**TRƯỜNG ĐẠI HỌC THỦY LỢI**

**Khoa Công nghệ thông tin**



**TÊN ĐỀ TÀI: Tìm Hiểu thuật toán hồi quy tuyến tính**

**Nhóm thực hiện: 1. Nguyễn Minh Chiều – 59Th1 - 175A071203**

**2.Ngô Văn Hải - 59Th1 - 175A07**

**Hà Nội, tháng 11 năm 2020**

Mục lục

[Phần 1: Tổng quan 2](#_Toc54860981)

[1. Giới thiệu về học máy 2](#_Toc54860982)

[2. Trình bày bài toán 3](#_Toc54860983)

[Đầu vào: 3](#_Toc54860984)

[Đầu ra: 4](#_Toc54860985)

[3. Trình bày phương pháp học máy được sử dụng trong bài tập lớn 4](#_Toc54860986)

[3.1. Giới thiệu 4](#_Toc54860987)

[3.2. Dạng của hồi quy tuyến tính 5](#_Toc54860988)

[3.3. Sai số dự đoán 5](#_Toc54860989)

[3.4. Hàm mất mát 5](#_Toc54860990)

[3.5. Nghiệm cho bài toán hồi quy tuyến tính 6](#_Toc54860991)

[Phần 2: Thực nghiệm 7](#_Toc54860992)

[1. Mô tả tập ví dụ huấn luyện và tập test 7](#_Toc54860993)

[2. Mô tả chương trình demo 8](#_Toc54860994)

[3. Mô tả kết quả dự đoán 14](#_Toc54860995)

[Kết luận 14](#_Toc54860996)

[Tài liệu tham khảo 15](#_Toc54860997)

# Phần 1: Tổng quan

## 1. Giới thiệu về học máy

Trong lĩnh vực trí tuệ nhân tạo có một nhánh nghiên cứu về khả năng tự học của máy tính được gọi là **học máy** (**machine learning**). Học máy (**machine learning**) có thể hiểu rằng nó là việc nghiên cứu và xây dựng các kỹ thuật giúp cho máy tính có thể tự học từ dữ liệu để giải quyết những vấn đề cụ thể mà không cần phải cài đặt các luật quyết định. Thường một chương trình máy tính cần các quy tắc, luật lệ để có thể thực thi được một tác vụ nào đó như dán nhãn cho các email là thư rác nếu nội dung email có chứa từ khoá “quảng cáo”. Nhưng với học máy, các máy tính có thể tự động phân loại các thư rác mà không cần chỉ trước bất kỳ quy tắc nào cả.

Nói chi tiết hơn một chút thì học máy là phương pháp vẽ các đường thể hiện mối quan hệ của tập dữ liệu. Ví dụ như đường ngăn cách 2 loại dữ liệu cho nhãn khác nhau, đường thể hiện xu hướng của giá nhà phụ thuộc vào diện tích,vị trí hay các đường phân cụm dữ liệu…

Hiện nay học máy được áp dụng rộng rãi bao gồm máy truy tìm dữ liệu, chẩn đoán y khoa, phát hiện thẻ tín dụng giả, phân tích thị trường chứng khoán, phân loại các chuỗi DNA, nhận dạng tiếng nói và chữ viết, dịch tự động, chơi trò chơi và cử động rô-bốt (robot locomotion)…

Các giải thuật học máy được phân ra làm 2 loại chính là:

+ Học có giám sát (Supervised Learning): Là phương pháp sử dụng những dữ liệu đã được gán nhãn từ trước để suy luận ra quan hệ giữa đầu vào và đầu ra. Các dữ liệu này được gọi là dữ liệu huấn luyện và chúng là cặp các đầu vào-đầu ra. Học có giám sát sẽ xem xét các tập huấn luyện này để từ đó có thể đưa ra dự đoán đầu ra cho 1 đầu vào mới chưa gặp bao giờ. Ví dụ dự đoán giá nhà, phân loại email…

+ Học không giám sát (Unsupervised Learning): Khác với học có giám sát, học không giám sát sử dụng những dữ liệu chưa được gán nhãn từ trước để suy luận. Phương pháp này thường được sử dụng để tìm cấu trúc của tập dữ liệu. Tuy nhiên lại không có phương pháp đánh giá được cấu trúc tìm ra được là đúng hay sai. Ví dụ như phân cụm dữ liệu, chiết xuất thành phần chính của một chất nào đó.

