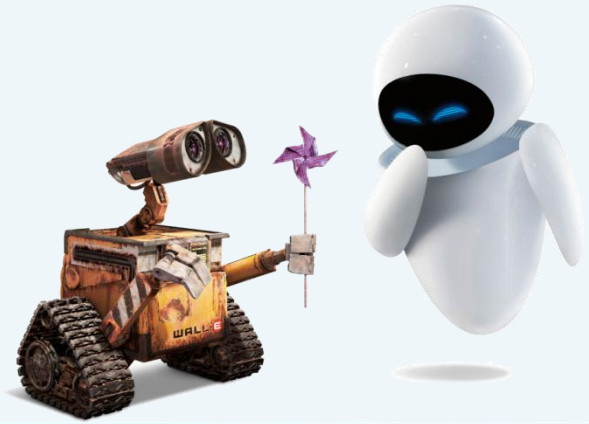
Trường Đại học Khoa học Tự nhiên
Khoa Công nghệ Thông tin

TÀI LIỆU LÝ THUYẾT MÁY HỌC

HỒI QUY TUYẾN TÍNH

Giảng viên: ThS. Lê Ngọc Thành
Email: lnthanh@fit.hcmus.edu.vn



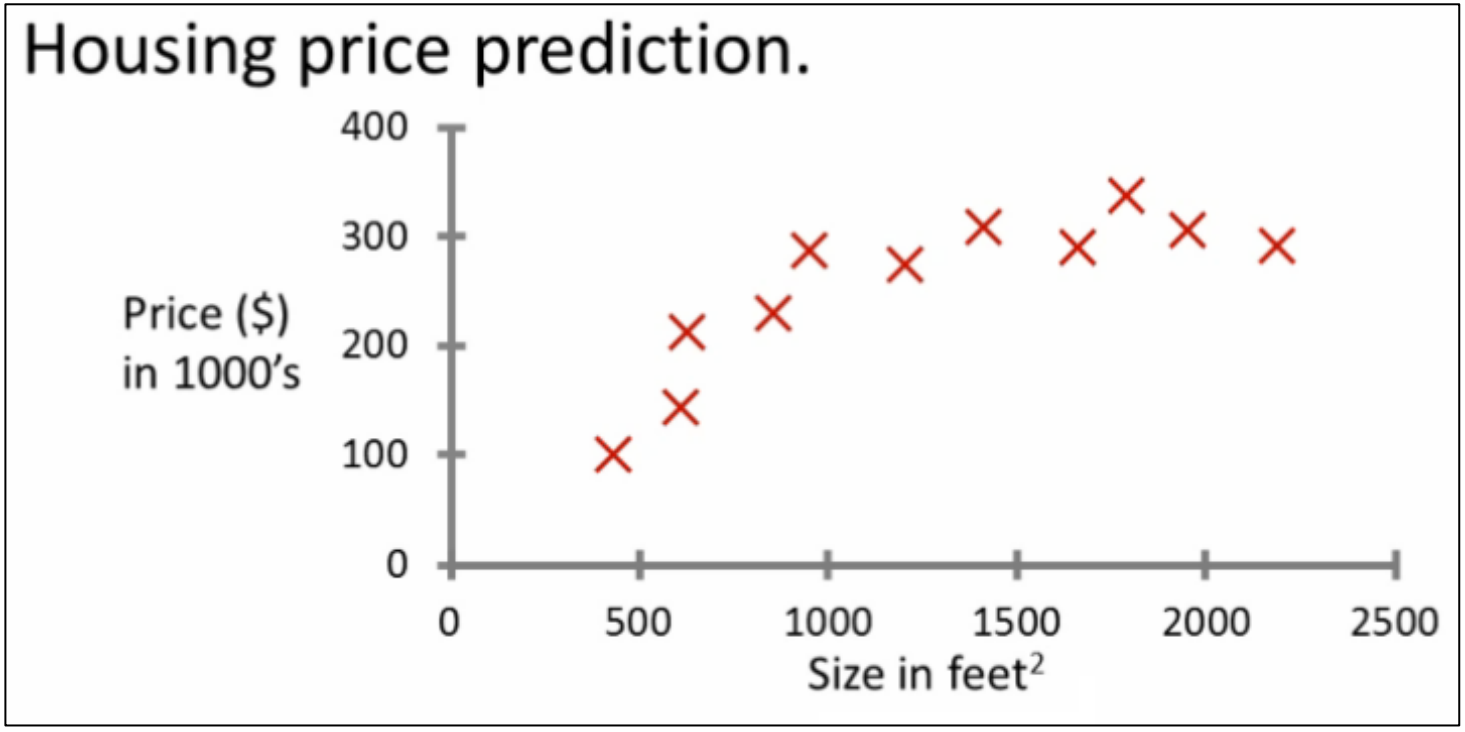
Winter 2012

Nội dung

- **Hồi quy tuyến tính**
  - Khái niệm
  - Phân biệt với mô hình phân lớp
  - Các loại mô hình tuyến tính
  - Ứng dụng
- Hồi quy tuyến tính với một biến
- Hồi quy tuyến tính với nhiều biến
- Hồi quy đa thức
- Biểu thức chuẩn

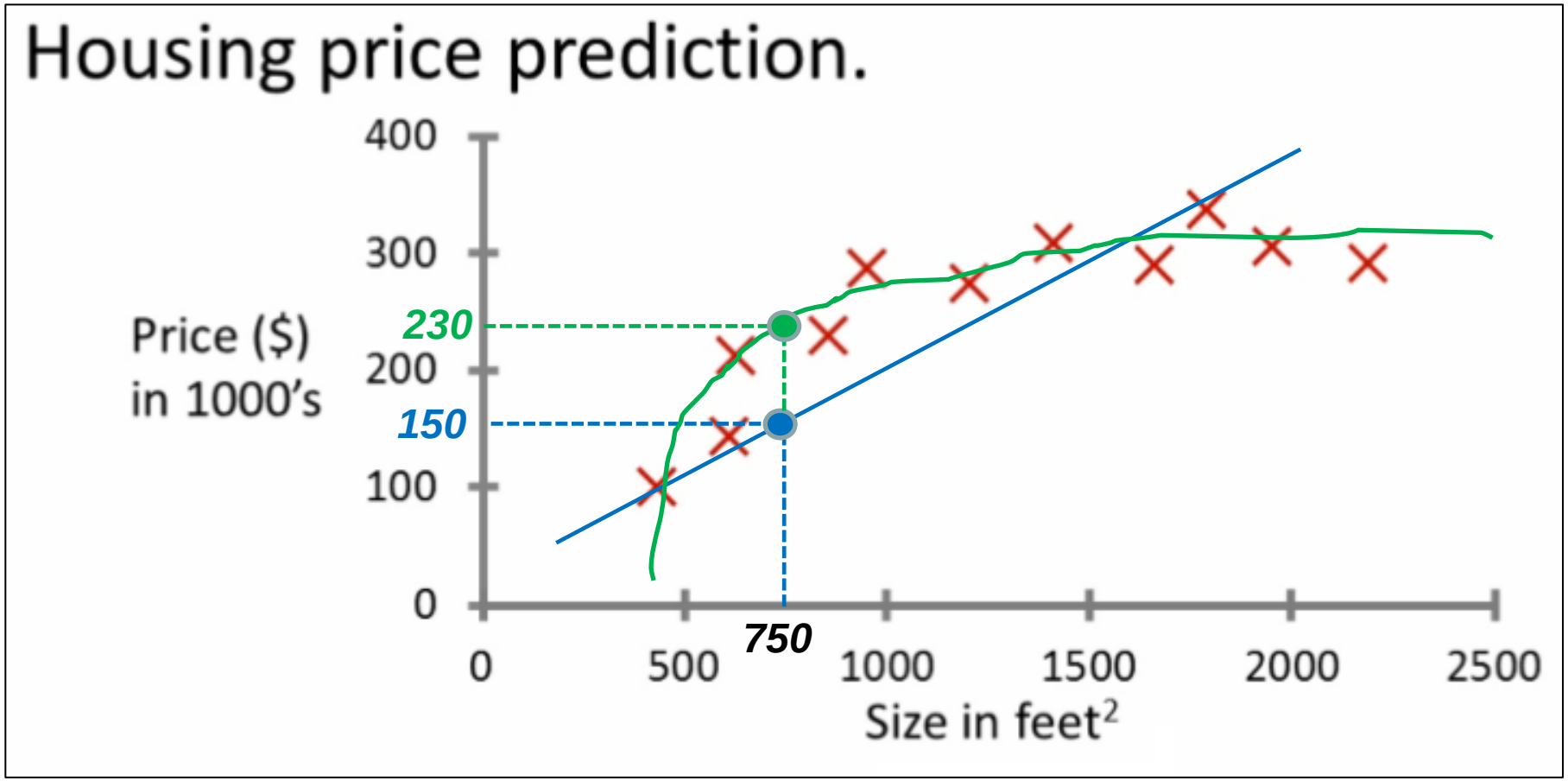
Tình huống 1

- Như thế nào để dự đoán giá nhà?
  - Tập hợp các dữ liệu liên quan đến giá nhà.
  - Chúng liên quan đến kích thước như thế nào?



- *Cho một căn nhà có kích thước 750 thước vuông, vậy giá mong đợi của nó là bao nhiêu?*

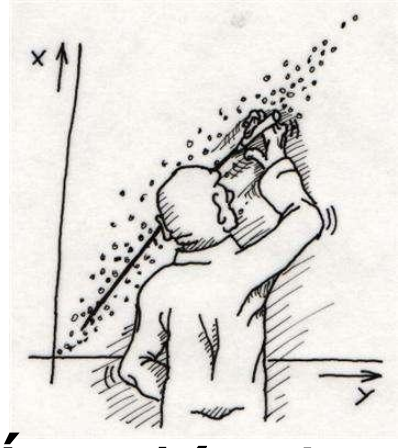
Tình huống 1 (tt)



- *Phương pháp giải quyết:*
  - Vẽ đường thẳng xuyên qua dữ liệu có sẵn
    - Giá nhà có thể là 150
  - Vẽ đường đa thức bậc 2
    - Giá nhà có thể là 230

Bài toán hồi quy

- Cho trước một tập dữ liệu đã có "*câu trả lời đúng*" hay đã cung cấp các giá trị output.
- Thuật toán sẽ học từ dữ liệu có sẵn này (training data) để rút ra được mô hình dự đoán (predictor).



- Nếu giá trị output là một ***giá trị liên tục***, ta có ***bài toán hồi quy*** (regression).

- Nếu giá trị output là ***rời rạc hữu hạn***, ta có ***bài toán phân lớp*** (classification).

Một số kí hiệu

- *Tập huấn luyện của giá nhà*

Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

m=47

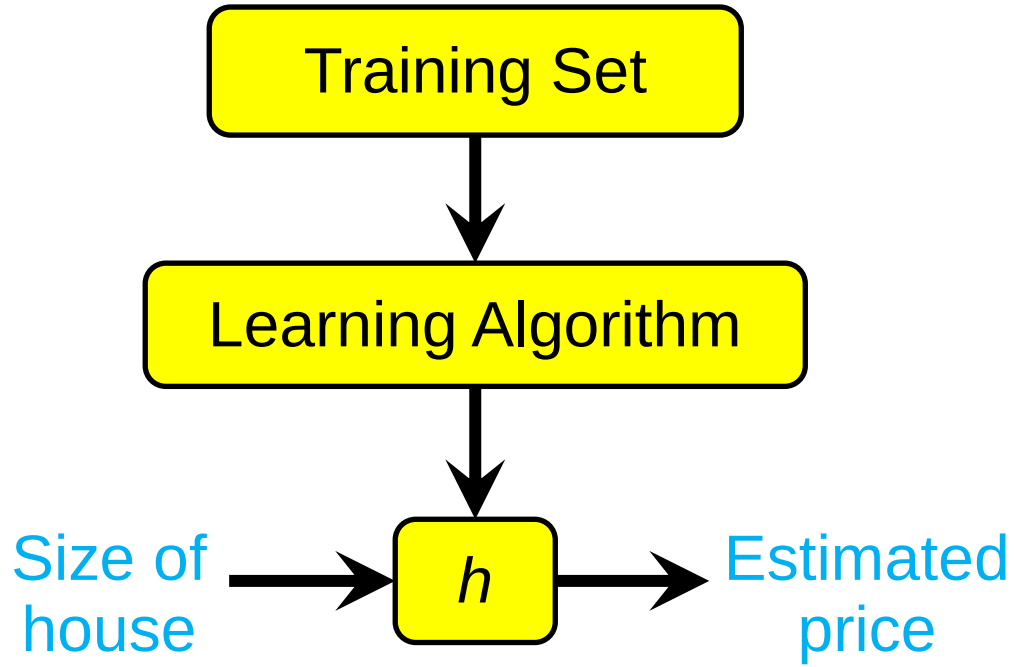
- Kí hiệu:
  - **m**: số mẫu huấn luyện
  - **x**: biến "input"/đặc trưng
  - **y**: biến "output"/biến "target"

(x,y): một mẫu huấn luyện

(x^i,y^i): mẫu huấn luyện thứ i (i=1,…,m)

$x^1 = 2104$
$y^1 = ?$

Hồi quy tuyến tính (1/2)



- Có dữ liệu học, cần một thuật toán học tốt để dự đoán giá trị output (liên tục).
- Giả thuyết (hypothesis), thuật toán đưa ra một hàm hồi quy (h) nhận giá trị input và trả ra giá trị dự đoán.
- Hàm đơn giản để giải quyết là ***hàm hồi quy tuyến tính***.

Hồi quy tuyến tính (2/2)

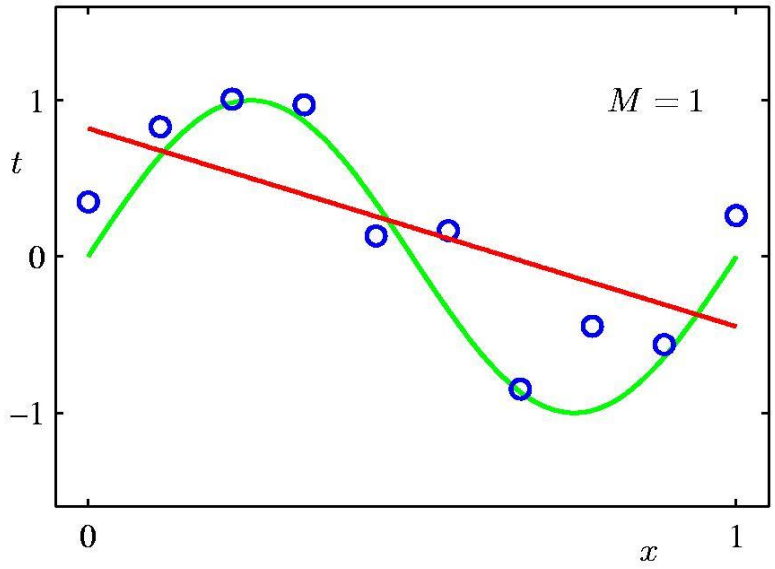
- Thể hiện hàm hồi quy tuyến tính:

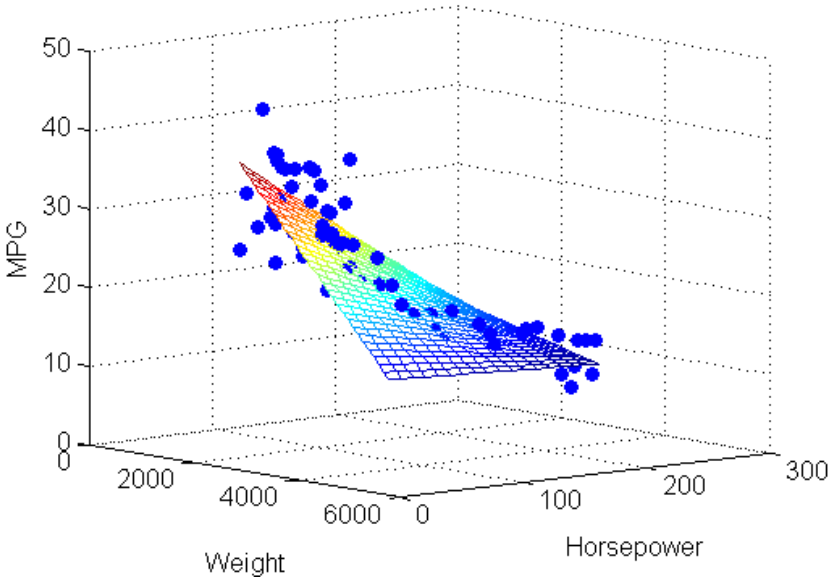
$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \ ...$$

- Hàm này là "tuyến tính" trên các tham số $\theta_0, \theta_1, ..., \theta_n$. Tham số cũng được gọi là trọng số (weight).
- Để đơn giản, hàm cũng được gọi là hàm tuyến tính của biến **x** (liên kết tuyến tính của các biến input).

Các loại hồi quy tuyến tính

- Thể đơn giản nhất:
  - Hồi quy tuyến tính đơn thức trên **một** biến input.
  - $h_\theta(x) = \theta_0 + \theta_1 x$
  - Univariate linear regression.
- Hồi quy đa thức trên một biến:
  - Ví dụ: $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$
  - Polinominal linear regression.
- Hồi quy trên nhiều biến input:
  - $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ \ldots$
  - Multivariate linear regression.



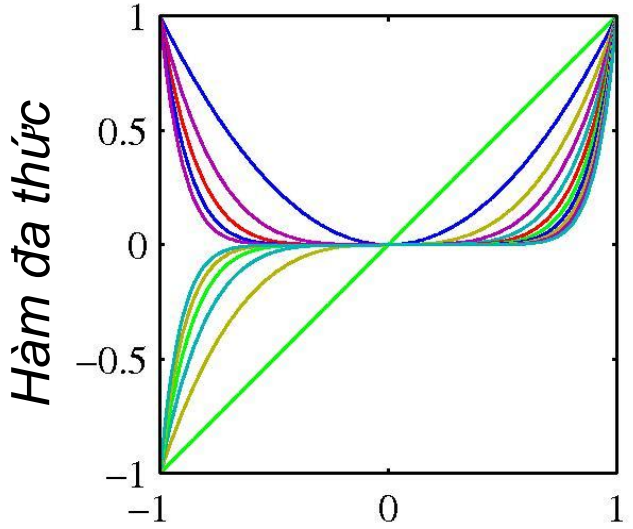


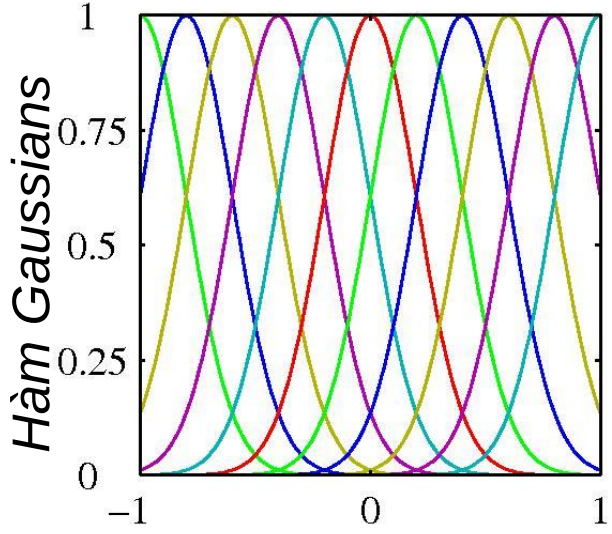
Hồi quy với hàm cơ sở

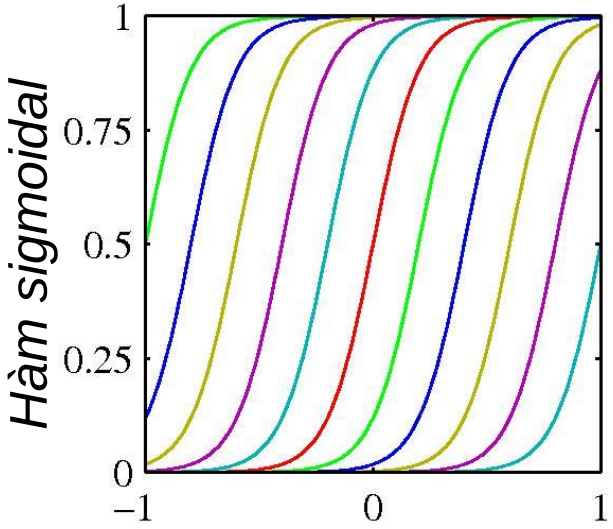
- Giá trị *input x* có thể là một giá trị thực. Tuy nhiên nó có thể được thể hiện qua các *hàm phi tuyến*, người ta gọi là ***hàm cơ sở*** (basic function). Kí hiệu: $\phi(x)$

$$h_\theta(x) = \theta_0 + \theta_1\phi_1(x) + \theta_1\phi_2(x) + \ \ldots$$

- Hàm hồi quy đa thức là trường hợp đặc biệt với hàm cơ sở $\phi(x) = \ x^2$ ...







