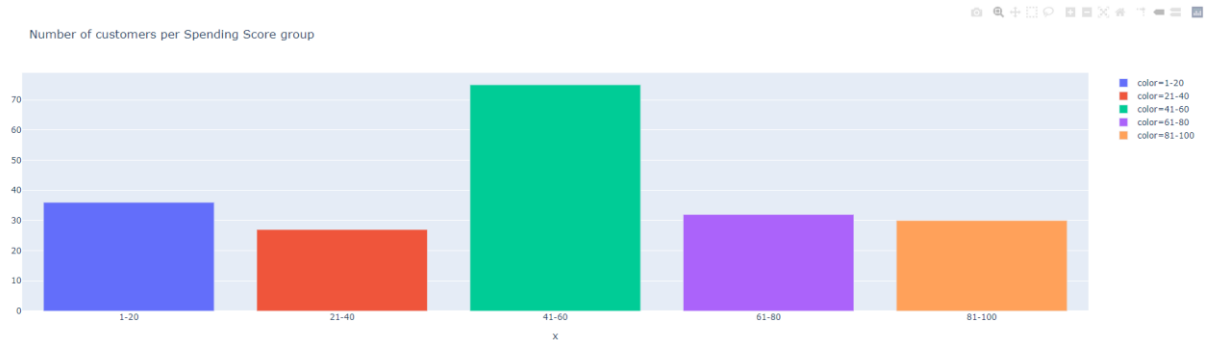
3.4.4 Phân tích thuộc tính Spending Score

Đây là thuộc tính được chính trung tâm đánh giá nó nói lên sự chị tiêu của khách hàng đối với trung tâm mua sắm. Nó là thuộc tính quan trọng nhất ảnh hưởng đến doanh thu và chiến lược tương lai của công ty.



Hình 3.4.4.1. Biểu đồ số lượng khách hàng của từng điểm mua sắm

Biểu đồ trên rất khó hình dung và không có mô hình rõ ràng. Vì vậy chúng ta cũng sẽ phân điểm theo từng lớp để dễ quan sát.



Hình 3.4.4.2. Biểu đồ số lượng khách hàng theo từng nhóm điểm

Ở cấp độ chung, chúng tôi có thể kết luận rằng phần lớn khách hàng có điểm chi tiêu trong khoảng 40 đến 60. Và phân bố khá đều ở các nhóm khác. Đây à biểu đồ quan trọng nhất trong quan điểm của trung tâm mua sắm.

CHƯƠNG 4: PHÂN KHÚC KHÁCH HÀNG

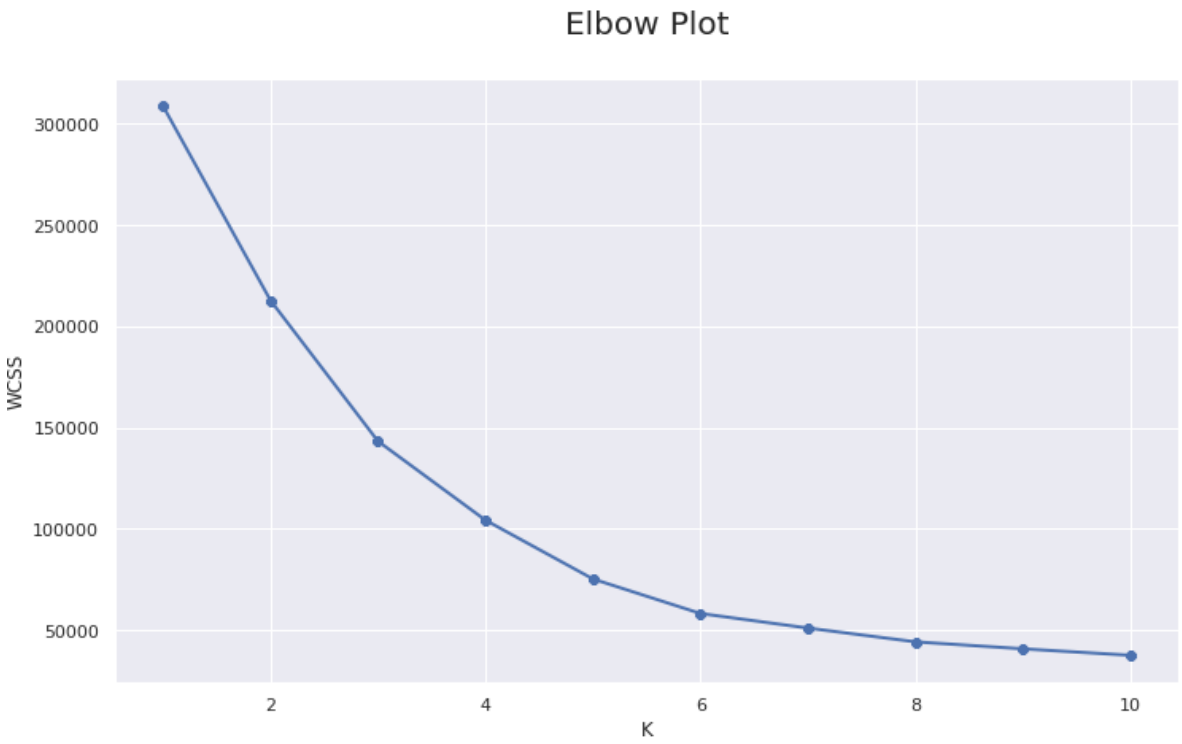
Trong chương này chúng tôi sẽ dựa vào các phân tích ở chương trước sẽ sử dụng các thuộc tính có thể ảnh hưởng đến bài toán để xây dựng mô hình phân lớp khách hàng. Ngày từ phần mở đầu chúng tôi đã xác định sử dụng thuật toán Kmean vì đây là thuật toán đơn giản và phù hợp nhất trong bài toán này.