Các giải thuật học máy được phân ra làm 2 loại chính là:

* Học có giám sát (Supervised Learning): Là phương pháp sử dụng những dữ liệu đã được gán nhãn từ trước để suy luận ra quan hệ giữa đầu vào và đầu ra. Các dữ liệu này được gọi là dữ liệu huấn luyện và chúng là cặp các đầu vào-đầu ra. Học có giám sát sẽ xem xét các tập huấn luyện này để từ đó có thể đưa ra dự đoán đầu ra cho 1 đầu vào mới chưa gặp bao giờ. Ví dụ dự đoán giá nhà, phân loại email.
* Học không giám sát (Unsupervised Learning): Khác với học có giám sát, học không giám sát sử dụng những dữ liệu chưa được gán nhãn từ trước để suy luận. Phương pháp này thường được sử dụng để tìm cấu trúc của tập dữ liệu. Tuy nhiên lại không có phương pháp đánh giá được cấu trúc tìm ra được là đúng hay sai. Ví dụ như phân cụm dữ liệu, chiết xuất thành phần chính của một chất nào đó.

Trong đề tài này nhóm em lựa chọn phương pháp hồi quy tuyến tính (Linear Regression) thuộc học có giám sát để thực hiện dự đoán chi phí bảo hiểm y tế.

## 2. Trình bày bài toán

Bài toán đặt ra là: liệu có thể dự đoán cân nặng của một người dựa vào yếu tố như : tuổi của họ không? (Trên thực tế, tất nhiên là không, còn phụ thuộc vào nhiều yếu tố khác nữa, chẳng hạn như mức độ sinh hoạt ).

Ở bài toán này chúng em sẽ giả sử rằng chúng ta có thể dự đoán được để so sánh giữa kết quả thực tế và kết quả dự đoán từ đó tính ra độ sai lệch thông qua hàm mất mát

### Input:

### Tập ví dụ huấn luyện tuổi cân nặng (từ 1-6 tuổi)

### Output:

Dự đoán cân nặng qua phương pháp Linear Regression (Hồi quy tuyến tính)

## 3. Trình bày phương pháp học máy được sử dụng trong bài tập lớn

### 3.1. Giới thiệu

Phương pháp học máy được sử dụng là hồi quy tuyến tính (Linear Regression) thuộc nhóm Supervised learning ( Học có giám sát )

* "Hồi quy tuyến tính" là một phương pháp thống kê để hồi quy dữ liệu với biến phụ thuộc có giá trị liên tục trong khi các biến độc lập có thể có một trong hai giá trị liên tục hoặc là giá trị phân loại. Nói cách khác "Hồi quy tuyến tính" là một phương pháp để dự đoán biến phụ thuộc (Y) dựa trên giá trị của biến độc lập (X). Nó có thể được sử dụng cho các trường hợp chúng ta muốn dự đoán một số lượng liên tục. Ví dụ, dự đoán giao thông ở một cửa hàng bán lẻ, dự đoán thời gian người dùng dừng lại một trang nào đó hoặc số trang đã truy cập vào một website nào đó v.v…
* Bài toán thực tế :

G/S chúng ta đã có số liệu thống kê từ 1338 mẫu bao gồm : x1 (Tuổi), x2 (Giới tính) , x3 (BMI) , x4 (Trẻ em), x5 (Người hút thuốc) , x6 (Khu vực) , x7 (Chi phí). Mục tiêu là mô hình hóa chi phí bảo hiểm y tế x7 dựa trên các yếu tố tác động chính xung quanh đối tượng ?

