BÁO CÁO ĐỀ TÀI HỌC MÁY

Phân Tích Tính Cách Khách Hàng

Lập Trình Nâng Cao – Th.S Ngô Thế Quyền

THÀNH VIÊN :

ĐỖ BÁ TRƯỜNG - 19000378

NGUYỄN MINH TUẤN - 19000371

TẠ QUỐC KHÁNH - 19000354

Mục Lục:

Tài liệu tham khảo:

https://machinelearningcoban.com/

https://www.kaggle.com/

Hands-on Machine Learning with Scikit Learn, Keras & TensorFlow – Aurélien Géron

Bộ dữ liệu:

https://drive.google.com/file/d/15Ecbtipul-6KZmj1pZKzwpXBxA987yo7/view?usp=sharing

1 Phát biểu bài toán

Tên đề tài: Phân Tích Tính Cách Khách Hàng

Đặc điểm: Phân tích tính cách khách hàng là một phân tích chi tiết về những khách hàng lí tưởng của một công ty. Nó giúp doanh nghiệp hiểu rõ hơn về khách hàng của mình và giúp họ dễ dàng sửa đổi các sản phẩm theo nhu cầu, hành vi của khách hàng. Phân tích tính cách khách hàng giúp doanh nghiệp sửa đổi sản phẩm của mình dựa trên khách hàng mục tiêu từ các loại phân khúc khách hàng khác nhau.

Bộ dữ liệu: gồm các trường chính là

1 ID: mã định danh khách hàng
2 Year_Birth: năm sinh
3 Education: trình độ học vấn
4 Marital_Status: trạng thái hôn nhân
5 Income: Thu nhập/năm
6 Kidhome: Số trẻ em trong hộ gia đình
7 Teenhome: Số thanh thiếu niên trong hộ gia đình
8 Dt_Customer: Ngày đăng ký khách hàng với công ty
9 Recency: Số ngày kể từ lần mua cuối cùng
10 Complain: Khách hàng có phàn nàn trong vòng 1 năm qua không (1:có/0:không)

Các sản phẩm:

11 MntWines: Số tiền chi cho rượu vang trong 2 năm qua
12 MntFruits: Số tiền chi cho trái cây trong 2 năm qua
13 MntMeatProducts: Số tiền chi cho thịt trong 2 năm qua
14 MntFishProducts: Số tiền chi cho cá trong 2 năm qua
15 MntSweetProducts: Số tiền chi cho đồ ngọt trong 2 năm qua
16 MntGoldProds: Số tiền chi cho vàng trong 2 năm qua

Chiến dịch khuyến mãi:

17 NumDealsPurchases: Số lần mua hàng được giảm giá

18 AcceptedCmp1: 1 nếu khách hàng chấp nhận ưu đãi trong chiến dịch thứ 1, 0 nếu không

19 AcceptedCmp2: 1 nếu khách hàng chấp nhận ưu đãi trong chiến dịch thứ 2, 0 nếu không

20 AcceptedCmp3: 1 nếu khách hàng chấp nhận ưu đãi trong chiến dịch thứ 3, 0 nếu không

21 AcceptedCmp4: 1 nếu khách hàng chấp nhận ưu đãi trong chiến dịch thứ 4, 0 nếu không

22 AcceptedCmp5: 1 nếu khách hàng chấp nhận ưu đãi trong chiến dịch thứ 5, 0 nếu không

23 Response: 1 nếu khách hàng chấp nhận đề nghị trong chiến dịch cuối cùng, 0 nếu không

24 NumWebPurchases: Số lượng mua hàng được thực hiện thông qua trang web của công ty

Địa điểm mua hàng:

25 NumCatalogPurchases: Số lần mua hàng được thực hiện bằng danh mục

26 NumStorePurchases: Số lượng mua hàng được thực hiện trực tiếp tại các cửa hàng

27 NumWebVisitsMonth: Số lượt truy cập vào trang web của công ty trong tháng trước

Các vấn đề chính cần giải quyết trong bài toán phân khúc khách hàng:

1 Đọc dữ liệu dang *.csv (Loading Data)
2 Làm sạch dữ liệu (Data Clearning)
3 Loại bỏ Exception (Drop Exception)
4 Trực quan hóa dữ liệu (Data Visualization)
5 Linear Regression Model cho một số features
6 Lựa chọn features để tiến hành phân cụm
7 Chuẩn hóa features table đã chọn (Data Standardization)
8 Tiến hành phân cụm: Kmeans Algorithm
9 Đánh giá Models
10 Kết luận, rút ra bảng kết quả

2 Phương pháp thực hiện

Các vấn đề đã nêu ra ở phần 1 sẽ được giải quyết theo từng phần trong phần 2.

Đọc dữ liệu *.csv

Sử dụng thư viện Pandas trong Python để tiến hành đọc file, file có kích thước là 2240 rows x 29 cols, sau đó lấy thông tin của các cột (name of col, num of null, type, unique).

