Linear Regression

Ngoc Hoang Luong

University of Information Technology (UIT), VNU-HCM

April 6, 2023

hồi quy tuyến tính

Lương Ngọc Hoàng

Trường Đại học Công nghệ Thông tin (UIT), ĐHQG-HCM

Ngày 6 tháng 4 năm 2023

Motivation

- Example: we want to predict the salary of NBA players in terms of certain variables: team, height, weight, position, years of experience, number of 2pts, number of 3pts, number of blocks, etc.
- We have information about some current NBA players:

Player	Height	Weight	Yrs Expr	2 Points	3 Points	Salary
1	...	...	...	...	...	...
2	...	...	...	...	...	...
3	...	...	...	...	...	...
...	...	...	...	...	...	...

- $\mathbf{x}_i$ denotes the vector of measurements of player i's statistics (height, weight, etc.). And, y_i denotes the salary of the i-th player.
- We assume the existence of some **unknown** function $f : \mathcal{X} \to y$ that determines the ideal salary.
- We seek a model $(\hat{f} : \mathcal{X} \to y)$, which we select from some set of candidate functions $h_1, h_2, \ldots, h_m$, that best approximates f.

Động lực

- Ví dụ: chúng tôi muốn dự đoán mức lương của các cầu thủ NBA theo một số biến nhất định: nhóm, chiều cao, cân nặng, vị trí, số năm làm việc kinh nghiệm, số điểm 2, số điểm 3, số khối, v.v.
- Chúng tôi có thông tin về một số cầu thủ NBA hiện tại:

Cầu thủ	Chiều cao	Cân nặng	Năm kinh nghiệm	2 điểm	3 điểm	Lương	
1	. . .	. . .	. . .	. . .	. . .	. . .	. . .
	. . .	. . .	. . .	. . .	. . .	. . .	. . .
2 3	. . .	. . .	. . .	. . .	. . .	. . .	. . .
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

- xi biểu thị vectơ đo lường số liệu thống kê của người chơi i (chiều cao, cân nặng.). Và, yi biểu thị mức lương của người chơi thứ i.
- Chúng ta giả sử tồn tại một hàm f X y chưa biết nào đó quyết định mức lương lý tưởng.
- Chúng tôi tìm kiếm một mô hình (ˆ f X y), mà chúng tôi chọn từ một số bộ các hàm ứng viên h1, h2, . . . , hm, gần đúng nhất với f.

The Intuition of Regression

- For simplicity, let's assume no inflation. We want to predict the salary of a new player.
- **Scenario 1**: We have no information on this new player. How would we predict the salary y_0?
- We can guess the salary using the historical average salary $\bar{y}$ of NBA players: $\hat{y}_0 = \bar{y}$.
- We use $\bar{y}$ as the typical score (i.e., a measure of center) as a plausible guess for y_0.
- We could also use the median of existing salaries, to disregard outliers.

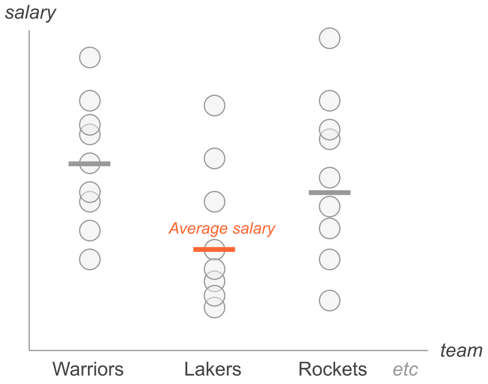
Trực giác hồi quy

- Để đơn giản, giả sử không có lạm phát. Chúng tôi muốn dự đoán lương của một cầu thủ mới.
- Tình huống 1: Chúng tôi không có thông tin gì về người chơi mới này. Làm sao chúng ta sẽ dự đoán mức lương y0?
- Chúng ta có thể đoán mức lương bằng cách sử dụng mức lương trung bình trước đây y¯ của các cầu thủ NBA: y^ 0 = y¯.
- Chúng tôi sử dụng y¯ làm điểm số điển hình (tức là thước đo trung tâm) như một dự đoán hợp lý cho y0.
- Chúng ta cũng có thể sử dụng mức lương trung bình hiện có để bỏ qua ngoại lệ.

VIET NAM NATIONAL UNIVERSITY HCMC
UNIVERSITY OF INFORMATION TECHNOLOGY
VNUHCM - UIT
3 / 35

The Intuition of Regression

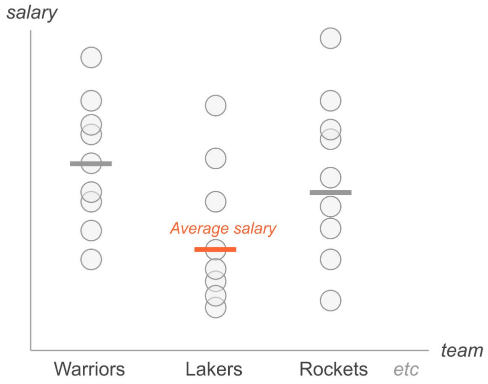
- **Scenario 2**: We know the new player will join LA Lakers. We can use this information to have a more educated guess for y_0.



- Instead of using the salaries of all players, we focus on the salaries of Laker's players: $y_0 = \texttt{avg}(\text{Laker's Salaries})$.

Trực giác hồi quy

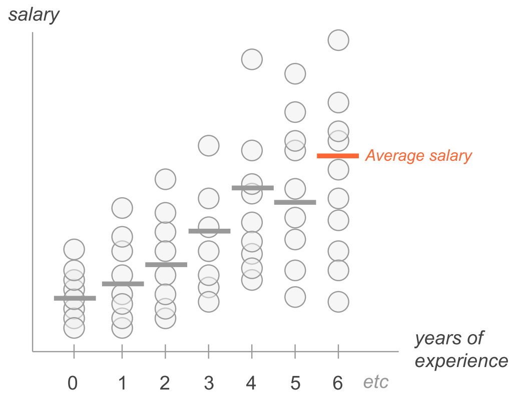
- Tình huống 2: Chúng tôi biết cầu thủ mới sẽ gia nhập LA Lakers. Chúng ta có thể sử dụng thông tin này để dự đoán chính xác hơn cho y0.



- Thay vì sử dụng tiền lương của tất cả người chơi, chúng tôi tập trung vào tiền lương của các cầu thủ của Laker: y0 = avg(Lương của Laker).

The Intuition of Regression

- **Scenario 3**: If we know this new player has 6 years of experience, we look at the average salaries of players with the same experience.

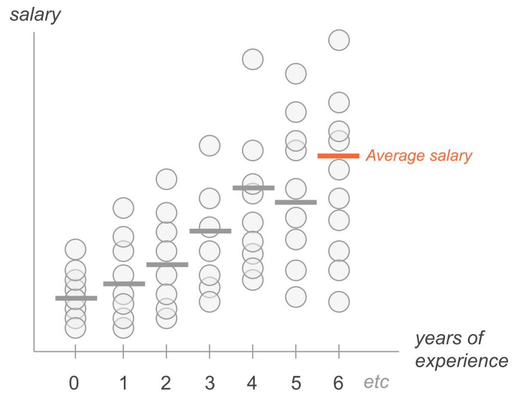


- In all examples, the predicted salary is a conditional mean:

$$\hat{y}_0 = \text{avg}(y_i | \mathbf{x}_i = \mathbf{x}_0)$$

Trực giác hồi quy

- Tình huống 3: Nếu chúng tôi biết người chơi mới này có 6 năm kinh nghiệm, chúng tôi sẽ xem xét mức lương trung bình của những người chơi có cùng kinh nghiệm.



