TRƯỜNG ĐẠI HỌC QUY NHƠN

KHOA CÔNG NGHỆ THÔNG TIN



**BÁO CÁO**

**KĨ THUẬT HỒI QUY TRONG KPDL**

Sinh Viên: Võ Duy Long, Nguyễn Văn Hòa

Lớp: Công Nghệ Thông Tin K43D

Chuyên ngành: Hệ Thống Thông Tin

Giảng Viên: ThS. Trần Thị Liên

Quy Nhơn, 06/2022

MỤC LỤC

**I: GIỚI THIỆU2**

**II: HỒI QUY TUYẾN TÍNH3**

1. Giới thiệu mô hình3
2. Hàm chi phi8
3. Cập nhập lại α và β 8
4. Giải thuật xây dựng mô hình 9
5. Xây dựng mô hình hồi quy tuyến tính đa biến 10

**III: HỒI QUY LOGISTIC26**

1. Khái niệm26
2. Hàm Logistic (sigmoid function)26
3. Hồi quy logistic là thuật toán phân loại tuyến tính27
4. Diễn giải tham số của mô hình hồi quy logistic28
5. Xác định sai số dự báo29
6. Ước lượng tham số - Phương pháp hợp lý cực đại 35

**I. Giời thiệu**

Hồi quy (Regression) là một khái niệm trong thống kê và học máy, được sử dụng để mô hình hóa mối quan hệ giữa biến phụ thuộc (đầu ra) và một hoặc nhiều biến độc lập (đầu vào). Hồi quy cố gắng tìm ra mô hình toán học hay cụ thể là tìm một hàm số bằng cách phân tích bộ dữ liệu được cung cấp. Mục tiêu cuối cùng của mô hình hồi quy là dự đoán biến phụ thuộc dựa vào các biến đầu ra.

Các thuật toán hồi quy phổ biến:

* **Hồi quy tuyến tính**
* **Hồi quy Logistic**
* Hồi quy đa thức
* Hồi quy ngưỡng
* Mạng nueral
* Hồi quy Ridge và Lasso
* …

**II: Hồi quy Tuyến tính:**

**1. Giới thiệu mô hình**

Hồi quy tuyến tính hay **Linear Regessoin** là một thuật toán học máy có giám sát được xây dựng dùng để dự đoán một giá trị đầu ra liên tiếp (biến phụ thuộc) dựa trên các giá trị đầu vào (biến độc lập).

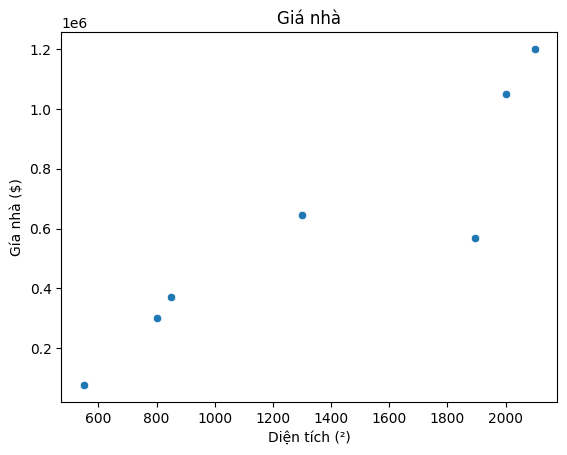
Ví dụ, ta có một mô hình hồi quy dự đoán giá nhà. Ta cung cấp biến độc lập ở đây là **diện tích** ngôi nhà, công việc của mô hình là cần dự đoán giá trị của biến phụ thuộc hay **giá nhà** dựa trên diện tích đã cho. Tất nhiên mô hình này được đào tạo bằng bộ dữ liệu tương đương là **diện tích** - **giá nhà** (trên thực tế, giá của ngôi nhà còn vào phụ thuộc nhiều yếu tố khác như: số phòng, vị trí địa lí, năm xây dựng, … bấy giờ ta sẽ xây dựng mô hình hồi quy tuyến tính đa biến **multiple linear regression**, có nghĩa biến độc lập lúc này bao gốm nhiều yếu tố)

Chú ý rằng, biến phụ thuộc phải là liên tục như câng nặng, tiền lương, giá cả, … chứ không thể phân lớp như giới tính (nam – nữ), tốt nghiệp (có – không). Riêng với các biến độc lập có thể liên tục hoặc phân lớp

Ví dụ: ta có dữ liệu về giá bán nhà như sau:

|  |  |  |
| --- | --- | --- |
|  | Diện tích(m²) | Giá($) |
| 0 | 2.000 | 1.050.000 |
| 1 | 800 | 300.000 |
| 2 | 850 | 370.000 |
| 3 | 550 | 78.000 |
| 4 | 2.100 | 1.200.000 |
| 5 | 1.300 | 645.000 |
| 6 | 1.893 | 567.000 |

Biểu diễn dữ liệu này bằng biểu đồ Scatter trong Python



Hình 1

Giả sử giờ ta có dữ liệu của một ngôi nhà mới với diện tích là 1540 m² thì ta cần dự đoán xem giá của ngôi nhà này là bao nhiêu dựa trên những dữ liệu đã cho.

Trước tiên, ta sẽ định nghĩa vài kí hiệu:

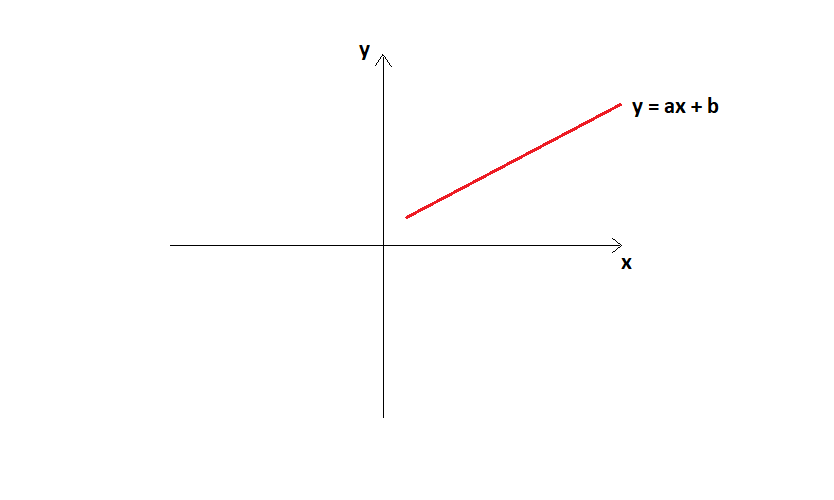
* xi : biến độc lập
* yi : biến phụ thuộc
* (xi,yi) : một cặp huấn luyện
* i = 1, 2, 3, …, n
* n là số cặp huấn luyện (x,y)

Với mô hình hồi quy tuyến tính, ta cố gắng xây dựng mối quan hệ giữa xi và yi, mối quan hệ này có thể được biểu diễn bằng một hàm giả thuyết