* Hàm dự đoán y = f(x) có dạng thế nào?
* Ở đây x = [x1,x2,x3,x4,x5,x6] là một vector hàng chứa thông tin input, 𝑦 là một số vô hướng (scalar) biểu diễn output.
* Một hàm số có thể mô tả mối quan hệ giữa chi phí bảo hiểm y tế với 6 thuộc tính đầu vào



* + f (x) = w1x1 + w2x2 + w3x3 + … + w6x6 + w0 (1)
  + Mối quan hệ y≈f(x) là một mối quan hệ tuyến tính (linear).
  + Bài toán trên là bài toán thuộc loại regression. Do đó, bài toán đi tìm các hệ số tối ưu {w1,w2,w3,..,w6,w0} được gọi là bài toán Linear Regression.
* **Chú ý 1:** y là giá trị thực của *outcome* (dựa trên số liệu thống kê chúng ta có trong tập *training data*), trong khi y^ là giá trị mà mô hình Linear Regression dự đoán được. Nhìn chung, y và y^ là hai giá trị khác nhau do có sai số mô hình, tuy nhiên, chúng ta mong muốn rằng sự khác nhau này rất nhỏ.
* **Chú ý 2:** *Linear* hay *tuyến tính* hiểu một cách đơn giản là *thẳng, phẳng*. Trong không gian hai chiều, một hàm số được gọi là *tuyến tính* nếu đồ thị của nó có dạng một *đường thẳng*. Trong không gian ba chiều, một hàm số được goi là *tuyến tính* nếu đồ thị của nó có dạng một *mặt phẳng*. Trong không gian nhiều hơn 3 chiều, khái niệm *mặt phẳng* không còn phù hợp nữa, thay vào đó, một khái niệm khác ra đời được gọi là *siêu mặt phẳng* (*hyperplane*). Các hàm số tuyến tính là các hàm đơn giản nhất, vì chúng thuận tiện trong việc hình dung và tính toán. Chúng ta sẽ được thấy trong các bài viết sau, *tuyến tính* rất quan trọng và hữu ích trong các bài toán Machine Learning. Kinh nghiệm cá nhân tôi cho thấy, trước khi hiểu được các thuật toán *phi tuyến* (non-linear, không phẳng), chúng ta cần nắm vững các kỹ thuật cho các mô hình *tuyến tính*.

### 3.2. Dạng của hồi quy tuyến tính

Trong phương trình (1) phía trên, nếu chúng ta đặt :

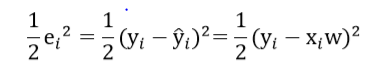
* w = [w1, w2, w3, … , w6, w0]T là vector (cột) hệ số cần phải tối ưu
* x =[ 1, x1, x2, x3,..., x6] là vector (hàng) dữ liệu input
* Số 1 ở đầu được thêm vào để phép tính đơn giản hơn và thuận tiện cho việc tính toán

 Khi đó, phương trình (1) có thể được viết lại dưới dạng:

y ≈ xw = y^

### 3.3. Sai số dự đoán

Chúng ta mong muốn rằng sự sai khác e giữa giá trị thực y và giá trị dự đoán y^ là nhỏ nhất. Nói cách khác, chúng ta muốn giá trị sau đây càng nhỏ càng tốt:

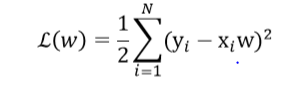


Trong đó hệ số ½ là để thuận tiện cho việc tính toán (khi tính đạo hàm thì số 1/2 sẽ bị triệt tiêu).

Chúng ta cần e2 vì ei=yi−y^i có thể là một số âm, việc nói ei nhỏ nhất sẽ không đúng vì khi e=−∞ là rất nhỏ nhưng sự sai lệch là rất lớn

### 3.4. Hàm mất mát

* Điều tương tự xảy ra với tất cả các cặp *(input, outcome)* (xi , yi ), i=1,2,…N , với N là số lượng dữ liệu quan sát được . Điều chúng ta muốn, tổng sai số là nhỏ nhất, tương đương với việc tìm w để hàm số sau đạt giá trị nhỏ nhất:



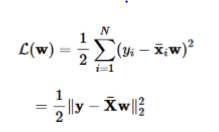
* Hàm số L(w) được gọi là **hàm mất mát** (loss function) của bài toán Linear Regression. Chúng ta luôn mong muốn rằng sự mất mát (sai số) là nhỏ nhất, điều đó đồng nghĩa với việc tìm vector hệ số w sao cho giá trị của hàm mất mát này càng nhỏ càng tốt. Giá trị của w làm cho hàm mất mát đạt giá trị nhỏ nhất được gọi là *điểm tối ưu* (optimal point), ký hiệu:



* Đặt : y = [y1;y2;…yN] là một véctơ cột chứa tất cả các output của dữ liệu huấn luyện

X = [x1;x2;…xN] là ma trận dữ liệu đầu vào mà mỗi hàng là một điểm dữ liệu

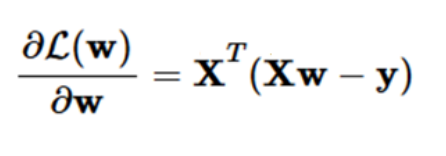
* Khi đó hàm số mất mát L(w) được viết dưới dạng ma trận đơn giản hơn:



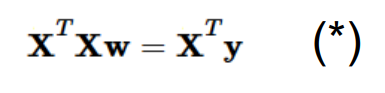
* với ∥z∥2 là Euclidean norm (chuẩn Euclid, hay khoảng cách Euclid), nói cách khác ∥z∥2 là tổng của bình phương mỗi phần tử của vector z.

### 3.5. Nghiệm cho bài toán hồi quy tuyến tính

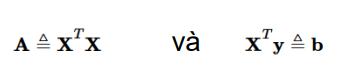
* Cách phổ biến nhất để tìm nghiệm cho một bài toán tối ưu là giải phương trình đạo hàm (gradient) bằng 0. Tất nhiên đó là khi việc tính đạo hàm và việc giải phương trình đạo hàm bằng 0 không quá phức tạp. Thật may mắn, với các mô hình tuyến tính, hai việc này là khả thi.
* Đạo hàm theo w của hàm mất mát là:



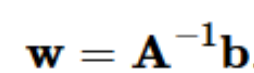
* Phương trình đạo hàm bằng 0 tương đương với:



Đặt:



* Nếu ma trận vuông A khả nghịch thì phương trình (\*) có nghiệm duy nhất



* Nếu ma trận A không khả nghịch (có định thức bằng 0) , trong trường hợp này thì hoặc phương trình vô nghiệm, hoặc là nó có vô số nghiệm

# Phần 2: Thực nghiệm

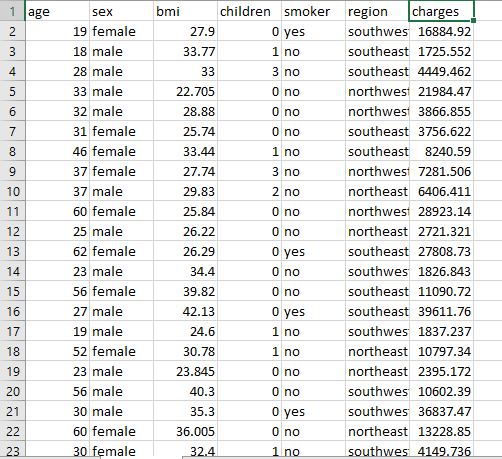
## 1. Mô tả tập ví dụ huấn luyện và tập test

Tâp dữ liệu đầu vào là bộ dữ liệu cho ’bảo hiểm’ y tế được lấy từ kho thông tin kaggle. Mục tiêu là mô hình hóa “chi phí “ cho bảo hiểm y tế dựa trên các dữ liệu được thu thập trước đó.

- Số trường hợp : 1338

- Số thuộc tính : 7

Các biến đầu vào là age = tuổi, sex = giới tính, bmi = Chỉ số khối cơ thể, children = Trẻ em, smoker = Người hút thuốc, region = Khu vực . Và biến đầu ra (dựa trên dữ liệu cảm giác) là charge = chi phí bảo hiểm y tế (usd). Dưới đây là một ảnh chụp màn hình của 1 phần của bộ dữ liệu :



Ở bài toán này , ta chia tập dữ liệu thành 2 tập : Huấn luyện (train) và tập thử nghiệm (test). Đào tạo tập huấn luyện và tập kiểm tra bằng cách sử dụng 30% dữ liệu được sử dụng để thử nghiệm và 70% cho huấn luyện.