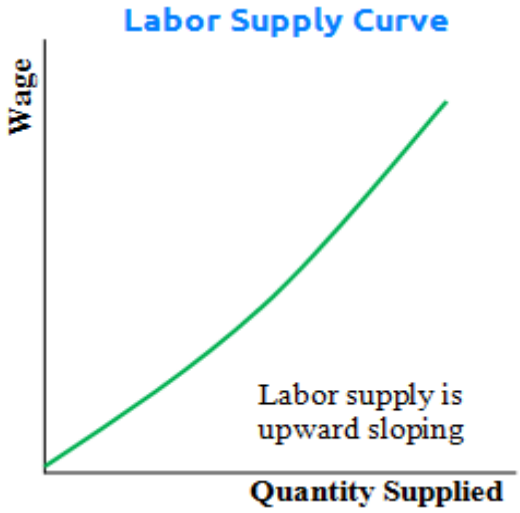
Ứng dụng của hàm hồi quy TT (1/4)

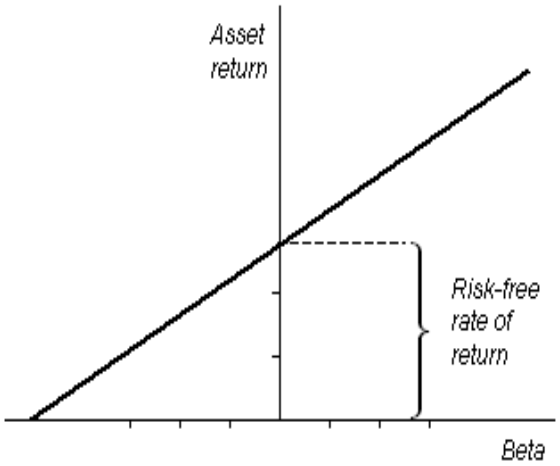
- Nếu mục tiêu là *dự đoán* hay dự báo (prediction/ forecasting), hồi quy tuyến tính dùng để "*khớp*" mô hình dự đoán với tập dữ liệu quan sát được của (**x**,**y**).

- Sau khi có được mô hình, với **x** mới (chưa có **y**), mô hình được sử dụng để đoán **y**.

Ứng dụng của hàm hồi quy TT (2/4)

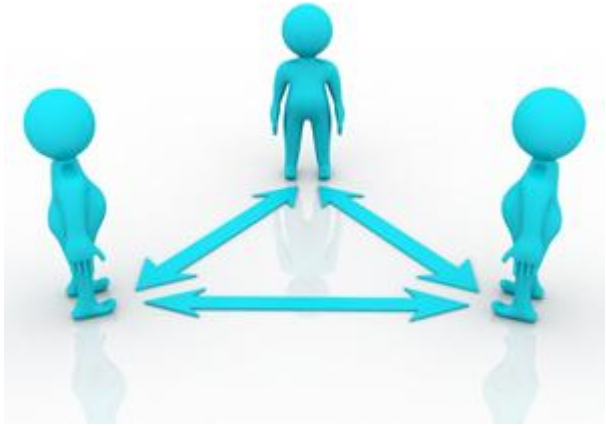
- Ví dụ ứng dụng dự đoán:
  - Dự đoán xu hướng (trend estimation) của giá dầu, GDP, cổ phiếu tăng hay giảm qua từng chu kỳ (time series)
  - Trong kinh tế, dự đoán chi tiêu tiêu dùng, đầu tư hàng tồn kho, định giá xuất khẩu, nhu cầu lao động…
  - Trong tài chính, được sử dụng để định lượng rủi ro ở mức hệ thống.





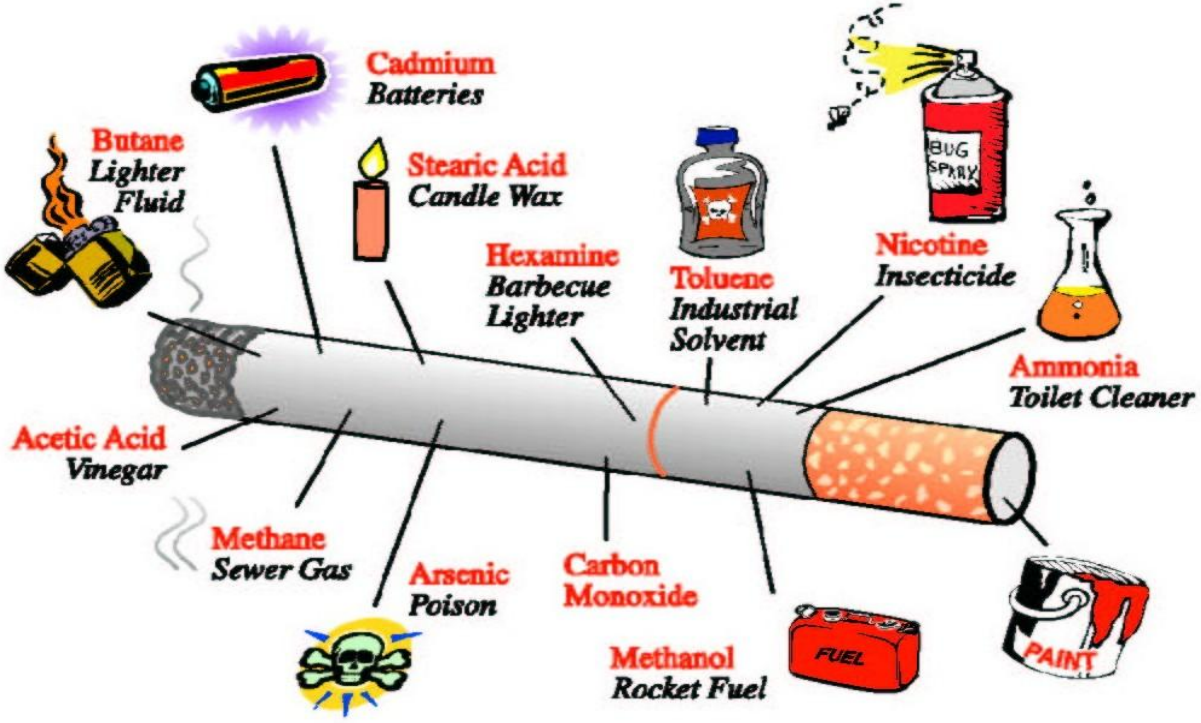
Ứng dụng của hàm hồi quy TT (3/4)

- Cho trước một biến **y** và tập các biến $\mathbf{x_1}$, $\mathbf{x_2}$, … có thể liên quan đến **y**, hồi quy tuyến tính có thể được áp dụng để:
  - Đánh giá *độ mạnh của mối quan hệ* **y** và $\mathbf{x_j}$.
  - Hoặc để đánh giá $\mathbf{x_j}$ nào hoàn toàn *không liên quan* đến **y**.
  - Hoặc *xác định tập con* nào của $\mathbf{x_j}$ chứa thông tin lặp lại về **y**.



Ứng dụng của hàm hồi quy TT (4/4)

- Ví dụ ứng dụng độ liên quan:
  - Tìm hiểu sự liên quan của hút thuốc đến tỷ lệ tử vong và bệnh tật.
  - Tác động của hút thuốc không phụ thuộc vào trình độ học vấn, giáo dục hay thu nhập.



Nội dung

- Hồi quy tuyến tính
- **Hồi quy tuyến tính với một biến**
  - Thể hiện mô hình
  - Hàm chi phí
  - Gradient Descent cho một biến
- Hồi quy tuyến tính với nhiều biến
- Hồi quy đa thức
- Biểu thức chuẩn

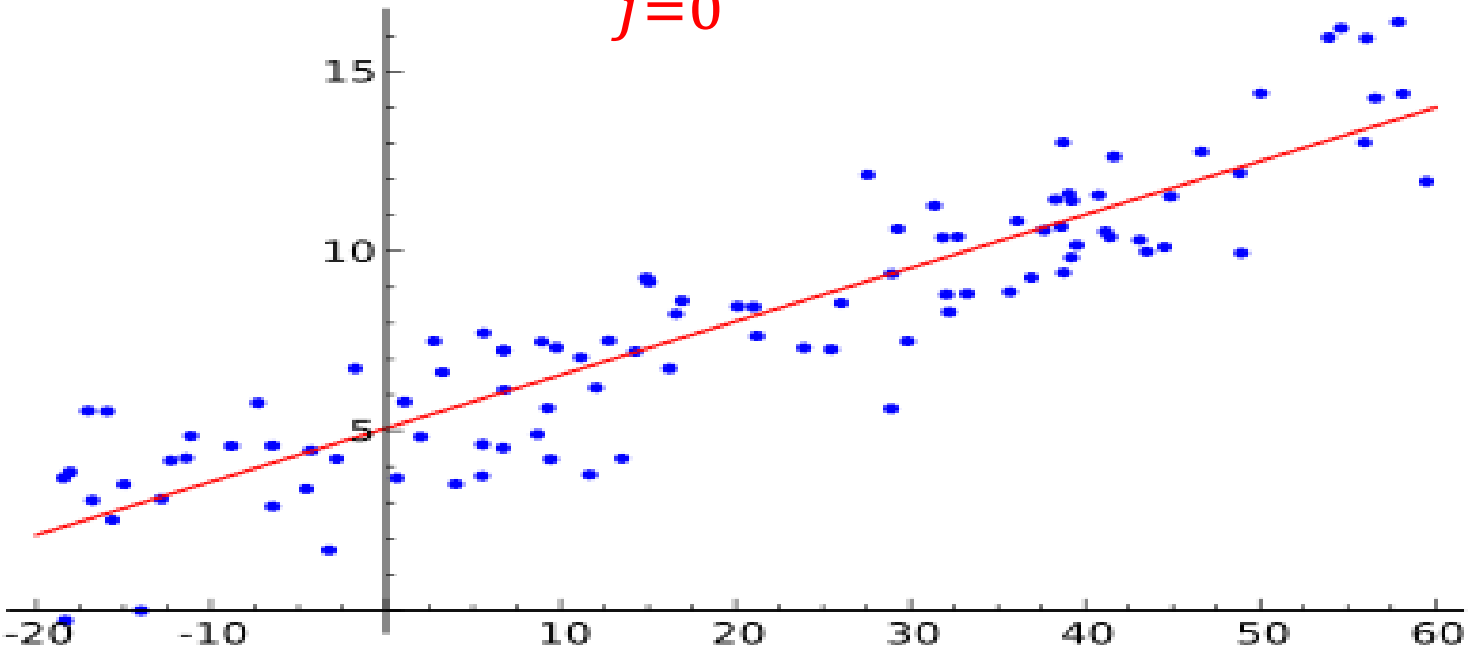
Thể hiện mô hình

- Hàm tuyến tính được thể hiện:

$$h_\theta(x) = \theta_0 + \theta_1 x$$

- Đặt $x_0 = 1$, ta có thể viết:

$$h_\theta(x) = \sum_{j=0}^{1} \theta_j x_j = \boldsymbol{\theta}^T \boldsymbol{x} = \boldsymbol{x}^T \boldsymbol{\theta}$$



Ví dụ hàm tuyến tính đơn biến

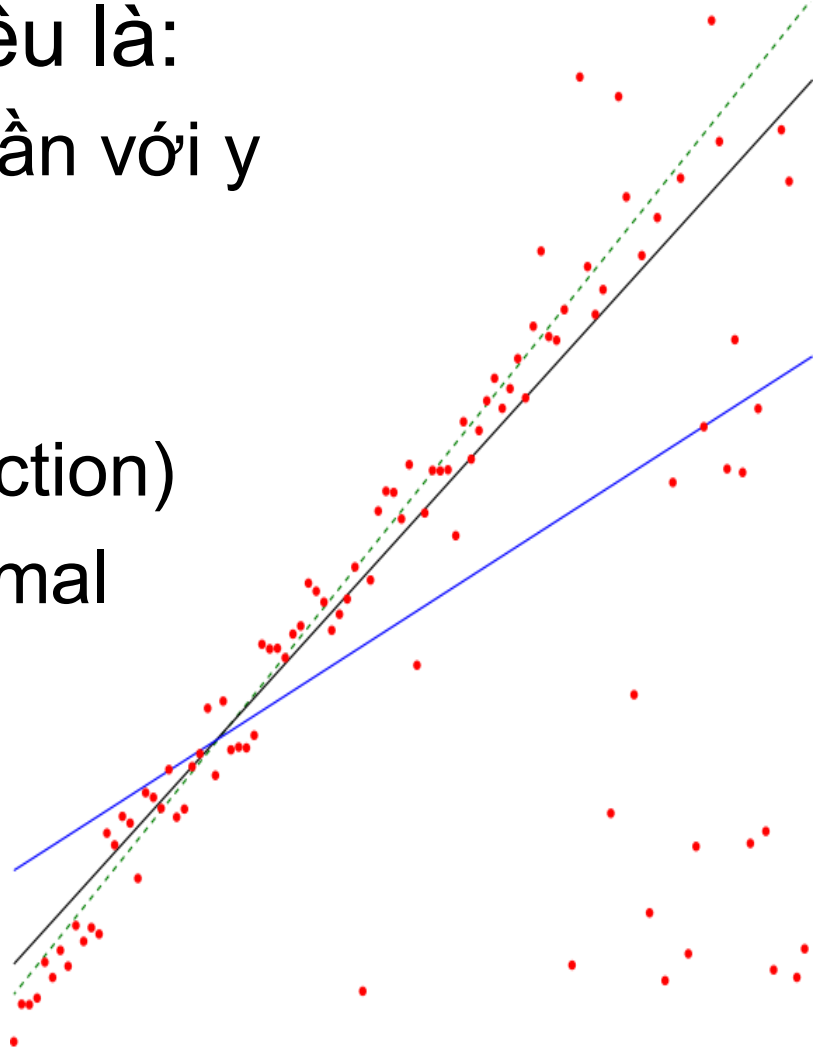
House sizes:

2104
1416
1534
852

$$h_\theta(x) = -40 + 0.25x$$

Học hồi quy tuyến tính

- Với dữ liệu cho trước, mục tiêu là:
  - *Học các tham số θ* để mà h_θ gần với y trong các mẫu huấn luyện
- Phương pháp học:
  - Dựa trên hàm chi phí (cost function)
  - Dựa trên biểu thức chuẩn (normal equation)
  - …
- Mỗi phương pháp học có thể ra các bộ tham số khác nhau.



Bài tập 1 – Xác định HQTT

- Cho dữ liệu giá nhà:

Size in feet² (x)	Price ($) in 1000's (y)
100	10
800	150

- Xác định hàm hồi quy tuyến tính đơn biến?

$$h_\theta(x) = -10 + 0.2x$$

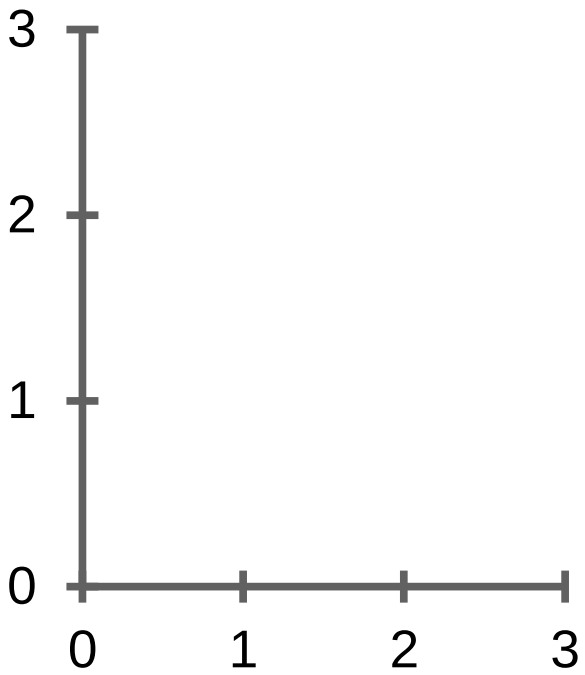
- Với dữ liệu?