***yi = α*xi *+ β***

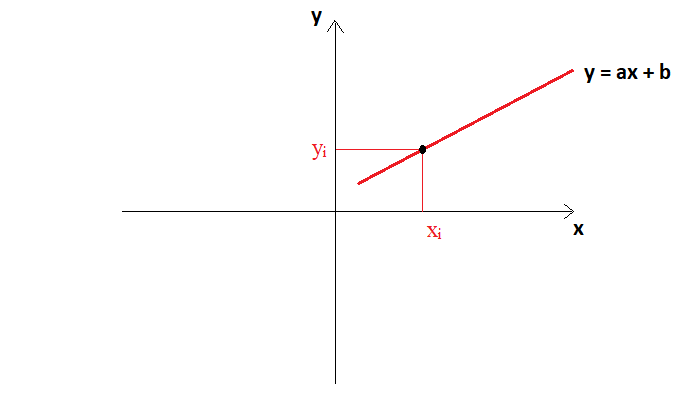
trong đó α và β là hai tham số của hàm mà ta chưa biết.

có thể thấy, hàm số phía trên là hàm bậc nhất mà nếu ta biểu diễn dưới không gian hai chiều thì sẽ được một đường thẳng với α và β được giả định cho trước



Hình 2

giả sử giờ có một xi bất kì, ta sẽ tính được yi tương ứng, cặp (xi,yi) được biểu diễn bằng một điểm trên đường thẳng y = αx + β

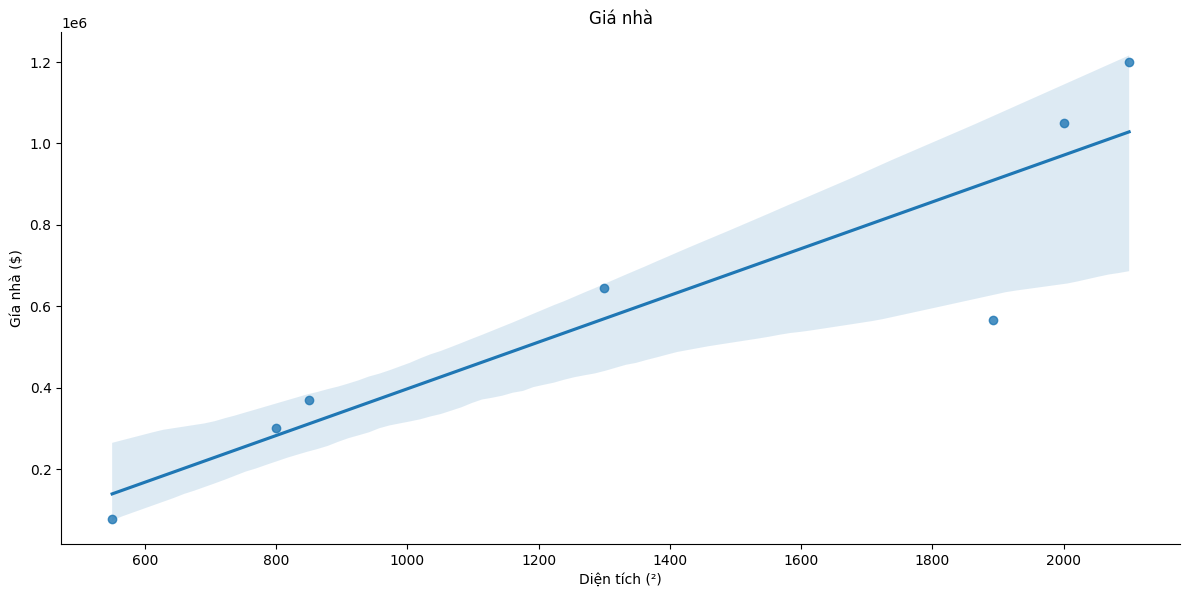


Hình 3

Quay lại với biểu đồ scatter giá nhà (hình 1) lúc nãy, nếu ta vẽ được một đường thẳng như vậy, ta dễ dàng dự đoán được giá của ngôi nhà mới dựa vào diện tích được cung cấp. Như ta thấy ở hình 3 cặp (xi,yi) nằm trên đường thẳng ***y = α*x *+ β*** hay nói cách khác đường thẳng này đi qua các cặp điểm (xi,yi). Với bộ dữ liệu có sẵn, các điểm (xi,yi) đã được cung cấp, làm thế nào ta có thể vẽ được đường thẳng

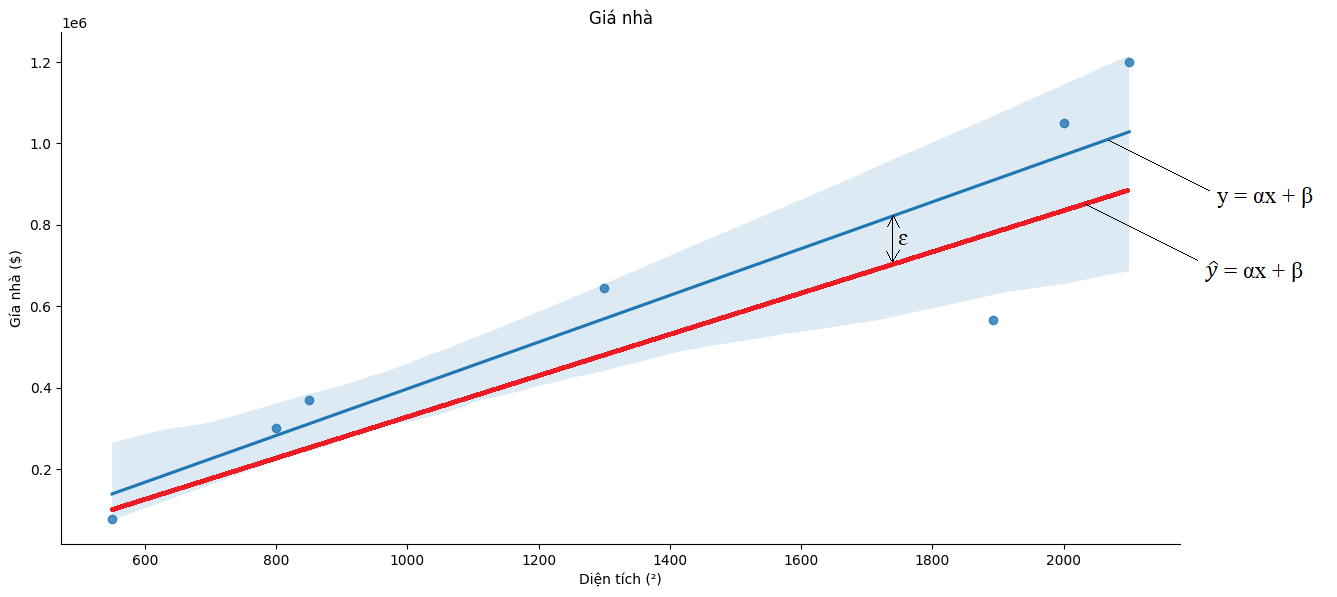
***y = α*x *+ β*** ?