4.1 Xây dựng mô hình phân lớp với 3 thuộc tính Age, Annual Income, Spending Score

4.1.1 Xác định số nhóm thích hợp

Phương pháp Elbow là một phương pháp thực nghiệm để tìm số lượng cụm tối ưu cho một tập dữ liệu. Trong phương pháp này, chúng tôi chọn một phạm vi các giá trị ứng viên của k, sau đó áp dụng phân cụm K-Means bằng cách sử dụng từng giá trị của k. Tìm khoảng cách trung bình của mỗi điểm trong một cụm đến tâm của nó và biểu diễn nó trong một biểu đồ. Chọn giá trị của k, trong đó **khoảng cách trung bình giảm đột ngột** .

Chúng tôi sẽ sử dụng biến WCSS là tổng bình phương khoảng cách giữa mỗi điểm và tâm trong một cụm ứng với số cụm K. Từ đó xây dựng nên biểu đồ Elbow.

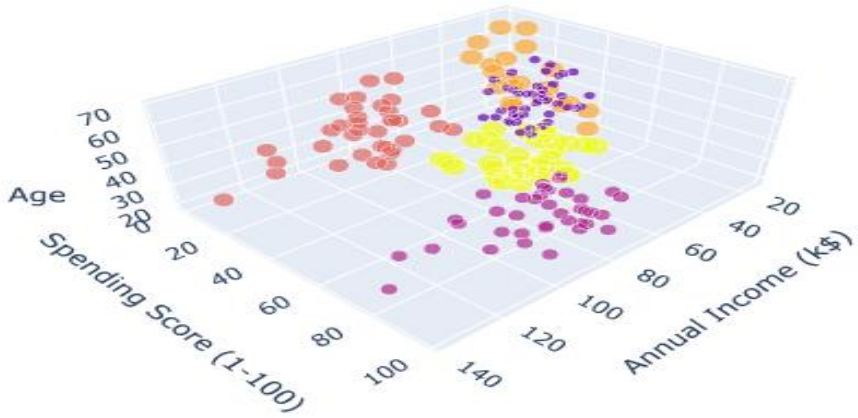


Hình 4.1.1. Đồ thị khuỷu tay ứng với 3 thuộc tính

Từ biểu đồ trên chúng tôi gặp khó trong việc xác định giá trị K. Vì các wcss không bị giảm đột ngột ở giá trị K nào cả. Nhưng quan sát kĩ với K=6. Đây là điểm đồ thị gần có dạng khủy tay nhất.

4.1.2 Áp dụng thuật toán Kmean để huấn luyện mô hình

Bắt đầu huấn luyện với số cụm là 6 và lấy các điểm trung tâm. Cuối cùng ta được một mô hình gồm 6 cụm như sau:



Hình 4.1.2. Phân lớp khách hàng trong không gian 3 chiều

➔ Từ biểu đồ Elbow cũng như trong không gian 3 chiều thì phân cụm dựa trên 3 thuộc tính cho kết quả không chính xác cũng như rất khó quan sát. Vì thế chúng ta sẽ loại bỏ bớt một thuộc tính.

4.2 Tìm thuộc tính tiêu biểu

Như đã phân tích ở chương 2 thuộc tính điểm chi tiêu là thuộc tính ảnh hưởng đến trực tiếp doanh thu cũng như trong bài toán này nó là thuộc tính tiêu biểu. Do đó chúng ta sẽ cần loại bỏ 1 trong 2 thuộc tính còn lại.

4.2.1 Phân tích mối tương quan của 2 thuộc tính Age và Annual Incom với thuộc tính Spending Core

4.2.1.1 Đánh giá mức độ tương quan giữa các thuộc tính.

```
1 # finding Correlation b/w features
2 dataset.corr()
```

	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
CustomerID	1.000000	-0.026763	0.977548	0.013835
Age	-0.026763	1.000000	-0.012398	-0.327227
Annual Income (k$)	0.977548	-0.012398	1.000000	0.009903
Spending Score (1-100)	0.013835	-0.327227	0.009903	1.000000

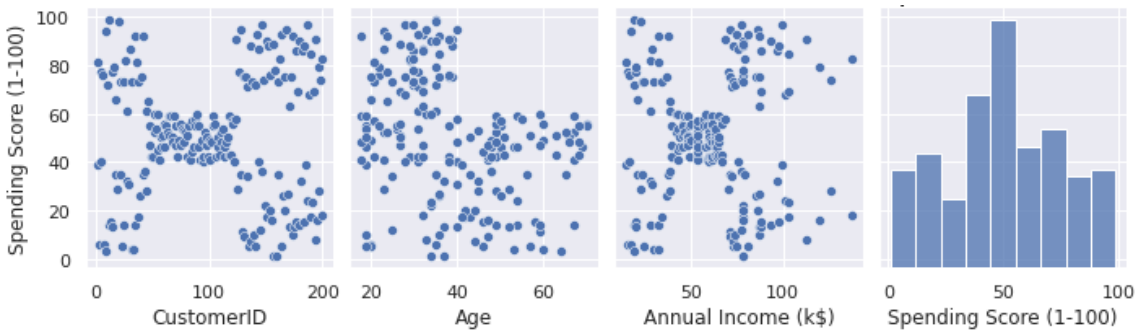
Bảng 4.2.1.1. Độ tương quan giữa các thuộc tính

.

Dựa vào bảng trên có thể thấy là độ tương quan giữa Age và Spending Score là tương quan nghịch còn với thuộc tính Annual Income có sự tương quan thuật thuận

4.2.1.2 Cụ thể hóa dữ liệu với các điểm thực tế

Do kích thước bộ dữ liệu của chúng ta nhỏ chỉ dừng lại ở 200 điểm. Vì thế việc cụ thể hóa ra có thể đơn gian thực hiện được. Từ đó đưa ra được cái nhìn trực quan nhất.