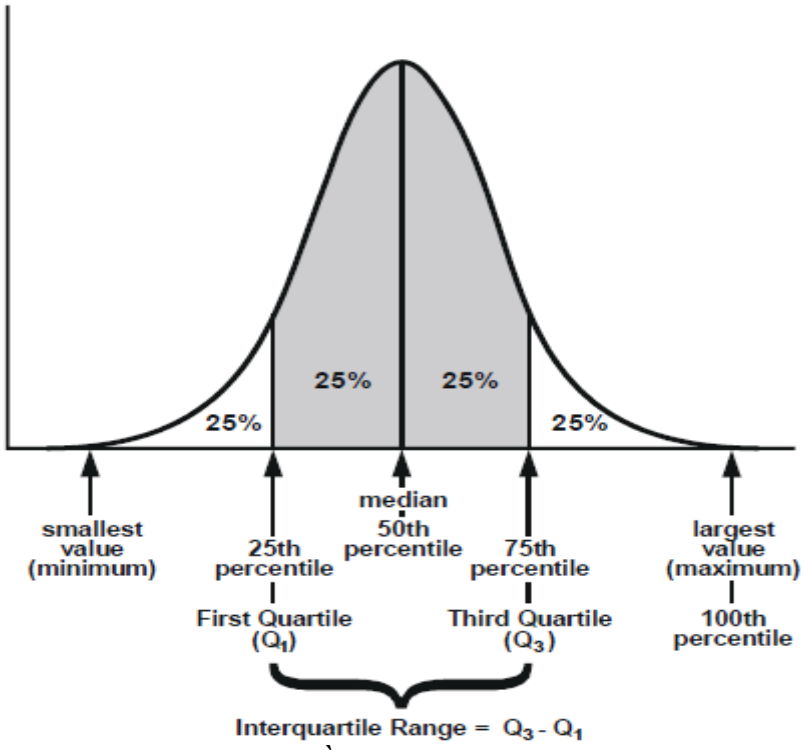
Làm sạch dữ liệu và loại bỏ các Exception

1. Gom nhóm các giá trị của cột ví dụ đối với cột Education ta sẽ được 2 nhóm là: Trước đại học (Post Graduation) và Sau đại học (Under Graduation), tương tự với cột Married_Status ta cũng có 2 nhóm là: Đã kết hôn (Married) và Độc thân (Single)
2. Thêm các cột mới vào bộ dữ liệu ví dụ như cột Year_Birth ta thêm cột Age, ...
3. Bỏ các cột thừa và không cần đến ra khỏi bộ dữ liệu.
4. Bỏ các hàng nếu cột của hàng null và bỏ dựa vào cột Age nếu Age > 100.
5. Bỏ các hàng có nhiễu: sử dụng hàm describe() trong thư viện Pandas trả về một bảng thống kê trên dữ liệu ta có được các giá trị như max, min, mean, count, std và tứ phân vị Q1 (25% dữ liệu <= Q1), Q2 (50% dữ liệu <= Q2), Q3 (75% dữ liệu <= Q3).

   Nhận thấy khoảng cách giữa giá trị max và Q3 khá lớn do đó có một số ngoại lệ xảy ra ta sẽ tiến hành loại bỏ các điểm đó.

   Mục đích của bước này là hạn chế sai lệch quá lớn khi áp dụng các mô hình học máy.

   Dựa vào bảng thống kê ta sẽ cần loại bỏ một số điểm dữ liệu trên cột Income, NumWebPurchases, …



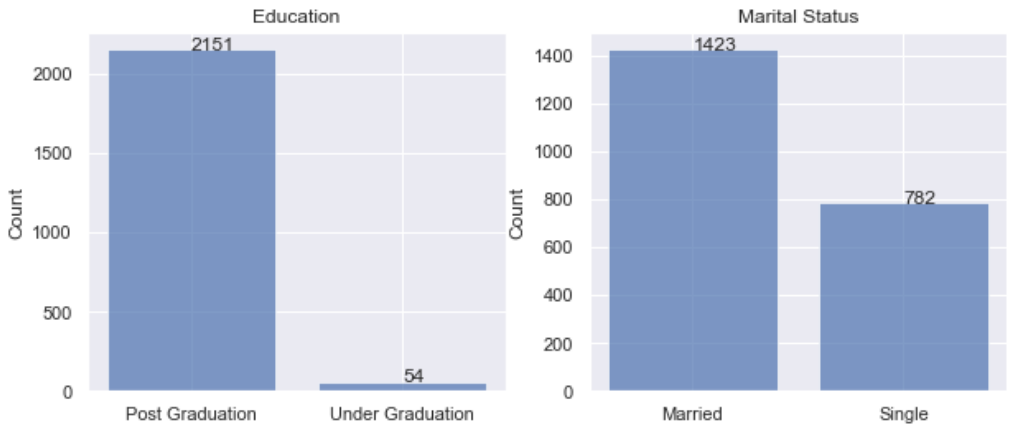
(*Đồ thị tứ phân vị*)