Size in feet² (x)	Price ($) in 1000's (y)
100	10
800	150
1534	315
852	178

$$h_\theta(x) = \theta_0 + \theta_1 x$$

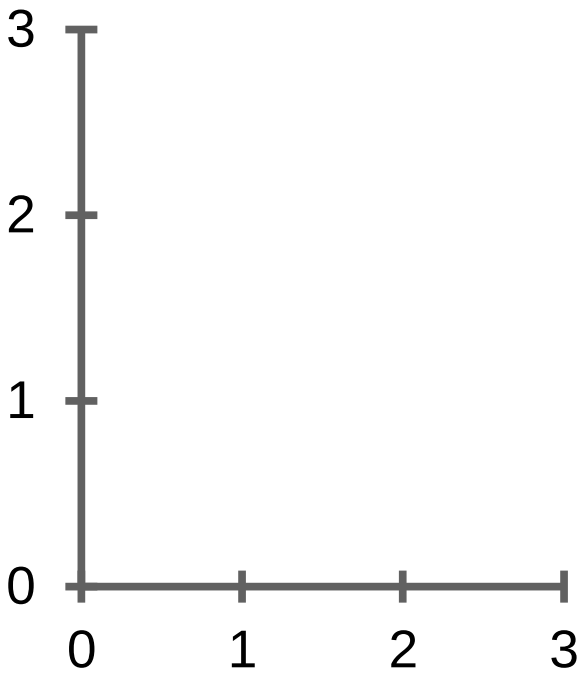
Một số dạng HQTT đơn biến

$$h_\theta(x) = \theta_0 + \theta_1 x$$



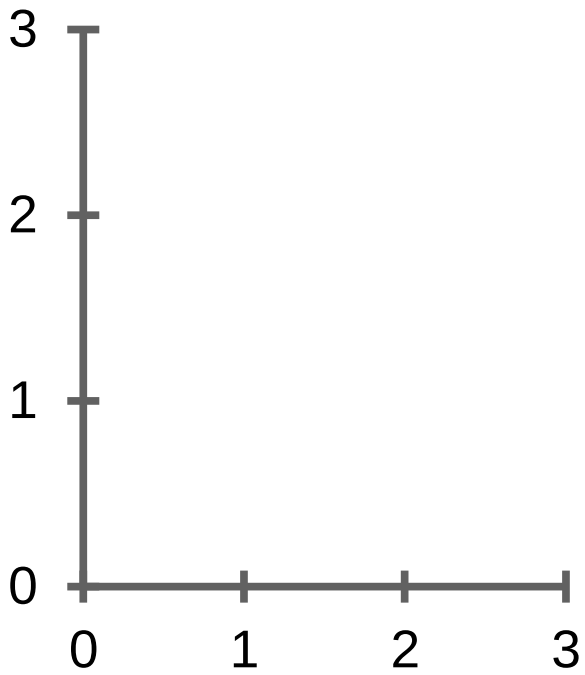
$\theta_0 = 1.5$

$\theta_1 = 0$



$\theta_0 = 0$

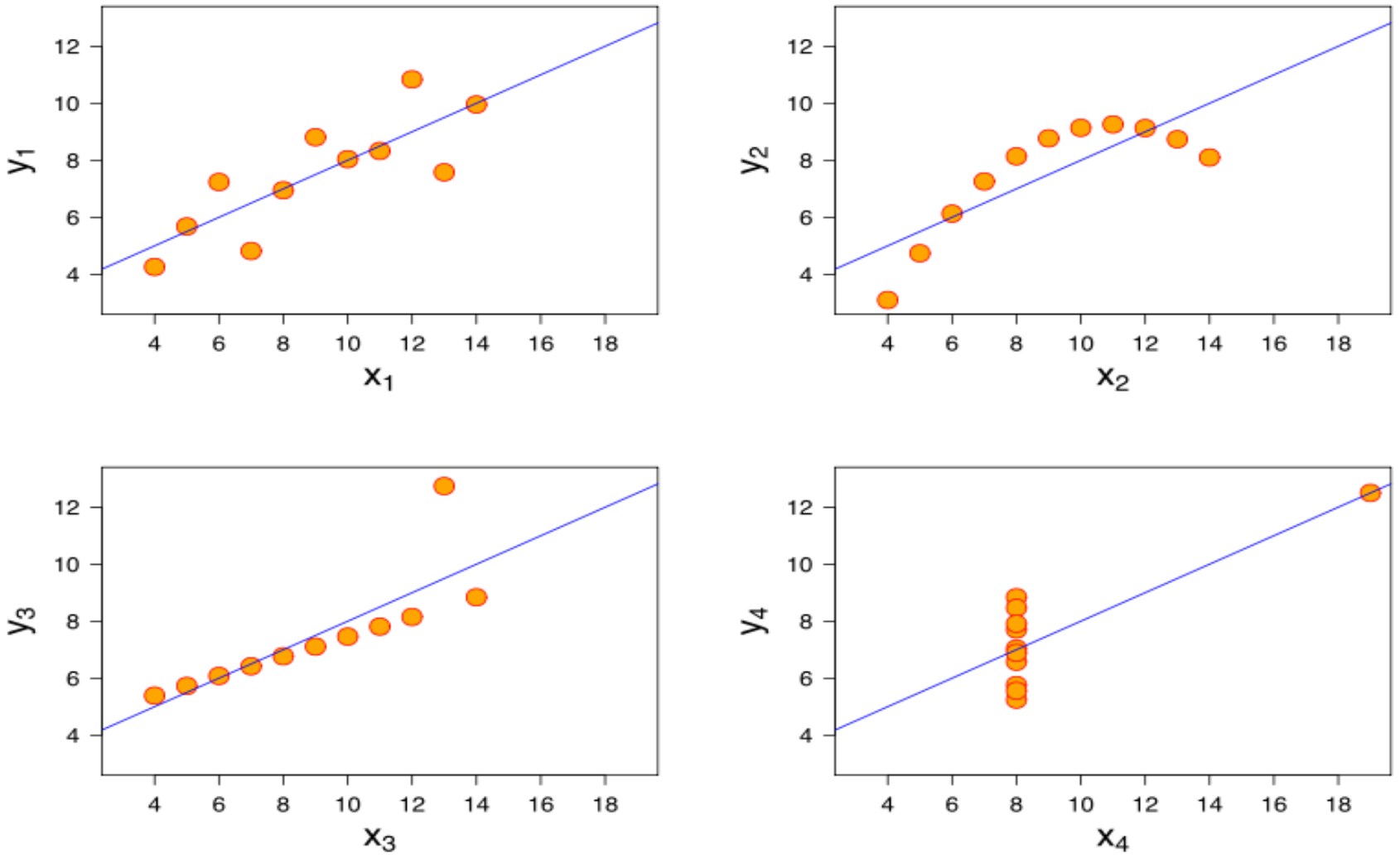
$\theta_1 = 0.5$



$\theta_0 = 1$

$\theta_1 = 0.5$

Một số dạng HQTT đơn biến (tt)



Tập dữ liệu trong Anscombe's quartet có cùng đường hồi quy tuyến tính nhưng dữ liệu lại phân bố khác nhau

Hàm chi phí

- Phương pháp học dựa trên việc đánh giá sự khác biệt giữa hàm h(x) so với y, gọi là ***hàm chi phí*** (cost function):

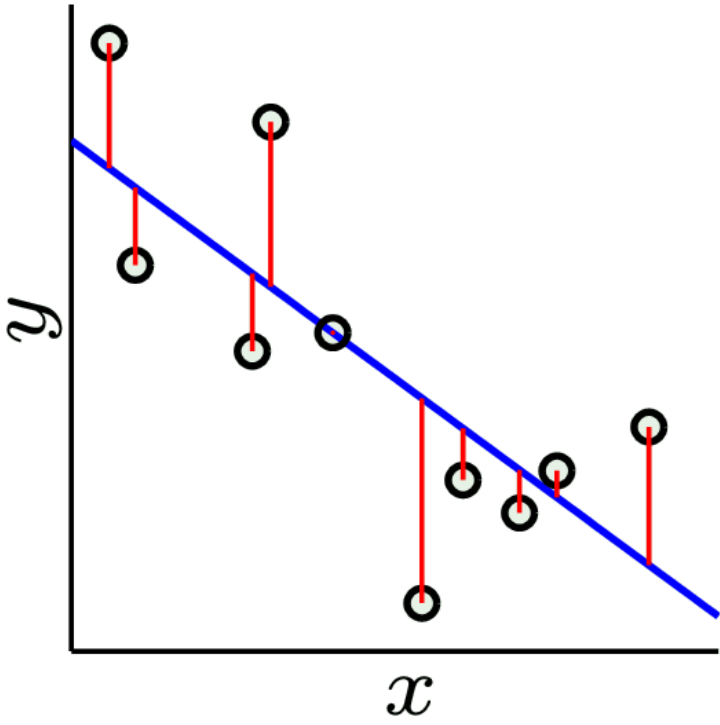
$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^i) - y^i\right)^2$$

với m là số mẫu được huấn luyện

$\frac{1}{2m}$*: dùng cho đạo hàm và chuẩn hóa*

h_θ*: hàm hồi quy tuyến tính đơn biến*

y^i*: output mong muốn*

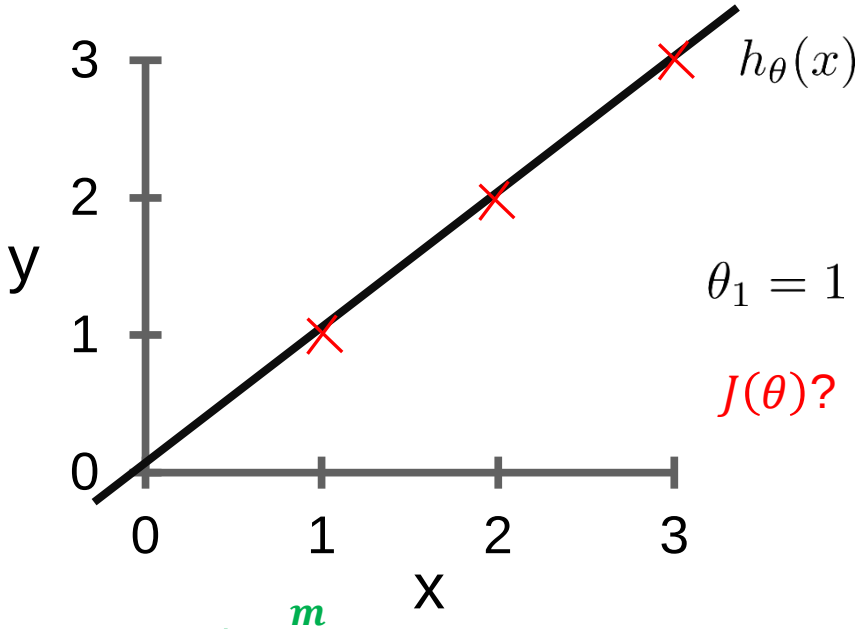


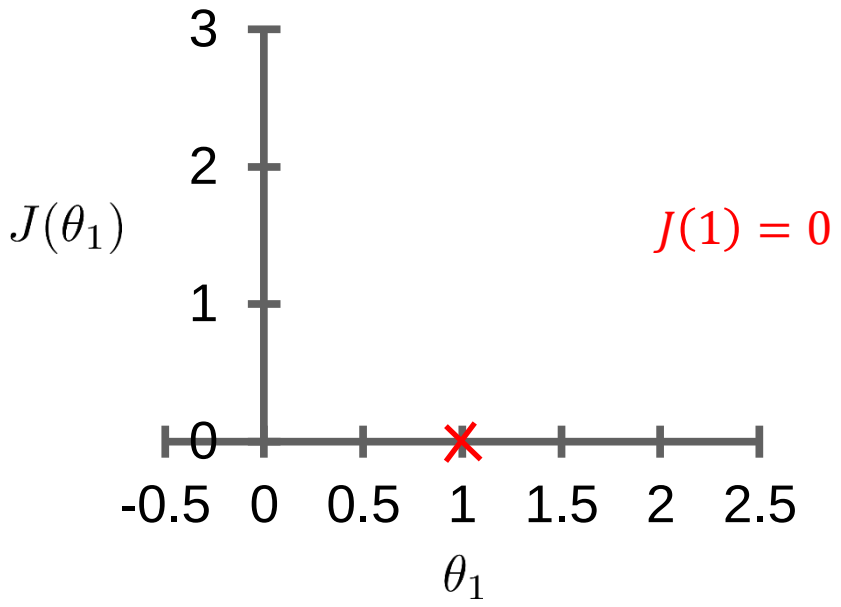
- Mục tiêu là làm cho hàm chi phí nhỏ nhất: $minimize_{\theta_0,\theta_1} J(\theta)$

Hình dạng hàm chi phí

- Đơn giản nhất, cho $\theta_0 = 0$:

$$h_\theta(x) = \theta_1 x$$





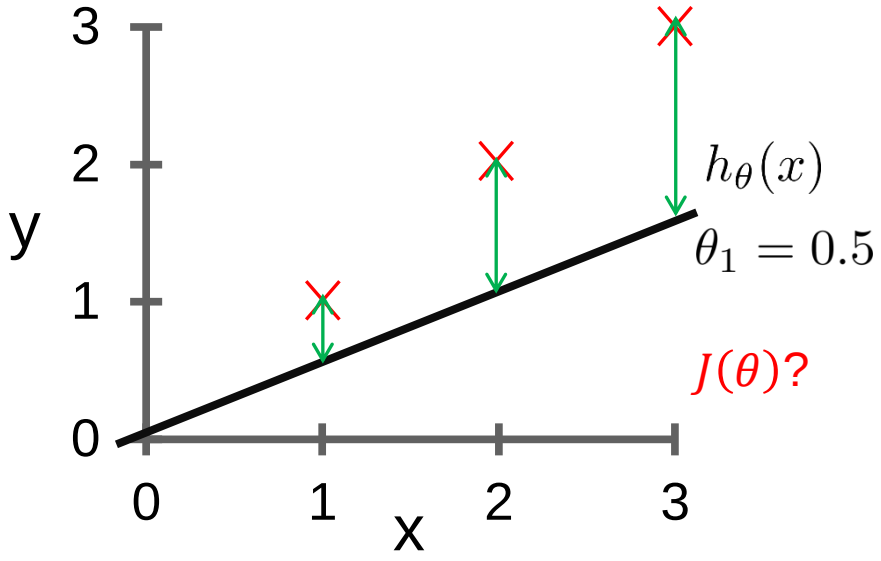
$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^i) - y^i\right)^2$$

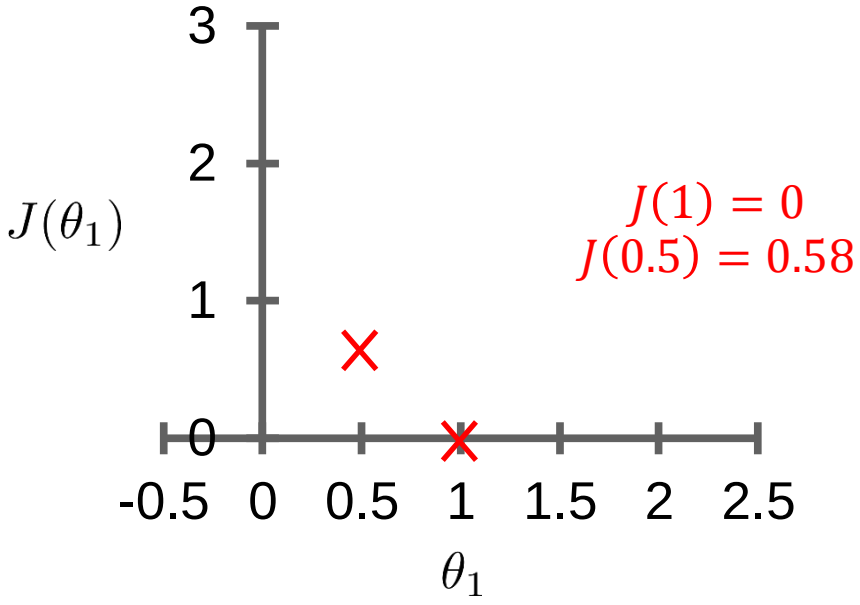
$$= \frac{1}{2m}\sum_{i=1}^{m}\left(\theta_1 x^i - y^i\right)^2$$

$$= \frac{1}{2m}\left(0^2 + 0^2 + 0^2\right) = 0$$

$\theta_1 = 0.5$?

Hình dạng hàm chi phí





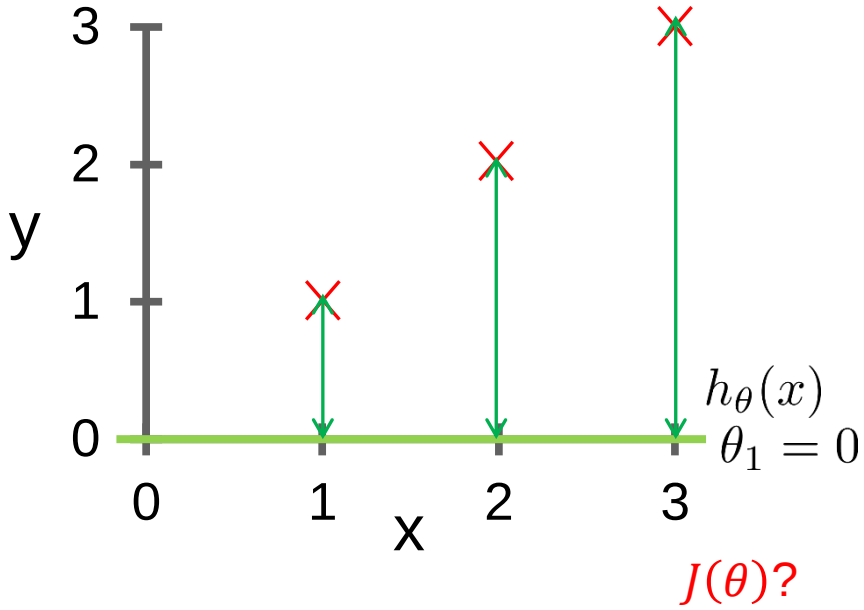
$$J(0.5) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^i) - y^i\right)^2$$

$$= \frac{1}{2 \times 3}\left((0.5-1)^2+(1-2)^2+(1.5-3)^2\right)$$

$$\approx 0.58$$

$\theta_1 = 0$?

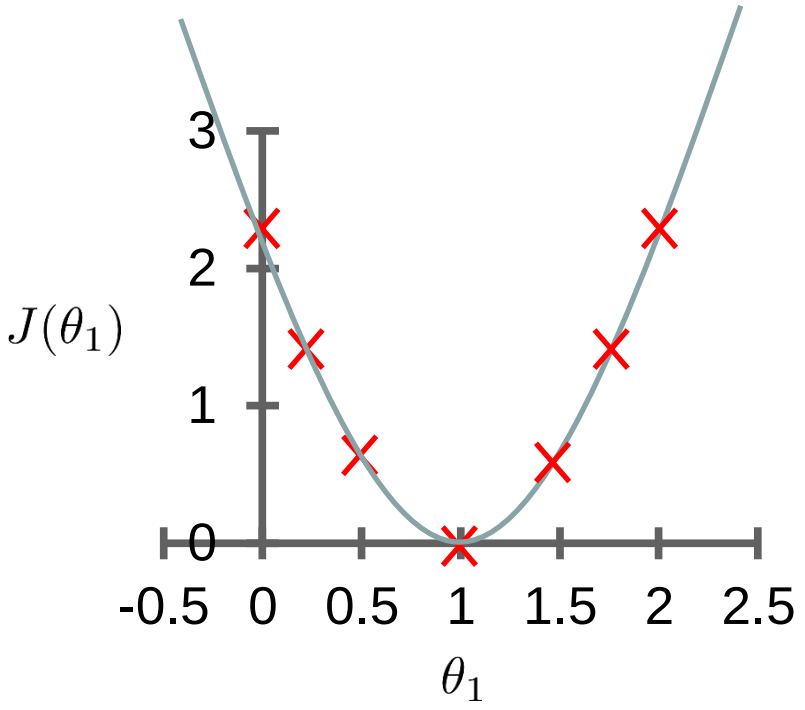
Hình dạng hàm chi phí



$J(\theta)$?

$$J(0) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^i) - y^i\right)^2$$

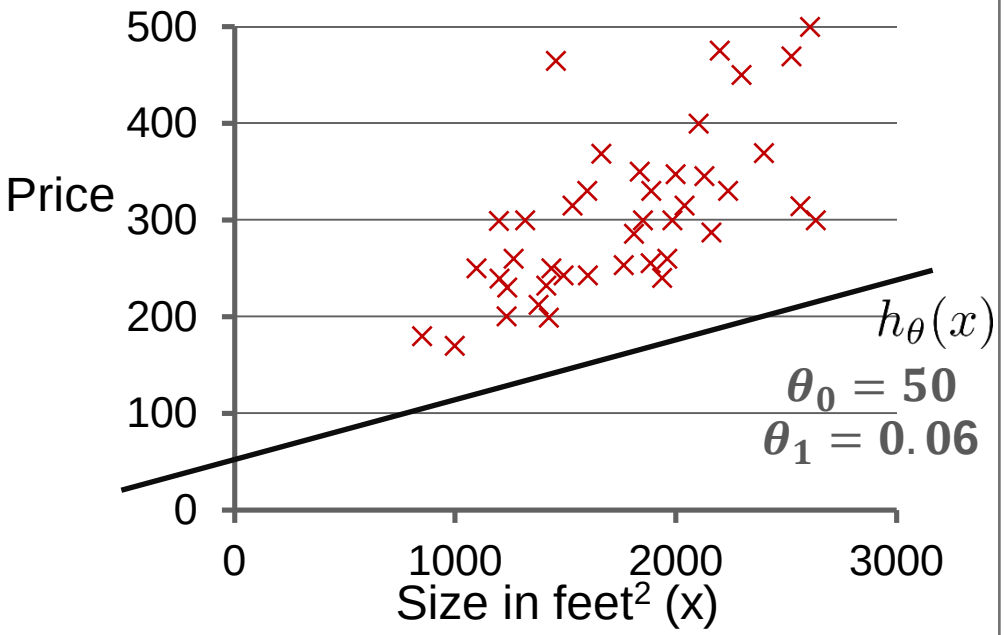
$$= \frac{1}{2 \times 3}\left(1^2 + 2^2 + 3^2\right) \approx 2.3$$



