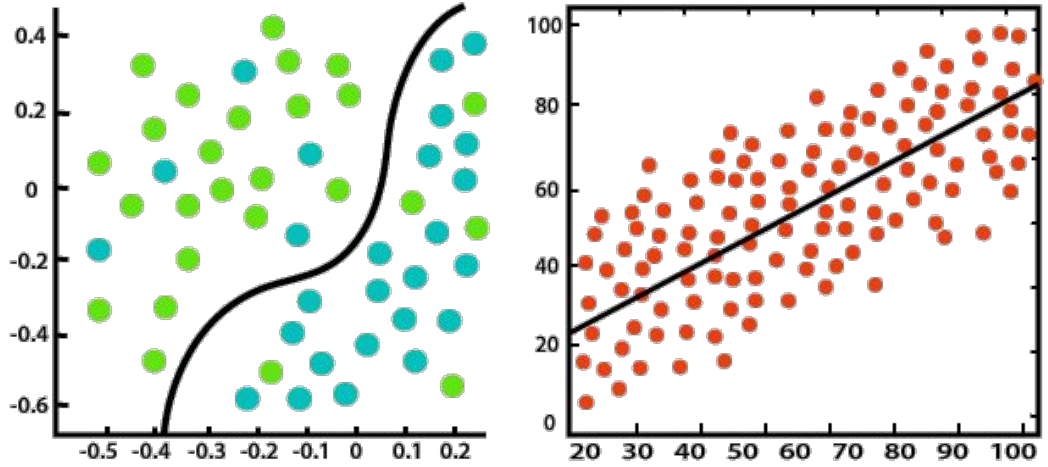
Chương 2
Hồi Quy Tuyến Tính

TS. Phạm Tuấn

ptuan@ute.udn.vn

Giới Thiệu

Chương này chúng ta sẽ ứng dụng hàm tuyến tính để giải quyết bài toán: **Hồi quy (Regression)**. Bài toán mà chúng ta học trong chương này đó là **học có giám sát**.

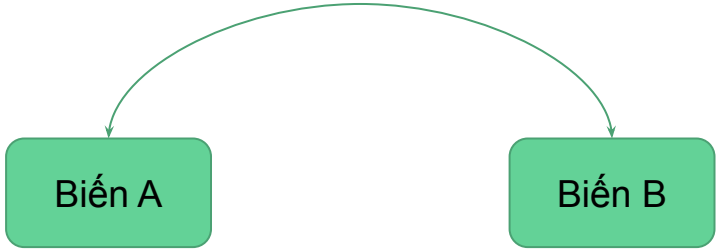


Giới Thiệu

Khi chúng ta đã có được dữ liệu với đa biến, một câu hỏi rất quan trọng là **làm thế nào tìm ra mối liên hệ giữa các biến**. Ví dụ: chúng ta có thể hỏi mối quan hệ giữa trọng lượng của mọi người và chiều cao, hoặc thời gian học tập và điểm kiểm tra....

Hồi quy là một tập hợp các kỹ thuật để ước tính các mối quan hệ giữa các biến.

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 gọi là tuyến tính nếu đồ thị của nó có dạng một mặt phẳng.



Giới thiệu

Chúng ta sẽ bắt đầu với Simple Linear regression nơi mà chỉ có 2 biến (không gian 2 chiều), ví dụ: cân nặng và chiều cao, cân nặng và size giày...