- Trong tất cả các ví dụ, mức lương dự đoán là giá trị trung bình có điều kiện:

y^ 0 = avg(yi xi = x0)

The Intuition of Regression

- The prediction is a conditional mean:

$$\hat{y}_0 = \texttt{avg}(y_i | \mathbf{x}_i = \mathbf{x}_0)$$

- But this strategy only works if we have data points $\mathbf{x}_i$ match the query point $\mathbf{x}_0$.
- The core idea of regression: Obtaining prediction $\hat{y}_0$ using quantities of the form $\texttt{avg}(y_i | \mathbf{x}_i = \mathbf{x}_0)$, which can be formalized as:

$$\mathbb{E}(y_i | x^*_{i1}, x^*_{i2}, \ldots, x^*_{ip}) \longrightarrow \hat{y}$$

where x^*_{ij} is the i-th measurement of the j-th variable.
- The **regression function**: a conditional expectation.

Trực giác hồi quy

- Dự đoán là trung bình có điều kiện:

y^ 0 = avg(yi xi = x0)

- Nhưng chiến lược này chỉ hoạt động nếu chúng ta có điểm dữ liệu xi khớp với điểm truy vấn x0.
- Ý tưởng cốt lõi của hồi quy: Dự đoán y^ 0 bằng cách sử dụng các đại lượng có dạng avg(yi xi = x0), có thể được viết thành:

E(yi xi1 , Xi2 , . . . , x ip) y^

trong đó x là phép đo thứ i của biến thứ j. ij

- Hàm hồi quy: kỳ vọng có điều kiện.

The Linear Regression Model

- In a regression model, we use one or more features X to predict the response Y.
- A linear regression model tells us how to combine the features into linear way to approximate the response.
- In the univariate case, we have a linear equation:

$$\hat{Y} = b_0 + b_1 X$$

- For a given individual i, we have:

$$\hat{y}_i = b_0 + b_1 x_i$$

or:

$$\hat{\mathbf{y}} = b_0 + b_1 \mathbf{x}$$

Mô hình hồi quy tuyến tính

- Trong mô hình hồi quy, chúng tôi sử dụng một hoặc nhiều tính năng X để dự đoán phản hồi Y .
- Mô hình hồi quy tuyến tính cho chúng ta biết cách kết hợp các đặc điểm thành cách tuyến tính để xấp xỉ đáp ứng. • Trong

trường hợp đơn biến, ta có phương trình tuyến tính:

Yˆ = b0 + b1X

- Với cá thể i cho trước, ta có:

yˆ i = b0 + b1xi

hoặc:

yˆ = b0 + b1x

VIET NAM NATIONAL UNIVERSITY HCMC
UNIVERSITY OF INFORMATION TECHNOLOGY
VNUHCM - UIT
7 / 35

The Linear Regression Model

- We can add an auxiliary constant feature in the form of a vector of 1's and use the matrix notations:

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{b}$$

where:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}; \quad \hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix}; \quad \mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \end{bmatrix}$$

- In the multivariate case, when $p > 1$, we have:

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & \dots & x_{1p} \\ 1 & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{bmatrix}; \quad \hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix}; \quad \mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_p \end{bmatrix}$$

Mô hình hồi quy tuyến tính

- Chúng ta có thể thêm một tính năng hằng số phụ ở dạng vectơ 1 và sử dụng các ký hiệu ma trận:

y^ = Xb

Ở đâu:

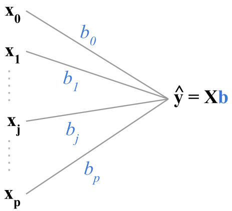
1x1 _
1 x2
X =
y^ =
y^ 1
y^ 2
; b = [b0 b1]
; 1 xn
y^ n

- Trong trường hợp đa biến, khi p > 1, ta có:

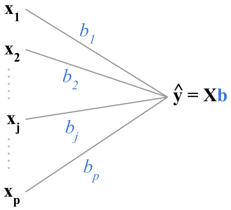
1x11 . _ . . x1p 1
x21 . . . x2p
X =
1xn1 . _ . . xnp
y^ 1
y^ 2
y^ =
y^ n
; b =
b0
b1
bp

VIET NAM NATIONAL UNIVERSITY HCMC
UNIVERSITY OF INFORMATION TECHNOLOGY
VNUHCM - UIT
8 / 35

The Linear Regression Model

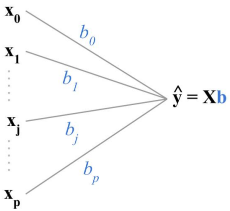


- If the predictors and the response are mean-centered, we can ignore the constant term x_0:

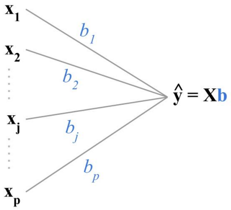


- How to obtain the vector of coefficients $\mathbf{b}$?

Mô hình hồi quy tuyến tính



- Nếu các yếu tố dự đoán và phản hồi tập trung vào giá trị trung bình, chúng ta có thể bỏ qua số hạng không đổi x0:



- Làm thế nào để có được véc tơ của các hệ số b?

The Error Measure

- We want the predictions $\hat{y}_i$ to be as close as possible to y_i.
- To measure how close $\hat{y}_i$ and y_i, the most common choise is the squared distance:

$$d^2(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2 = (\hat{y}_i - y_i)^2 = (\mathbf{b}^\top \mathbf{x}_i - y_i)^2$$

- To measure the overall error, we can use:
  - The sum of squared errors (SSE):

$$\text{SSE} = \sum_{i=1}^{n} d^2(y_i, \hat{y}_i)$$

  - The mean squared error (MSE):

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} d^2(y_i, \hat{y}_i)$$

Biện pháp lỗi

• Chúng tôi muốn dự đoán yˆ i càng gần với yi càng tốt . • Để đo mức độ gần gũi của yˆ i và yi , lựa chọn phổ biến nhất là bình phương khoảng cách:

ngày² (yi , yˆ i) = (yi yˆ i)² = (yˆ i yi)² = (b xi yi)²

• Để đo lỗi tổng thể, chúng ta có thể sử dụng: • Tổng bình phương lỗi (SSE):

SSE = N ngày² (yi , yˆ i) tôi = 1

• Sai số trung bình bình phương (MSE):

MSE = 1/n N i=1 ngày² (yi , yˆ i)