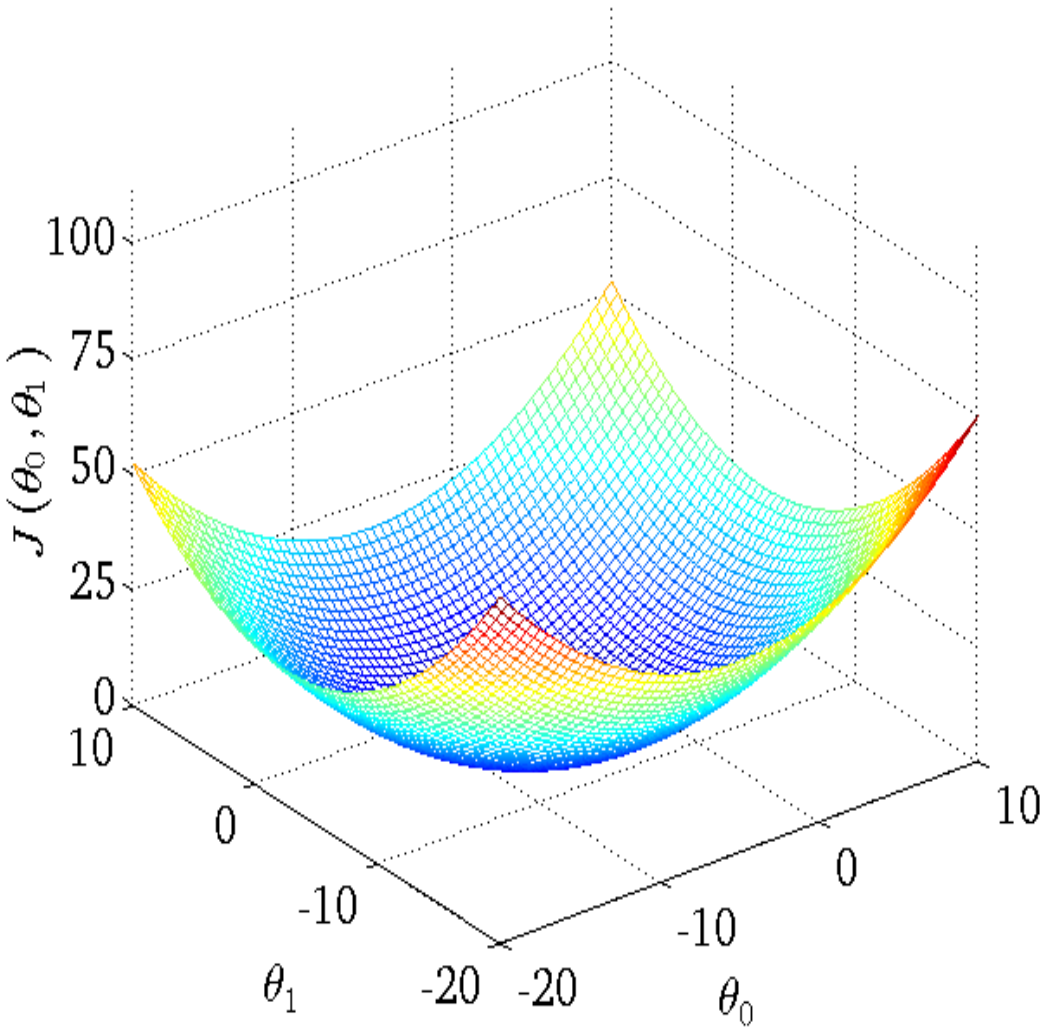
$minimize_{\theta_1} J(\theta)$ tại $\theta_1 = 1$

Hình dạng hàm chi phí

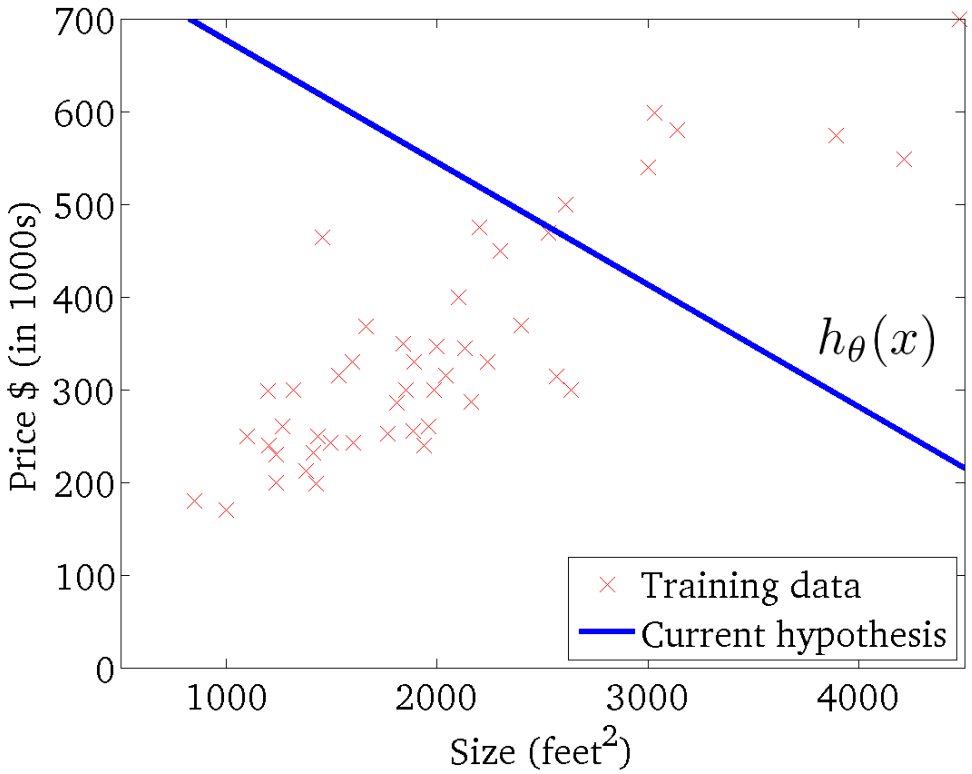
- Với θ_0, θ_1 bất kỳ:



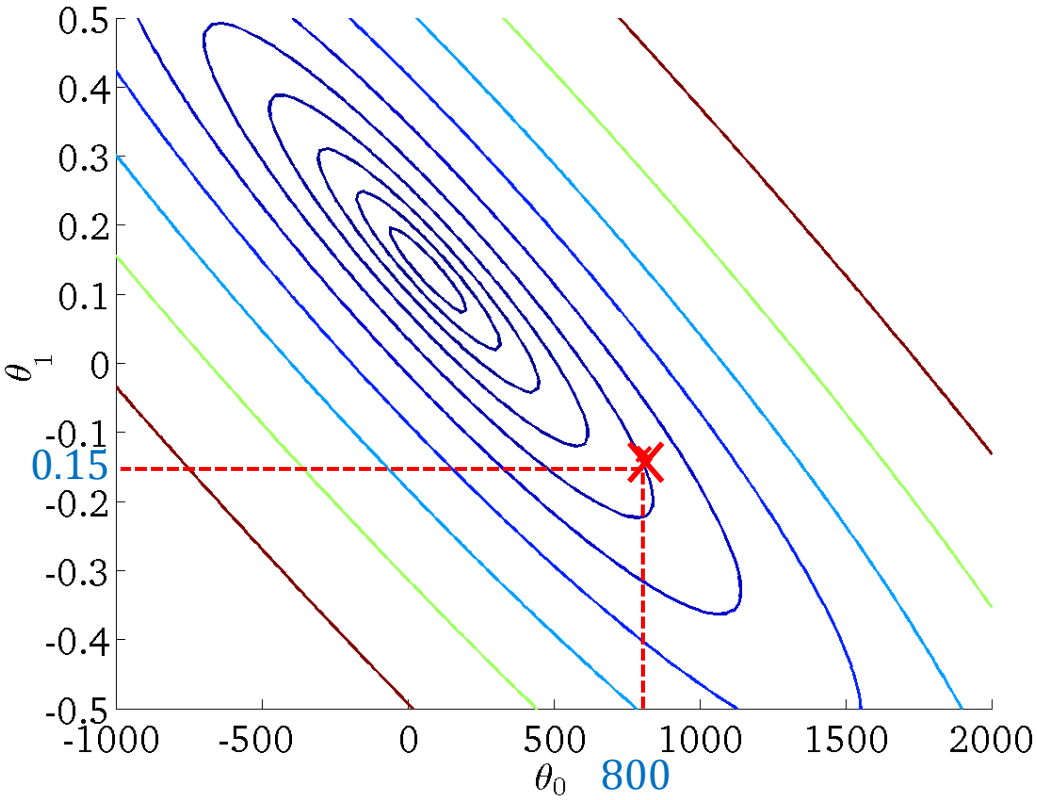
$$h_\theta(x) = 50 + 0.06x$$



Hình dạng hàm chi phí

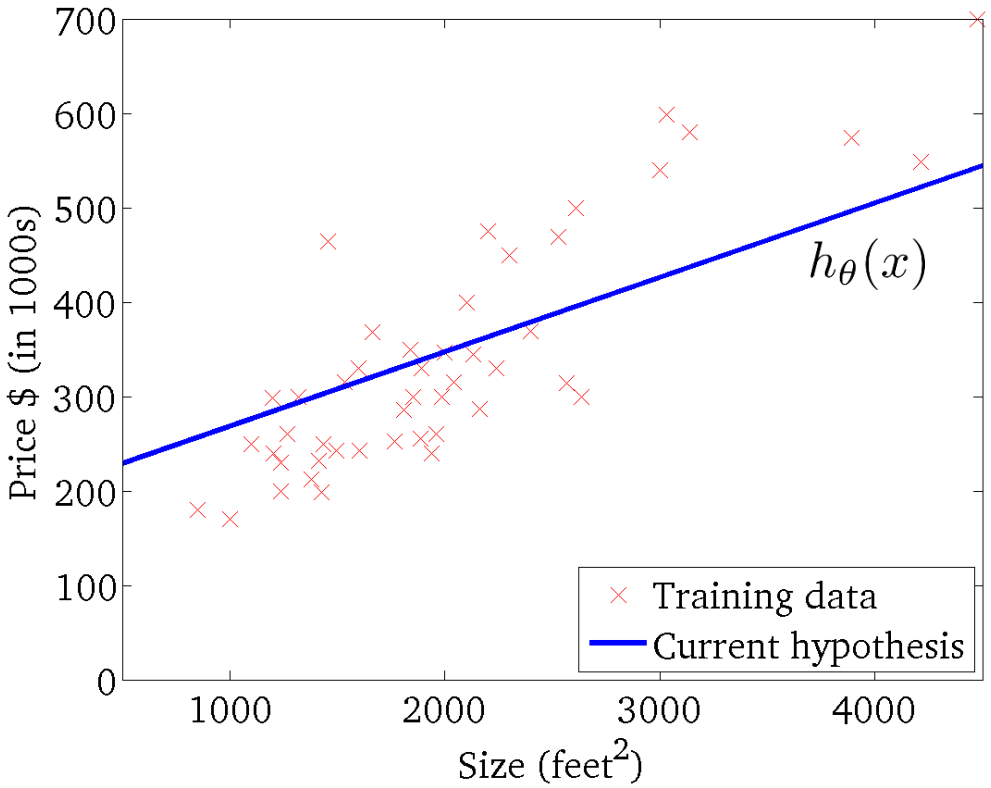


Hàm chi phí J_θ của hàm hồi quy h_θ khi chiếu lên θ_0 và θ_1

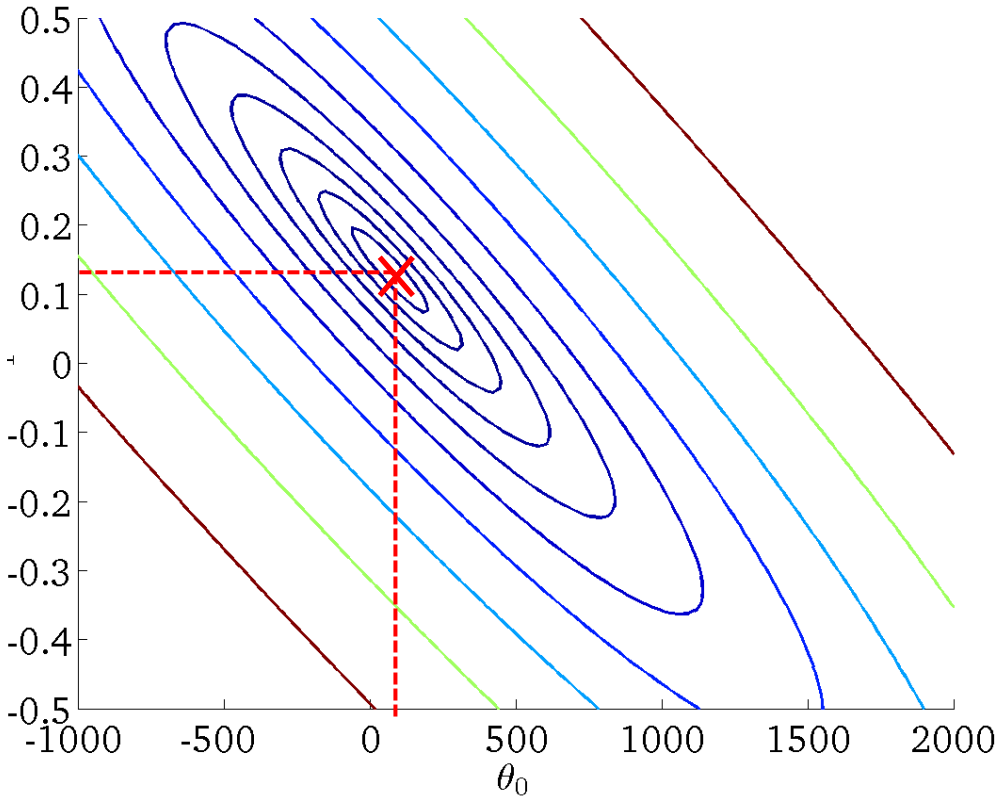


Mỗi vòng elip hay mỗi màu đại diện cho cùng giá trị hàm chi phí J_θ nhưng mỗi vị trí khác nhau thể hiện các θ_0, θ_1 khác nhau (contour figures/plots)

Hình dạng hàm chi phí



Hàm chi phí J_θ của hàm hồi quy h_θ khi chiếu lên θ_0 và θ_1

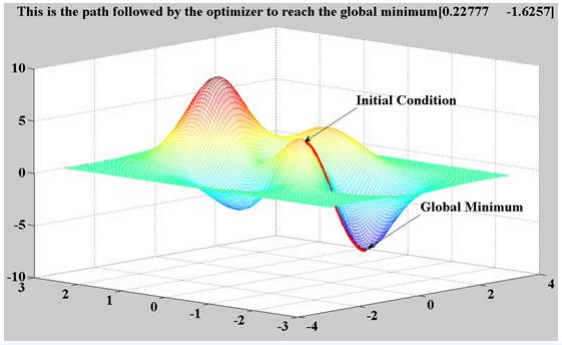


$$\boldsymbol{minimize_{\theta_0,\theta_1} J(\theta)}$$

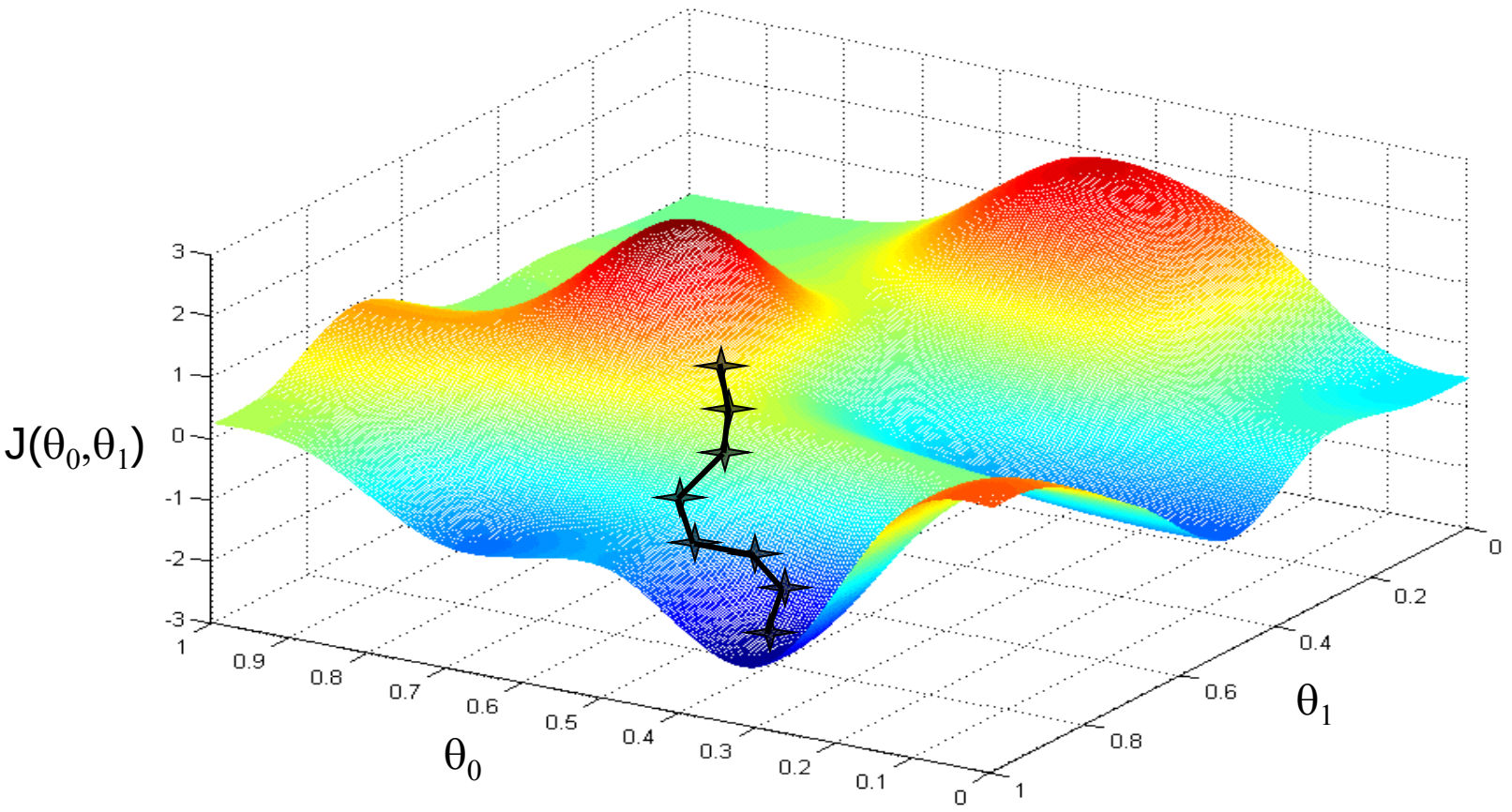
Phương pháp thử và sai các giá trị θ_0, θ_1 cho đến khi đạt $minimize_{\theta_0,\theta_1} J(\theta)$ liệu có hiệu quả?

Gradient Descent

- Có nhiều hàm $J(\theta_0, \theta_1)$ nhưng mục tiêu là tìm hàm $\text{minimum}\, J(\theta_0, \theta_1)$
- *Phương pháp gradient descent:*
  - Bắt đầu với bất kì giá trị nào của θ_0, θ_1 (thường chọn = 0).
  - Thay đổi θ_0, θ_1 để giảm $J(\theta_0, \theta_1)$ cho đến khi đạt được giá trị tối thiểu.
  - Mỗi lần thay đổi tham số θ_0, θ_1, *chọn gradient* (đạo hàm) mà *giảm $J(\theta_0, \theta_1)$ nhiều nhất* có thể.