Sau khi làm sạch dữ liệu và loại bỏ các exception ta có kích thước bộ dữ liệu mới là 2205 rows x 23 cols

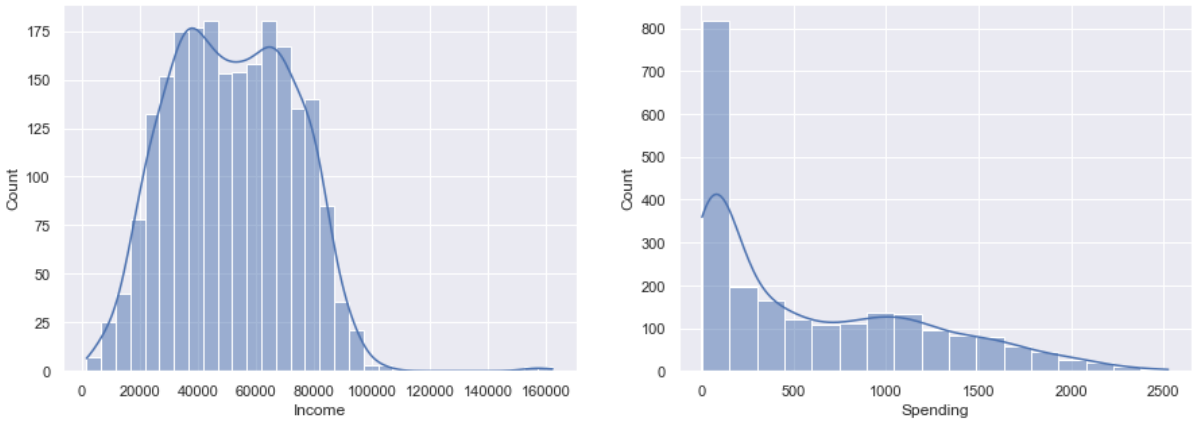
Trực quan dữ liệu

Sử dụng các thư viện của Python như matplotlib, plotly, seaborn để dựng các đồ thị cho ta cái nhìn tổng quan hơn về dữ liệu

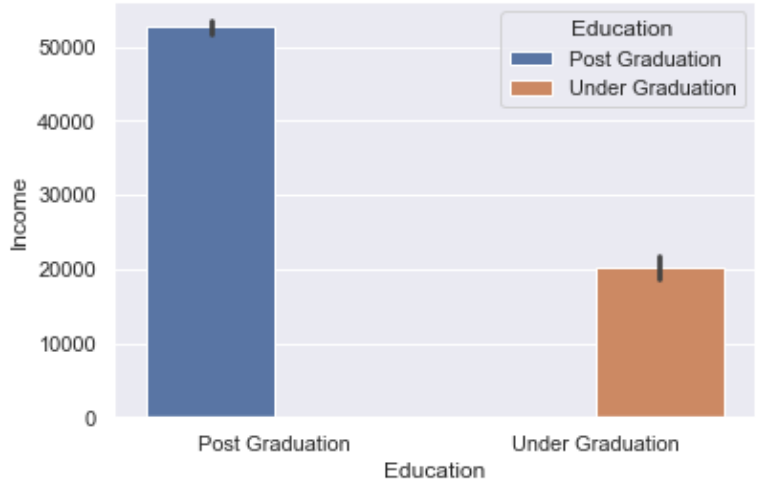
Một số đồ thị về khách hàng:



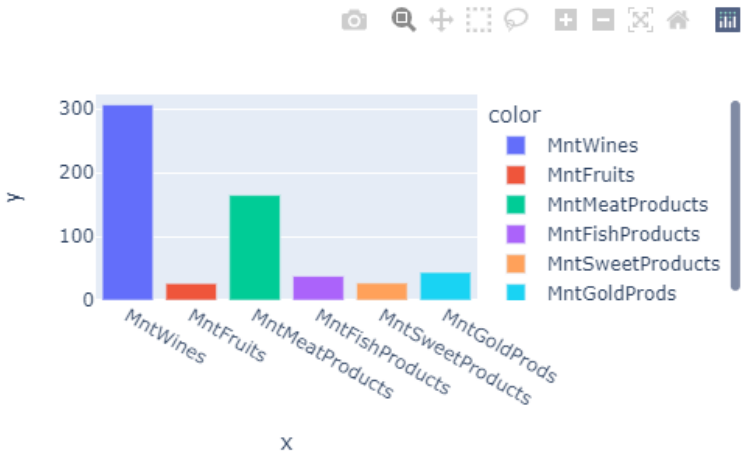
(đồ thị bar: Education và Marital Status)



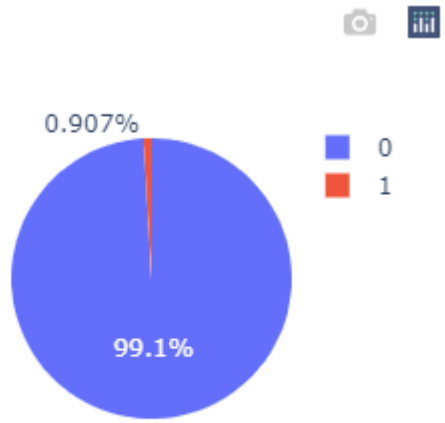
(đồ thị histogram: mức lương (Income) và chi tiêu (Spending) của khách hàng)