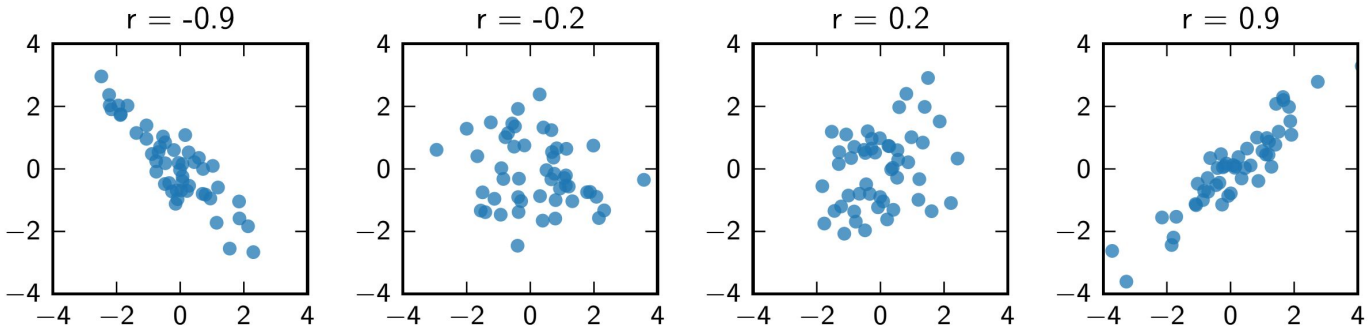
Simple linear regression sẽ tìm ra mối quan hệ giữa 2 biến trên bằng cách cố gắng khớp (fit) đường thẳng $y = w_0 + x^*w_1$ đối với dữ liệu của chúng ta.

Nếu dữ liệu chúng ta có $(N + 1)$ biến thì hàm số sẽ như thế nào ?

Vậy chúng ta sẽ có lợi ích như thế nào sau khi tìm ra mối quan hệ đó ?

Correlation coefficient

Điều kiện nào mà chúng ta có thể tìm được mối quan hệ đó bằng Linear regression ?



r is the **correlation coefficient**

Hàm mô hình

Đường thẳng $y = w_0 + x*w_1$.

x được gọi là **independent variable**, hay còn gọi biến quan sát được

y được gọi là **dependent variable** or **response variable**.

w_1 is the slope of the line: this is one of the most important quantities in any linear regression analysis. A value very close to 0 indicates little to no relationship; large positive or negative values indicate large positive or negative relationships, respectively.

w_0 là điểm khởi đầu.

Sai số

Trong phương trình phía trên, chúng ta đặt **w** = $[w_0, w_1]^T$ là **vector cột** của hệ số cần tối ưu.

Và **x** = $[1, x_1]$ là **vector hàng** của dữ liệu đầu vào. Khi đó phương trình trên được viết thành:

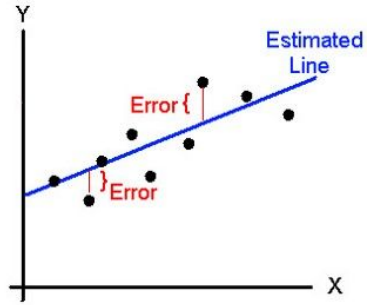
x(1*2) * w(2*1) = y(1*1)

$$y \approx \mathbf{x}\mathbf{w} = \hat{y}$$

Trong đó **y** là giá trị thực tế, và **ŷ** là giá trị đầu ra của hàm tuyến tính (hay còn gọi giá trị dự đoán).

Thực tế, luôn luôn có sự chênh lệch giữa **y** và **ŷ**, và sự chênh lệch đó gọi là **Sai số (e)**.

$$e_i = y_i - \hat{y}_i$$



Hàm mất mát

Để xây dựng mô hình tuyến tính, chúng ta cần ***N* điểm** tập hợp dữ liệu như sau (x_1,y_1), (x_2,y_2),..., (x_n,y_n). Đây là các điểm dữ liệu được thu thập từ trước.

Chúng xây dựng mô hình tuyến tính thông qua việc giải quyết bài toán tối ưu sau:

$$\min_{w_0, w_1} \sum_{i=1}^{n} \left(y_i - \left(w_0 + x_i w_1 \right) \right)^2$$

Hàm trên được miêu tả như sau: chúng ta phải tìm cực tiểu của hàm số theo các biến số w_0 và w_1

Lời giải cho hàm mất mát

$$\min_{w_0, w_1} \sum_{i=1}^{n} \left(yi - \left(w_0 + x_i w_1 \right) \right)^2$$

Bài toán tối ưu này có tên gọi là **least-squares linear regression problem,** Đây cũng chính là hàm mất mát cho mô hình hồi quy tuyến tính.