Hình 4.2.1.2. Trực quan hóa các điểm theo các cặp thuộc tính

➔ Dựa vào hình trên ta chỉ quan tâm 3 thuộc tính. Và rõ ràng chúng ta có thể đưa ra nhận xét biểu đồ giữa thu nhập và điểm chi tiêu các điểm phân cụm rõ thành từng cụm hơn tuổi và điểm chi tiêu. Do đó chúng ta sẽ sử dụng Annual Icome và Spending Score để xây dựng mô hình phân lớp cuối cùng.

4.3 Xây dựng mô hình phân lớp với 2 thuộc tính tiêu biểu

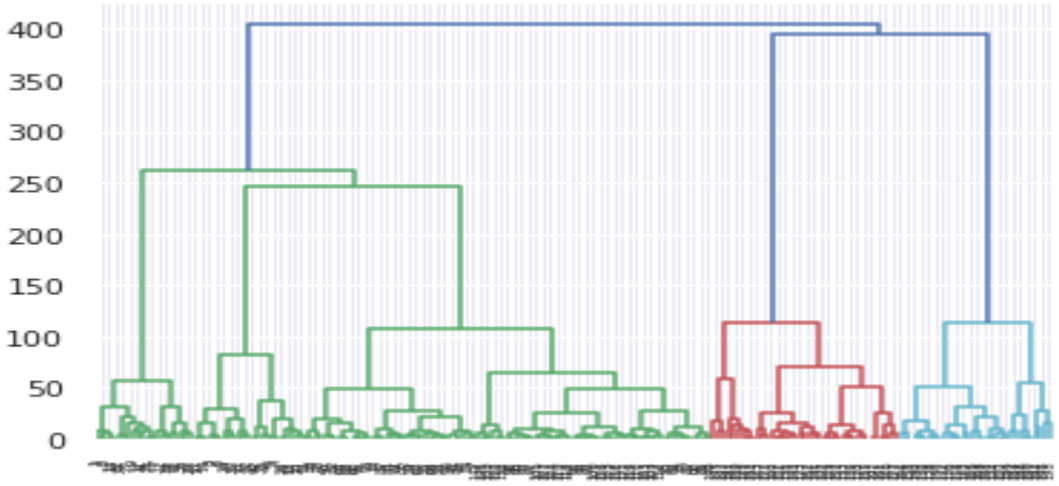
4.3.1 Tìm số cụm k tối đa

Do chỉ còn sử dụng 2 thuộc tính để huấn luyện do đó chúng ta sẽ sử dụng 2 phương pháp để tìm ra số cụm K tối ưu nhất để để sử dụng thuật toán Kmean.

```
# Giá trị đầu vào là 2 thuộc tính Annual Income (K$) và Spending Score
X = dataset.iloc[:, [3, 4]].values
```

Hình 4.2.1.1. Lấy giá trị đầu vào để huấn luyện.

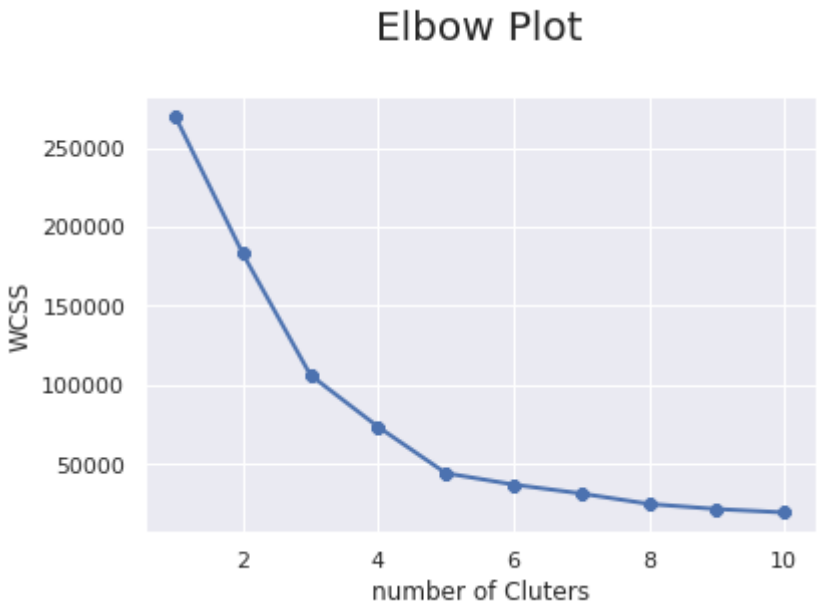
Chúng ta sẽ thể hiện mối quan hệ thứ bậc của các điểm dữ liệu dựa vào 2 thuộc tính là Annual Income và Spending core bằng cách sử dụng biểu đồ Dendrogram.



Hình 4.2.1.2.Biểu đồ dendrogram cho 2 thuộc tính

Dựa vào biểu đồ trên có thể thấy ra có thể chia số nhóm của dữ liệu thành 3 hoặc 5 nhóm.

Tiếp theo chúng ta sử dụng lại đồ thị Elow để xác định số cụm tối ưu.



Hình 4.2.1.3. Đồ thị khuỷu tay cho 2 thuộc tính

.

Dựa vào sự quan sát 2 hình trên có thể thấy 5 nhóm là số nhôm tối ưu cho bộ dữ liệu này.