để hình dùng, ta sẽ dùng thư viện seaborn của python để vẽ được đường thẳng như thế:



Hình 4

dễ dàng nhận thấy, đường thẳng này đi qua giữa các điểm (xi,yi). nếu không dùng thư viện thì sao, giả sử bằng mắt thường, ta thử vẽ một đường thẳng làm sao đi qua giữa các điểm trên biểu đồ



Hình 5

ta sẽ kí hiệu = αx + β là phương trình đường thẳng mà ta dự đoán, y = αx + β là phương trình đường thẳng thực tế. Có thể thấy rằng, mắt thường khó có thể kẻ được đường thẳng chính xác với thực tế, và giữa dự đoán với thực tế sẽ có một độ lệch ɛ= . Ta sẽ gọi độ lệch này là **lỗi**. Nếu ta giảm được ɛ về mức càng gần bằng 0 thì của ta càng chính xác. Vấn đề là làm sao để giảm giá ɛ này ?

rõ ràng là ta không thể thay đổ thông số x, y vì đây là dữ liệu. vì thế ta cần phải thay đổi hai hệ số α và β

**2. Hàm chi phí:**

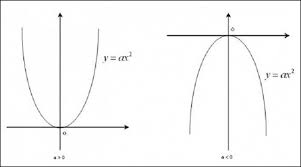
Với mỗi xi ta sẽ có giá trị dự đoán i  biểu diễn dưới hàm số i = αxi + β và giá trị y thực tế, lỗi của giá trị này là ɛi , tổng các lỗi được gọi là chi phí. Nếu ta giảm chi phí này xuống càng thấp thì giá trị dự đoán của ta càng chính xác. Trong mô hình hồi quy tuyến tính, ta sẽ huấn luyện mô hình với số lần nhất định, mỗi lần huấn luyện ta sẽ cập nhập lại hai tham số α và β và tính toán lại chi phí, đến khi chi phí giảm đển mức chấp nhận được, ta sẽ ngừng đào tạo và sử dụng α và β cuối cùng đề làm tham số cho phương trình y = αx + β, đến đây mô hình của ta được coi như hoàn tất.

Quay với lại hàm chi phí, ta sẽ dùng nguyên tắc **Trung bình Bình phương Lỗi – Mean Square Error (MSE)** để tính toán, mô tả bằng công thức như sau:

i – (αxi + β))2

**3. Cập nhập lai α và β:**

MSE là phương trình bậc 2, nếu ta biểu diễn trên không gian 2 chiều ta sẽ được một đường parabol có dạng



MSE đạt giá trị càng nhỏ khi đạo hàm của nó càng gần bằng 0, ta cần tính toán lại 2 tham số α và β (ban đầu ta sẽ tự cho khi xây dựng mô hình) dựa vào từng điểm trên đường cong parabol. Mỗi điểm này ta sẽ gọi là tốc độ học **learning rate**.

Nếu giải đạo hàm theo α và β ta sẽ được 2 phương trình:

Sau khi đã tính α’ và β’ mới, ta tiến hành cập nhập lại hai tham số này theo công thức với learning rate là giá trị của tốc độ học mà ta tự cho trước:

α = (α - α’) \* learning rate

β = (β – β’) \* learning rate

Đến cuối cùng, ta sẽ có α và β tối ưu nhất, hàm số của ta cũng như thế mà hoàn thành.

**4. Giải thuật xây dựng mô hình:**

**Đầu vào:**

- x: tập biến độc lập

- y: tập biến phụ thuộc

- α, β

- learning\_rate: tốc độ học

- loop: số lần học

- x\_new: biến độc lập mới, dùng đề dự đoán biến phụ thuộc mới

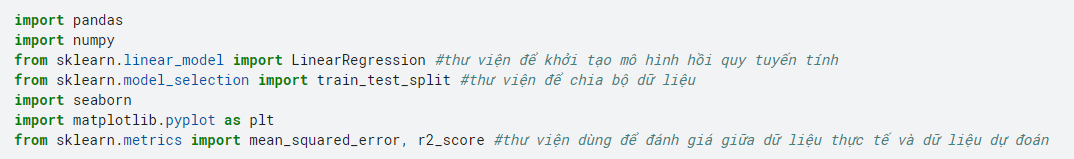
**Đầu ra:** giá trị phụ thuộc mới y\_new

|  |
| --- |
| **BEGIN:**  **for** mỗi giá trị **k** trong số lần học **loop**:  //tính toán hàm chi phi  **for** mỗi giá trị **i** trong độ dài len(x) của tập biến độc lập **x**:  tổng\_lỗi += (y[i] – (a \* x[i] + b))2  trung\_bình\_lỗi = tổng\_lỗi / len(x)  **end for**  //cập nhập lại α và β  **for** mỗi giá trị **i** trong độ dài len(x) của tập biến độc lập **x**:  tổng\_α += -2 \* x[i] \* (y[i] – (a \* x[i] + b))  tổng\_b += -2 \* (y[i] – (a \* x[i] + b))  **end for**  α = (tổng\_α / n) \* learning\_rate  β = (tổng\_β / n) \* learning\_rate  **end for**  //dự đoán giá trị phụ thuộc y\_new  **return** α \* x\_new + β  **END** |

**5. Xây dựng mô hình hồi quy tuyến tính đa biến (Multiple linear regression) với thư viện *sklearn.linear\_model***

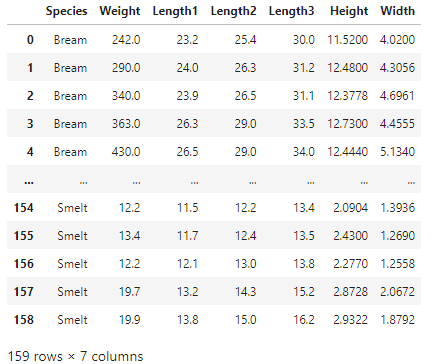
Chúng ta sẽ sử dụng bộ dữ liệu trong **thị thường cá** để mô tả việc xây dựng mô hình

* **Trước tiên ta sẽ import thư viện**



* Thư viện ***sklearn.linear\_model* import *LinearRegression***: đây là thư viện chính để xây dựng mô hình hồi quy tuyến tính
* Thư viện ***sklearn.model\_selection* import *train\_test\_split***: dùng để chia một bộ dữ liệu thành 2 phần **train** và **test**
* Thư viện ***sklearn.metric* import *mean\_squared\_error, r2\_score***: dùng để đánh giá dữ liệu giữa dự đoán và thực tế
* **Import data:**





Giải thích dữ liệu: bộ dữ liệu này mô tả các kích thước và cân nặng của từng loài cá

+ Species: tên của loài cá

+ Weight: cân nặng của cá

+ Length1, Length2, Length3, Height, Width: các kích thước của cá

🡺 Với bộ dữ liệu này ta sẽ xây dựng mô hình hồi quy tuyến tính để dự đoán Weight (câng nặng) của cá dựa trên các thông số về kích thước.

* **Thống kê và tiền xử lý dữ liệu (đây là bước quan trọng trước khi xây dựng mô hình):**

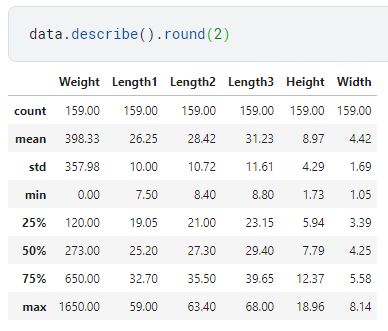
Cụ thể việc cần làm là:

+ Thay thế hoặc loại bỏ các giá trị rỗng

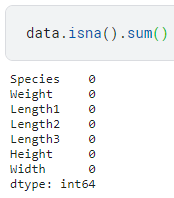
+ Mã hóa hoặc phân bin dữ liệu về dạng số (đối với dữ liệu phân lớp)

+ Xử lý outliers (ngoại lai)

* Thống kê dữ liệu:

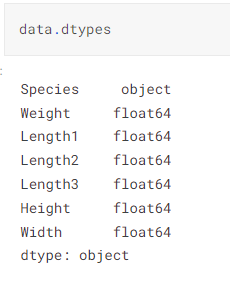


* Kiểm tra dữ liệu rỗng:

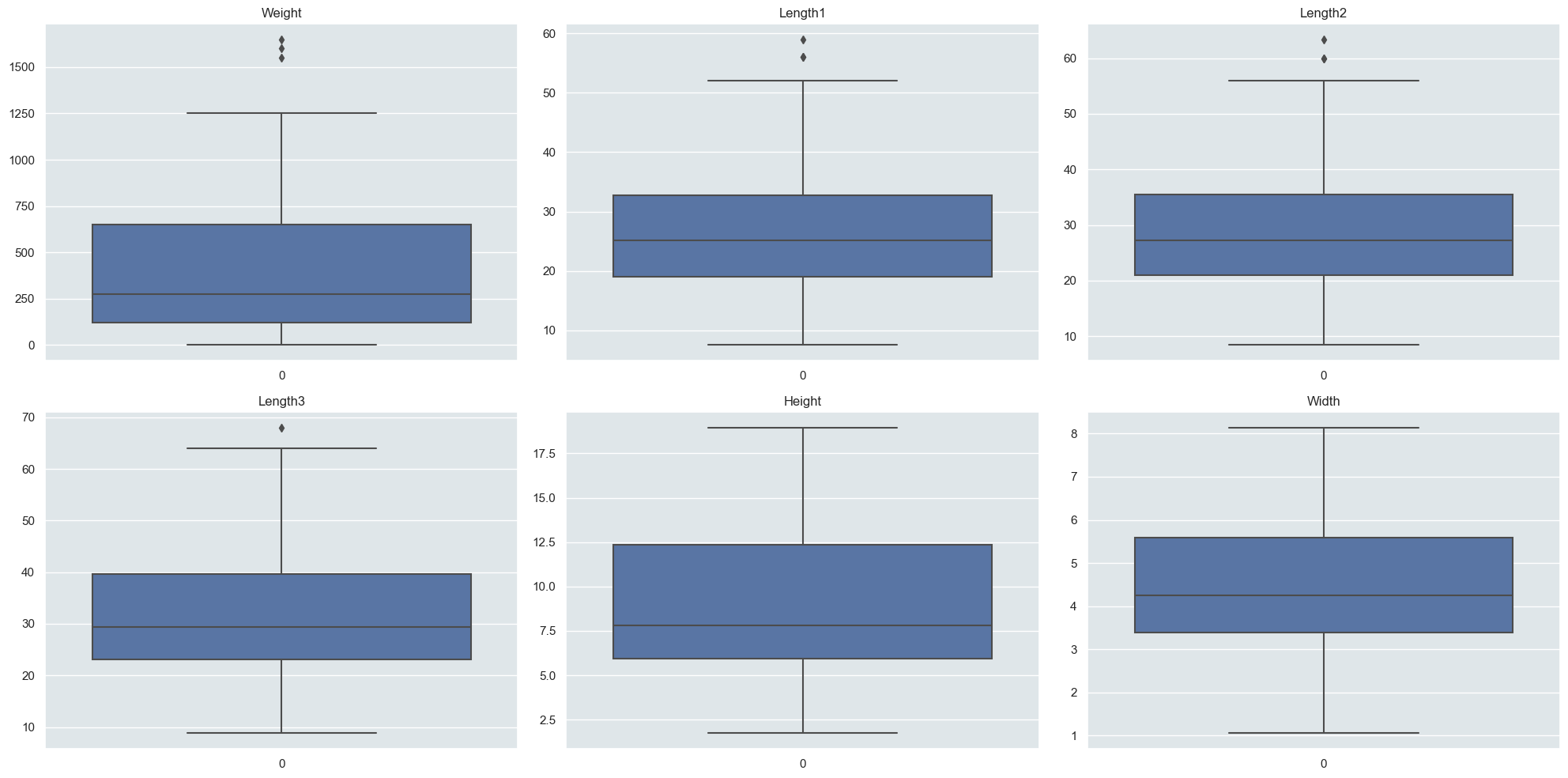


🡺 Không có giá trị rỗng trong các cột

* Kiểm tra dữ liệu nhiễu



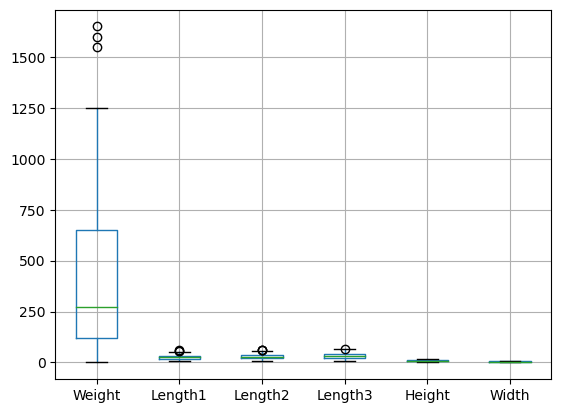
* Kiểm tra outliers (ngoại lai):Sử dụng biểu đồ boxplot để quan sát



|  |  |
| --- | --- |
|  |  |

Có thể dùng hàm boxplot của dataframe để xem ngoại lai của tất cả các cột (kiểu dữ liệu numeric và các thông số min max trong các cột không quá chênh lệch) trong dataframe:





🡺 Ta thấy các đặc trưng như: *Lenght1, Lenght2, Lenght3, Weight* chứa outliers*, Width* và *Height* thì không, ta tiến hành xử lý outliers

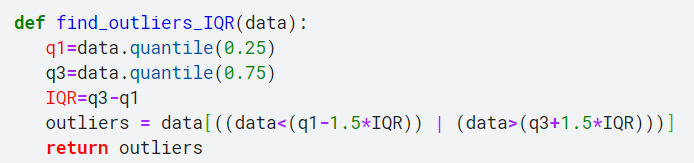
* **Loại bỏ outliers**:

Outliers ảnh hưởng rất nhiều đến tính đúng đắn của mô hình khi xây dựng. ta cần xử lý outliers bằng cách:

+ Tìm các vị trí chứa giá trị outliers

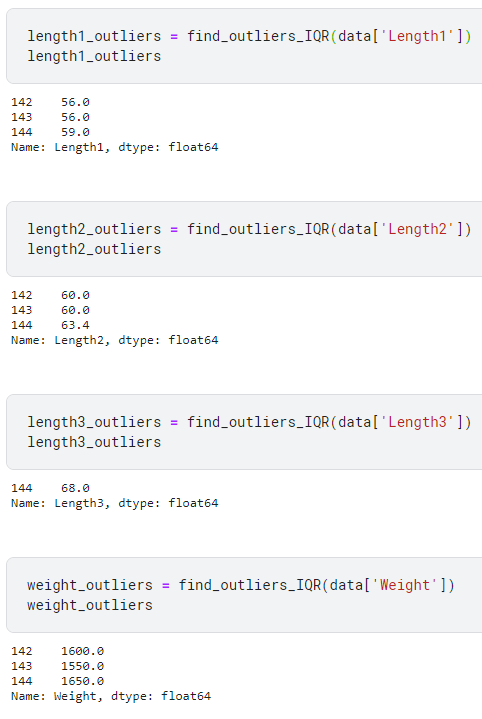
+ Kiểm tra, chỉnh sửa hoặc loại bỏ nếu cần thiết

* Dựng hàm tìm outliers dùng **interquartile range** **(IQR)**



Hàm nhận tham số đầu và là data cần tìm outliers, tính chỉ số IQR dựa và tứ vị phân và trả về outliers của data đó