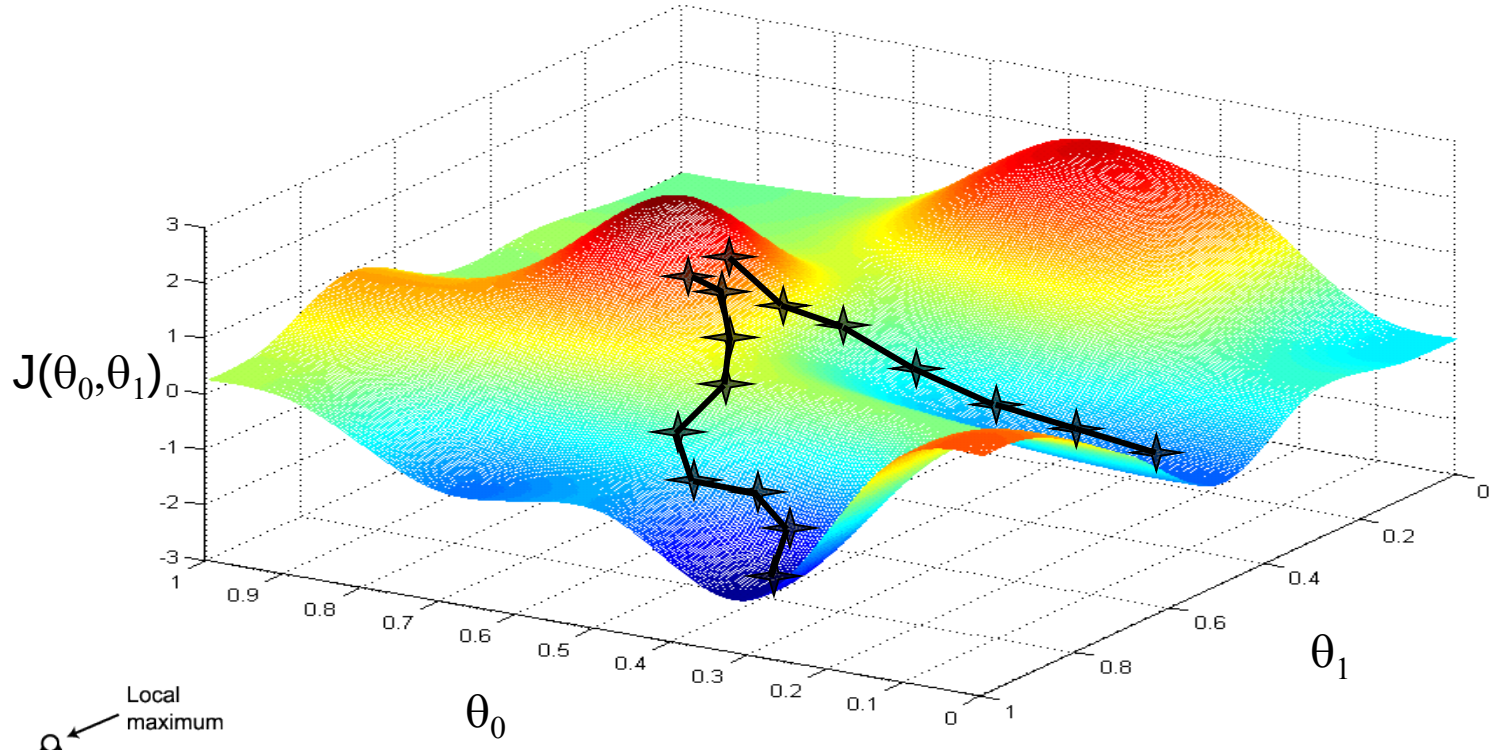


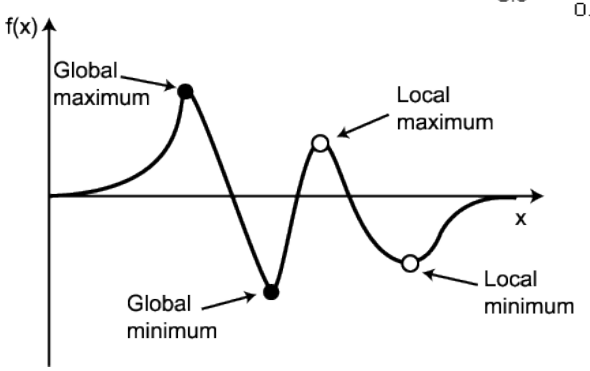
Minh họa Gradient Descent



Nhận xét Gradient Descent

- Điểm bắt đầu ở đâu quyết định giá trị nhỏ nhất đạt được (cực tiểu địa phương).





Thuật toán gradient descent

repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad (\text{for } j = 0 \text{ and } j = 1)$$

}

- Quá trình cập nhật các θ_j phải đồng thời:

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$
$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$
$$\theta_0 := \text{temp0}$$
$$\theta_1 := \text{temp1}$$

- Cập nhật không đúng:

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$
$$\theta_0 := \text{temp0}$$
$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$
$$\theta_1 := \text{temp1}$$

Thuật toán gradient descent (tt)

repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad (\text{for } j = 0 \text{ and } j = 1)$$

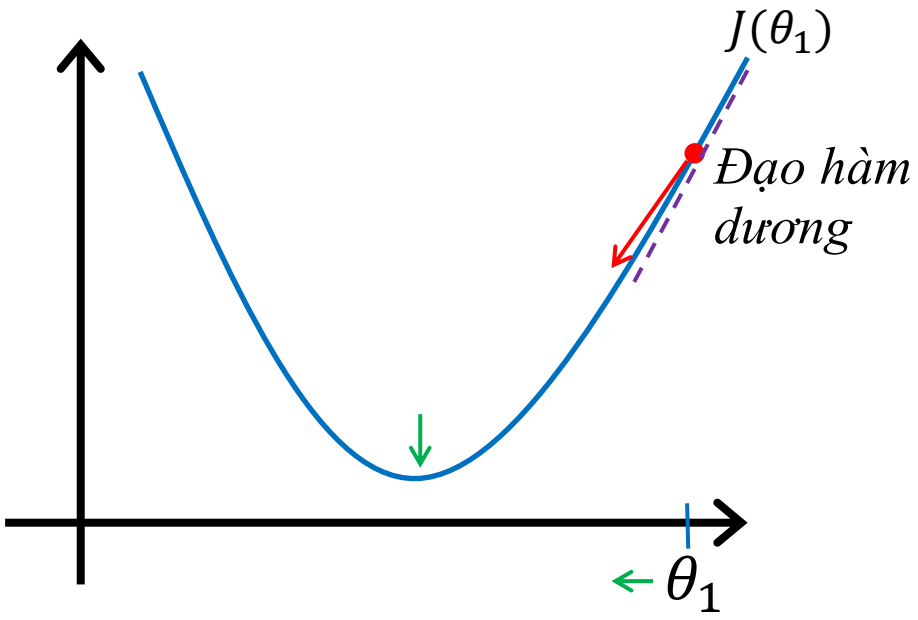
}

α: gọi là tỉ lệ học (>0)

$\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$: đạo hàm từng phần ứng với θ_0, θ_1. Ý nghĩa của đạo hàm:

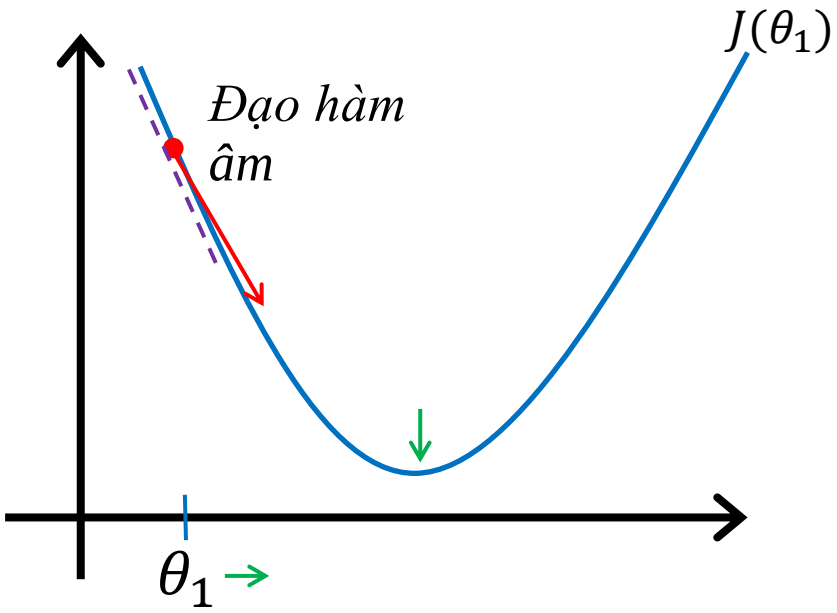
- Là *tiếp tuyến* tại điểm trên đường thẳng, nói lên *xu hướng thay đổi* của điểm dữ liệu.
- Di chuyển hướng xuống sẽ là *đạo hàm âm*, vì vậy sẽ cập nhật J(θ_j) đến giá trị nhỏ hơn. Và ngược lại.

Đạo hàm từng phần



$$\theta_1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$

$$\theta_1 = \theta_1 - \alpha \times (số\ dương)$$



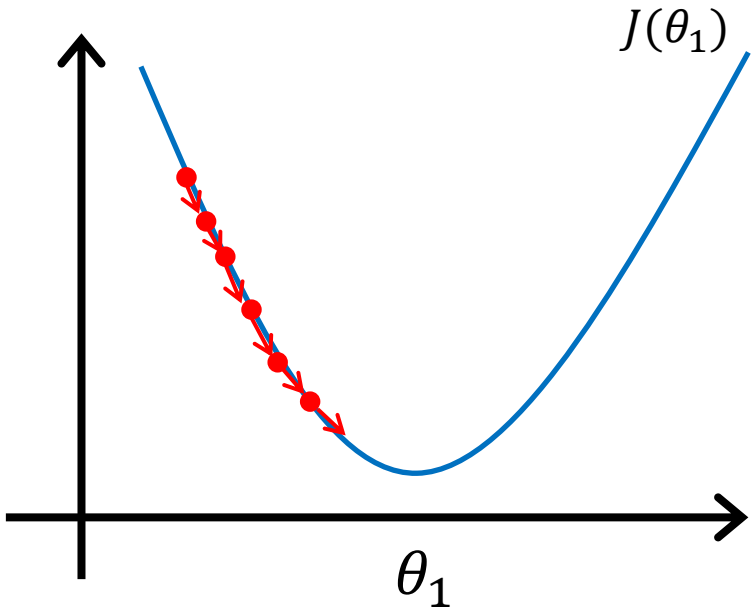
$$\theta_1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$

$$\theta_1 = \theta_1 - \alpha \times (số\ âm)$$

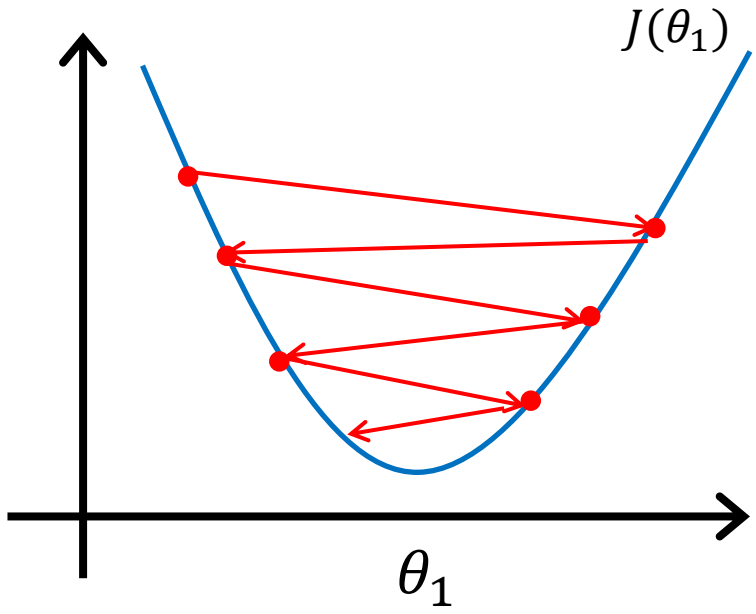
Hệ số học α

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_1)$$

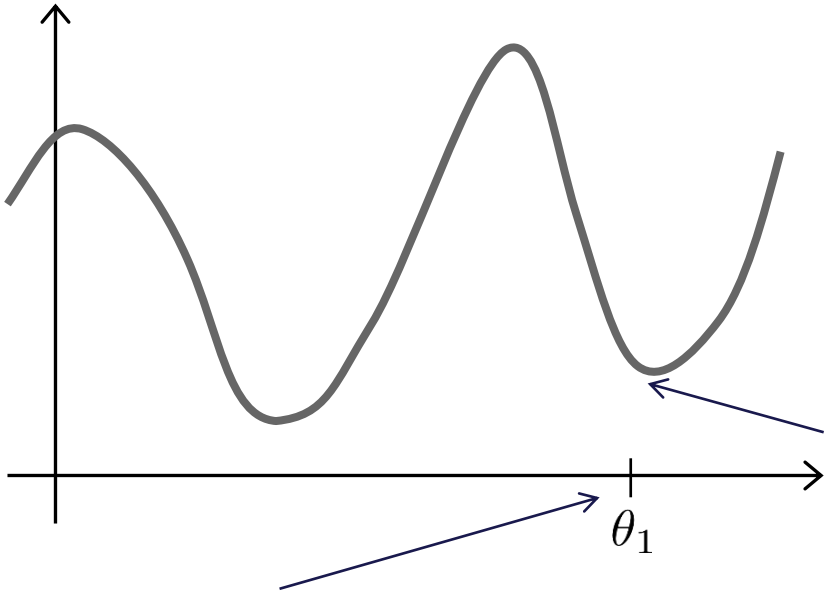
- Nếu hệ số học *α quá nhỏ*, gradient descent sẽ lấy các bước *thay đổi nhỏ*, dẫn đến *chậm hội tụ*.



- Nếu hệ số học *α quá lớn*, gradient descent sẽ có thể *nhảy vượt qua điểm cực tiểu*. Nó có thể dẫn đến *không hội tụ*, thậm chí còn làm *xấu đi*.



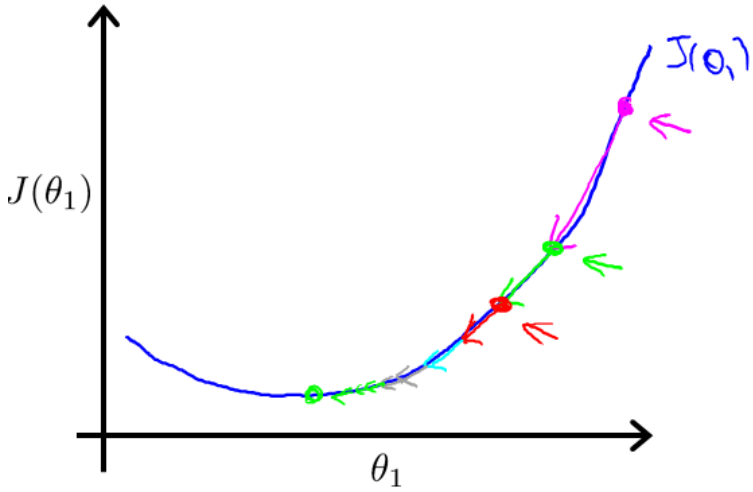
Cực tiểu địa phương và toàn cục



- Gradient descent có thể *hội tụ tại cực tiểu địa phương* thậm chí với hệ số học α cố định.

$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

- Khi càng *gần một cực tiểu*, gradient descent sẽ *tự động có bước nhảy nhỏ hơn* nên ta không cần thay đổi hệ số học α theo thời gian.



Hồi quy tuyến tính với gradient descent

- Đạo hàm từng phần cho hàm chi phí:

$$\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) = \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^i) - y^i \right)^2$$

$$= \frac{\partial}{\partial \theta_j} \frac{1}{2m} \sum_{i=1}^{m} \left(\underline{\theta_0 + \theta_1 x^i} - y^i \right)^2$$

- Với j=0: $\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^i) - y^i \right)$
- Với j=1: $\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^i) - y^i \right) . x^i$

Thuật toán gradient descent

Cách tính đạo hàm từng phần của hàm chi phí

$$
\begin{aligned}
\frac{\partial}{\partial \theta_j} J(\theta) &= \frac{\partial}{\partial \theta_j} \frac{1}{2} (h_\theta(x) - y)^2 \\
&= 2 \cdot \frac{1}{2} (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} (h_\theta(x) - y) \\
&= (h_\theta(x) - y) \cdot \frac{\partial}{\partial \theta_j} \left(\sum_{i=0}^{n} \theta_i x_i - y \right) \\
&= (h_\theta(x) - y)\, x_j
\end{aligned}
$$

repeat until convergence {

$$
\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)
$$

$$
\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) \cdot x^{(i)}
$$

}

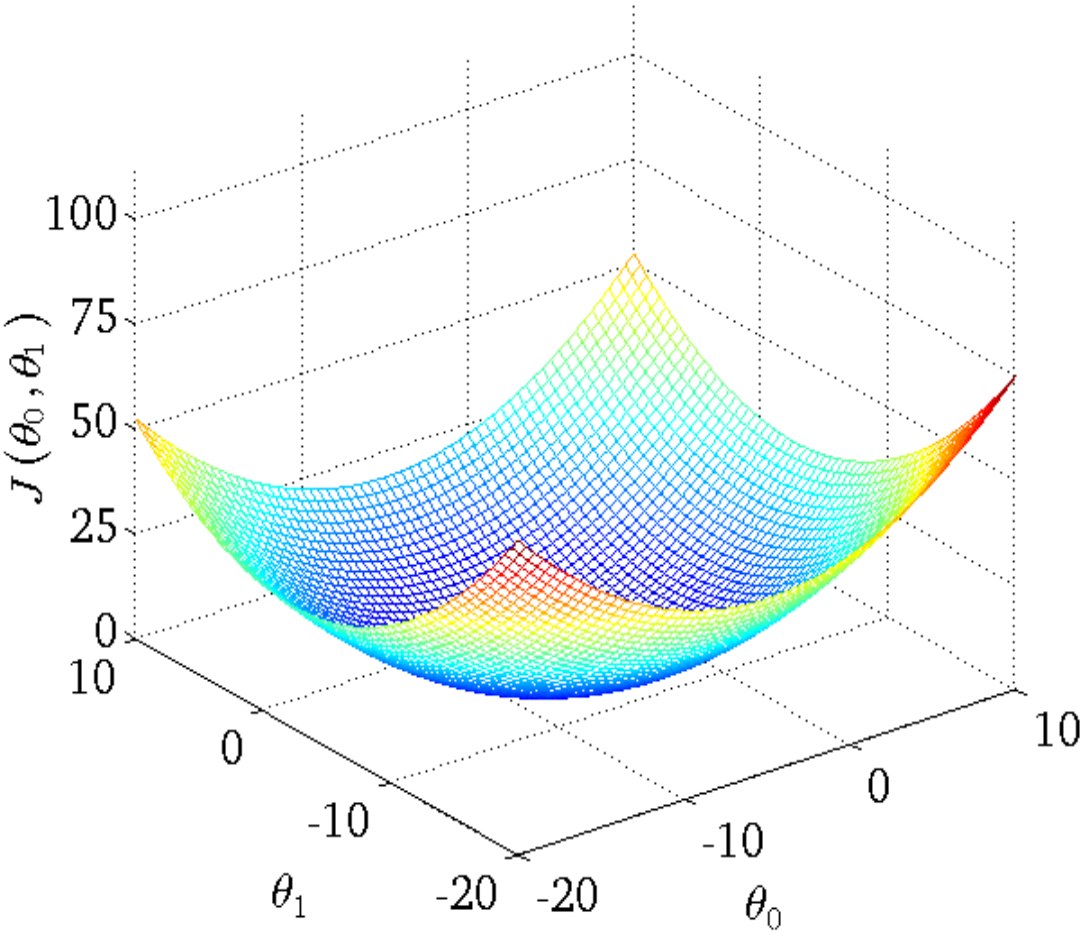
Cập nhật θ_0 và θ_1 một cách đồng thời

Đây gọi là luật cập nhật LMS (least mean squares)