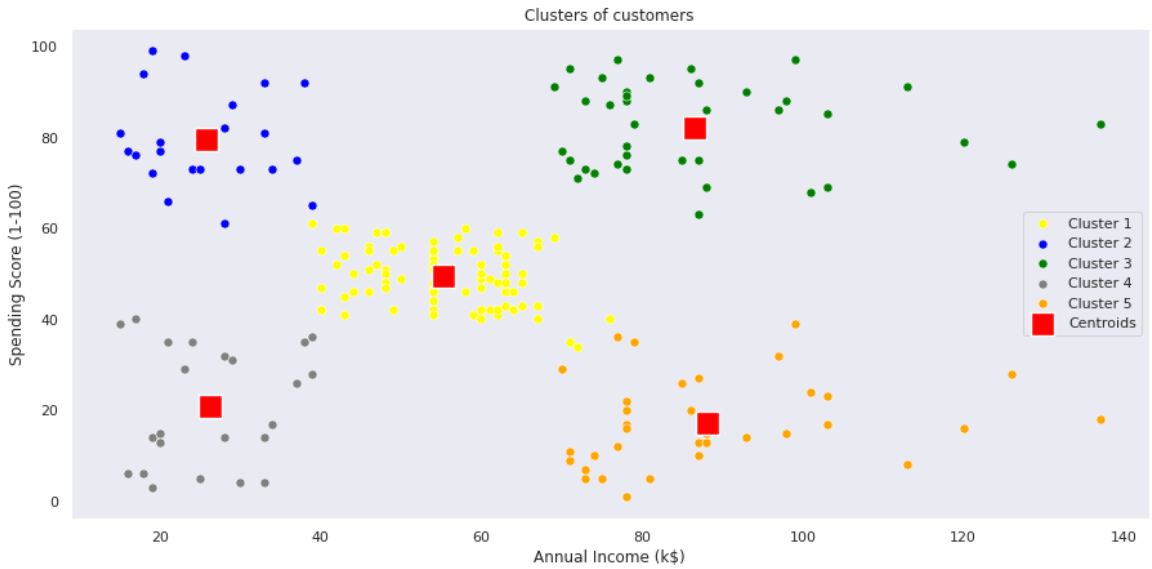
4.3.2 Huấn luyện phân lớp với thuật toán K-means

Như đã tìm hiểu ở phần trên ta sẽ tiến hành huấn luyện mô hình K-means với số k=5.

```
kmeans = KMeans(n_clusters = 5, random_state = 0)
y_kmeans = kmeans.fit_predict(X)
```

Hình 4.2.2.1. Huấn luyện với mô hình K-means

Cuối cùng ta sẽ thu được một bộ dữ liệu đã được phân từng nhóm dựa theo 2 thuộc tính là Annual Income và Spending core.Trực quan hóa kết quả ta sẽ thu được hình ảnh cụ thể như sau.



Hình 4.2.2.2. Phân bố của từng lớp khách hàng

Có thể thấy các lớp được phân bố khá rõ ràng, có thể nhận thấy bằng mắt thường. Từ những lớp chúng ta vừa phân ra chúng ta sẽ thêm thuộc tính **Clusters** cho từng khách hàng. Do đó trong bài toán tiếp theo là dự báo bộ dữ liệu sẽ trở thành bộ dữ liệu có nhãn. Chúng ra sẽ sử dụng các thuật toán có giám sát để dự đoán ngược lại thuộc tính này.

Dựa vào mức thu nhập và mức chi tiêu được chia thành 5 nhóm như sau:

- Cụm 1- Thu nhập trung bình Chi tiêu trung bình = Tiêu chuẩn.
- Cụm 2- Thu nhập thấp và chi tiêu cao = Bất cẩn.
- Cụm 3- Thu nhập cao và chi tiêu cao = Mục tiêu.
- Cụm 4- Thu nhập thấp và chi tiêu thấp = Hợp lý.
- Cụm 5- Thu nhập cao chi tiêu thấp = Cẩn thận

Sau khi thêm vào thuộc tính Cluster ta sẽ có bộ dữ liệu mới tên là Cluster-Customer.csv

```
dataset["Cluster"]=y_kmeans + 1
dataset.to_csv("Cluster_Customers.csv")
```

Hình 4.2.2.3. Tạo file Custer_Customer

.

CHƯƠNG 5: DỰ ĐOÁN PHÂN KHÚC KHÁCH HÀNG

Trong chương này chúng ta sẽ làm quen với bộ dữ liệu mới được hoàn thành từ bộ dữ liệu ban đầu với thuộc tính Clusters mà chúng ra huấn luyện được từ chương 4.

5.1 Bộ dữ liệu

Ở phần này chúng ta sẽ làm quen với bộ dữ liệu mới này.

```
1
2 import pandas as pd
3 import numpy as np
4 dataset = pd.read_csv("/content/Cluster_Customers.csv")
5 dataset
```

Hình 5.1.1. Đọc dữ liệu từ file csv

CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)	Cluster
1	Male	19	15	39	4
2	Male	21	15	81	2
3	Female	20	16	6	4
4	Female	23	16	77	2
5	Female	31	17	40	4

Hình 5.1.2. 5 điểm dữ liệu đầu tiên của bộ dữ liệu

Chúng ta sẽ thử kiểm tra lại để tìm ra bộ dữ liệu mới có bị thiếu giá trí nào hay không.

```
1 # checking if there is any NULL data
2 dataset.isnull().any()
```

```
Unnamed: 0              False
CustomerID              False
Gender                  False
Age                     False
Annual Income (k$)      False
Spending Score (1-100)  False
Cluster                 False
dtype: bool
```

Hình 5.1.3. Kiểm tra giá trị thiếu của bộ dữ liệu

➔ Bộ dữ liệu mới hoàn toàn hoàn hảo để sử dụng cho huấn luyện để phân lớp khách hàng.

5.2 Tiền xử lý

Như đã phân tích ở chương 4 chúng ta sẽ sử dụng 2 thuộc tính Annual Income và Spending core để làm giá trị đầu vào.

Tất cả các phương pháp học máy được so sánh đều sử dụng bộ dữ liệu được phân chia giống nhau với tỉ lệ train: test = 8: 2 và kết quả được báo cáo dựa trên tập test.

```
# Phân chia bộ dữ liệu
X = dataset.iloc[:, [3, 4]].values
Y = dataset.iloc[:, -1].values
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test =train_test_split(X, Y, train_size = 0.8, random_state = 0)
```

Hình 5.2. Phân chia bộ dữ liệu thành test và train với tỷ lệ 2:8

Có thể thấy giá trị đầu vào là X, giá trị đầu ra là Y.