The Error Measure

- Let $e_i = (y_i - \hat{y}_i)$
- We have:

$$
\begin{aligned}
\texttt{MSE} &= \frac{1}{n}\sum_{i=1}^{n} e_i^2 = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{b}^\top \mathbf{x}_i - y_i)^2 = \frac{1}{n}(\mathbf{X}\mathbf{b} - \mathbf{y})^\top(\mathbf{X}\mathbf{b} - \mathbf{y}) \\
&= \frac{1}{n}\|\mathbf{X}\mathbf{b} - \mathbf{y}\|^2 = \frac{1}{n}\|\hat{\mathbf{y}} - \mathbf{y}\|^2 \\
&= \frac{1}{n}\|e\|^2 = \frac{1}{n}(\hat{\mathbf{y}} - \mathbf{y})^\top(\hat{\mathbf{y}} - \mathbf{y})
\end{aligned}
$$

- The MSE is proportional to the squared norm of the residual vector $\mathbf{e} = \hat{\mathbf{y}} - \mathbf{y}$

Biện pháp lỗi

- Cho ei = (yi
y^ i) • Ta có:

MSE = 1/n ∑ N i=1 tôi 2 = 1/n ∑ N i=1 (y^ i yi) 2

= 1/n ∑ N i=1 (b xi yi) 2 = 1/N (Xb y) (Xb y)

= 1/n Xb y 2 = 1/N y^ y 2

= n1/N e 2 = 1/N (y^ y) (y^ y)

- MSE tỷ lệ với bình phương chuẩn của vectơ dư
e = y^ y

The Least Squares Algorithm

- In (ordinary least squares regression, we minimize the mean of squared errors (MSE).
- We compute the gradient of MSE wrt $\mathbf{b}$.

$$\begin{aligned}\nabla \mathrm{MSE}(\mathbf{b}) &= \frac{\partial}{\partial \mathbf{b}} \mathrm{MSE}(\mathbf{b}) \\ &= \frac{\partial}{\partial \mathbf{b}} \left(\frac{1}{n} \mathbf{b}^\top \mathbf{X}^\top \mathbf{X} \mathbf{b} - \frac{2}{n} \mathbf{b}^\top \mathbf{X}^\top \mathbf{y} + \frac{1}{n} \mathbf{y}^\top \mathbf{y} \right) \\ &= \frac{2}{n} \mathbf{X}^\top \mathbf{X} \mathbf{b} - \frac{2}{n} \mathbf{X}^\top \mathbf{y}\end{aligned}$$

- Equating to zero we have the **Normal Equations**:

$$\mathbf{X}^\top \mathbf{X} \mathbf{b} = \mathbf{X}^\top \mathbf{y}$$

- This is a system of n equations with $p+1$ unknowns (including the constant term b_0).

Thuật toán bình phương nhỏ nhất

- Trong (hồi quy bình phương nhỏ nhất thông thường, chúng tôi tối thiểu hóa giá trị trung bình của lỗi bình phương (MSE).
- Chúng tôi tính toán độ dốc của MSE wrt b.

MSE(b) = MSE(b)

= b / b (1/N 2 b X Xb − b/N X y + 1/N y y)

= 2/N 2 X Xb − X/N y

- Tương đương với 0, chúng ta có các Phương trình Chuẩn tắc:

X Xb = X y

- Đây là hệ n phương trình với p + 1 ẩn số (kể cả hằng số b0).

The Least Squares Algorithm

$$\mathbf{X}^\top \mathbf{X}\mathbf{b} = \mathbf{X}^\top \mathbf{y}$$

- If $\mathbf{X}^\top \mathbf{X}$ is invertible, then the vector of regression coefficients $\mathbf{b}$:

$$\mathbf{b} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

- We can compute the response vector:

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{b} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y} = \mathbf{H}\mathbf{y}$$

- The hat matrix $\mathbf{H} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top$ is an **orthogonal projector**:
  - It is symmetric.
  - It is idempotent.
  - Its eigenvalues are either 0 or 1.

Thuật toán bình phương nhỏ nhất

X Xb = X y

- Nếu X X khả nghịch thì vectơ hệ số hồi quy b:

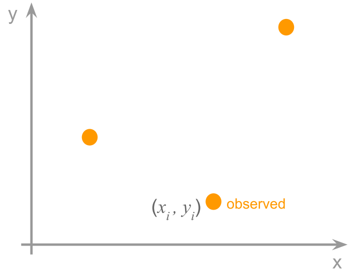
b = (X X) 1X y

- Chúng ta có thể tính toán vectơ phản hồi:

y^ = Xb = X(X X) 1X y = Hy

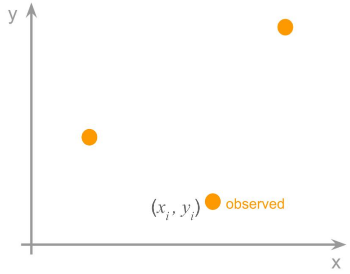
- Ma trận mũ H = X(X X) 1X là một phép chiếu trực giao: • Nó là ma trận đối xứng. •
  Nó là idempotent.
  - Các giá trị riêng của nó là 0 hoặc 1.

Geometries of OLS - Rows Perspective



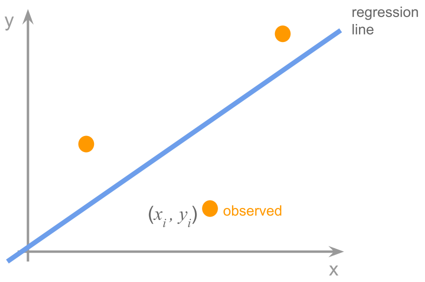
- Assume we have the response Y and one predictor X (i.e., $p = 1$).

Hình học của OLS - Phối cảnh hàng



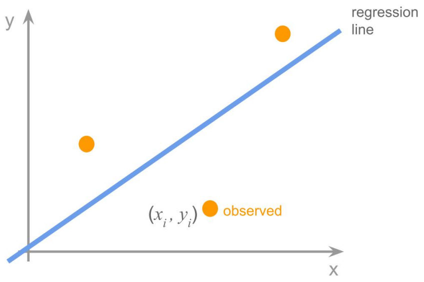
- Giả sử chúng ta có câu trả lời Y và một yếu tố dự đoán X (nghĩa là p = 1).

Geometries of OLS - Rows Perspective



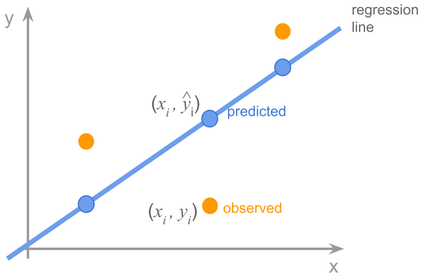
- We predict y_i by linear combining the inputs $\hat{y}_i = b_0 + b_1 x_i$. In 2D, the fitted model is a line.
- In 3D, the fitted model would be a plan. In higher dimensions, it would be a hyperplane.

Hình học của OLS - Phối cảnh hàng



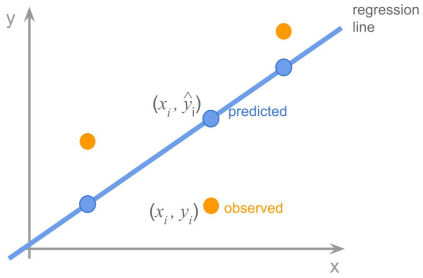
- Chúng tôi dự đoán yi bằng cách kết hợp tuyến tính các yếu tố đầu vào yˆ i = b0 + b1xi . Trong 2D, mô hình được trang bị là một dòng.
- Ở chế độ 3D, mô hình được trang bị sẽ là một kế hoạch. Ở các chiều cao hơn, nó sẽ là một siêu phẳng.