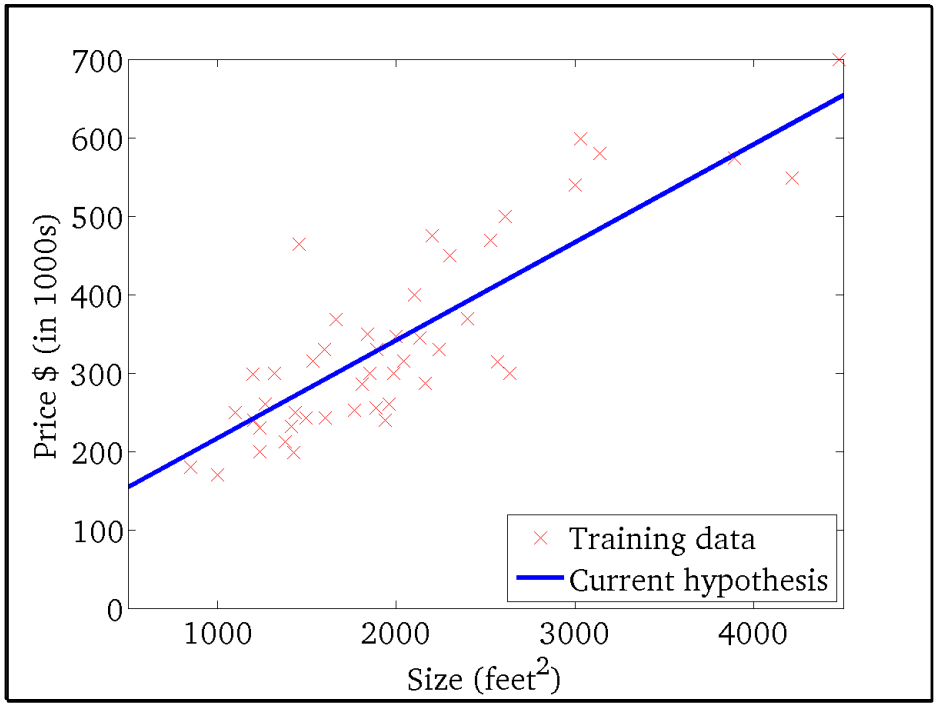
Hàm chi phí hồi quy tuyến tính

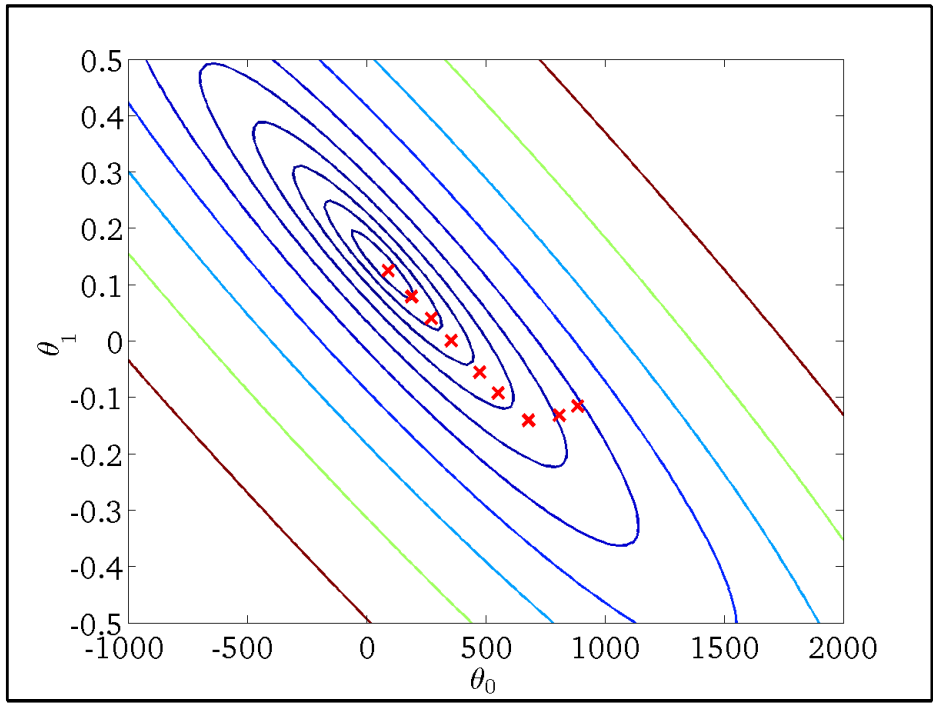
Hàm chi phí hồi quy tuyến tính có gặp phải vấn đề cực tiểu địa phương không? Cách giải quyết?

- Hàm chi phí hồi quy tuyến tính luôn là một *hàm lồi* (convex function), nghĩa là luôn có *một cực tiểu duy nhất*.
  - Dạng hình giống cái "tô".
  - Chỉ có một tối ưu toàn cục, nên gradient descent sẽ luôn hội tụ tại điểm tối ưu toàn cục.



Gradient descent HQTT





Các loại gradient descent (1/2)

repeat until convergence {

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) \cdot x^{(i)}$$

}

- Thuật toán gradient descent ở trên được gọi là *"batch" gradient descent*:
  - "batch" (hàng loạt): mỗi bước cập nhật của gradient descent sử dụng tất cả mẫu huấn luyện $\sum_{i=1}^{m}$.

- Ngoài ra, còn có *"stochastic" gradient descent* ("incremental" gradient descent):
  - Các tham số được cập nhật theo gradient descent liên quan đến chỉ một mẫu huấn luyện

Loop {

for i=1 to m, {

$$\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)}) \right) x_j^{(i)} \qquad \text{(for every } j\text{)}.$$

}

}

Các loại gradient descent (2/2)

Batch Gradient Descent	Stochastic Gradient Descent
- Đợi *có hết dữ liệu* rồi mới cập nhật các tham số.	- Có thể bắt đầu tiến trình ngay khi *có một dữ liệu*.
- Stochastic có tham số θ đạt *gần đến cực tiểu nhanh hơn* so với Batch. - Stochastic *hiếm khi hội tụ đến cực tiểu* và tham số θ sẽ làm cho hàm chi phí dao động xung quanh cực tiểu. Tuy nhiên thực tế, gần đạt đến giá trị cực tiểu cũng đã đủ tốt. - Vì vậy, đối với *tập dữ liệu lớn*, người ta thường áp dụng *phương pháp stochastic* nhiều hơn so với batch.	

Nội dung

- Hồi quy tuyến tính
- Hồi quy tuyến tính với một biến
- **Hồi quy tuyến tính với nhiều biến**
  - Đa đặc trưng
  - Hồi quy nhiều biến
  - Gradient Descent cho nhiều biến
- Hồi quy đa thức
- Biểu thức chuẩn

Đa biến

- Đa biến ≡ đa đặc trưng (multiple feature)
- Hàm hồi quy tuyến tính đa biến:

Size (feet²)	Price ($1000)
2104	460
1416	232
1534	315
852	178
…	…

Size (feet²)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

$$h_\theta(x) = \theta_0 + \theta_1 x$$

(*x là kích thước nhà*)

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = ?$$

(**x** *là kích thước nhà, số phòng, số tầng, tuổi nhà*)

Một số kí hiệu

Size (feet²) x_1	Number of bedrooms x_2	Number of floors x_3	Age of home (years) x_4	Price ($1000) y
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

$m = 47$

$n = 4$

- Kí hiệu:
  - n: số đặc trưng.
  - x^i: input của mẫu huấn luyện thứ *i*.
  - x_j^i: giá trị của đặc trưng *j* trong mẫu huấn luyện thứ *i*.

Hàm hồi quy tuyến tính đa biến

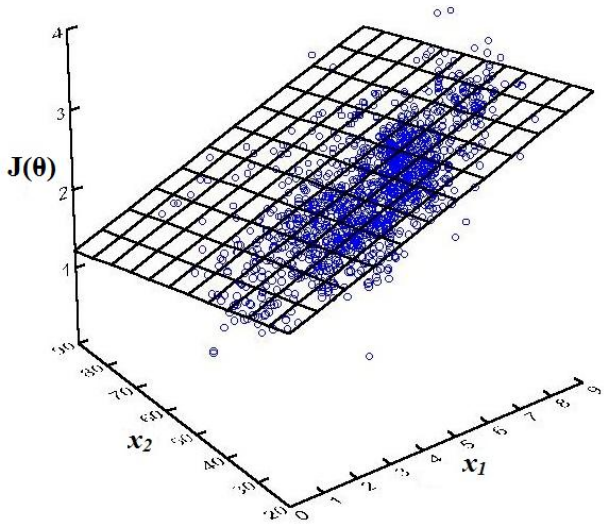
- Hàm hồi quy tuyến tính đa biến (multivariate linear regression):

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ \ldots + \theta_n x_n$$

  – Ví dụ:

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = 80 + 0.1x_1 + 0.01x_2 + 3x_3 - 2x_4$$

Size (feet²) x_1	Number of bedrooms x_2	Number of floors x_3	Age of home (years) x_4	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…



Hàm hồi quy tuyến tính đa biến

- Đặt $x_0 = 1$ và:

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} \in R^{n+1}, \mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \in R^{n+1}$$

Vector tham số (parameter vector)

Vector đặc trưng (feature vector)

- Ta có thể viết:

$$\overbrace{[\theta_0 \quad \theta_1 \quad \theta_2 \ \ldots \ \theta_n]}^{\boldsymbol{\theta}^T} \times \overbrace{\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}^{\mathbf{x}} \Rightarrow h_{\boldsymbol{\theta}}(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n = \boldsymbol{\theta}^T \mathbf{x}$$

Gradient descent

- Để xác định *các tham số* cho hồi quy tuyến tính nhiều biến, ta cũng dựa trên hàm chi phí và gradient descent.
- *Hàm chi phí:*

$$J(\theta_0, \theta_1, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

- *Gradient descent:*

Repeat $\{$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \ldots, \theta_n)$$

$\}$ (simultaneously update for every $j = 0, \ldots, n$)

Đạo hàm từng phần

Repeat $\{$

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

$\}$

(simultaneously update for every $j = 0, \ldots, n$)

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_0^{(i)}$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_1^{(i)}$$

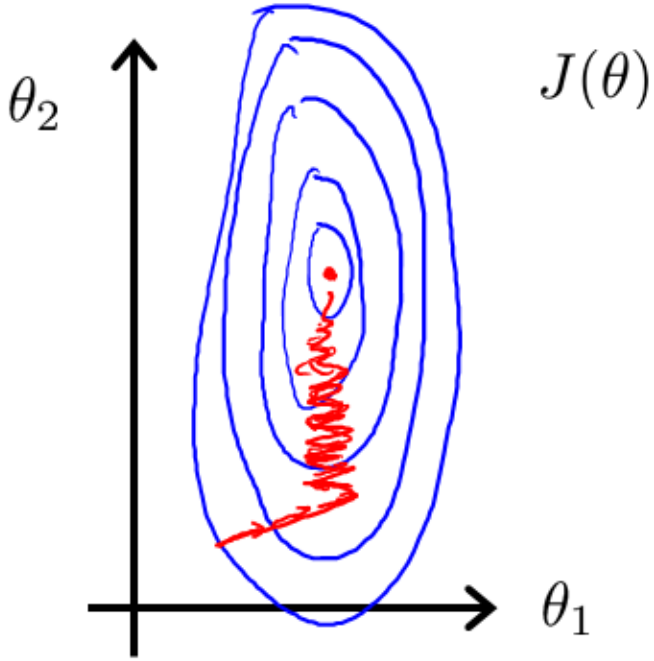
$$\theta_2 := \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_2^{(i)}$$

$\ldots$

Hội tụ của gradient descent

- Các đặc trưng có miền giá trị *chênh lệch nhau* nhiều có thể dẫn đến *chậm hội tụ* đến tối ưu toàn cục.
- Ví dụ: $x_1 = size\ (0 - 2000 feet^2)$

$$x_2 = number\ of\ bedrooms\ (1 - 5)$$

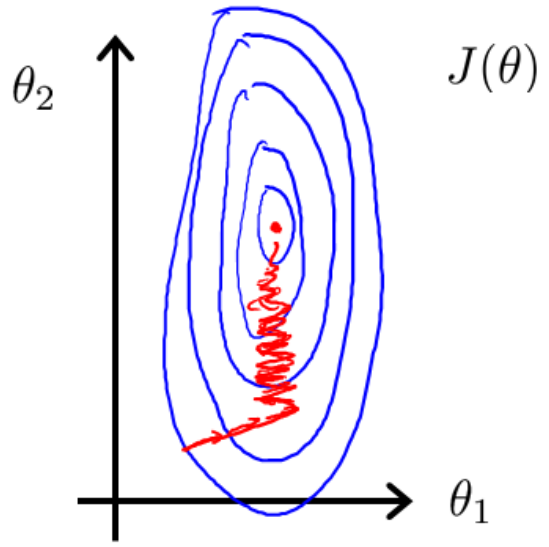


Hội tụ của gradient descent (tt)

- Cần đảm bảo các đặc trưng có cùng tỉ lệ tương tự nhau.
  - Hội tụ của gradient descent trung bình sẽ nhanh hơn.

E.g. x_1 = size (0-2000 feet^2)

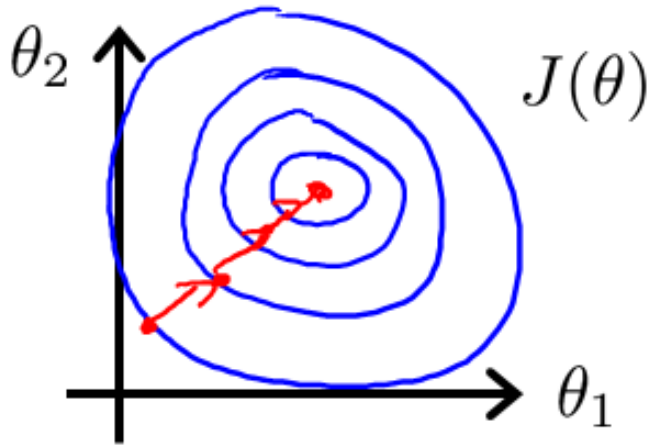
x_2 = number of bedrooms (1-5)



$$x_1 = \frac{size(feet^2)}{2000}$$

$$x_1 = \frac{number\ of\ bedrooms}{5}$$

$0 \le x_1 \le 1$ $\quad$ $0 \le x_2 \le 1$



Scale đặc trưng

- Chuẩn hóa trung bình (mean normalization):
  - Thay thế x_i bằng: $\frac{x_i-\mu}{max}$, $\frac{x_i-\mu}{max-min}$, $\frac{x_i-min}{max-min}$, $\frac{x_i-\mu}{\sigma}$ (ngoài trừ x_0)
  - Ví dụ:

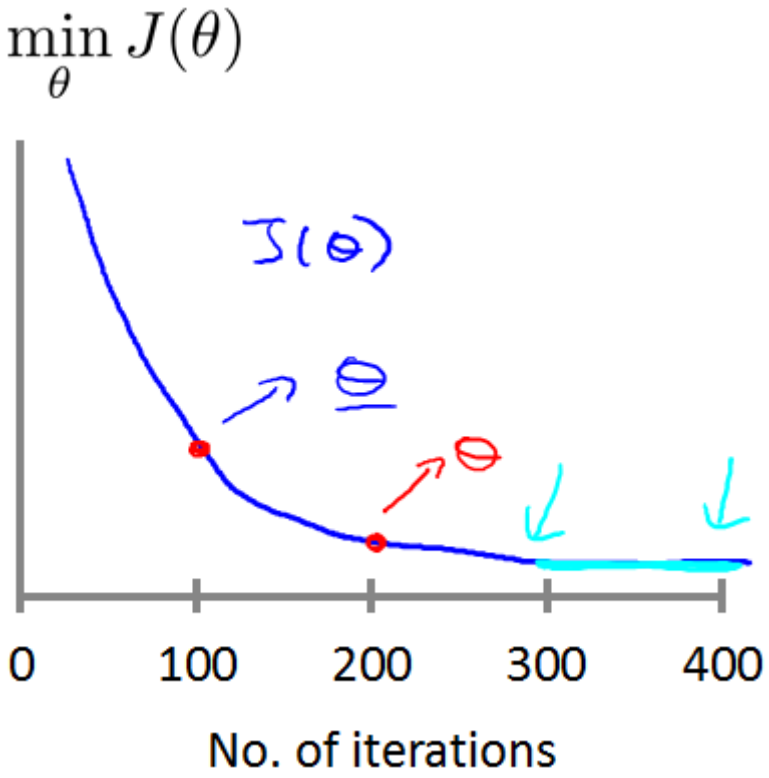
$$x_1 = \frac{size-1000}{2000}$$

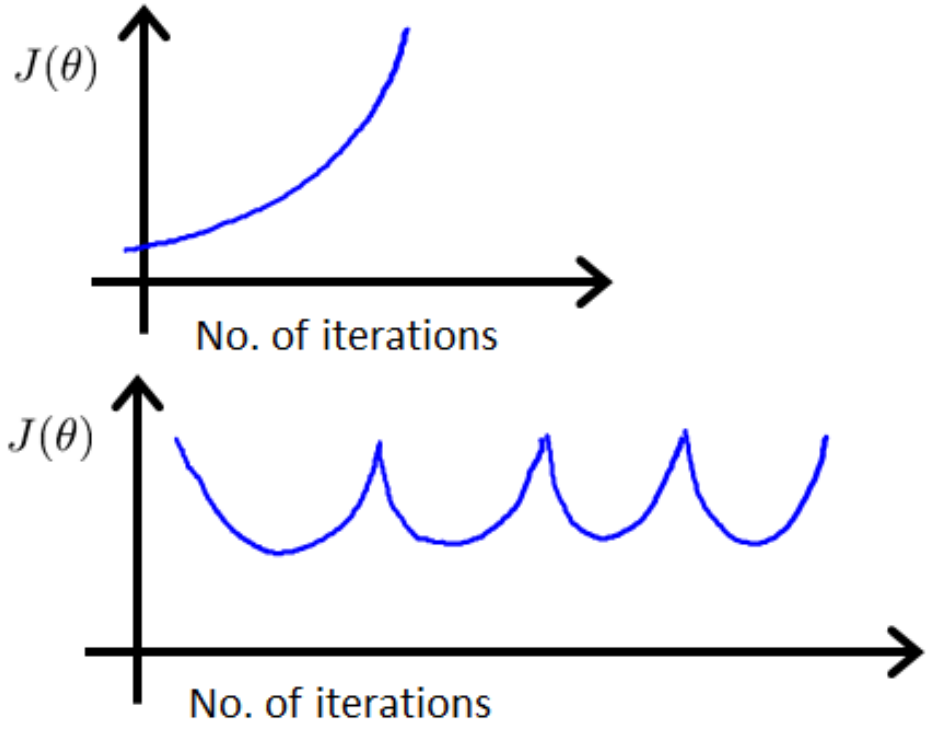
$$x_2 = \frac{\#bedrooms-2}{5}$$

$$-0.5 \leq x_1 \leq 0.5, -0.5 \leq x_2 \leq 0.5$$

Vấn đề của gradient descent

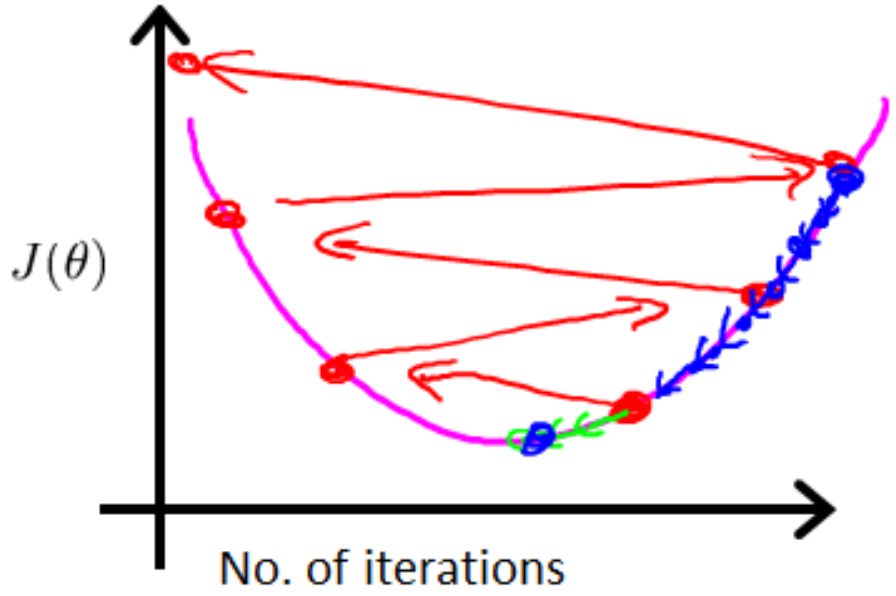
- Liệu gradient descent có giảm sau mỗi lần lặp?
- Lặp bao nhiêu lần thì đủ?
- Như thế nào để chọn hệ số học α?