Import train_test_split từ sklearn.model_selection để phân chia tập dữ liệu.

5.3 Huấn luyện mô hình

Tại đây, ta sẽ sử dụng các tham số với giá trị mặc định để huấn luyện cho cả 3 mô hình được đưa ra là: Kernel SVM, Naïve Bayes, Random Forest Classification để có thể so sánh và đánh giá khách quan hiệu suất của các mô hình trên.

```
# Huấn luyện với mô hình Naiv Bayes.
from sklearn.naive_bayes import GaussianNB
classifier_1 = GaussianNB()
classifier_1.fit(X_train, Y_train)

# Huấn luyện với mô hình RamdomForest.
from sklearn.ensemble import RandomForestClassifier
classifier_2 = RandomForestClassifier(n_estimators = 10, criterion ='entropy', random_state = 0)
classifier_2.fit(X_train, Y_train)

# Huấn luyện với mô hình
from sklearn.svm import SVC
classifier_3 = SVC()
classifier_3.fit(X_train, Y_train)

```

Hình 5.2. Huấn luyện 3 mô hình với tham số mặc định

- Sử dụng lớp **GaussianNB** trong module **sklearn.naive_bayes** để huấn luyện mô hình **Naïve Bayes.**
- Sử dụng lớp **SVC** trong module **sklearn.svm** để huấn luyện mô hình **Kernel SVM.**

- Sử dụng lớp **RandomForestClassifier** trong module **sklearn.ensemble** để huấn luyện mô hình **RandomForest**.

5.4 Đánh giá mô hình

Ở phần này chúng tôi sử dụng 2 phương pháp cơ bản nhất đã được học để sử dụng đánh giá các mô hình.

Chúng ta sẽ sử dụng 2 phương pháp để đánh giá mô hình đó chính là :

- Accuracy
- Confusion matrix

Accuracy: chúng ta sẽ sử dụng tập test và đưa ra kết quả dự đoán, sau đó Accuracy sẽ đánh giá độ chính xác của dự đoán và đưa đưa kết quả mà chúng ta mong muốn.
- Sử dụng lớp **accuracy_score** trong model **sklearn.metrics** để đánh giá mô hình

Confusion matrix: sử dụng tập test, đưa ra kết quả dự đoán và sau đó cho chúng ta một ma trận nhầm lẫn . Rồi từ ma trận đó chúng ta có thể đưa ra các phương pháp để đánh giá độ chính xác mà mô hình đã dự đoán được.
- Sử dụng lớp confusion_matrix trong model **sklearn.metrics** để đánh giá mô hình

Chúng ta sẽ có kết quả dự đoán đúng sẽ nằm trên đường chéo chính, còn những dự đoán nằm ngoài đường chéo chính sẽ là những dự đoán không chính xác.
Từ ma trận trên bằng nhiều phương pháp đánh giá khác nhau chúng ta sẽ có các kết quả khác nhau như:

- Precision
- Recall
- F1.

5.4.1 Naive Bayes

- Kết quả của Accuracy thu được khi chạy thuật toán là: 0.825.
- Đây là ma trận nhầm lẫn mà chúng ta thu được:

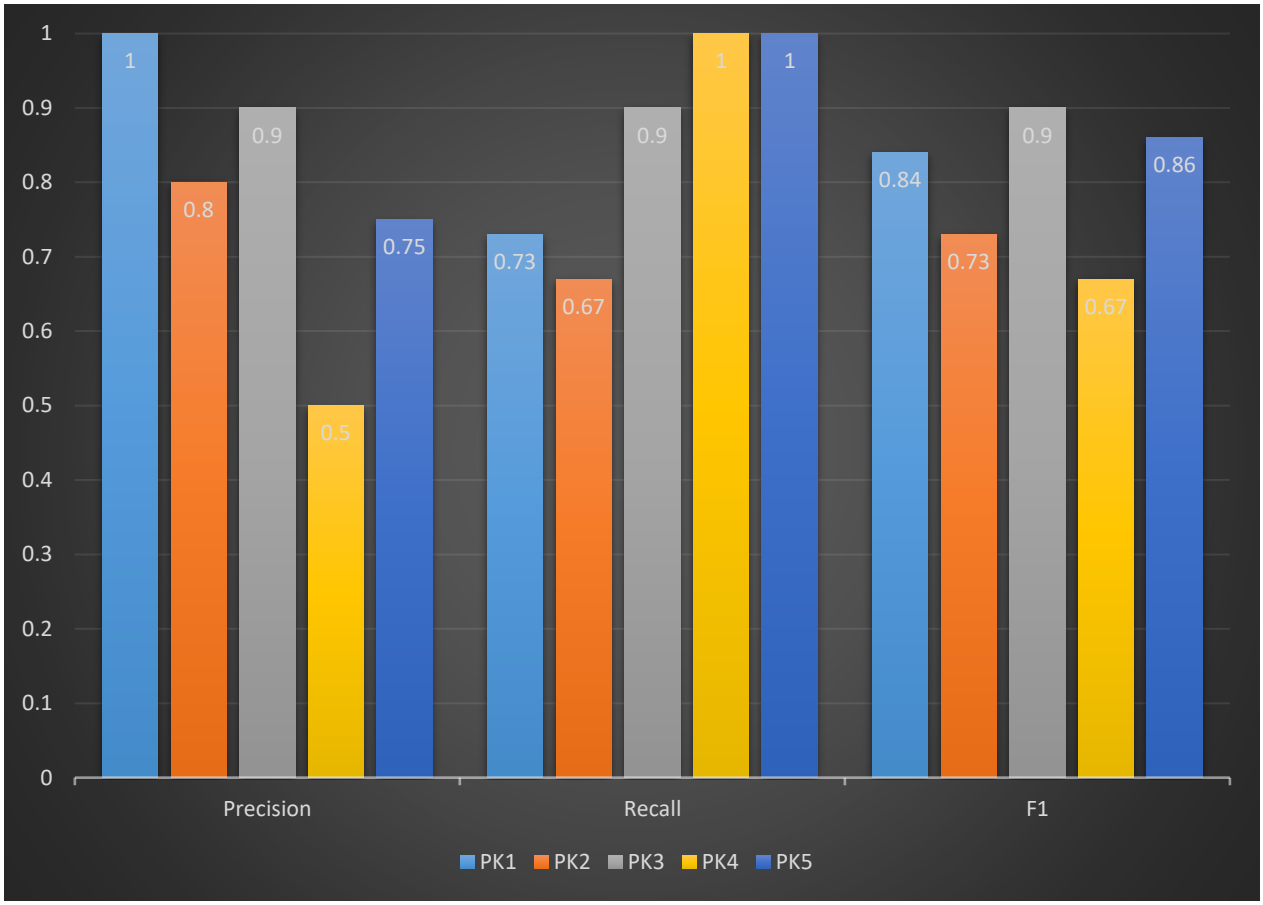
```
array([[11,  0,  0,  0,  0],
       [ 1,  4,  0,  0,  0],
       [ 1,  0,  9,  0,  0],
       [ 1,  2,  0,  3,  0],
       [ 1,  0,  1,  0,  6]])
```