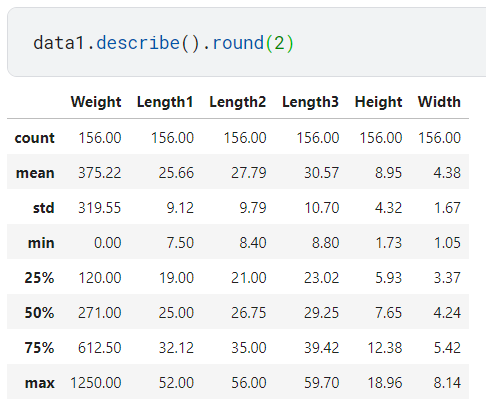
* Sử dụng hàm này và tìm outliers trên các đặc trưng



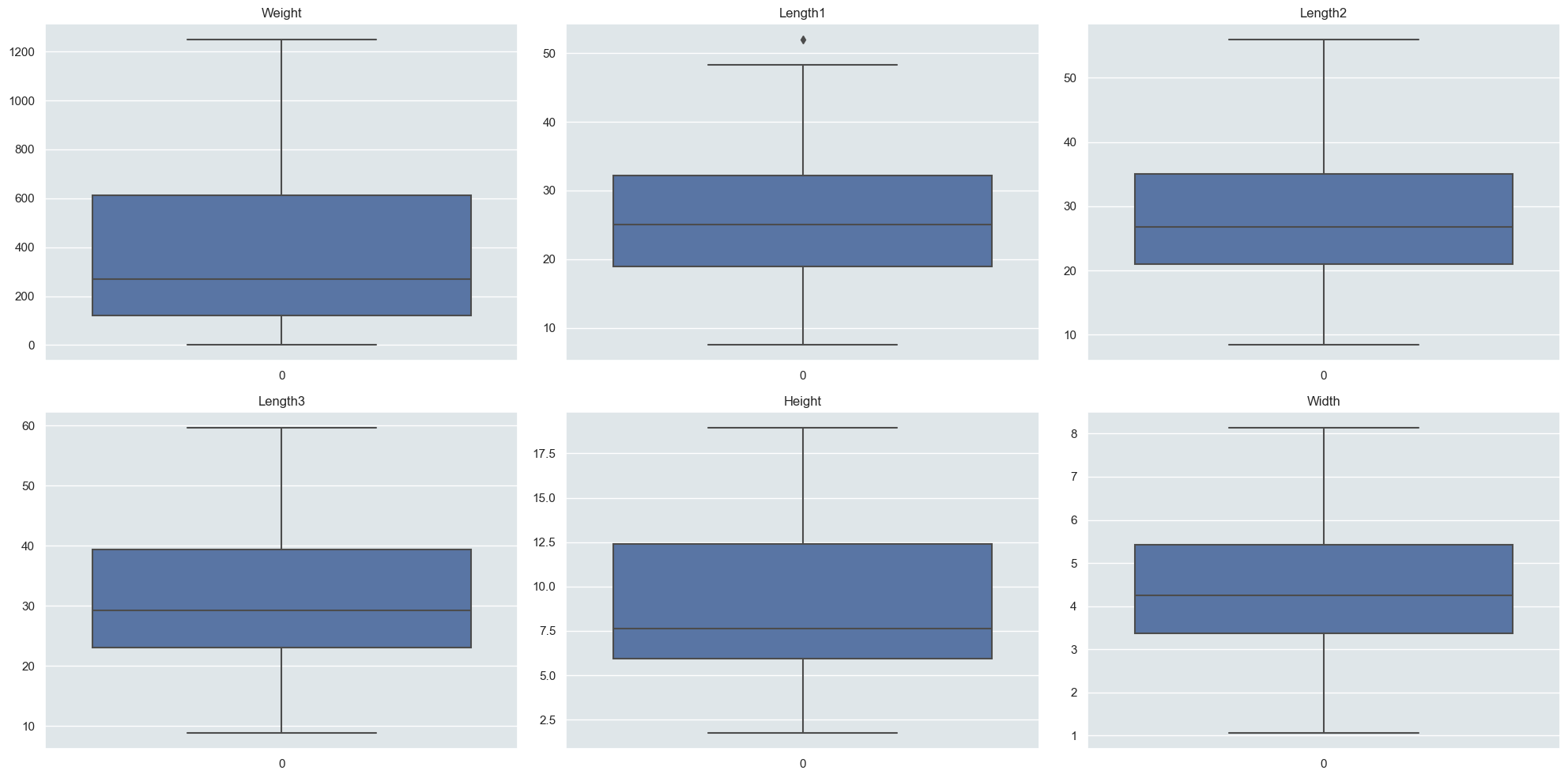
🡺 Nhận thấy rằng các hàng (records) tại index = [142,143,144] chứa giá trị ngoại lai trên tất cả các cột đang xét, vì vậy ta tiến hành loại bỏ những hàng này