Geometries of OLS - Rows Perspective



- With a model, we obtain predicted value $\hat{y}_i$.
- Some predicted values are equal to the observed values.
- Some predicted values are higher than the observed values.
- Some predicted values are smaller than the observed values.

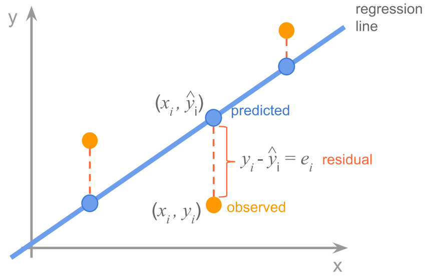
Hình học của OLS - Phối cảnh hàng



• Với một mô hình, chúng ta thu được giá trị dự đoán yˆ i . • Một số giá trị dự đoán bằng giá trị quan sát. • Một số giá trị dự đoán cao hơn giá trị quan sát. • Một số giá trị dự đoán nhỏ hơn giá trị quan sát.

VIET NAM NATIONAL UNIVERSITY HCMC
UNIVERSITY OF INFORMATION TECHNOLOGY
VNUHCM - UIT
16 / 35

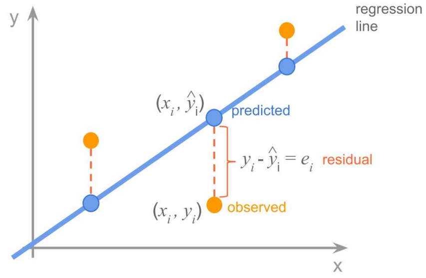
Geometries of OLS - Rows Perspective



- Given a set of data points, we want the line that minimizes the squares of the errors $e_i = \hat{y}_i - y_i$, which are known as the **residuals**.
- We want to find parameters $b_0, b_1, \ldots, b_p$ that minimize the squared norm of the vector of residuals.

$$\sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 = \sum_{i=1}^{n} (b_0 + b_1 x_i - y_i)^2$$

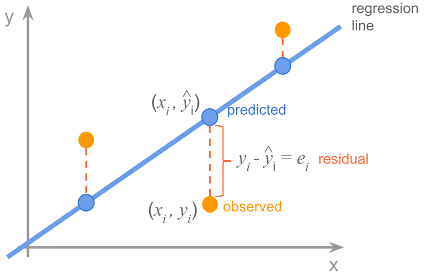
Hình học của OLS - Phối cảnh hàng



- Cho trước một tập hợp các điểm dữ liệu, chúng tôi muốn đường tối thiểu hóa bình phương của các sai số ei = yˆ i yi , được gọi là phần dư.
- Ta muốn tìm tham số b0, b1, . . . , bp mà cực tiểu bình phương của véc tơ của phần dư.

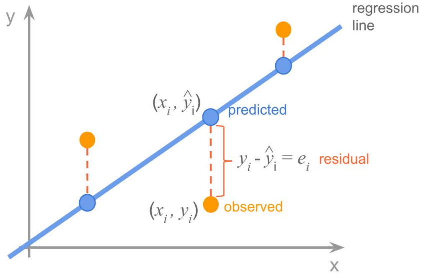
N 2 = N (yˆ i yi) 2 = N (b0 + b1xi yi) 2
tôi _ tôi = 1 tôi = 1 tôi = 1

Geometries of OLS - Rows Perspective



$$
\begin{aligned}
\|\mathbf{e}\|^2 &= \|\hat{\mathbf{y}} - \mathbf{y}\|^2 \\
&= \|\mathbf{Xb} - \mathbf{y}\|^2 \\
&= (\mathbf{Xb} - \mathbf{y})^\top (\mathbf{Xb} - \mathbf{y}) \\
&\propto \text{MSE}
\end{aligned}
$$

Hình học của OLS - Phối cảnh hàng



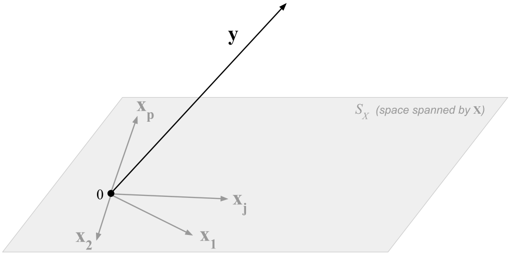
e 2 = yˆ 2

y = Xb y2

= (Xb y) (Xb y)

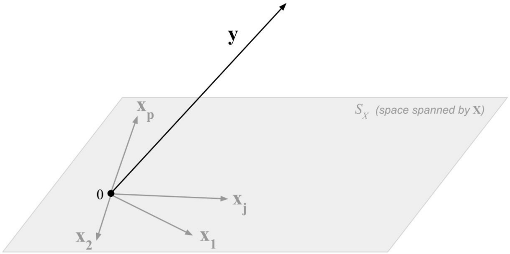
MSE

Geometries of OLS - Columns Perspective



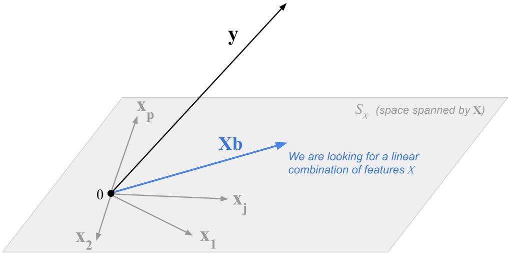
- Consider the variables in the n-dimensional spaces, both the response and the predictors.
- The X variables span some subspace $\mathbb{S}_X$. This subspace does not contain the response Y, unless Y is a linear combination of $X_1, X_2, \ldots, X_p$.

Hình học của OLS - Phối cảnh cột



- Xét các biến trong không gian n chiều, cả biến phản ứng và dự đoán.
- Các biến X mở rộng một số không gian con SX. Không gian con này không chứa phản ứng Y X1, , trừ khi Y là tổ hợp tuyến tính của X2, . . . , Xp.

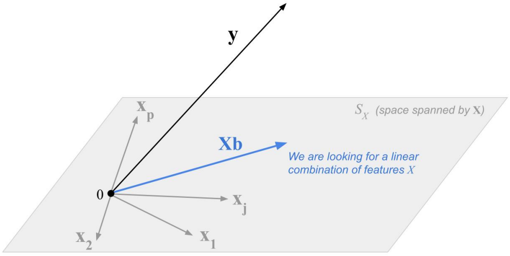
Geometries of OLS - Columns Perspective



- There are an infinite number of linear combinations of $X_1, X_2, \dots, X_p$.
- We want a linear combination $\mathbf{Xb}$ that best approximates $\mathbf{y}$.

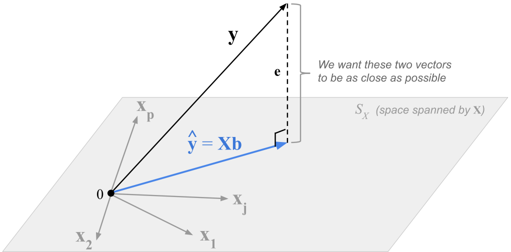
Machine Translated by Google

Hình học của OLS - Phối cảnh cột



- Có vô số tổ hợp tuyến tính của X1, X2, . . . , Xp.
- Chúng ta muốn có một tổ hợp tuyến tính Xb gần đúng nhất với y.