- Từ ma trận trên ta có thể thấy :
  Số lượng điểm dữ liệu được dự đoán đúng là: 33
  Số lượng điểm dữ liệu được dự đoán không đúng là: 7
  Từ đó chúng ta có thể tính được tỷ lệ điểm phân loại đúng là : 0,825

Ta có bảng sau:

	Precision	**Recall**	**F1**
Phân khúc thứ 1	1.00	0.73	0.84
Phân khúc thứ 2	0.80	0.67	0.73
Phân khúc thứ 3	0.90	0.90	0.9
Phân khúc thứ 4	0.50	1.00	0.67
Phân khúc thứ 5	0.75	1.00	0.86
Trung bình	0.79	0.86	0.8

Bảng 5.4.1. Classification Report- NB trên tập Test



Hình 5.4.1 Biểu đồ giá trị độ đo Classification Report - NB trên tập Test

Nhận xét:

Đối với mô hình Naive Bayes, ta nhận thấy:

− Kết quả đánh giá trung bình của mô hình phân loại là 82,5%

− Từ kết quả precision, recall và F1-score của mô hình sử dụng Naive Bayes được thể hiện trong Bảng và hình. Mô hình phân loại khá tốt đối với các phân khúc khách hàng , nhưng vẫn có sự chênh lệch rất lớn ở chúng, lên đến 50% (50% và 100% ở precision)

− Điều này cũng có thể hiểu được do sự không cân đối của bộ dữ liệu

→ *Tuy độ chính xác tương đối cao nhưng do sự chênh lệch độ chính xác ở precision của lớp 4 quá lớn cho nên thuật toán này không tối ưu.*

5.4.2 Random Forest

-Kết quả của Accuracy thu được khi chạy thuật toán là: 0.8

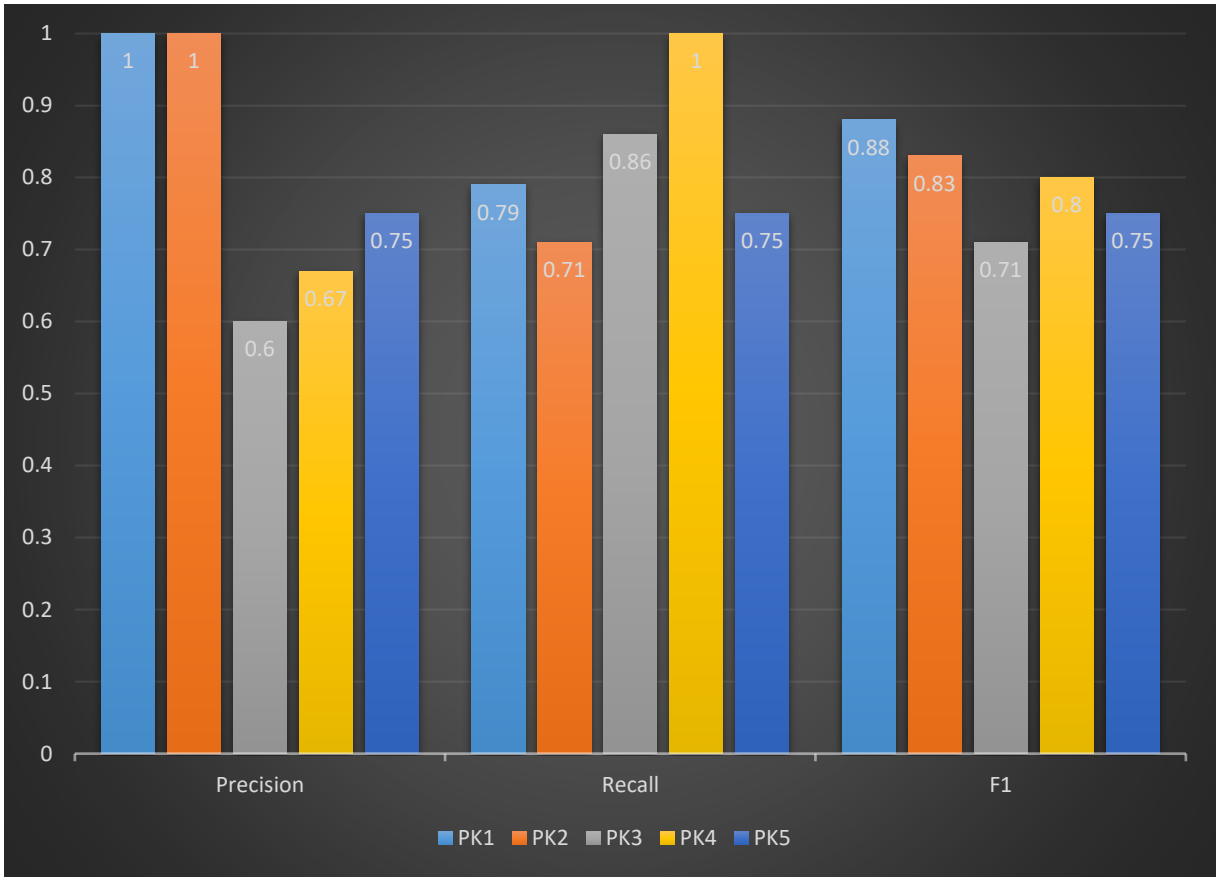
-Đây là ma trận nhầm lẫn mà chúng ta thu được:

```
array([[11,  0,  0,  0,  0],
       [ 0,  5,  0,  0,  0],
       [ 2,  0,  6,  0,  2],
       [ 0,  2,  0,  4,  0],
       [ 1,  0,  1,  0,  6]])
```