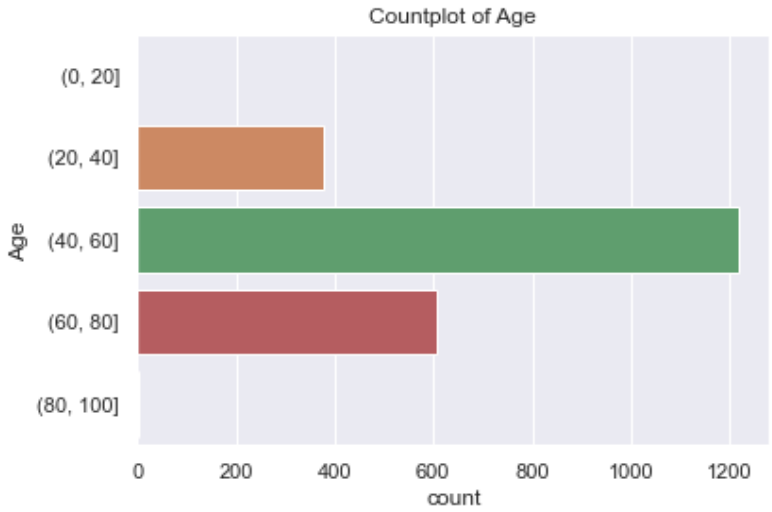
(đồ thị bar: mức lương trung bình của những khách hàng có học vấn trước đại học và sau đại học)



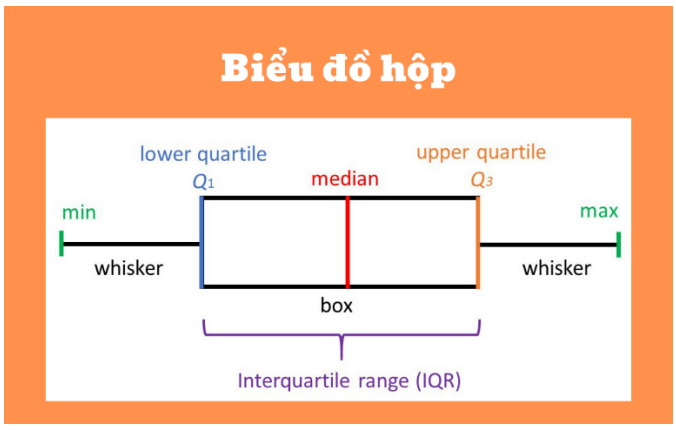
(đồ thị bar: Lượng chi tiêu trung bình các sản phẩm)



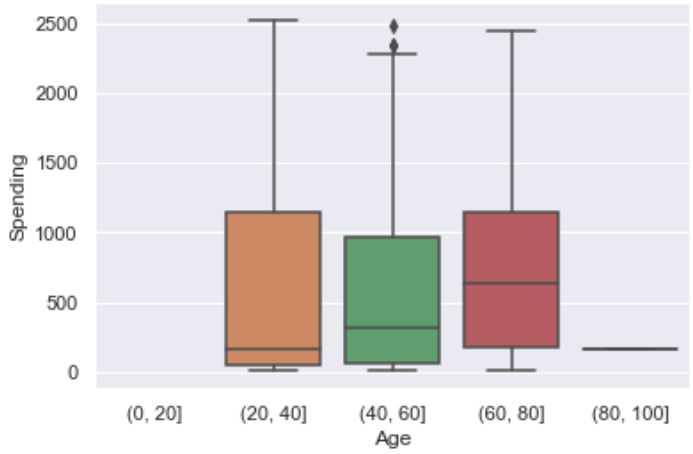
(đồ thị pie: tỷ lệ phàn nàn về các sản phẩm, 1 : có/0: không)



(đồ thị bar: các khoảng tuổi và số lượng khách hàng nằm trong khoảng đó)



Biểu đồ box: diễn tả 5 vị trí phân bố của dữ liệu, biểu diễn các giá trị quan trọng của dãy số một cách trực quan, dễ hiểu. Các giá trị gồm min, max, tứ phân vị thứ nhất (Q1), trung vị, tứ phân vị thứ ba (Q3)



(đồ thị box: độ tuổi và thông tin thống kê liên quan đến lượng chi tiêu (Spending))

Linear Regression (Hồi quy tuyến tính)

Là thuật toán học máy thuộc nhóm thuật toán học có giám sát (Supervised Learning), được sử dụng để dự đoán các giá trị liên tục.