## 2. Mô tả chương trình demo

\* Hiểu dữ liệu

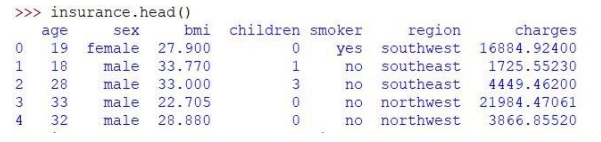
Input:

import pandas as pd

insurance = pd.read\_csv("insurance.csv")

insurance.head()

Output :

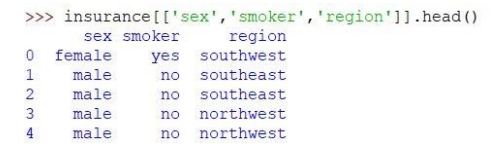


Lưu ý: - Các thuật toán hồi quy dường như chỉ hoạt động trên các tính năng được biểu thị dưới dạng số.

- Bằng cách nhìn vào tập dữ liệu, ta thấy các cột ‘sex’, ‘smoker’, ‘region’ có định dạng chuỗi nên ta có thể chuyển đổi thành các giá trị số sau đây:

#Hiển thị 5 dòng đầu

insurance[[‘sex’,’smoker’,’region’]].head()



# Thay thế giá trị string thành số

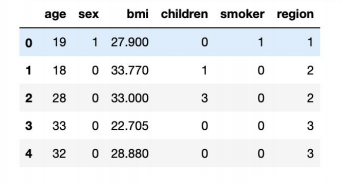
Input :

insurance['sex'] = insurance['sex'].apply({'male':0, 'female':1}.get)

insurance['smoker'] = insurance['smoker'].apply({'yes':1, 'no':0}.get) insurance['region'] =

insurance['region'].apply({'southwest':1, 'southeast':2, 'northwest':3, 'northeast':4}.get)

Output :



- Tạo một số sơ đồ đơn giản để kiểm tra dữ liệu.

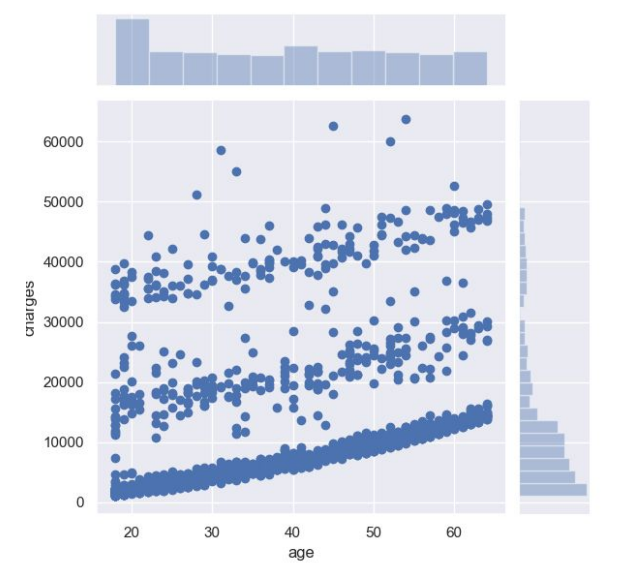
import seaborn as sns

# Sự tương quan giữa 'charges' và 'age'

Input :

sns.jointplot(x=insurance['age'],y=insurance['charges'])

OutPut



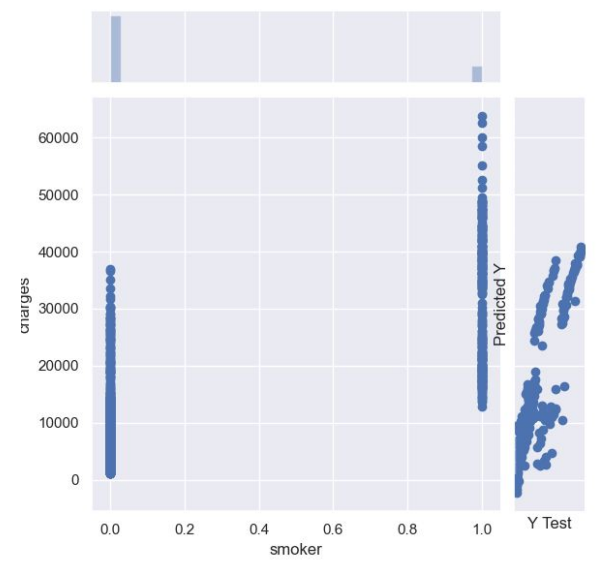
-Nhận xét : Ta thấy ở đây khi tuổi (age) tăng thì phí bảo hiểm y tế cũng có xu hướng tăng.