Cho tập hợp *N* điểm các dữ liệu, lời giải tối ưu cho phương trình trên như sau:

$$w_1 = \frac{\sum_{i=1}^{n} x_i y_i - \frac{1}{n} \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i^2 - \frac{1}{n} \left(\sum_{i=1}^{n} x_i \right)^2}$$

$$w_0 = \frac{1}{n} \sum_{i=1}^{n} y_i - w_1 \frac{1}{n} \sum_{i=1}^{n} x_i$$

Đánh giá mô hình

Để đánh giá hiệu suất mô hình đối với dữ liệu cho trước, chúng ta sử dụng thông số **R-squared** để thực hiện điều đó và công thức của R-squared là:

$$R^2 = 1 - \frac{RSS}{TSS}$$

Trong đó **RSS** là từ viết tắt cho **Residual sum of squares**, và **TSS** là đại diện cho **Total Sum of Squares**

$$RSS = \sum_{i=1}^{n} e_i^2$$

e = sai số

$$\text{TSS} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

TSS = total sum of squares
n = number of observations
y_i = value in a sample
$\bar{y}$ = mean value of a sample

Đặt tính của R-squared

R-squared có vài đặt tính như sau:

- Giá trị luôn nằm trong khoảng 0 và 1.
- Giá trị càng lớn tức là gần 1 thì mô hình tuyến tính có hiệu suất tốt và độ biến đổi của dữ liệu là thấp.
- Giá trị càng nhỏ tức là gần 0 thì mô hình tuyến tính có hiệu suất tồi và dữ liệu của chúng ta gần như xuất hiện ngẫu nhiên.

Yếu điểm của Hồi quy tuyến tính

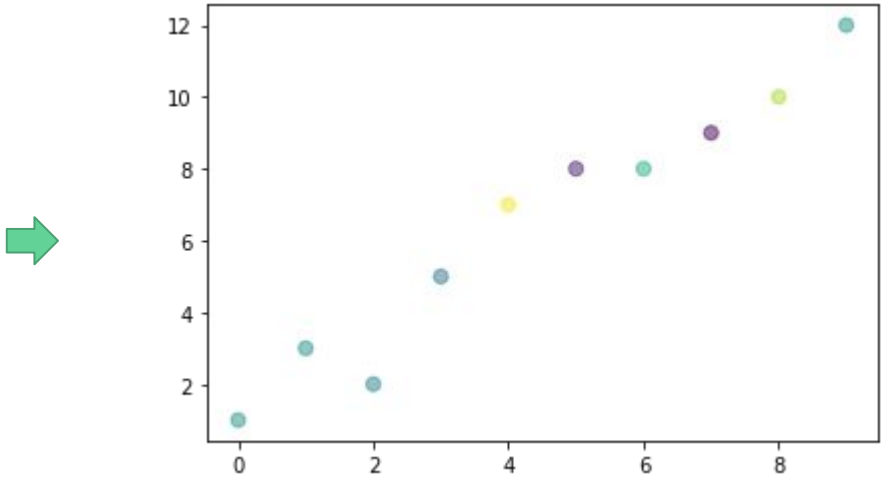
Mô hình Hồi quy tuyến tính là một mô hình đơn giản nhất trong hệ thống Máy học. Do độ phức tạp của mô hình hồi quy tuyến tính thấp nên nó chỉ phù hợp với những dữ liệu đơn giản, và bộc lộ những nhược điểm đối với những tập dữ liệu phức tạp.