* Thống kê lại dữ liệu mới

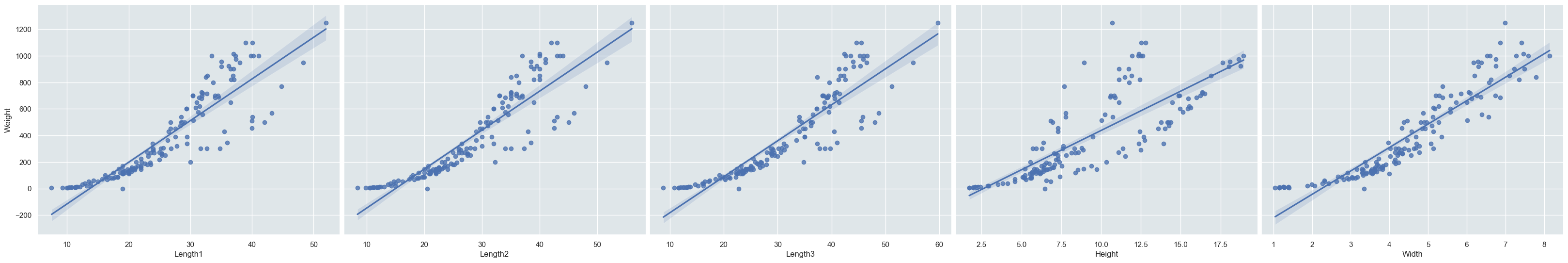


* Kiểm tra lại outliers trên từng đặc trưng bằng biểu đồ

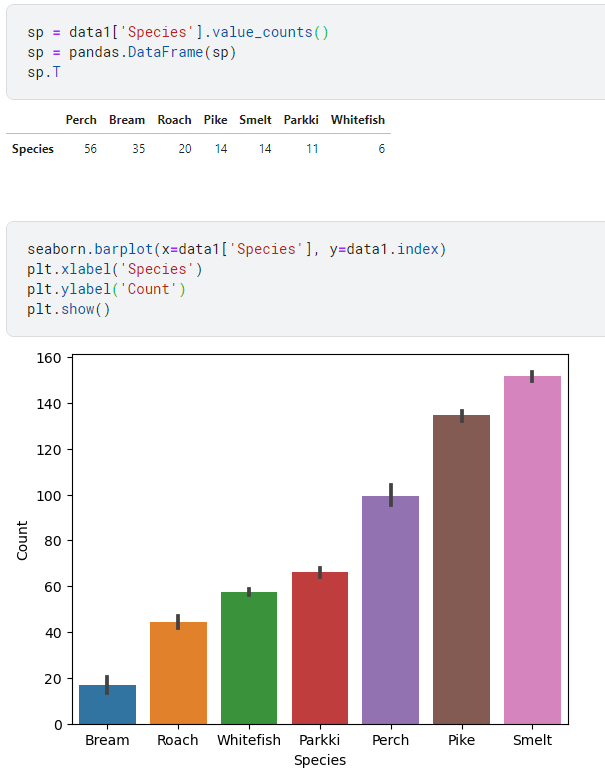


🡺 outliers đã được loại bỏ

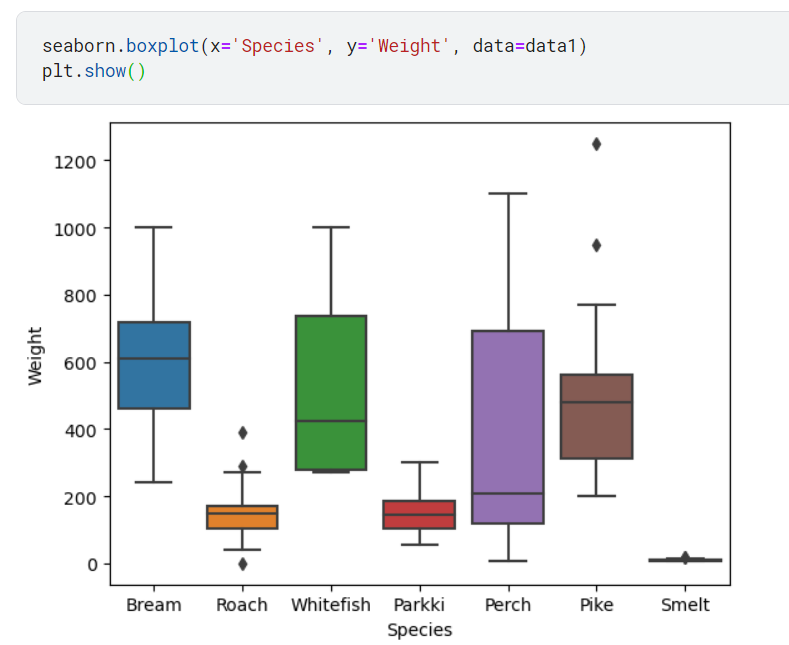
* Một vài biểu đồ để trực quan hóa dữa liệu
* biểu đồ lmplot giữa cột Weight và các đặc trưng về kích thước, cả 2 có sự phụ thuộc theo tỉ lệ thuận



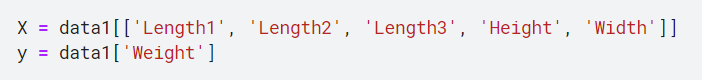
* Thống kê các loại cá và số lượng mỗi loài trong tập dữ liệu



* Biểu đồ boxplot giữa Species và Weight



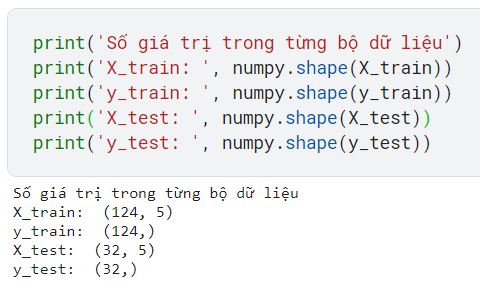
* **Xây dựng mô hình**
* Xác định biến độc lập và biến phụ thuộc, ở đây cân nặng của cá phụ thuộc vào kích thước nên biến phụ thuộc (y) và biến độc lập (X) sẽ là:



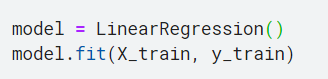
* Chia dữ liệu thành 2 phần (không giao nhau) **train** và **test** theo tỉ lệ 8:2



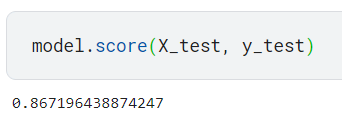
* Xem số giá trị trong từng bộ dữ liệu



* Xây dựng mô hình



* Dùng mô hình để đánh giá mức độ phụ thuộc của biến phụ thuộc với biến độc lập



🡺 0.86 tương đương với 86% giá trị của biến phụ thuộc phụ thuộc vào biến độc lập

* Hàm số giả thuyết của mô hình hồi quy tuyến tính đa biến (multiple linear regression) có dạng:

y = α1X1 + α2X2 + α3X3 + ... + αnXn + β

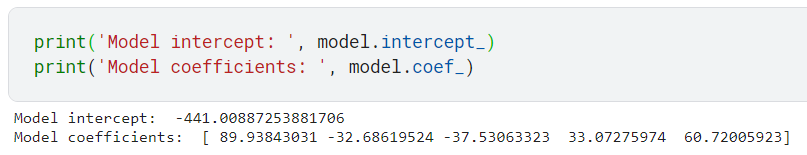
+ X1 – Xn: các biến độc lập

+ y: biến phụ thuộc

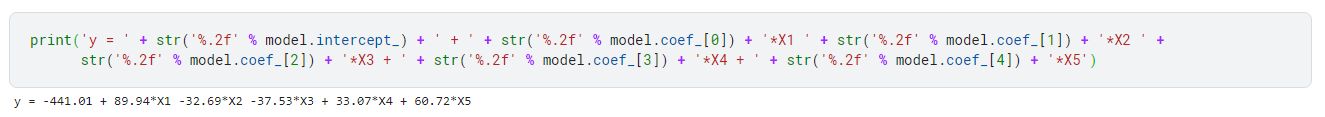
+ α và β: các tham số không thay đổi của hàm

+ n: số lượng biến độc lập

ta sẽ dùng hàm **intercept\_** và **coef\_** để tìm các giá trị **β** và **α**



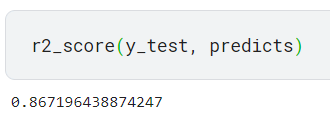
hiển thị hàm số giá thuyết



* **Dùng mô hình để dự đoán**

****

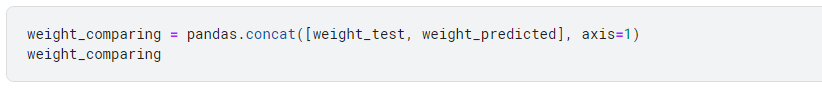
* **Đánh giá kết quả dự đoán**
* Đánh giá mức độ phụ thuộc của **predicts** và **y\_test**, ở đây là mức độ tương đương của các giá trị giữa 2 tập

****

**🡺** có 86% giá trị tương đương giữa 2 tập

* Tạo 1 **Dataframe** để so sánhgiữa 2 giá trị là **Weight** - thực tế và **Predicted Weight** - dự đoán