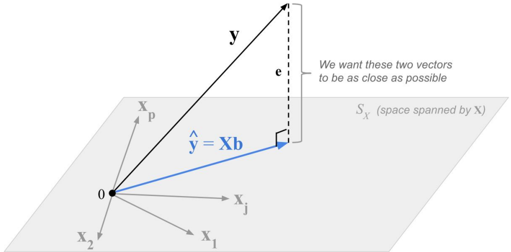
Geometries of OLS - Columns Perspective



- We want a mix of features $\hat{\mathbf{y}} = \mathbf{X}\mathbf{b}$ that is the closest approximation to $\mathbf{y}$.
- The difference between $\hat{\mathbf{y}}$ and $\mathbf{y}$ is: $\mathbf{e} = \hat{\mathbf{y}} - \mathbf{y}$.
- We want $\hat{\mathbf{y}}$ such that the squared norm $\|\mathbf{e}\|^2$ is as small as possible.

$$\min \|\mathbf{e}\|^2 = \min \|\hat{\mathbf{y}} - \mathbf{y}\|^2 \propto \min \mathtt{MSE}$$

Hình học của OLS - Phối cảnh cột



• Chúng tôi muốn kết hợp các đặc trưng yˆ = Xb gần nhất gần đúng với y. •

Hiệu giữa yˆ và y là: e = yˆ y. • Ta muốn yˆ

sao cho chuẩn bình phương e 2 là càng nhỏ càng tốt.

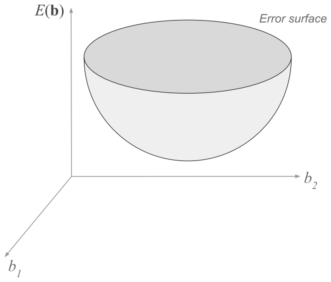
phút e 2 = cực tiểu yˆ y 2 tối thiểu MSE

Geometries of OLS - Parameters Perspective

- From the point of view of parameters $\mathbf{b}$, we can classify the order of each term in the Mean Squared Error (MSE):

$$E(\mathbf{y},\hat{\mathbf{y}}) = \frac{1}{n}\Big(\underbrace{\mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b}}_{\text{Quadratic Form}} - \underbrace{2\mathbf{b}^\top\mathbf{X}^\top\mathbf{y}}_{\text{Linear}} + \underbrace{\mathbf{y}^\top\mathbf{y}}_{\text{Constant}}\Big)$$

- Since $\mathbf{X}^\top\mathbf{X}$ is positive semidefinite, we know that $\mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b} \geq 0$.
- Assume we have only two predictors X_1 and X_2, then $\mathbf{b} = (b_1, b_2)$. The MSE will be a paraboloid in (E, b_1, b_2) space.



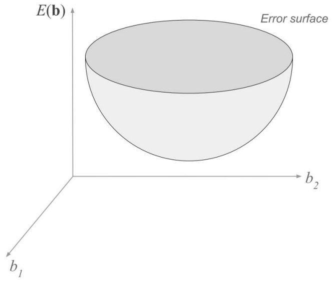
Hình học của OLS - Phối cảnh tham số

- Từ quan điểm của tham số b, chúng ta có thể phân loại thứ tự của từng thuật ngữ trong Lỗi bình phương trung bình (MSE):

E(y,yˆ) = 1/N (b X Xb 2b X y + y y)

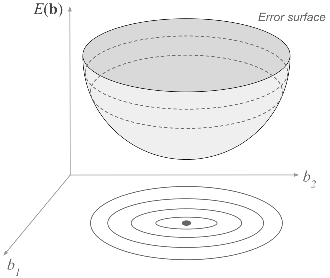
Dạng bậc hai tuyến tính Không thay đổi

- Vì X X là nửa xác định dương, nên ta biết rằng b X Xb ≥ 0.
- Giả sử ta chỉ có hai biến dự đoán X1 và X2, khi đó b = (b1, b2). MSE sẽ là một paraboloid trong không gian (E, b1, b2) .



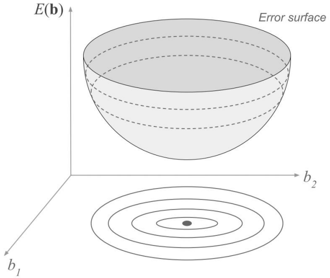
Geometries of OLS - Parameters Perspective

- Since $\mathbf{X}^\top\mathbf{X}$ is positive semidefinite, we know that $\mathbf{b}^\top\mathbf{X}^\top\mathbf{X}\mathbf{b} \geq 0$.
- Assume we have only two predictors X_1 and X_2, then $\mathbf{b} = (b_1, b_2)$. The MSE will be a paraboloid in (E, b_1, b_2) space.
- Imagine we get horizontal slices of the MSE surface. For each slice, we can project it onto the plane spanned by parameters b_1 and b_2. The resulting projections are like a topographic map, with error contours on this plane.



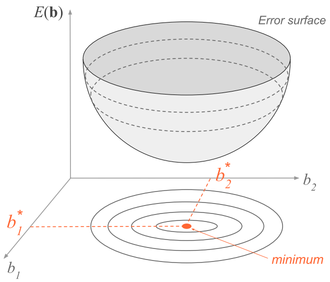
Hình học của OLS - Phối cảnh tham số

- Vì X X là nửa xác định dương, nên ta biết rằng b X Xb ≥ 0.
- Giả sử ta chỉ có hai biến dự đoán X1 và X2, khi đó b = (b1, b2). MSE sẽ là một paraboloid trong không gian (E, b1, b2) .
- Hãy tưởng tượng chúng ta có các lát cắt ngang của bề mặt MSE. Đối với mỗi lát cắt, chúng ta có thể chiếu nó lên mặt phẳng kéo dài bởi các tham số b1 và b2. Các phép chiếu kết quả giống như một bản đồ địa hình, với các đường viền lỗi trên mặt phẳng này.



Geometries of OLS - Parameters Perspective

- Assume we have only two predictors X_1 and X_2, then $\mathbf{b} = (b_1, b_2)$. The MSE will be a paraboloid in (E, b_1, b_2) space.
- Imagine we get horizontal slices of the MSE surface. For each slice, we can project it onto the plane spanned by parameters b_1 and b_2. The resulting projections are like a topographic map, with error contours on this plane.

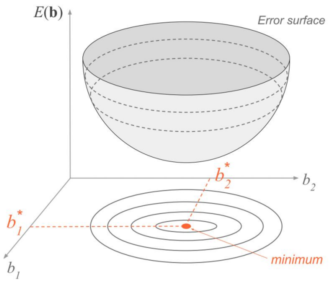


- The minimum of the error surface at point (b_1^*, b_2^*).
- Assuming $\mathbf{X}^\top\mathbf{X}$ invertible, $\mathbf{b}^* = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$.

Hình học của OLS - Phối cảnh tham số • Giả sử chúng ta chỉ có hai biến dự đoán X1 và X2, khi đó b = (b1, b2).

MSE sẽ là một paraboloid trong không gian (E, b1, b2) .