Phương trình đường hồi quy tuyến tính dạng tổng quát:

$y = w_0 + w_1x_1 + w_2x_2 + ... + w_nx_n = \bar{X}w$

$\bar{X} = [1 \ x_1 \ x_2 \ x_3 \ ... \ x_n]$ – $\bar{X}$ mở rộng (các cột dữ liệu)

$w = [w_0 \ w_1 \ w_2 \ \dots \ w_n]^T$ – Hằng số dự đoán

Phương trình mô tả mối quan hệ giữa input (x) và output (y).

Mục tiêu của mô hình là tìm được w (sử dụng phương pháp toán học) :

Ta có hàm mất mát (Loss function) :

$$L(w) = \frac{1}{2n}\sum_{i=0}^{i=n}(y_i - \bar{X}_i w)^2$$

Điều ta muốn là tổng sai số là nhỏ nhất, tương đương với việc tìm w để hàm số L(w) đạt min.

L(w) được viết dưới dạng ma trận đơn giản hơn:

$$L(w) = \frac{1}{2n}\left|\left|y - \bar{X}w\right|\right|^2$$

||z|| : là chuẩn euclidean

Tìm nghiệm của một bài toán tối ưu là giải phương trình đạo hàm bằng 0:

$$\frac{\delta L(w)}{w} = \frac{1}{n} \bar{X}^T (\bar{X}w - y)$$

Ta có $\frac{\delta L(w)}{w} = 0$:

$$\bar{X}^T(\bar{X}\text{w} - \text{y}) = 0 \Rightarrow \bar{X}^T\bar{X}\text{w} = \bar{X}^T\text{y} = \text{b} \ (*)$$

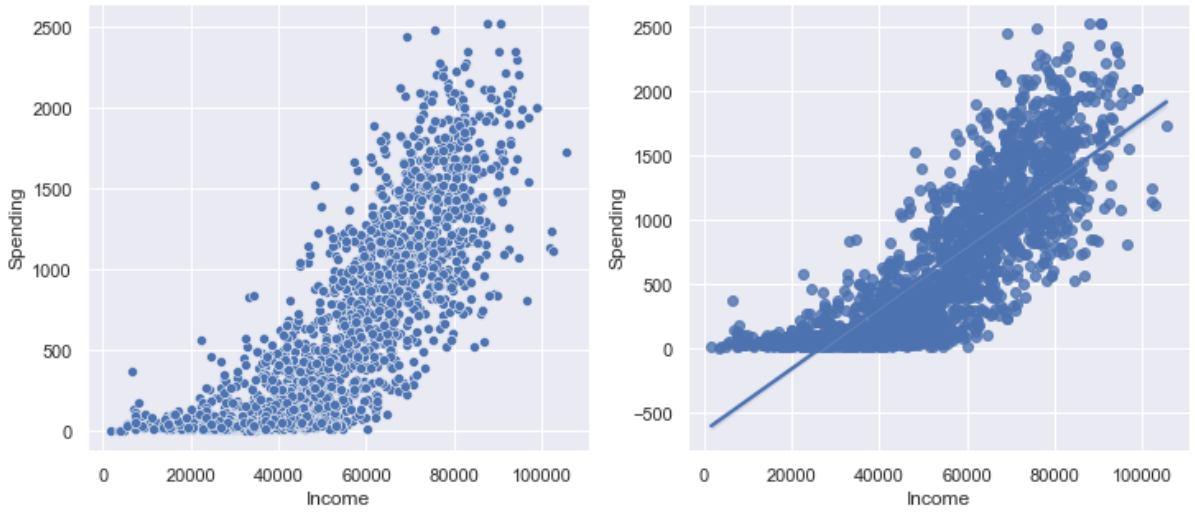
Nếu ma trận A = $\bar{X}^T\bar{X}$ là khả nghịch (định thức khác 0), thì phương trình (*) có nghiệm duy nhất :

$$\text{w} = \text{A}^{-1}\,\text{b}$$

Nếu ma trận A không khả nghịch (có định thức bằng 0) vậy hệ phương trình tuyến tính trong trường hợp này là vô nghiệm hoặc vô số nghiệm. Khi đó ta sử dụng khái niệm giả nghịch đảo, với khái niệm giả nghịch đảo điểm tối ưu của bài toán Linear Regression có dạng :