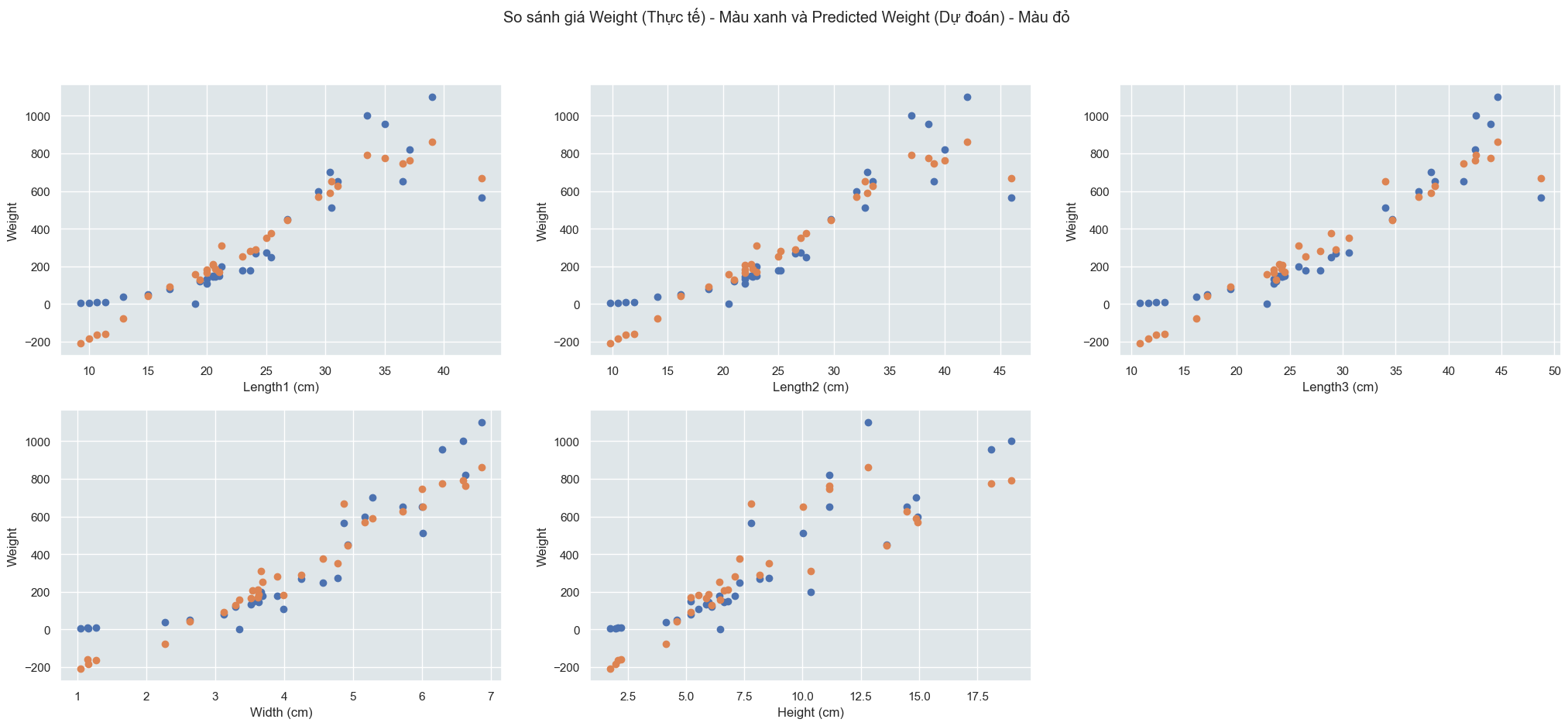






🡺 từ so sánh trên, ta có thể thấy các giá trị dự đoán có xu hướng sai khi Weight nhỏ

* Sử dụng biểu đồ scatter để xem xét mức độ chênh lệnh giữa dự đoán và thực tế



|  |  |
| --- | --- |
|  |  |

**III: Hồi quy Logistic:**

1. **Khái niệm**

Hồi quy Logistic (**Logistic Regression**) là một phương pháp hồi quy trong thống kê, được sử dụng để dự đoán xác suất xảy ra của một biến phụ thuộc nhị phân (binary) dựa trên các biến độc lập. Biến phụ thuộc trong hồi quy Logistic chỉ có hai giá trị đối với mỗi quan sát, ví dụ như 0 hoặc 1, true hoặc false.

*Ví dụ 1*: dựa vào kết quả thi của học sinh, ta sẽ dự đoán học sinh sẽ vượt qua kỳ thi hay thất bại dựa trên số giờ ngủ và số giờ học.

|  |  |  |
| --- | --- | --- |
| Studied | Slept | Passed |
| 4.85 | 9.63 | 1 |
| 8.62 | 3.23 | 0 |
| 5.43 | 8.23 | 1 |
| 9.21 | 6.34 | 0 |

*Ví dụ 2*: Dự báo xác suất KH có khả năng vỡ

* Biến phụ thuộc là vỡ nợ/không vỡ nợ (1/0)
* Dự báo xác xuất vỡ nợ là P(y=1)
* Dự báo xác xuất không vỡ nợ là P(y=0) = 1 – P(y=1)
* Các biến độc lập (x1, x2, …, xk) có thể là :
* Thu nhập hàng tháng
* Tuổi
* Tình trạng hôn nhân
* Tài sản bảo đảm

……..

1. **Hàm logistic (sigmoid function)**

**P(y=1) =**

|  |  |
| --- | --- |
| P: | Xác suất |
| y: | Biến phụ thuộc |
| x: | Biến độc lập |
| z: | hàm số |

A picture containing diagram

Description automatically generated

Hàm trả về giá trị xác suất trong khoảng (0,1), trong khi đó chúng ta cần dự báo ra giá trị y = 1/0 (vỡ nợ/ không vỡ nợ). nên ta cần một ngưỡng giá trị để ánh xạ nó thành 1/0.

Có thể sử dụng một giá trị ngưỡng t (0 < t < 1), thông thường cho t = 0.5

* Nếu P(y=1) ≥ t: dự báo y = 1
* Nếu P(y=1) < t: dự báo y = 0

1. **Hồi quy logistic là thuật toán phân loại tuyến tính**

Đặt z = β0 + β1x1 + β2x2 + … + βkxk (hàm tuyến tính)

Thì khi đó hàm logistic: **P(y=1) = =**

* Dự báo y = 1 nếu P(y=1) ≥ 0.5 :
* ≥ 0.5 ⬄ z ≥ 0 ⬄ β0 + β1x1 + β2x2 + … + βkxk ≥ 0
* Dự báo y = 0 nếu P(y=1) <0.5 :
* < 0.5 ⬄ z < 0 ⬄ β0 + β1x1 + β2x2 + … + βkxk < 0

*Ví dụ 3*: giả sử hồi quy logistic với :

* 2 biến x1, x2 (biến độc lập)
* Hệ số ước lượng β : β0 = -3, β1 =1, β2 = 1

Thì dự báo hồi quy của ta lúc này sẽ là :

* Dự báo y = 1 nếu -3 + x1 + x2 ≥ 0
* Dự báo y = 0 nếu -3 + x1 + x2 < 0
* Đối với điểm (1,1) : -3 + 1 + 1 = -1 < 0 => dự báo y = 0
* Đối với điểm (3,1) : -3 + 3 + 1 = 1 > 0 => dự báo y = 1

Chart, scatter chart

Description automatically generated

1. **Diễn giải tham số của mô hình hồi quy logistic**

|  |  |
| --- | --- |
| Hàm logistic | P(y=1) = |
| Cách viết khác của hàm logistic | Odds = |
|  | Log(Odds) = |
| Tỷ số Odds | Odds = |

* Tỷ số Odds và xác suất y = 1 là đồng biến, tỷ số Odds càng lớn khả năng xảy ra càng cao.
* Khi biến x tăng 1 đơn vị thì giá trị log(Odds) tăng β đơn vị
* Khi biến x tăng 1 đơn vị thì tỷ số Odds tăng lên exp(β) lần
* Điều này cho thấy nếu β > 0 thì x càng lớn, xác suất y = 1 càng lớn và ngược lại.