#Sự tương quan giữa 'charges' and 'smoker'

Input :

sns.jointplot(x=insurance['age'],y=insurance['charges'])

Output :



Nhận xét : Ta thấy phí BHYT cho người hút thuốc (smoker) cao hơn so với người ko hút thuốc (non-smoker)

\* Training mô hình hồi quy tuyến tính

**Bước 1:** Đầu tiên sẽ chia dữ liệu thành mảng 'X' có chứa các đặc trưng và mảng 'y' với biến mục tiêu.

Input :

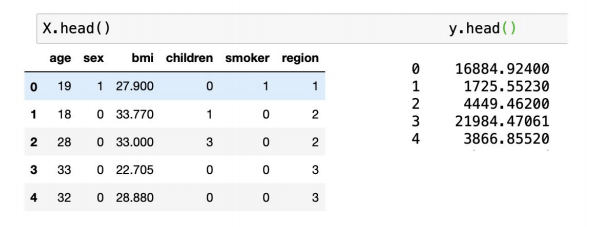
#Các đặc trưng

X = insurance[['age', 'sex', 'bmi', 'children','smoker','region']]

#Biến dự đoán

y = insurance['charges']

Output



**Bước 2:** Ta sẽ chia tập dữ liệu (insurance.csv) thành tập huấn luyện(training set) và tập kiểm tra (testing set). Sau đó, ta sẽ huấn luyện (train) mô hình trên tập huấn luyện và sau đó sử dụng tập kiểm tra để đánh giá mô hình (Dự đoán biến 'y').

Đầu tiên nhập các thư viện cần thiết :

Input :

#Nhập mô hình

from sklearn.model\_selection import train\_test\_split

#Tách train và test data

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.3)

'Test\_size' thể hiện rằng 30% dữ liệu chuyển đến test data và phần còn lại tới train data.

Có thể nhanh chóng kiểm tra số lượng hồ sơ cho cả tập dữ liệu đào tạo và tập dữ liệu thử nghiệm.

Output :

len(X\_test) # 402

len(X\_train) # 936

len(Y\_Pred) # 1338

**Bước 3:** Huấn luyện và kiểm tra mô hình

Bây giờ, ta có một bộ dữ liệu test và train, chúng ta có thể đánh giá mô hình bằng cách sử dụng hồi quy tuyến tính như dưới đây.

#Nhập mô hình

from sklearn.linear\_model import LinearRegression

LR= LinearRegression()

#Fit hồi quy tuyến tính bằng chuyển training dataset

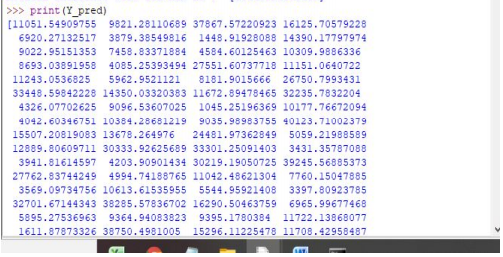
LR.fit(X\_train,Y\_train)

**Bước 4:** Dự đoán từ mô hình

#Dự đoán biến target cho test datset

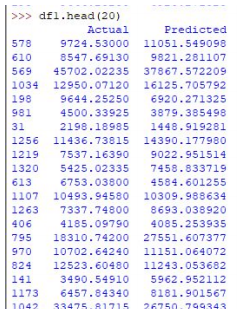
Y\_pred = model.predict(X\_test)

Ở đây chúng ta chuyển 'X\_test' cho mô hình để dự đoán 'Y\_test'



**Bước 5:** So sánh kết quả

Ta kiểm tra sự khác biệt giữa giá trị thực tế và giá trị dự đoán.



So sánh các kết quả 'dự đoán' này với kết quả ‘thực tế’ bằng cách vẽ đồ thị.