- Hãy tưởng tượng chúng ta có các lát cắt ngang của bề mặt MSE. Đối với mỗi lát cắt, chúng ta có thể chiếu nó lên mặt phẳng kéo dài bởi các tham số b1 và b2. Các phép chiếu kết quả giống như một bản đồ địa hình, với các đường viền lỗi trên mặt phẳng này.

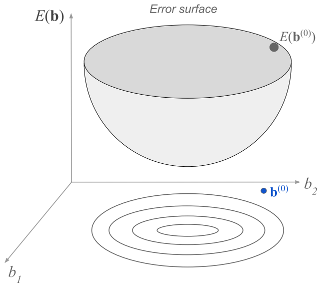


- Cực tiểu của bề mặt sai số tại điểm (b • Giả sử 1 , 2b).

X X khả nghịch, b = (X X) 1X y.

Idea of Gradient Descent

- Start with an random point $\mathbf{b}^{(0)} = (b_1^{(0)}, b_2^{(0)})$ of model parameters.
- Evaluate the error function at this point $E(\mathbf{b}^{(0)})$. This gives a location somewhere on the loss surface.

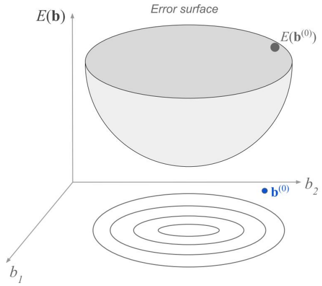


Ý tưởng về Gradient Descent

(0) • Bắt đầu với một điểm ngẫu nhiên b (0) 1 , b 2 (0)) của các tham số mô hình.

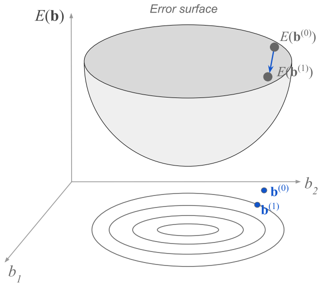
= (b • Đánh giá hàm lỗi tại điểm này E(b (0)).

vị trí ở đâu đó trên bề mặt mất mát.



Idea of Gradient Descent

- Start with an random point $\mathbf{b}^{(0)} = (b_1^{(0)}, b_2^{(0)})$ of model parameters.
- Evaluate the error function at this point $E(\mathbf{b}^{(0)})$. This gives a location somewhere on the loss surface.
- We get a new vector $\mathbf{b}^{(1)}$ so that we "move down" the surface to obtain a new position $E(\mathbf{b}^{(1)})$.

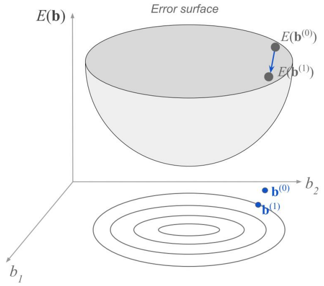


Ý tưởng về Gradient Descent

(0) • Bắt đầu với một điểm ngẫu nhiên b 1 , b2(0)) của các tham số mô hình.

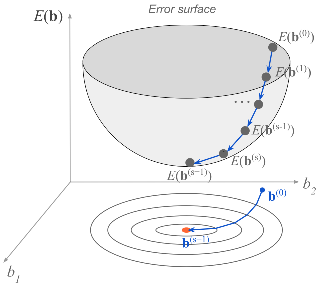
(0) = (b • Đánh giá hàm lỗi tại điểm này E(b (0)).
vị trí ở đâu đó trên bề mặt mất mát.

• Chúng ta có một vectơ b (1) mới để chúng ta "di chuyển xuống" bề mặt để có được một vị trí mới E(b (1)).



Idea of Gradient Descent

- Start with an random point $\mathbf{b}^{(0)} = (b_1^{(0)}, b_2^{(0)})$ of model parameters.
- Evaluate the error function at this point $E(\mathbf{b}^{(0)})$. This gives a location somewhere on the loss surface.
- We get a new vector $\mathbf{b}^{(1)}$ so that we "move down" the surface to obtain a new position $E(\mathbf{b}^{(1)})$.
- At each step s, we obtain a new vector $\mathbf{b}^{(s)}$ that yields an error $E(\mathbf{b}^{(s)})$ that is closer to the minimum of $E(\mathbf{b})$.



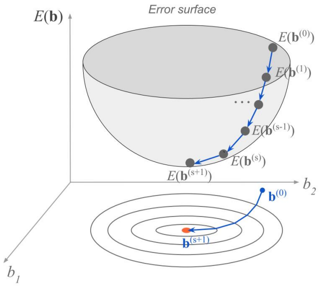
- Eventually, we should get very close to the minimum $\mathbf{b}^*$.

Ý tưởng về Gradient Descent

- Bắt đầu với một điểm ngẫu nhiên b (0) = (0) 1 , b2(0)) của các tham số mô hình.

(b • Đánh giá hàm sai số tại điểm này E(b (0)).
vị trí ở đâu đó trên bề mặt mất mát.

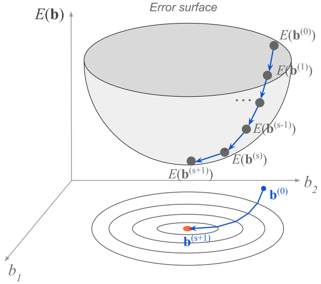
- Chúng ta có một vectơ b (1) mới để chúng ta "di chuyển xuống" bề mặt để có được một vị trí mới E(b (1)).
- Tại mỗi bước s, chúng ta thu được một véc tơ b (s) mới tạo ra lỗi E(b (s)) gần với giá trị nhỏ nhất của E(b).



- Cuối cùng, chúng ta sẽ tiến rất gần đến b .

Idea of Gradient Descent

- Keep in mind that we don't see the surface.
- We only have local information at the current point $\mathbf{b}^{(s)}$ we evaluate the error function $E(\mathbf{b}^{(s)})$.



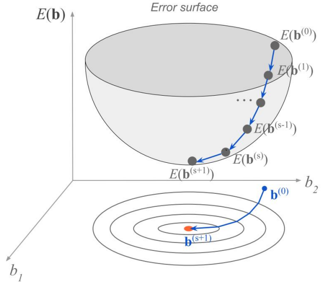
- Imagine we need to get to the bottom of a valley from the top of a mountain at night.
- We touch our surroundings to feel which direction the slope of the terrain goes down.
- We identify the direction that gives the steepest descent.

Ý tưởng về Gradient Descent

• Hãy nhớ rằng chúng ta không nhìn thấy bề mặt. • Chúng

ta chỉ có thông tin cục bộ tại điểm hiện tại b (s) đánh giá hàm lỗi E(b chúng tôi

(s)).



• Hãy tưởng tượng chúng ta cần đi xuống đáy thung lũng từ đỉnh núi vào ban đêm. • Chúng

ta chạm vào môi trường

xung quanh để cảm nhận độ dốc của địa hình đi xuống theo hướng nào. • Chúng tôi xác

định hướng đi xuống

dốc nhất.