1. **Xác định sai số dự báo**

5.1. Ma trận nhầm lẫn (confusion matrix) : So sánh giá trị thực tế và giá

|  |  |  |
| --- | --- | --- |
|  | Dự báo = 0 | Dự báo = 1 |
| Thực tế = 0 | Âm tính thực  (TN – True Negative) | Dương tính giả  (FP – False Positive) |
| Thực tế = 1 | Âm tính giả  (FN – False Negative) | Dương tính thực  (TP – True Positive) |

Lưu ý:

* N là số quan sát
* Độ chính xác toàn thể (% khả năng dự đoán chính xác): (TN + TP)/N
* Sai số toàn thể (% khả năng dự đoán sai): (FP + FN)/N
* Độ nhạy (Sensitivity) : % khả năng dự đoán chính xác xảy ra y =1
* Độ nhạy = Dương tính thực / (Thực tế = 1) = TP/(TP+FN)
* Sai số âm tính giả = 1 – độ nhạy = FN/(TP+FN)
* Độ đặc hiệu (Specifitity) : % khả năng dự đoán chính xác xảy ra y =0
* Độ đặc hiệu = Âm tính thực / (Thực tế = 0) = TN/(TN + FP)
* Sai số dương tính giả = 1 – độ đặc hiệu = FP/(TN+FP)
* Mô hình tốt là mô hình có độ nhạy lớn và độ đặc hiệu lớn

*Ví dụ 4* : dự báo khả năng KH vỡ nợ hay không vỡ nợ

Mô hình dự báo

|  |  |  |
| --- | --- | --- |
|  | Dự báo = 0 | Dự báo = 1 |
| Thực tế = 0 | 50 | 10 |
| Thực tế = 1 | 20 | 20 |

* Xét trên 100 quan sát, thì dự báo 50 vỡ nợ và 20 không vỡ nợ là chính xác
* Độ chính xác toàn thể: (50+20)/100 = 80%
* Sai số toàn thể: (20+10)/100 = 30%

Để xem độ chính xác toàn thể ở trên là chính xác hay không thì ta cần so sánh độ chính xác với mô hình cơ sở ( mô hình cơ sở là mô hình ta sẽ dự báo các giá trị đa số trong mẫu) trong trường hợp này giá trị đa số là các KH thực sự không vỡ nợ (50) => độ chính xác mô hình cơ sở là 50%. Do đó, mô hình dự báo của ta tốt hơn so với mô hình cơ sở.

* Độ nhạy: KH thực sự vỡ nợ chiếm bao nhiêu % được dự báo vỡ nợ?
* Độ nhạy: 20/(20+20) = 50%
* Sai số âm tính giả: 1 – độ nhạy = 1 – 50% = 50%
* Độ đặc hiệu: KH thực sự không vỡ nợ chiếm bao nhiêu % dự báo vỡ nợ?
* Độ đặc hiệu: 50/(50+10) = 83%
* Sai số dương tính giả: 1 – độ đặc hiệu = 1 – 83% = 17%

5.2. Sự đánh đổi độ nhạy và độ đặc hiệu

5.2.1 chọn giá trị ngưỡng phù hợp

Thông thường giá trị ngưỡng t thường chọn là 0.5, nhưng không thực sự tối ưu nhất nên ta cần xác định ngưỡng t cho phù hợp. Việc lựa chọn giá trị ngưỡng t còn tùy thuộc vào sai số nào quan trọng hơn trong mô hình.

* TH1: Nếu muốn giảm sai số dương tính giả (dự báo vỡ nợ trong khi KH không vỡ nợ) thì chọn giá trị ngưỡng t lớn (vd: 0.7): dự báo KH vỡ nợ ít hơn. Điều này làm cho ngân hàng tin tưởng và cho nhiều KH vay hơn => Độ đặc hiệu lớn, độ nhạy nhỏ.
* TH2: Nếu muốn giảm sai số âm tính giả (dự báo không vỡ nợ trong khi KH vỡ nợ) thì chọn giá trị ngưỡng t nhỏ (vd: 0.3): dự báo KH vỡ nợ nhiều hơn. Điều này làm cho ngân hàng chọn lọc và hạn chế cho nhiều KH vay hơn => Độ đặc hiệu nhỏ, độ nhạy lớn.

Mục đích của việc chọn giá trị ngưỡng là làm sao cho tổng chi phí phát sinh là nhỏ nhất:

* Gọi a là chi phí của KH vỡ nợ khi dự báo không vỡ nợ (chi phí nợ xấu, chi phí xử lí khoảng nợ đó,…)
* Gọi b là chi phí của KH không vợ nợ khi dự báo vỡ nợ (mất lãi phí mà ta nhận được nếu cho khách hàng đó vay)
* Tổng chi phí = a\*âm tính giả + b\*dương tính giả

5.2.2. Đường ROC

Đường ROC (Receive Operator Charateristic Curve): Thể hiện sự đánh đổi độ nhạy và độ đặc hiệu khi giá trị ngưỡng thay đổi.

* Trục x: sai số dương tính giả (1 - đặc hiệu)
* Trục y: độ nhạy
* Khi độ nhạy tăng thì sai số dương tính giả tăng và ngược lại

Chart, scatter chart

Description automatically generated

5.2.3. Hệ số AUC

Hệ số AUC (Area Under the ROC Curve) là diện tích phần nằm dưới đường ROC, AUC thể hiện khả năng dự báo của mô hình đúng hơn so với Độ chính xác của mô hình trong trường hợp mẫu không cân bằng.

Chart, histogram

Description automatically generated

* AUC = 1: Mô hình hoàn hảo
* AUC = 0.5: Mô hình ngẫu nhiên

Chart

Description automatically generated

1. **Ước lượng tham số - Phương pháp hợp lý cực đại**

* Phương pháp hồi quy Logistic:
* P(y=1) = hβ (x)=

Các tham số β cần phải ước lượng từ số liệu, kí hiệu hβ (x) : trong đó β(x) là các vector

* P(y=1) = hβ (x)
* P(y=0) = 1 - hβ (x)
* Viết gọn :

P(y |x,β) = (hβ (x))y (1- hβ (x))1-y

* Hàm hợp lý (xác suất xảy ra đồng thời y1, y2,…yn):

L(β) = P(y1, y2,…yn|x, β) = P(yi |xi,β) = (hβ (xi))yi (1- hβ (xi))1-yi

* Log của hàm hợp lý (Log(axby) = xlog(a)+ ylog(b)):

Log(L(β)) = yilog(hβ (xi)) + 1-yilog(1- hβ (xi))]

*Chú ý :* hàm log hợp lý ở trên là hàm của tham số β, xi và yi là các hằng số đã biết từ dữ liệu.

* Xác định vector β sao cho hàm LogL(β) đạt cực đại. <= ý tưởng của phương pháp cực đại.

Thông thường ta sẽ đạo hàm của hàm log với giá trị β với từng cái β0, β1,… và xét đạo hàm bằng 0 sau đó giải cho β. tuy nhiên, bài toán hồi quy logistic có đặc điểm là không có công thức tường minh cho ước lượng β như hồi quy tuyến tính. Ta ước lượng bằng cách sử dụng các thuật toán toán tối ưu hóa (lập đi lặp lại để tìm giá trị tối ưu : Gradient Descent, phương pháp Newton-Raphson, Solver của Excel,..)