$$\text{w} = \text{A}^{+}\,\text{b} = (\bar{X}^T\bar{X})^{+}\,(\bar{X}^T y)$$

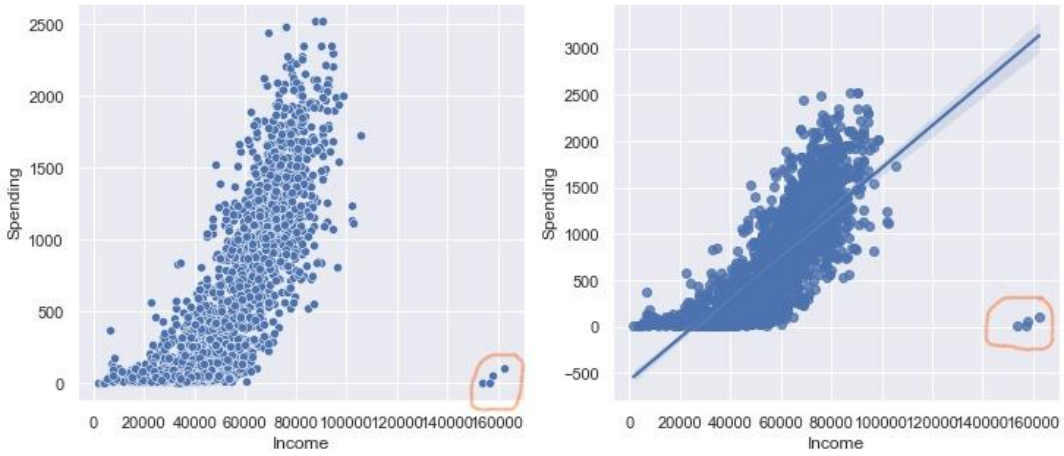
Ta sẽ kẻ một đường hồi quy tuyến tính trên các điểm dữ liệu (ví dụ trong không gian 2 chiều, tương tự với không gian n chiều, n > 2), các điểm dữ liệu trên đồ thị nằm càng gần đường hồi quy tuyến tính thì y dự đoán sẽ càng gần với y.



Công thức MAE (Mean Absolute Error): trị tuyệt đối độ lệch trung bình.

$$\text{mae} = \frac{1}{n}\sum_{i=0}^{i=n} |y_{predict}[i] - y[i]|$$

Nhược điểm của thuật toán: thuật toán nhạy cảm với điểm dữ liệu nhiễu, với điểm nhiễu khiến cho dự đoán bị sai lệnh khá lớn, ví dụ như trong hình có một số điểm nhiễu:



Ta có thể sử dụng thư viện Sklearn trong Python để tiếp cận với mô hình Linear Regression một cách nhanh chóng cho bài toán :

```
from sklearn.linear_model import LinearRegression
```

Đối với bài toán phân khúc khách hàng ta sẽ sử dụng mô hình Linear Regression trong thư viện sklearn.linear_model đối với 2 cột là x : Income và y : Spending.

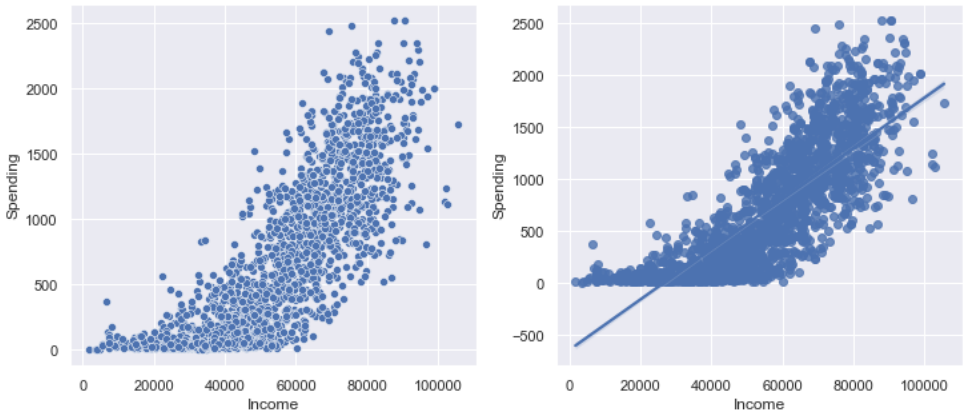
Ta chia dữ liệu thành 2 bộ dữ liệu: Trainning set (80%) và Testing set (20%)

Trainning set là tập dữ liệu sử dụng để huấn luyện mô hình học máy

Testing set là tập dữ liệu dùng cho việc kiểm thử

Sau đó ta có giá trị mae = 244.71 (giá trị dự đoán bị lệch trung bình là 244.71 (khá lớn))

Đồ thị như sau:



Lựa chọn features

Ta tiến hành lựa chọn các cột để phân khúc khách hàng dựa vào các tiêu chí như mức lương (Income), lượng chi tiêu (Spending) và thời gian gắn bó với công ty (Time_withCompany).

	Income	Spending	Time_withCompany
0	58138.0	1617	113.100000
1	46344.0	27	94.766667
2	71613.0	776	101.400000
3	26646.0	53	95.633333
4	58293.0	422	96.366667

Chuẩn hóa features

Bước chuẩn hóa là quan trọng do có cột chứa dữ liệu nhỏ và có cột chứa dữ liệu lớn, chuẩn hóa là một yêu cầu chung đối với nhiều công cụ ước tính học máy.

Điểm tiêu chuẩn của một mẫu x được tính như sau:

$$z = \frac{(x-u)}{s}$$

u: là giá trị trung bình của các mẫu

s: là độ lệch tiêu chuẩn của các mẫu

Việc căn giữa và chia tỷ lệ diễn ra độc lập trên từng cột bằng cách tính toán các số liệu thống kê liên quan trên các mẫu.