Giải quyết

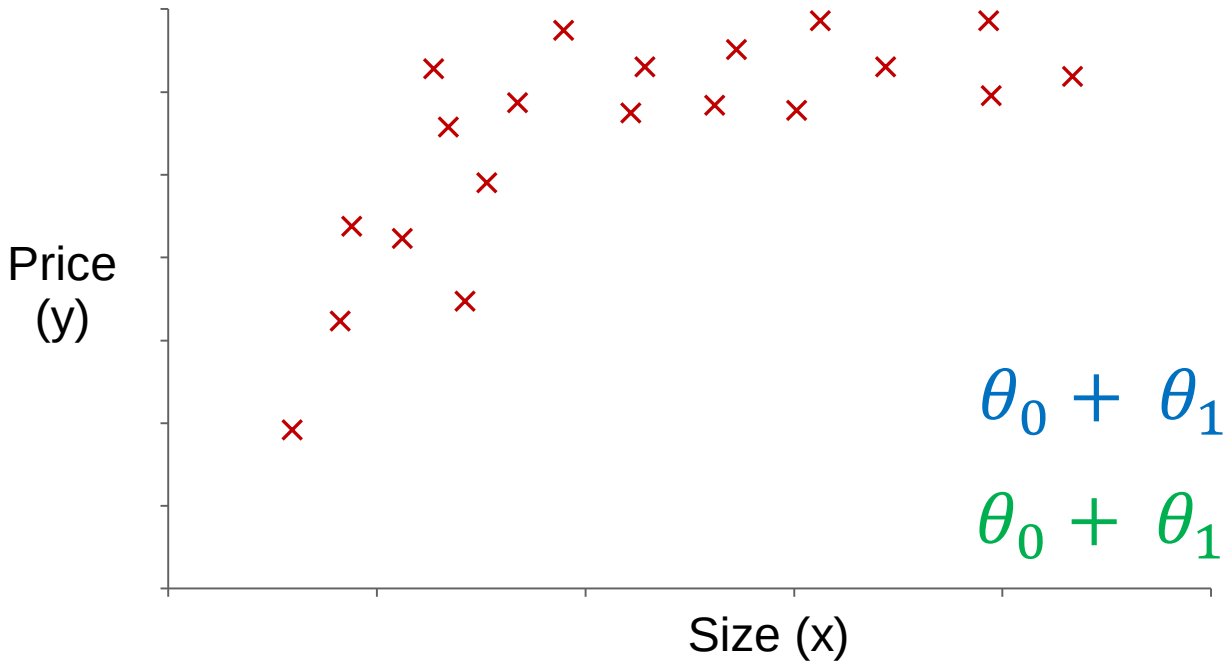
- Đặt ra một ngưỡng hội tụ ε, nếu $J(\theta)$ nhỏ hơn ngưỡng này thì dừng.
- Sử dụng hệ số học α đủ nhỏ. Tuy nhiên:
  - Nếu α quá nhỏ: chậm hội tụ
  - Nếu α quá lớn: $J(\theta)$ có thể không giảm; thậm chí không hội tụ; hoặc có thể rất chậm hội tụ.



Nội dung

- Hồi quy tuyến tính
- Hồi quy tuyến tính với một biến
- Hồi quy tuyến tính với nhiều biến
- **Hồi quy đa thức**
- Biểu thức chuẩn

Hồi quy đa thức



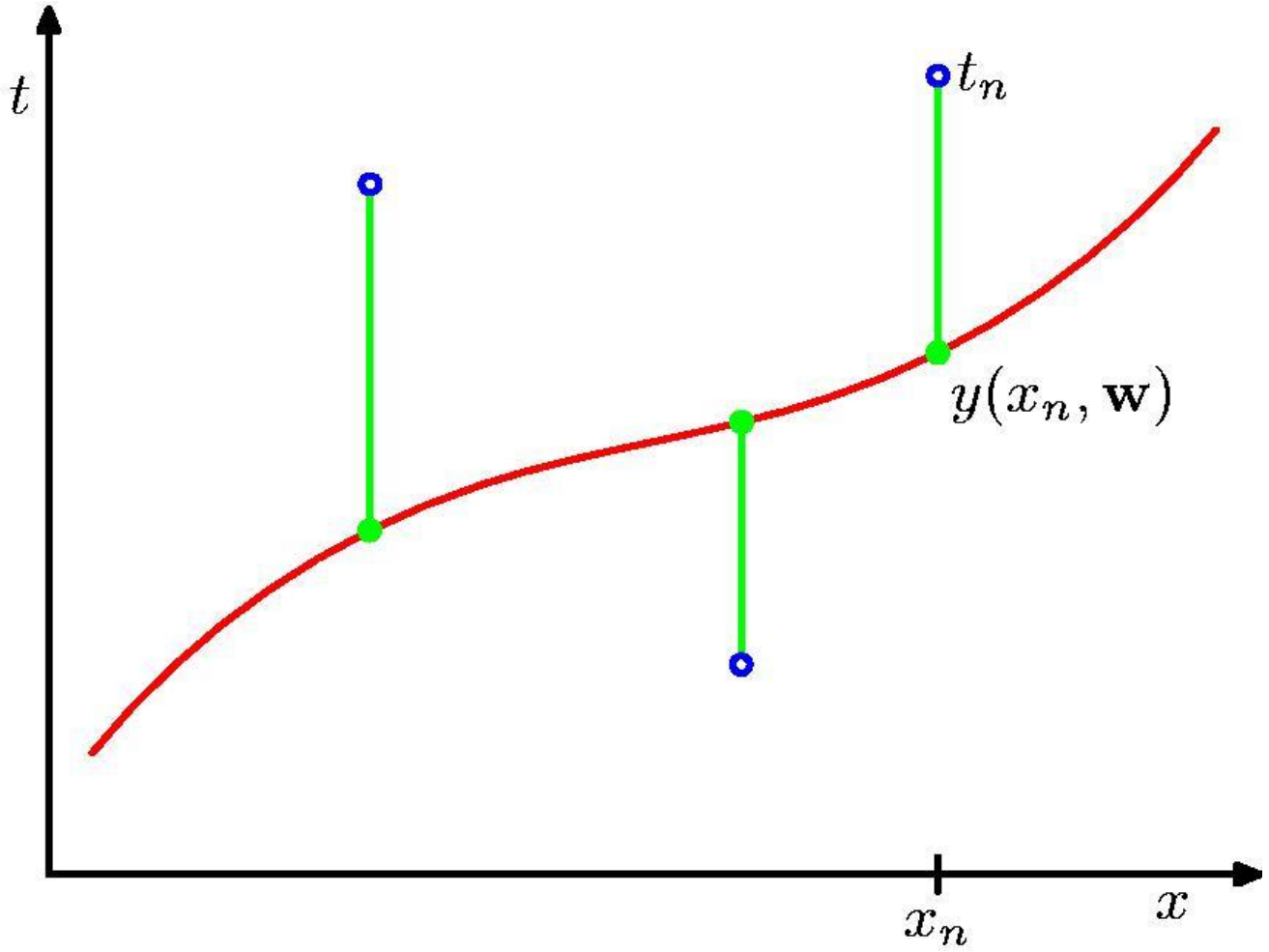
$$\theta_0 + \theta_1 x + \theta_2 x^2$$

$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3$$

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$
$$= \theta_0 + \theta_1 (size) + \theta_2 (size)^2 + \theta_3 (size)^3$$

$$x_1 = (size)$$
$$x_2 = (size)^2$$
$$x_3 = (size)^3$$

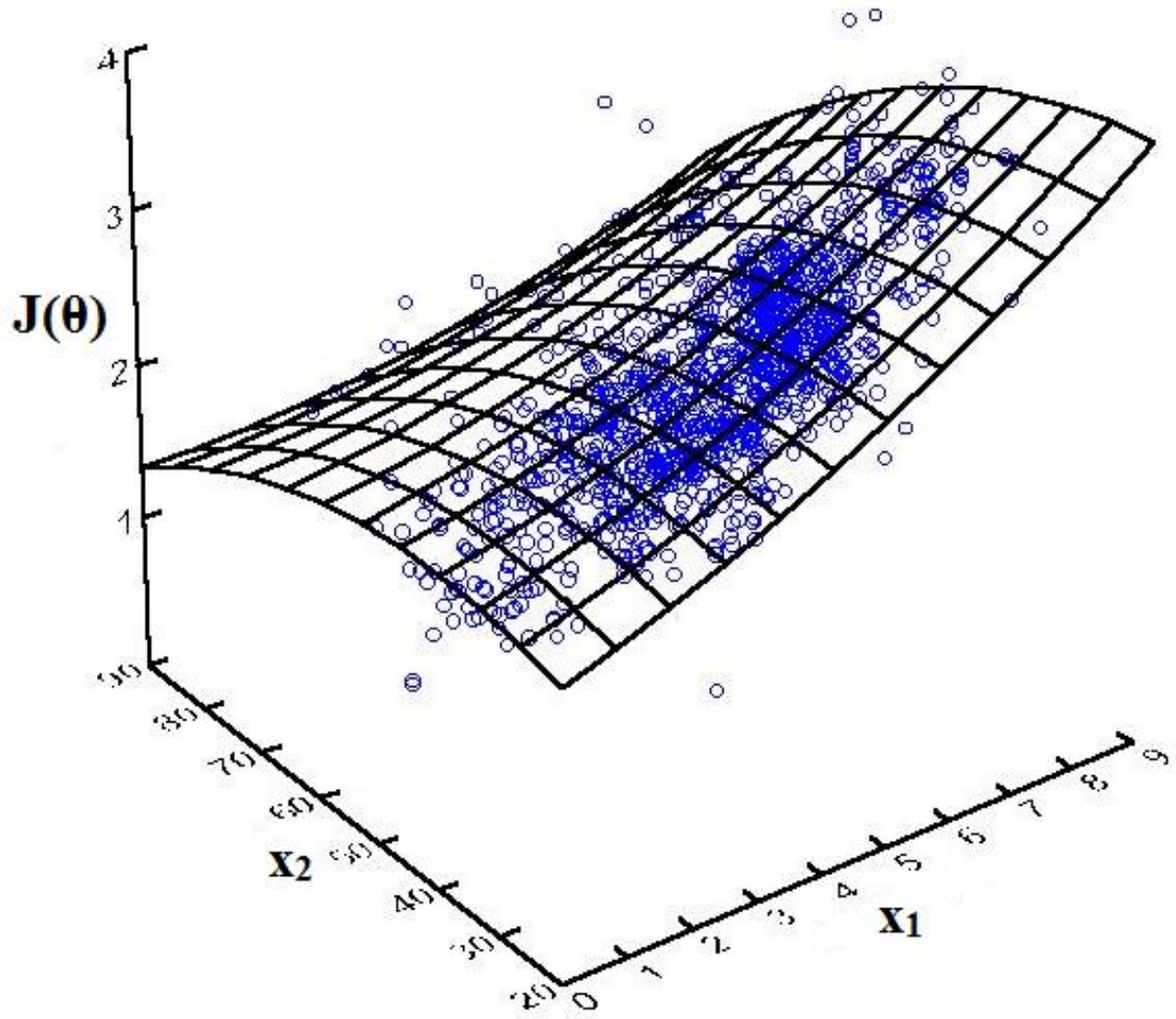
Hàm chi phí



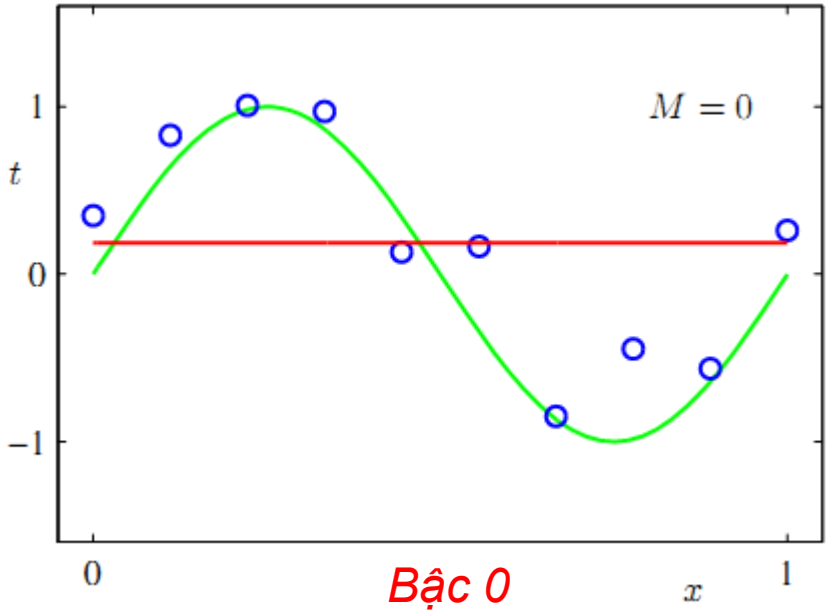
$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2$$

Hàm chi phí (tt)