Moving Down an Error Surface

- "Moving down an error surface" means that we generate the new vector $\mathbf{b}^{(s+1)}$ from the current point $\mathbf{b}^{(s)}$ using the formula:

$$\mathbf{b}^{(s+1)} = \mathbf{b}^{(s)} + \alpha \mathbf{v}^{(s)}$$

- $\mathbf{v}^{(s)}$ is the vector indicating the direction we move at step s. We can consider $\mathbf{v}^{(s)}$ to be a unit vector.
- α the **step size**, indicating how far we move along direction $\mathbf{v}^{(s)}$.

$$\begin{aligned}
\mathbf{b}^{(1)} &= \mathbf{b}^{(0)} + \alpha \mathbf{v}^{(0)} \\
\mathbf{b}^{(2)} &= \mathbf{b}^{(1)} + \alpha \mathbf{v}^{(1)} \\
&\vdots \\
\mathbf{b}^{(s+1)} &= \mathbf{b}^{(s)} + \alpha \mathbf{v}^{(s)}
\end{aligned}$$

- We assume a *constant* step size α. More sophisticated versions of gradient descent allow *variable* step size.
- The direction $\mathbf{v}^{(s)}$ changes at each step s. How do we find $\mathbf{v}^{(s)}$?

Di chuyển xuống bề mặt lỗi

- "Di chuyển xuống bề mặt lỗi" có nghĩa là chúng ta tạo vectơ b (s+1) mới từ điểm b (s) hiện tại bằng cách sử dụng công thức: b (s+1) = b (s)

+ αv (S)

- v (s) là véc tơ chỉ phương ta di chuyển ở bước s. Chúng tôi có thể xem xét v (s) là một vectơ đơn vị.

(s) • α **kích thước bước,** cho biết chúng ta di chuyển bao xa theo hướng vb .

(1) = b (0) (0) + αv

b (2) = b (1) (1) + αv

b (s+1) = b (s) + αv (S)

- Chúng tôi giả sử kích thước bước không đổi α. Các phiên bản phức tạp hơn của giảm độ dốc cho phép kích thước bước thay

đổi. • Hướng v (s) thay đổi ở mỗi bước s. Làm thế nào để chúng tôi tìm thấy v (s) ?

The direction of $\mathbf{v}^{(s)}$

- What does it mean for $\mathbf{b}^{(s+1)}$ getting closer to the minimum?
- We want the error $E(\mathbf{b}^{(s+1)})$ is less than the error $E(\mathbf{b}^{(s)})$.
- The difference $\Delta E_{\mathbf{b}}$ should be as negative as possible

$$\Delta E_{\mathbf{b}} = E(\mathbf{b}^{(s+1)}) - E(\mathbf{b}^{(s)}) = E(\mathbf{b}^{(s)} + \alpha \mathbf{v}^{(s)}) - E(\mathbf{b}^{(s)})$$

- Apply Taylor series expansion of $E(\mathbf{b}^{(s)} + \alpha \mathbf{v}^{(s)})$, we get:

$$\begin{aligned}\Delta E_{\mathbf{b}} &= E(\mathbf{b}^{(s)}) + \nabla E(\mathbf{b}^{(s)})^\top (\alpha \mathbf{v}^{(s)}) + O(\alpha^2) - E(\mathbf{b}^{(s)}) \\ &= \alpha \nabla E(\mathbf{b}^{(s)})^\top \mathbf{v}^{(s)} + O(\alpha^2) \approx \alpha \nabla E(\mathbf{b}^{(s)})^\top \mathbf{v}^{(s)}\end{aligned}$$

- The last term involves the inner product between the gradient and a unit vector: $[\nabla E(\mathbf{b}^{(s)})]^\top \mathbf{v}^{(s)}$
- Denote $\mathbf{u} = \nabla E(\mathbf{b}^{(s)})$, we need to find $\mathbf{v}$ to make $\mathbf{u}^\top \mathbf{v}$ as negative as possible.

hướng của v (S)

• Điều đó có ý nghĩa gì khi b (s+1) tiến gần đến giá trị nhỏ nhất? • Chúng tôi muốn lỗi E(b (s+1)) nhỏ hơn lỗi E(b (s)). • Chênh lệch Eb phải càng âm càng tốt

Eb = E(b (s+1)) E(b (s)) = E(b (s) + αv (s)) E(b (s))

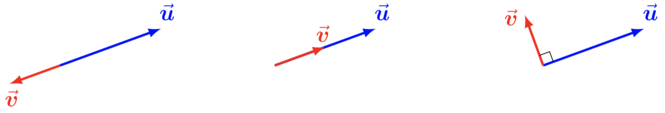
• Áp dụng khai triển chuỗi Taylor của E(b (s) + αv (s)), ta có:

Eb = E(b (s)) + E(b (s)) (αv (s)) + 2) E(b (s))

= α E(b (s)) v (s) + O(α 2 O(α) ≈ α E(b (s)) v (s)

• Số hạng cuối cùng liên quan đến tích bên trong giữa gradient và một vectơ đơn vị: [E(b (s))] v (s) • Biểu thị u = E(b (s)), ta cần tìm v để biến u v thành âm càng tốt.

The direction of $\mathbf{v}^{(s)}$



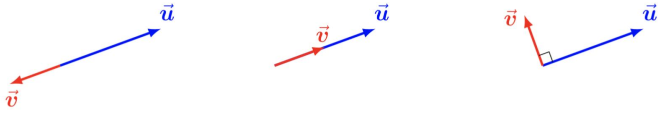
antiparallel parallel orthogonal

- When $\mathbf{u}$ and $\mathbf{v}$ are parallel, then $\mathbf{u}^\top\mathbf{v} = \|\mathbf{u}\|$ (we assume $\mathbf{v}$ to be a unit vector).
- When $\mathbf{u}$ and $\mathbf{v}$ are antiparallel, then $\mathbf{u}^\top\mathbf{v} = -\|\mathbf{u}\|$.
- When $\mathbf{u}$ and $\mathbf{v}$ are orthogonal, then $\mathbf{u}^\top\mathbf{v} = 0$.
- In any case, we have that:

$$\mathbf{u}^\top\mathbf{v} \geq -\|\mathbf{u}\|$$

- The least we can get is $-\|\mathbf{u}\|$ when $\mathbf{v}$ is the opposite direction of $\mathbf{u} = \nabla E(\mathbf{b}^{(s)})$, i.e., the gradient of $E(\mathbf{b})$ at step s.

hướng của v (S)



antiparallel parallel orthogonal

- Khi u và v song song thì u v = u (ta giả sử v là một đơn vị véc

tơ). • Khi u và v đối song song thì u v =

u . • Khi u và v trực giao thì u v =

0. • Trong mọi trường hợp, ta có:

u v ≥ u

- Giá trị nhỏ nhất chúng ta có thể nhận được là u khi v ngược hướng với u = E(b (s)), nghĩa là gradient của E(b) tại bước s.