Sử dụng StandardScaler() trong thư viện sklearn.preprocessing để dùng cho việc chuẩn hóa dữ liệu. Dữ liệu sau khi chuẩn hóa

	Income	Spending	Time_withCompany
0	0.298240	1.681498	1.527764
1	-0.261178	-0.962057	-1.189818
2	0.937392	0.283240	-0.206547
3	-1.195501	-0.918829	-1.061351
4	0.305592	-0.305325	-0.952647

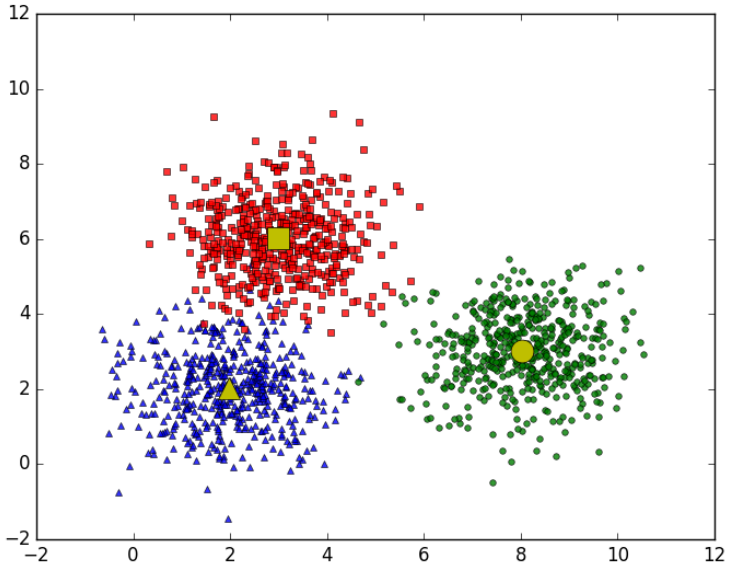
Kmeans Algorithm

Là thuật toán học máy thuộc nhóm thuật toán học không giám sát (Unsupervised Learning).

Trong thuật toán K-means clustering, chúng ta không biết nhãn (label) của từng điểm dữ liệu. Mục đích là làm thế nào để phân dữ liệu thành các cụm (cluster) khác nhau sao cho dữ liệu trong cùng một cụm có tính chất giống nhau.

Ví dụ: một công ty muốn tạo ra những chính sách ưu đãi cho những nhóm khách hàng khác nhau dựa trên sự tương tác giữa mỗi khách hàng với công ty đó (số năm là khách hàng; số tiền khách hàng đã chi trả cho công ty; độ tuổi; giới tính; thành phố; nghề nghiệp; ….). Giả sử công ty đó có rất nhiều dữ liệu của rất nhiều khách hàng nhưng chưa có cách nào chia toàn bộ khách hàng đó thành một số nhóm/cụm khác nhau. Vấn đề này sẽ được thuật toán K-means Clustering giải quyết. Sau khi đã phân ra được từng nhóm thì nhân viên công ty đó có thể lựa chọn ra một vài khách hàng trong mỗi nhóm để quyết định xem mỗi nhóm tương ứng với nhóm khách hàng nào.

Ý tưởng đơn giản nhất về cluster (cụm) là tập hợp các điểm ở gần nhau trong một không gian nào đó (không gian này có thể rất nhiều chiều trong trường hợp thông tin về một điểm dữ liệu là rất lớn). Hình bên dưới là một ví dụ về 3 cụm dữ liệu (cluster).



Giả sử mỗi cluster có một điểm đại diện (center) màu vàng. Và những điểm xung quanh mỗi center thuộc vào cùng nhóm với center đó. Một cách đơn giản nhất, xét một điểm bất kỳ, ta xem xét điểm đó gần với center nào nhất thì nó thuộc về cùng nhóm center đó.

Mục đích cuối cùng của thuật toán phân nhóm này là: từ dữ liệu đầu vào và số lượng nhóm chúng ta muốn tìm, hãy chỉ ra center của mỗi nhóm và phân các điểm dữ liệu vào các nhóm tương ứng. Giả sử thêm rằng mỗi điểm dữ liệu chỉ thuộc vào đúng một nhóm.

Tóm tắt thuật toán:

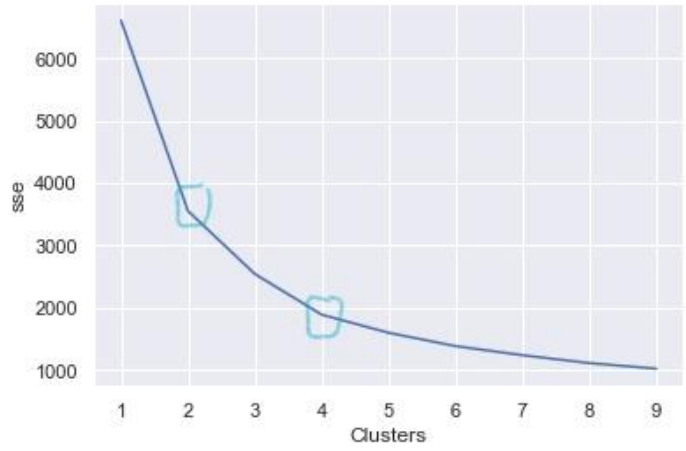
Đầu vào: dữ liệu X và số lượng cluster cần tìm K.