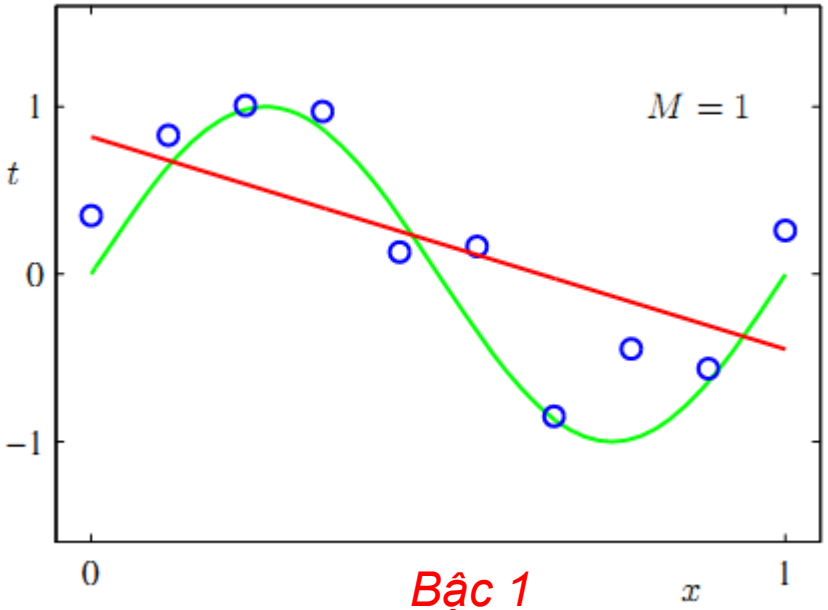
- Hàm chi phí cho đa thức nhiều biến



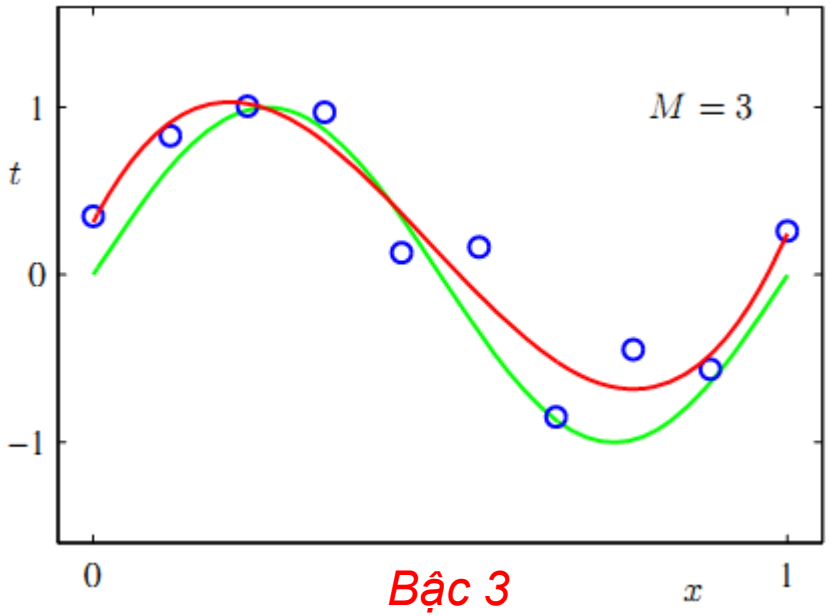
Bậc đa thức



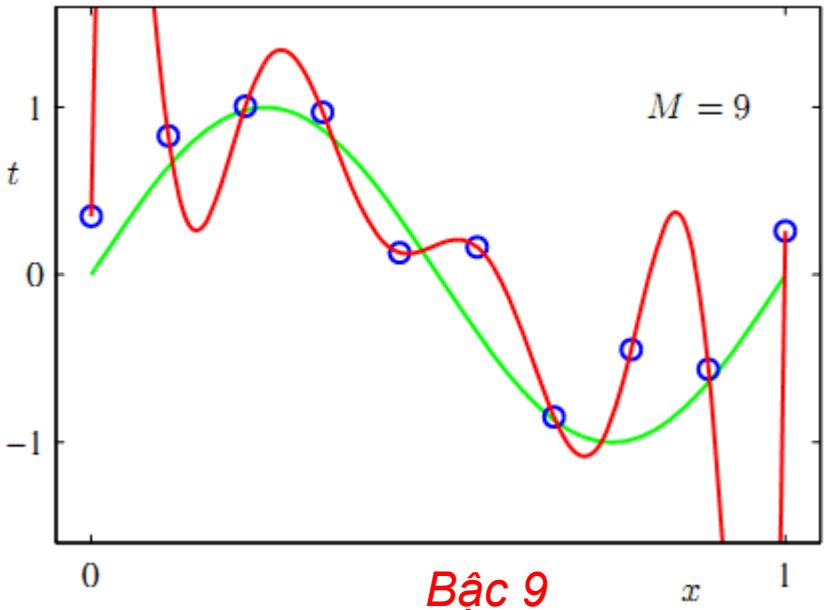
Bậc 0



Bậc 1

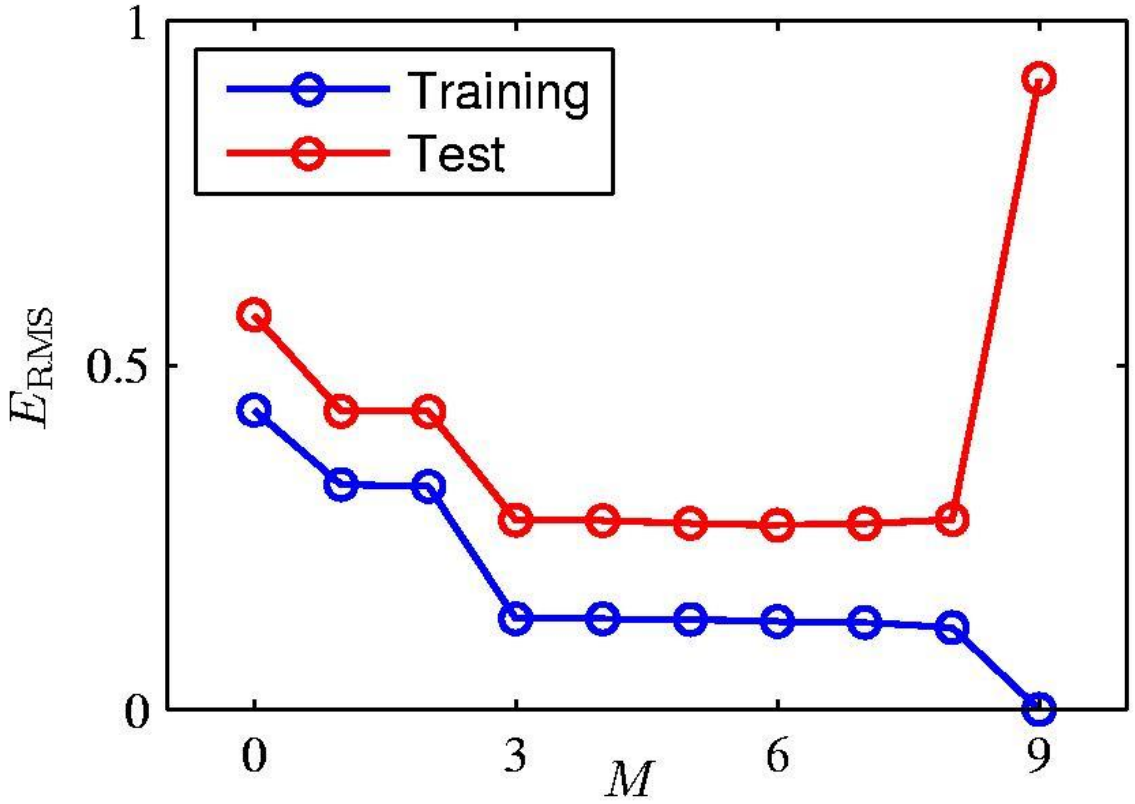


Bậc 3



Bậc 9

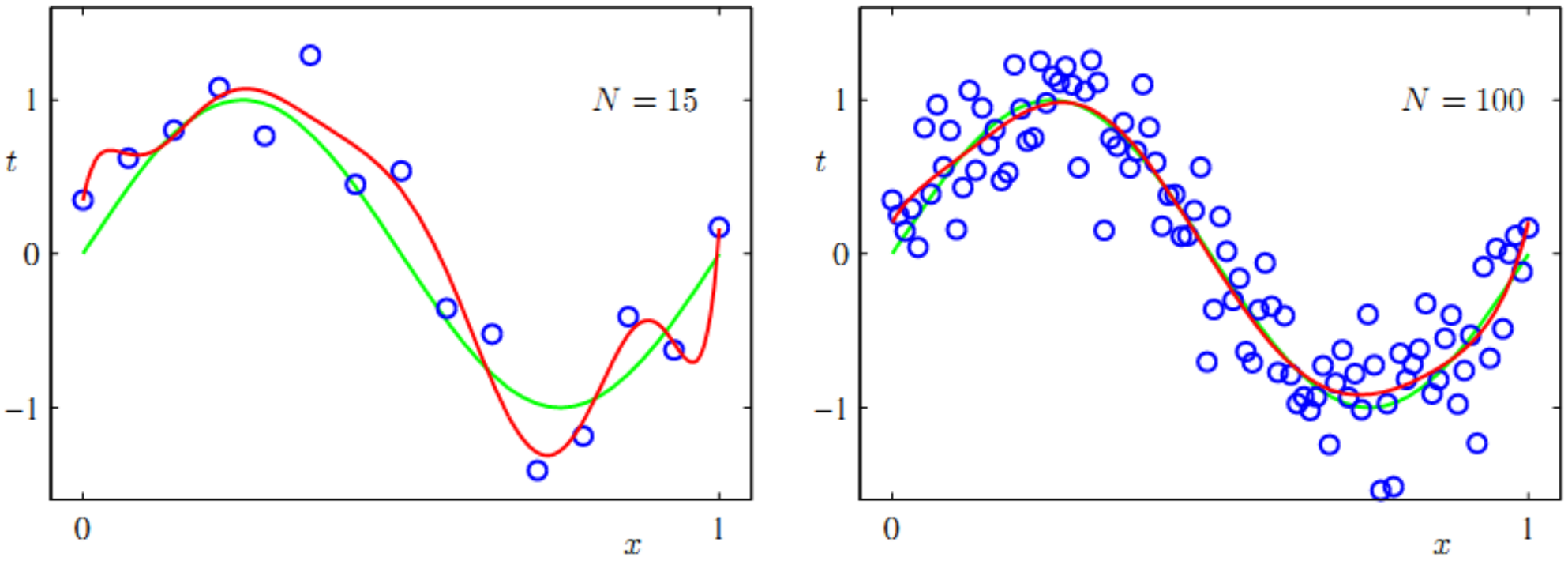
Vấn đề Over-fitting



Root-Mean-Square (RMS) Error: $E_{\mathrm{RMS}} = \sqrt{2E(\mathbf{w}^\star)/N}$

Một cách để khắc phục over-fitting là sử dụng phương pháp *regularization*

Bậc đa thức với số lượng mẫu



Đa thức bậc 9 (màu đỏ) với số lượng mẫu khác nhau (N=15 và N=100)

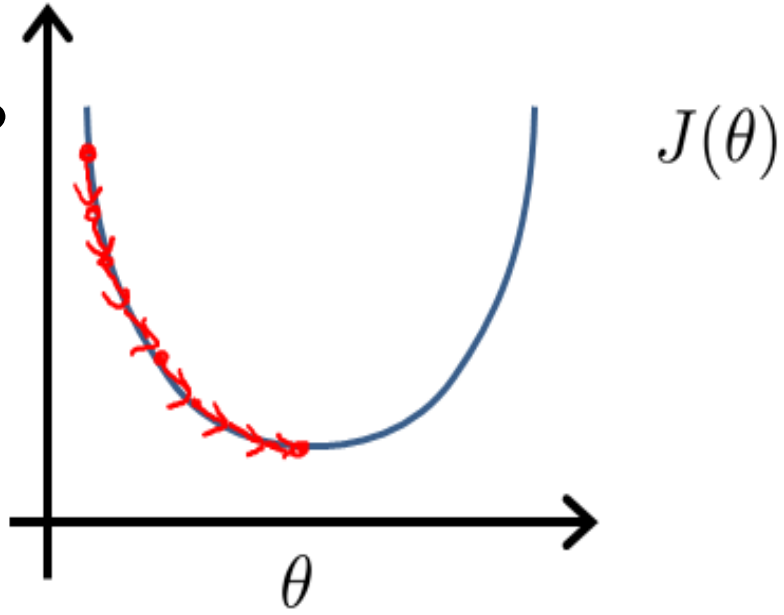
Một heuristic: số lượng điểm dữ liệu không nên *nhỏ hơn 5 hay 10 lần* số lượng tham số trong mô hình đa thức → mô hình càng phức tạp khi dữ liệu tăng → không hiệu quả

Nội dung

- Hồi quy tuyến tính
- Hồi quy tuyến tính với một biến
- Hồi quy tuyến tính với nhiều biến
- Hồi quy đa thức
- **Biểu thức chuẩn**
  - Khái niệm
  - Đạo hàm ma trận
  - Toán tử “trace”
  - Cực tiểu hàm chi phí

Một số vấn đề gradient descent

- **Gradient descent:**
  - Vector $\boldsymbol{\theta}$ *khởi tạo là bao nhiêu?*
  - *Hệ số học* α*?*
  - *Bao nhiêu vòng lặp?*
  - *Khi nào hội tụ?*
  - *Có hội tụ không?*
  - *Ngưỡng* ε *nên là bao nhiêu? …*



- Từ các vấn đề trên, ***biểu thức chuẩn*** (normal equation) cung cấp một giải pháp tốt hơn.
  - Phương pháp giải quyết $\boldsymbol{\theta}$ dựa trên phân tích.
  - Cũng có những thuận lợi và bất lợi riêng.

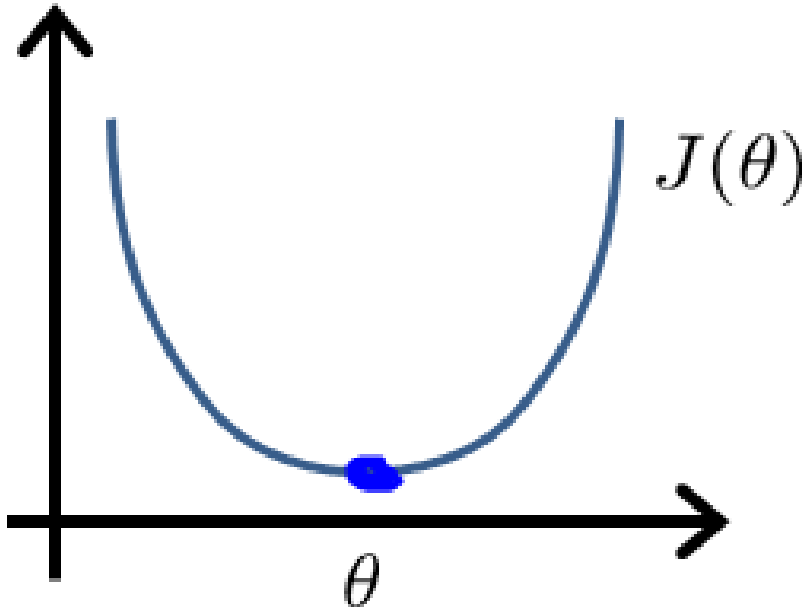
Cực trị hàm số

- Xem xét hàm chi phí đơn giản sau:

$$J(\theta) = a\theta^2 + b\theta + c \quad (với\ \theta\ là\ số\ thực)$$

- Tìm θ để minimize $J(\theta)$

$$\frac{\partial}{\partial\theta}J(\theta) = 0$$



Cực trị hàm số

- Tổng quát:

$$\theta \in \mathbb{R}^{n+1} \qquad J(\theta_0, \theta_1, \ldots, \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

$$\frac{\partial}{\partial \theta_j} J(\theta) = \cdots = 0 \quad \text{(for every } j \text{)}$$

  - Lấy đạo hàm từng phần và cho bằng 0 để tìm điểm cực trị.
  - Từ đó rút ra được $\boldsymbol{\theta}$ làm minimize $J(\theta)$
- Việc tính đạo hàm trên toàn tập mẫu huấn luyện sẽ có thể phức tạp và tốn thời gian.
  - Mục tiêu vẫn cần rút ra $\boldsymbol{\theta}$ để làm minimize $J(\boldsymbol{\theta})$, tiến trình có thể khác nhau.

Đạo hàm ma trận

- Cho hàm $f: R^{m \times n} \to R$, đạo hàm của f trên A:

$$\nabla_A f(A) = \begin{bmatrix} \frac{\partial f}{\partial A_{11}} & \cdots & \frac{\partial f}{\partial A_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial A_{m1}} & \cdots & \frac{\partial f}{\partial A_{mn}} \end{bmatrix}$$

- Ví dụ: $f(A) = \frac{3}{2}A_{11} + 5A_{12}^2 + A_{21}A_{22}$ với ma trận $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$, đạo hàm của $f(A)$:

$$\nabla_A f(A) = \begin{bmatrix} \frac{3}{2} & 10A_{12} \\ A_{22} & A_{21} \end{bmatrix}$$

Toán tử "trace"

- Cho ma trận A, "trace" của A là tổng các phần tử trên đường chéo:

$$\mathrm{tr}A = \sum_{i=1}^{n} A_{ii}$$

- Nếu a là số thực, $\mathrm{tr}a = a$
- Một số tính chất:
  - $\mathrm{tr}AB = \mathrm{tr}BA$ hay $\mathrm{tr}ABC = \mathrm{tr}CAB = \mathrm{tr}BCA$
  - $\mathrm{tr}A = \mathrm{tr}A^T$
  - $\mathrm{tr}(A + B) = \mathrm{tr}A + \mathrm{tr}B$
  - $\mathrm{tr}aA = a\mathrm{tr}A$

Biểu thức đạo hàm của "trace"

- Biểu thức đạo hàm (tự c/m):

$$\nabla_A \mathrm{tr} AB = B^T \quad (1)$$

$$\nabla_{A^T} f(A) = (\nabla_A f(A))^T \quad (2)$$

$$\nabla_A \mathrm{tr} ABA^T C = CAB + C^T AB^T \quad (3)$$

$$\nabla_A |\mathrm{A}| = |A| \, (A^{-1})^T \quad (4)$$

- Từ (2) và (3):

$$\nabla_{A^T} \mathrm{tr} ABA^T C = B^T A^T C^T + BA^T C \quad (5)$$

Ma trận thiết kế

- Ma trận thiết kế (design matrix) cho giá trị nhập của các mẫu huấn luyện:

$$X = \begin{bmatrix} (\mathrm{x}^1)^T \\ (\mathrm{x}^2)^T \\ \vdots \\ (\mathrm{x}^m)^T \end{bmatrix}$$

- Gọi $\vec{y}$ là vector m-chiều chứa các giá trị output tương ứng với các mẫu:

$$\vec{y} = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^m \end{bmatrix}$$

Hàm chi phí

- Do $h_\theta(\mathbf{x}) = \mathbf{x}^T\boldsymbol{\theta}$:

$$X\boldsymbol{\theta} - \vec{y} = \begin{bmatrix} (\mathbf{x}^1)^T\boldsymbol{\theta} \\ (\mathbf{x}^2)^T\boldsymbol{\theta} \\ \vdots \\ (\mathbf{x}^m)^T\boldsymbol{\theta} \end{bmatrix} - \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^m \end{bmatrix} = \begin{bmatrix} h_\theta(\mathbf{x}^1) - y^1 \\ h_\theta(\mathbf{x}^2) - y^2 \\ \vdots \\ h_\theta(\mathbf{x}^m) - y^m \end{bmatrix}$$

- Với vector z bất kỳ, $z^T z = \sum_i z_i^2$

$$\frac{1}{2}(X\boldsymbol{\theta} - \vec{y})^T(X\boldsymbol{\theta} - \vec{y}) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(\mathbf{x}^i) - y^i)^2 = J(\theta)$$

Đạo hàm hàm chi phí

- Áp dụng:
  - Tính chất $\mathrm{tr}a = a$ cho bước 3 đạo hàm.
  - Tính chất $\mathrm{tr}A = \mathrm{tr}A^T$ cho bước 4 đạo hàm.
  - Bước 5 sử dụng biểu thức (5) $\nabla_{A^T}\mathrm{tr}ABA^TC = B^TA^TC^T + BA^TC$ với $A^T = \theta$, $B = B^T = X^TX$, $C = I$ và biểu thức (1) $\nabla_A\mathrm{tr}AB = B^T$.
- Đạo hàm hàm chi phí:

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \nabla_\theta \frac{1}{2}(X\theta - \vec{y})^T(X\theta - \vec{y}) \\
&= \frac{1}{2}\nabla_\theta \left(\theta^T X^T X\theta - \theta^T X^T \vec{y} - \vec{y}^T X\theta + \vec{y}^T\vec{y}\right) \\
&= \frac{1}{2}\nabla_\theta \,\mathrm{tr}\left(\theta^T X^T X\theta - \theta^T X^T \vec{y} - \vec{y}^T X\theta + \vec{y}^T\vec{y}\right) \\
&= \frac{1}{2}\nabla_\theta \left(\mathrm{tr}\,\theta^T X^T X\theta - 2\mathrm{tr}\,\vec{y}^T X\theta\right) \\
&= \frac{1}{2}\left(X^T X\theta + X^T X\theta - 2X^T\vec{y}\right) \\
&= X^T X\theta - X^T\vec{y}
\end{aligned}
$$

- Tìm cực trị bằng cách cho đạo hàm =0:

$$X^T X\theta = X^T\vec{y}$$

$$\theta = (X^T X)^{-1}X^T\vec{y}$$

Gradient descent vs. normal equation

Gradient descent	Normal equation
- Cần chọn hệ số học α. - Cần chạy nhiều vòng lặp. - Làm việc tốt thậm chí khi số chiều (số đặc trưng) n lớn	- Không cần chọn hệ số học. - Không cần nhiều vòng chạy. - Chỉ cần tính $(X^T X)^{-1}$ - Độ phức tạp $O(n^3)$ nên chậm khi số n lớn. - $X^T X$ có thể không khả nghịch (khắc phục bằng cách tránh trùng lắp đặc trưng, giảm số lượng đặc trưng hay áp dụng *regularization*).

Giải thích theo xác suất (1/2)

- Giả sử dữ liệu được phân bố theo xác suất chuẩn (gaussian):

$$p(y^{(i)}|x^{(i)};\theta) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)$$

- Hàm likelihood:

$$\begin{aligned} L(\theta) &= \prod_{i=1}^{m} p(y^{(i)} \mid x^{(i)};\theta) \\ &= \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right). \end{aligned}$$

Giải thích theo xác suất (2/2)

- Tìm maximum likelihood thông qua hàm log:

$$\begin{aligned}
\ell(\theta) &= \log L(\theta) \\
&= \log \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}\right) \\
&= \sum_{i=1}^{m} \log \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}\right) \\
&= m \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{\sigma^2} \cdot \frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - \theta^T x^{(i)})^2
\end{aligned}$$

- Như vậy, ta thấy rằng maximum likelihood, đồng nghĩa với việc minimum:

$$\frac{1}{2} \sum_{i=1}^{m} (y^{(i)} - \theta^T x^{(i)})^2$$

HQTT có đánh trọng cục bộ

- Hồi quy tuyến tính có đánh trọng cục bộ có dạng hàm chi phí sau:

$$\sum_i w^{(i)}(y^{(i)} - \theta^T x^{(i)})^2$$

- Trọng số sẽ đánh giá độ ưu tiên cho từng điểm dữ liệu.
  - Những điểm có trọng số cao thì thuật toán sẽ cố chọn θ để làm cho hàm chi phí nhỏ.
  - Những điểm có trọng số thấp thì thuật toán sẽ gần như bỏ qua (điểm nhiễu).
- Một cách chọn trọng số [3]: $w^{(i)} = \exp\left(-\frac{(x^{(i)} - x)^2}{2\tau^2}\right)$

Tài liệu tham khảo

[1] Christopher.M.Bishop, Chương 3, “Pattern Recognition and Machine Learning”, 2007.

[2] Andrew Ng, Lecture 2 & 5, “Machine Learning Courses”, 2011

[3] Andrew Ng, Lecture Notes 1, “Machine Learning Courses”, CS229

[4] Wikipedia, Linear Regression, http://en.wikipedia.org/wiki/Linear_regression