Input :

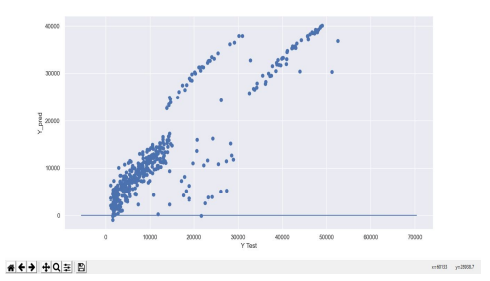
import matplotlib.pyplot as plt

plt.scatter(Y\_test,Y\_pred)

plt.xlabel('Y Test')

plt.ylabel('predicted Y')

Output:



Tiếp đó ta tính độ xác thực :

Input :

from sklearn.model\_selection import cross\_val\_score, train\_test\_split

r2\_score(Y\_test, Y\_pred)

Output :

0.7911113876316934

Lỗi bình phương gốc

Input :

from sklearn.metrics import mean\_squared\_error

import math

from math import sqrt

mse = mean\_squared\_error(Y\_test, Y\_pred)

rmse = sqrt(mse)

Output : Root Mean Squared Error: 5672.5601216471205

**Bước 6:**

Bây giờ thử dự đoán phí bảo hiểm của khách hàng mới.

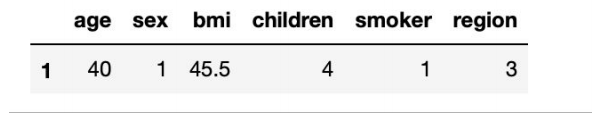
#Dự đoán phí BHYT cho khách hàng mới: Khách hàng tên Nam

Input : data = {'age' : 40, 'sex' : 1, 'bmi' : 45.50, 'children' : 4, 'smoker' : 1, 'region' : 3}

index = [1]

Nam\_df = pd.DataFrame(data,index)

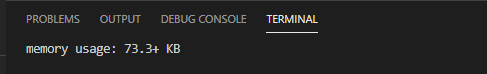
Nam\_df



prediction\_Nam = model.predict(Nam\_df)

print("Medical Insurance cost for Nam is : ",prediction\_Nam)

Output :



## 3. Mô tả kết quả dự đoán

Với lượng dữ liệu thu thâp và kết quả sau tính toán với phương pháp hồi quy tuyến tính ta có kết quả :

- Cho ra được biểu đồ tương quan giữa các yếu tố ảnh hưởng đến chi phí bảo hiểm

- Đồ thị so sánh giữa kết quả dự đoán và kết quả thực tế

- Sai số kiểm tra sự khác biệt giữa giá trị thực tế và giá trị dự đoán

- Kết quả thực tế luôn bị ảnh hưởng bởi nhiều yếu tố khách quan khác nhau

# Kết luận

* Mục tiêu của hồi quy là xác định các giá trị của các tham số để tối thiểu hóa tổng các giá trị còn lại bình phương cho tập hợp các quan sát.
* Do hồi quy tuyến tính bị hạn chế trong việc kết hợp các hàm tuyến tính (đường thẳng / mặt phẳng) với dữ liệu, nên nó không phù hợp với dữ liệu trong thế giới thực vì các kỹ thuật tổng quát hơn như mạng thần kinh có thể mô hình hóa các hàm phi tuyến tính. Nhưng hồi quy tuyến tính có một số lợi thế thú vị:

● · Hồi quy tuyến tính là phương pháp được sử dụng rộng rãi nhất, và nó được hiểu rõ.

● · Đào tạo một mô hình hồi quy tuyến tính thường nhanh hơn nhiều so với các phương pháp như mạng lưới thần kinh.

● · Các mô hình hồi quy tuyến tính rất đơn giản và yêu cầu bộ nhớ tối thiểu để thực hiện, vì vậy chúng hoạt động tốt trên các bộ điều khiển nhúng có không gian bộ nhớ hạn chế.

● · Bằng cách kiểm tra độ lớn và dấu hiệu của các hệ số hồi quy, bạn có thể suy ra cách biến dự báo ảnh hưởng đến kết quả mục tiêu.

# Tài liệu tham khảo

1. Slides bài giảng học máy ( PGS.TS Nguyễn Hữu Quỳnh)

<https://machinelearningcoban.com/>

1. https://www.kaggle.com/mirichoi0218/insurance