-Từ ma trận trên ta có thể thấy:
Số lượng điểm dữ liệu được dự đoán đúng là: 32
Số lượng điểm dữ liệu được dự đoán không đúng là: 8
Từ đó chúng ta có thể tính được tỷ lệ điểm phân loại đúng là: 0,8
Ta có bảng sau:

	Precision	**Recall**	**F1**
Phân khúc thứ 1	1.00	0.79	0.88
Phân khúc thứ 2	1.00	0.71	0.83
Phân khúc thứ 3	0.60	0.86	0.71
Phân khúc thứ 4	0.67	1.00	0.80
Phân khúc thứ 5	0.75	0.75	0.75
Trung bình	0.84	0.82	0.79

Bảng 5.2: Classification Report- RF trên tập Test



Hình 5.2 Biểu đồ giá trị độ đo Classification Report- RF trên tập Test

Nhận xét:

Đối với mô hình Random Forest, ta nhận thấy:

− Kết quả đánh giá trung bình của mô hình phân loại là 80% .

− Từ kết quả precision, recall và F1-score của mô hình sử dụng Random Forest được thể hiện trong Bảng và hình. Mô hình phân loại tương đối tốt đối với bộ dữ liệu, nhưng bên cạnh đó còn sự chênh lệch giữa các phân khúc với nhau khá nhiều (vd: 60% và 100%).

-> Độ chính xác ở mức tương đối chấp nhận được là 0.8 cùng với không có sự chênh lệch độ chính xác ở lớp 4 nhiều cho nên thuật toán này là thích hợp cho mô hình.

5.4.3 Kernel SVM

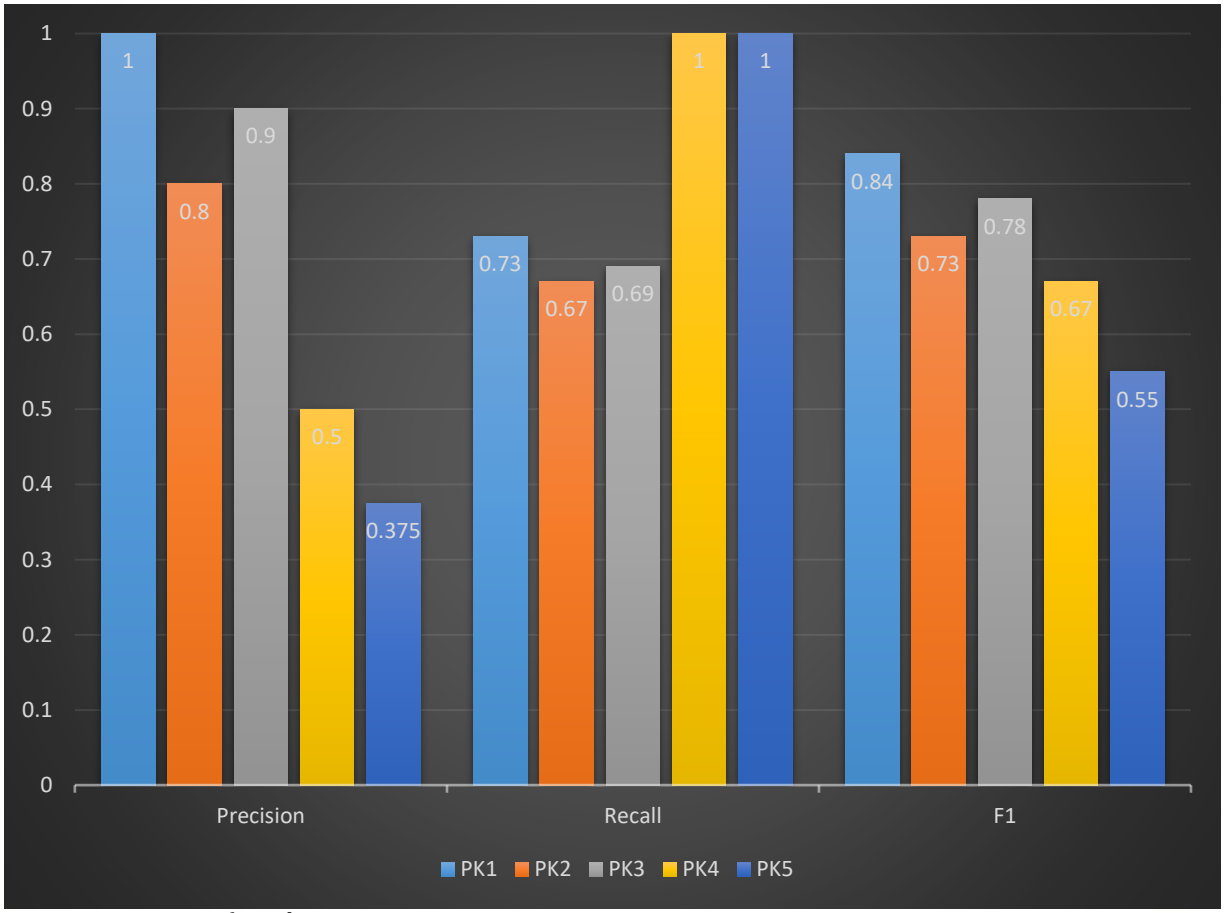
- Kết quả của Accuracy thu được khi chạy thuật toán là: 0.75.
- Đây là ma trận nhầm lẫn mà chúng ta thu được:

```
array([[11,  0,  0,  0,  0],
       [ 1,  4,  0,  0,  0],
       [ 1,  0,  9,  0,  0],
       [ 1,  2,  0,  3,  0],
       [ 1,  0,  4,  0,  3]])
```