- Hồi quy tuyến tính yêu cầu mối quan hệ giữa các biến phải là tuyến tính.
- Hồi quy tuyến tính bị ảnh hưởng mạnh bởi các **điểm dữ liệu phân cực**.
- Mô hình hồi quy tuyến tính đạt hiệu suất cao nếu sai số tuân theo hàm phân bổ Gaussian. $\varepsilon \sim \mathcal{N}(0, \sigma^2)$

Ví dụ trên Python

Chúng ta tạo một số điểm dữ liệu giả như sau:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
# import seaborn as sns
x = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
y = np.array([1, 3, 2, 5, 7, 8, 8, 9, 10, 12])
# number of observations/points
n = np.size(x)
#Lets plot a scatter plot for the given values
colors = np.random.rand(n)
area = 50
plt.scatter(x, y, area, colors, alpha=0.5)
```



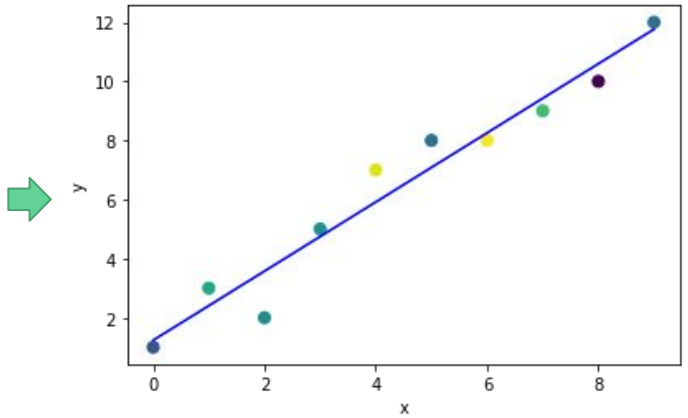
Ví dụ trên Python

Chúng ta dùng thư viện Sklearn để giải bài toán với điểm dữ liệu như trên

```
from sklearn.linear_model import LinearRegression
reg = LinearRegression().fit(x.reshape(-1,1), y)
print('w1 using Sklearn library is:', reg.coef_[0])
print('w0 using Sklearn library is:', reg.intercept_)

# plot data points
plt.scatter(x, y, area, colors, marker = "o")
# predicted response vector
y_pred = reg.intercept_ + reg.coef_[0]*x
# plotting the regression line
plt.plot(x, y_pred, color = "b")

# putting labels
plt.xlabel('x')
plt.ylabel('y')
#show plot
plt.show()
```



Ví dụ trên Python

Chúng ta cũng sử dụng một function có sẵn trong Sklearn để tính toán giá trị **R-squared**

```
from sklearn.metrics import r2_score
r2Score = r2_score(y, y_pred)
print('R2 score:', r2Score)
```

R2 score: 0.95

Như kết quả ở trên R-squared có giá trị rất gần 1 do đó, mô hình hồi quy tuyến tính hoạt động tốt với dữ liệu cho trước.

Bài tập nhóm (số 1)

~~Diễn giải và chứng minh làm sao tìm được nghiệm w_0 và w_1 như ở slide số 9. **(2 điểm)**~~

Tạo 10 điểm dữ liệu ngẫu nhiên và vẽ chúng lên trục toạ độ. **(0.5 điểm)**

Tính w_0, w_1 và R-squared dùng thư viện Sklearn **(0.5 điểm)**

Tính w_0 và w_1 với công thức ở slide số 9 và in ra. **(2 điểm)**

Vẽ đường thẳng với w_0 và w_1 . **(1 điểm)**

Tính R-squared với công thức ở slide số 10 và in ra. **(1 điểm)**

Tạo thêm 1 điểm dữ liệu cách rất xa 10 điểm dữ liệu ban đầu và tính lại w_0 và w_1 (công thức) và in ra. **(1 điểm)**

Vẽ đường thẳng mới với w_0 và w_1 mới. **(1 điểm)**

Tính lại R-squared và in ra. **(1 điểm)**