

Introduction

Hồi quy tuyến tính là một thuật toán học có giám sát, trong đó đầu ra dự đoán là liên tục và có độ dốc không đổi.

Nó được sử dụng để dự đoán các giá trị trong một **phạm vi liên tục**, (ví dụ: doanh số, giá cả) thay vì cố gắng phân loại chúng thành các danh mục hay nhóm (ví dụ: mèo, chó).

Có hai loại chính:

Simple regression

Multivariable regression

1

Simple regression

Simple regression giống như một phương trình đường thẳng mà bạn đã từng học ở phổ thông

$$y = mx + b$$

Trong đó **m** và **b** là các biến mà chúng ta cố gắng “**học**” để với mỗi giá trị input của **x** chúng ta có một giá trị dự đoán của y đúng nhất có thể.

2

Simple regression

dataset

Company	Radio (\$)	Sales
Amazon	37.8	22.1
Google	39.3	10.4
Facebook	45.9	18.3
Apple	41.3	18.5

3

Tạo một dự đoán

Radio: Biến độc lập. Trong machine learning gọi là *features*

Weight: Hệ số của biến độc lập. Trong machine learning gọi là Weight

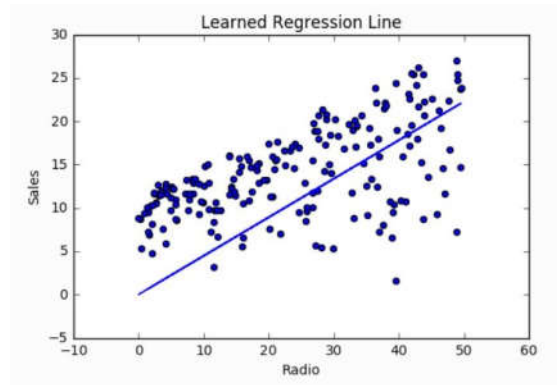
Bias: giá trị lệch để bù đắp những sai số

$$Sales = Weight \cdot Radio + Bias$$

4

Kết quả cần đạt được

Tìm được giá trị Weight và
Bias để có
một đường thẳng phù hợp
nhất với tập dữ liệu



5

Cost function

Chúng ta cần 1 hàm chi phí để tối ưu weight

Hàm lỗi **MSE** (measures squared error) hàm này đo sự sai khác bằng cách lấy trung bình của bình phương giữa giá trị dự đoán và giá trị thực tế.

Đầu ra là một số duy nhất thể hiện chi phí của tập các trọng số (weight) hiện tại

Mục tiêu của chúng ta là tối thiểu MSE để tăng độ chính xác hiện tại

6

Math

N là số điểm dữ liệu, số sample

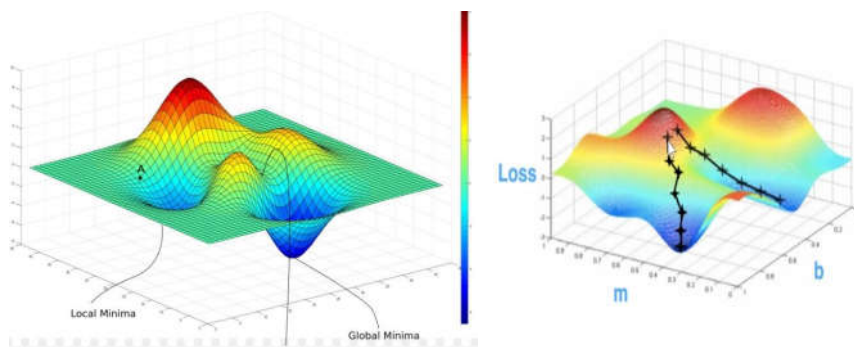
$mx_i + b$ Giá trị dự đoán

y_i Là giá trị thực

$$MSE = \frac{1}{N} \sum_{i=1}^n (y_i - (mx_i + b))^2$$

7

Gradient descent

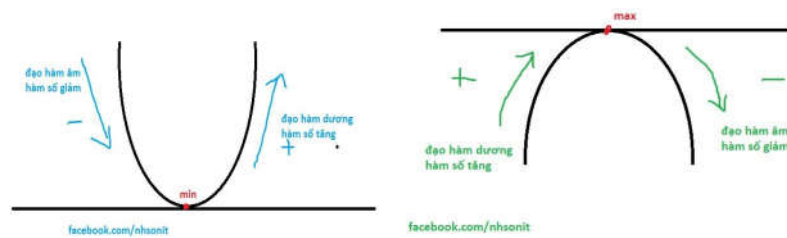


8

Gradient descent

Điểm local minimum là điểm có đạo hàm: $f'(x) = 0$

Đạo hàm phía bên trái local minimum là không dương, và bên phải là không âm.
Max thì ngược lại



9

Gradient descent

Hàm chi phí

$$MSE = \frac{1}{N} \sum_{i=1}^n (y_i - (mx_i + b))^2$$

Chúng ta có thể tính toán độ dốc của hàm chi phí này là:

$$\begin{aligned} f'(m, b) &= \begin{bmatrix} \frac{df}{dm} \\ \frac{df}{db} \end{bmatrix} = \begin{bmatrix} \frac{1}{N} \sum -x_i \cdot 2(y_i - (mx_i + b)) \\ \frac{1}{N} \sum -1 \cdot 2(y_i - (mx_i + b)) \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{N} \sum -2x_i(y_i - (mx_i + b)) \\ \frac{1}{N} \sum -2(y_i - (mx_i + b)) \end{bmatrix} \end{aligned}$$

10

Gradient descent

Để tối ưu hàm chi phí chúng ta cần tính lại weight và bias mới tối ưu hơn.

Chúng ta thực hiện việc tính weight và bias mới bằng cách

lấy weight và bias hiện tại - (tính weight và bias với mỗi điểm dữ liệu rồi lấy trung bình của tất cả các điểm dữ liệu sau đó nhân với learning rate)

learning rate là tốc độ học

11

Training

Đào tạo, training một model là quá trình lặp đi lặp lại cải thiện phương trình dự đoán của bạn bằng cách lặp qua tập dữ liệu nhiều lần.

Update lại weight và bias với **Gradient descent**

Việc training kết thúc khi đạt đến ngưỡng lỗi chấp nhận được hoặc những lần lặp tiếp theo không thể giảm được chi phí

12