- Từ ma trận trên ta có thể thấy:
  Số lượng điểm dữ liệu được dự đoán đúng là: 30.
  Số lượng điểm dữ liệu được dự đoán không đúng là: 10.
  Từ đó chúng ta có thể tính được tỷ lệ điểm phân loại đúng là: 0.75.
  Ta có bảng sau:

	Precision	**Recall**	**F1**
Phân khúc thứ 1	1.00	0.73	0.84
Phân khúc thứ 2	0.80	0.67	0.73
Phân khúc thứ 3	0.90	0.69	0.78
Phân khúc thứ 4	0.5	1.00	0.67
Phân khúc thứ 5	0.375	1.00	0.55
Trung bình	0.715	0.82	0.71

Bảng 5.4.3. Classification Report- Kernel SVM trên tập Test



Hình 5.4.3 Biểu đồ giá trị độ đo Classification Report- Kernel SVM trên tập Test

Nhận xét:

Đối với mô hình Kernel SVM, ta nhận thấy:

− Kết quả đánh giá trung bình của mô hình phân loại là 82,5%.

− Từ kết quả precision, recall và F1-score của mô hình sử dụng Kernel SVM được thể hiện trong Bảng và hình. Mô hình phân loại tương đối tốt đối với các phân khúc, nhưng vẫn có sự chênh lệch rất lớn ở chúng, lên đến 62.5% (37.5% và 100% ở precision).

− Điều này cũng có thể hiểu được do sự không cân đối của bộ dữ liệu và số lượng dữ liệu của tập test quá ít nên độ chính xác chưa đạt mức tối ưu.

➔ *Dọ độ chính xác của thuật toán này thấp cho nên không lựa chọn cho mô hình này.*

CHƯƠNG 6: KẾT LUẬN

Trong bài báo cáo này, chúng tôi đã trình bày được tầm quan trọng của việc phân khúc khách hàng trong kinh doanh trong Trung tâm thương mại. Sử dụng các kiến thức phân tích, xử lý, huấn luyện mô hình. Chúng tôi đã giới thiệu được về bộ dữ liệu khách hàng của một trung tâm thương mại. Từ đó chúng tôi phân tích được tầm quan trọng của các thuộc tính sau đó đưa ra được 2 thuộc tính tiêu biểu nhất là **Annual Incom(Thu nhập hàng năm)** và **Spending Score(Điểm chi tiêu)** để phân khúc ra được 5 nhóm khách hàng tiêu biểu là:

- Cụm 1- Thu nhập trung bình Chi tiêu trung bình = Tiêu chuẩn.
- Cụm 2- Thu nhập thấp và chi tiêu cao = Bất cẩn.
- Cụm 3- Thu nhập cao và chi tiêu cao = Mục tiêu.
- Cụm 4- Thu nhập thấp và chi tiêu thấp = Hợp lý.
- Cụm 5- Thu nhập cao chi tiêu thấp = Cẩn thận.

Sau đó chúng tôi thêm thuộc tính cụm cho mỗi điểm dữ liệu. Được bộ dữ liệu mới có tên là **Cluters_customers.csv.** Sử dụng 3 thuật toán học máy để huấn luyện dự đoán thuộc tính cụm ta được bảng sau:

Model -Features	**Precision**	**Recall**	**Accuracy**
Naive Bayes	0.79	0.86	**0.825**
Random Forest	0.84	0.82	**0.8**
Kernel SVM	0.715	0.82	**0.75**

Bảng 6.1. Bảng so sánh kết quả độ đo các mô hình phân loại

Từ kết quả đánh giá huấn luyện chúng tôi nhận thấy thuật toán Random Forest là tối ưu cho bài toán này. Từ kết quả của mô hình tôi hy vọng có thể giúp cho các doanh

nghiệp định hình được khách hàng mục tiêu. Cuối cùng, đưa ra được các chiến lược kinh doanh phù hợp.

TÀI LIỆU THAM KHẢO

[1] Các yếu tố ảnh hưởng đến hành vi của người tiêu dùng: (uef.edu.vn)
[2] https://sentory.vn/confusion-matrix-la-gi/

[3]https://viblo.asia/p/phan-lop-bang-random-forests-trong-python-djeZ1D2QKWz

[4] https://www.stdio.vn/computer-vision/gioi-thieu-ve-mo-hinh-svm-D15jcg

[5]https://bigdatauni.com/tin-tuc/cac-phuong-phap-danh-gia-trong-thuat-toan-clustering.html

[6] https://ichi.pro/vi/dendrogram-la-gi-173606807999262

[7] Customer Segmentation using Kmeans, HC & DBSCAN | Kaggle

[8] PV8, "What is difference between metrics.r2_score and acccuracy_score," Stack Overflow, 2019. [Online]. Available: https://stackoverflow.com/questions/58163026/what-is-difference-between-metrics-r2-score-and-acccuracy-score.

[9] H. D. Thoi, "Machine Learning cho người mới bắt đầu (Part 2)," VIBLO, 2018. [Online]. Available: https://viblo.asia/p/machine-learning-cho-nguoi-moi-bat-dau-part-2- naQZR1WXKvx.