Đầu ra: các center M và label vector cho từng điểm dữ liệu Y.

1. Chọn K điểm bất kỳ làm các center ban đầu.
2. Phân mỗi điểm dữ liệu vào cluster có center gần nó nhất.
3. Nếu việc gán dữ liệu vào từng cluster ở bước 2 không thay đổi so với vòng lặp trước nó thì ta dừng thuật toán.
4. Cập nhật center cho từng cluster bằng cách lấy trung bình cộng của tất các điểm dữ liệu đã được gán vào cluster đó sau bước 2.
5. Quay lại bước 2.

Đối với bài toán phân khúc khách hàng:

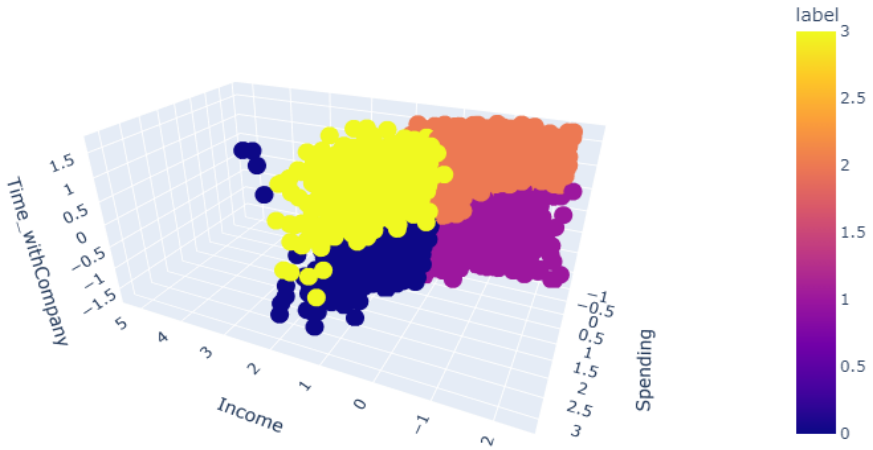
Xác định số cụm K bằng cách áp dụng phương pháp điểm khuỷu tay, như trong hình:



Điểm khuỷu tay là điểm mà đường đi qua nó bị gấp khúc và bị thay đổi (giống như khuỷu tay của người)

Ta có thể thấy các điểm khuỷu tay được khoanh tròn tương ứng là clusters = 2 và clusters = 4, ta sẽ chọn K = 4 vì dữ liệu qua điểm này bị thay đổi nhiều hơn so với điểm clusters = 2 và vì ta cần 4 cụm khách hàng để có thể bao quát hết được các khách hàng

Ta sẽ tiến hành phân cụm theo các bước của thuật toán Kmeans dựa vào các features đã chọn. Ta có kết quả như trong hình:



Ta cũng có thể tiếp cận với thuận toán Kmeans nhanh hơn thông qua thư viện sklearn.cluster

Đánh giá Models

Đối với Linear Regression Model có giá trị mae khá lớn do mẫu có nhiều điểm nhiễu.

Đối với Kmeans Algorithm ta có bảng kết quả như sau:

Cluster	Income	Spending	Time with company
0	High	High	Short time
1	Low	Low	Short time
2	Low	Low	A long time
3	High	High	A long time

Từ bảng kết quả thu được ta có thể thêm cột label vào bộ dữ liệu và có thể xác định được khách hàng thuộc nhóm nào để cung cấp ưu đãi cũng như giới thiệu sản phẩm dựa trên các phân khúc khách hàng

	Income	Spending	Time_withCompany	label
0	0.298240	1.681498	1.527764	3
1	-0.261178	-0.962057	-1.189818	1
2	0.937392	0.283240	-0.206547	0
3	-1.195501	-0.918829	-1.061351	1
4	0.305592	-0.305325	-0.952647	1

Đưa về dữ liệu ban đầu:

	Income	Spending	Time_withCompany	label
0	58138.0	1617	113.100000	3
1	46344.0	27	94.766667	1
2	71613.0	776	101.400000	0
3	26646.0	53	95.633333	1
4	58293.0	422	96.366667	1

Thư viện sử dụng trong đề tài học máy Phân tích tính cách khách hàng:

sklearn (https://scikit-learn.org/stable/)

seaborn (https://seaborn.pydata.org/index.html)

matplotlib (https://matplotlib.org/)

plotly (https://plotly.com/graphing-libraries/)

scipy (https://scipy.org/)

pandas (https://pandas.pydata.org/)

numpy (https://numpy.org/)

Công cụ sử dụng:

Jupyter notebook

Vscode

Github