The direction of $\mathbf{v}^{(s)}$

$$\Delta E_{\mathbf{b}} = \alpha \nabla E(\mathbf{b}^{(s)})^\top \mathbf{v}^{(s)} + O(\alpha^2) \geq -\alpha \| E(\mathbf{b}^{(s)}) \|$$

- To make $\Delta E_{\mathbf{b}}$ as negative as possible, $\mathbf{v}^{(s)}$ should be parallel to the opposite direction of the gradient $\nabla E(\mathbf{b}^{(s)})$:

$$\mathbf{v}^{(s)} = -\frac{\nabla E(\mathbf{b}^{(s)})}{\|\nabla E(\mathbf{b}^{(s)})\|}$$

- The norm division is to make $\mathbf{v}^{(s)}$ a unit vector. However, we don't need to implement this normalization because it can be absorbed into the step size α.
- **Gradient Descent**: We are descending in the direction opposite to the gradient of the error function.

hướng của v (S)

Eb = α E(b (s)) v (s) + O(α 2) ≥ α E(b (s))

Để làm cho Eb càng âm càng tốt, v ngược (s) phải song song với • hướng với gradient E(b (s)):

v (S) = E(b (s)) E(b(s))

(s) • Phép chia chuẩn là để tạo va véc tơ đơn vị. Tuy nhiên, chúng tôi không cần thực hiện chuẩn hóa này vì nó có thể được hấp thụ vào kích thước bước α. •

Gradient Descent: Chúng ta đang giảm dần theo hướng ngược lại với gradient của hàm lỗi.

GD for Linear Regression in Vector-Matrix Notation

- The error function $E(\mathbf{b})$:

$$E(\mathbf{b}) = \frac{1}{n}(\mathbf{y} - \mathbf{X}\mathbf{b})^\top(\mathbf{y} - \mathbf{X}\mathbf{b}) = \frac{1}{n}(\mathbf{b}\mathbf{X}^\top\mathbf{X}\mathbf{b} - 2\mathbf{b}^\top\mathbf{X}^\top\mathbf{y} + \mathbf{y}^\top\mathbf{y})$$

- The formula for its gradient $\nabla E(\mathbf{b})$ is:

$$\nabla E(\mathbf{b}) = \frac{1}{n}(2\mathbf{X}^\top\mathbf{X}\mathbf{b} - 2\mathbf{X}^\top\mathbf{y}) = \frac{2}{n}\mathbf{X}^\top(\mathbf{X}\mathbf{b} - \mathbf{X}^\top\mathbf{y})$$

- The algorithm:
  1. Initialize $\mathbf{b}^{(0)} = (b_0^{(0)}, b_1^{(0)}, \dots, b_p^{(0)})$
  2. For $s = 0, 1, 2, \dots$ do:
     - Update model parameters $\mathbf{b}$:

$$\mathbf{b}^{(s+1)} = \mathbf{b}^{(s)} - \alpha\nabla E(\mathbf{b}^{(s)}) = \mathbf{b}^{(s)} - \alpha\left[\frac{2}{n}\mathbf{X}^\top(\mathbf{X}\mathbf{b}^{(s)} - \mathbf{X}^\top\mathbf{y})\right]$$

     - When there is little change between $\mathbf{b}^{(k+1)}$ and $\mathbf{b}^{(k)}$ (for some k), we assume the algorithm **converged** and $\mathbf{b}^* = \mathbf{b}^{(k+1)}$.

GD cho hồi quy tuyến tính trong ký hiệu ma trận vectơ • Hàm lỗi E(b):

E(b) = T/N (y Xb) (y Xb) = 1/N (bX Xb 2b X y + y y)

- Công thức cho độ dốc E(b) của nó là:

E(b) = 1/N (2X Xb 2X y) = 2/N X (Xb X y)

- Thuật toán:
  1. Khởi tạo b (0) = (b 0 (0), b1(0), . . . , bp(0))
  2. Với s = 0, 1, 2, . . . làm:
     - Cập nhật thông số mô hình b:

b (s+1) = b (s) (s) α E(b) = b (s) α[2/N X (Xb(s) X y)]

     - Khi có ít thay đổi giữa b (k+1) và b (k) (với k nào đó), (k+1) ta giả sử thuật toán hội tụ và b= b .

GD for Linear Regression in Pointwise Notation

- The error function $E(\mathbf{b})$:

$$E(\mathbf{b}) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \mathbf{b}^\top \mathbf{x}_i\right)^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}(y_i - b_0x_{i0} - b_1x_{i1} - \ldots - b_jx_{ij} - \ldots - b_px_{ip})$$

- The partial derivate wrt b_j is:

$$\frac{\partial E(\mathbf{b})}{\partial b_j} = -\frac{2}{n}\sum_{i=1}^{n}(y_i - b_0x_{i0} - b_1x_{i1} - \ldots - b_jx_{ij} - \ldots - b_pxip)x_{ij}$$

$$= -\frac{2}{n}\sum_{i=1}^{n}\left(y_i - \mathbf{b}^\top \mathbf{x}_i\right)x_{ij}$$

GD cho hồi quy tuyến tính trong Pointwise Notation

- Chức năng báo lỗi E(b):

E(b) = 1/n N i=1 (yi b xi) 2 = 1/n N i=1 (yi y^ i) 2

= 1/n N i=1 (yi b0xi0 b1xi1 ··· bjxij ··· bpxip)

- Đạo hàm riêng wrt bj là:

E(b)/bj = 2/n N i=1 (yi b0xi0 b1xi1 ··· bjxij ··· bpxip)xij

= 2/n N i=1 (yi b xi) xij

GD for Linear Regression in Pointwise Notation

1. Initialize $\mathbf{b}^{(0)} = (b_0^{(0)}, b_1^{(0)}, \ldots, b_p^{(0)})$
2. For $s = 0, 1, 2, \ldots$ do:
   - Update model parameters $\mathbf{b}$:

$$b_j^{(s+1)} = b_j^{(s)} + \alpha \cdot \frac{\partial}{\partial b_j} E(\mathbf{b}^{(s)})$$

$$= b_j^{(s)} + \alpha \cdot \frac{2}{n} \sum_{i=1}^{n} \left(y_i - [\mathbf{b}^{(s)}]^\top \mathbf{x}_i \right) x_{ij}$$

     for all $j = 0, 1, \ldots, p$ simultaneously.
   - Store these elements into the vector:

$$\mathbf{b}^{(s+1)} = (b_0^{(s+1)}, b_1^{(s+1)}, \ldots, b_p^{(s+1)})$$

   - When there is little change between $\mathbf{b}^{(k+1)}$ and $\mathbf{b}^{(k)}$ (for some k), we assume the algorithm **converged** and $\mathbf{b}^* = \mathbf{b}^{(k+1)}$.

GD cho hồi quy tuyến tính trong Pointwise Notation

1 Khởi tạo b (0) = (b 0 (0) , b(0) 1 , . . . , b(0) P)

2 Với s = 0, 1, 2, . . . làm: •

Cập nhật thông số mô hình b:

+ α (s+1) jj (s) b = b 2 (s) E(b) bj

= b + α (s) j N n i=1 (yi [b (S)] xi) xij

với mọi j = 0, 1, . . . , p đồng thời. • Lưu trữ các phần tử này vào vector:

b (s+1) = (b 0 (s+1) , b(s+1) 1 , . . . , b(s+1) P)

• Khi có ít thay đổi giữa b (k+1) và b (k) (với k nào đó), = b (k+1) ta giả sử thuật toán hội tụ và b .

VIET NAM NATIONAL UNIVERSITY HCMC
UNIVERSITY OF INFORMATION TECHNOLOGY
VNUHCM - UIT