Stochastic Gradient Descent (SGD)

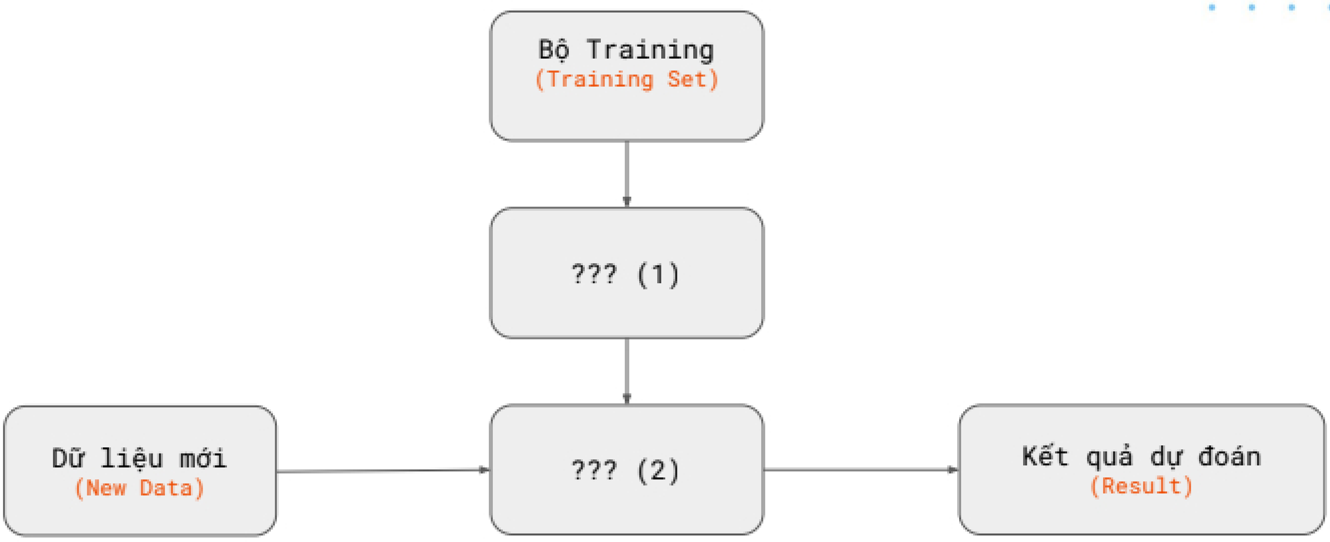
Stochastic Gradient Descent (SGD)

PROTON X

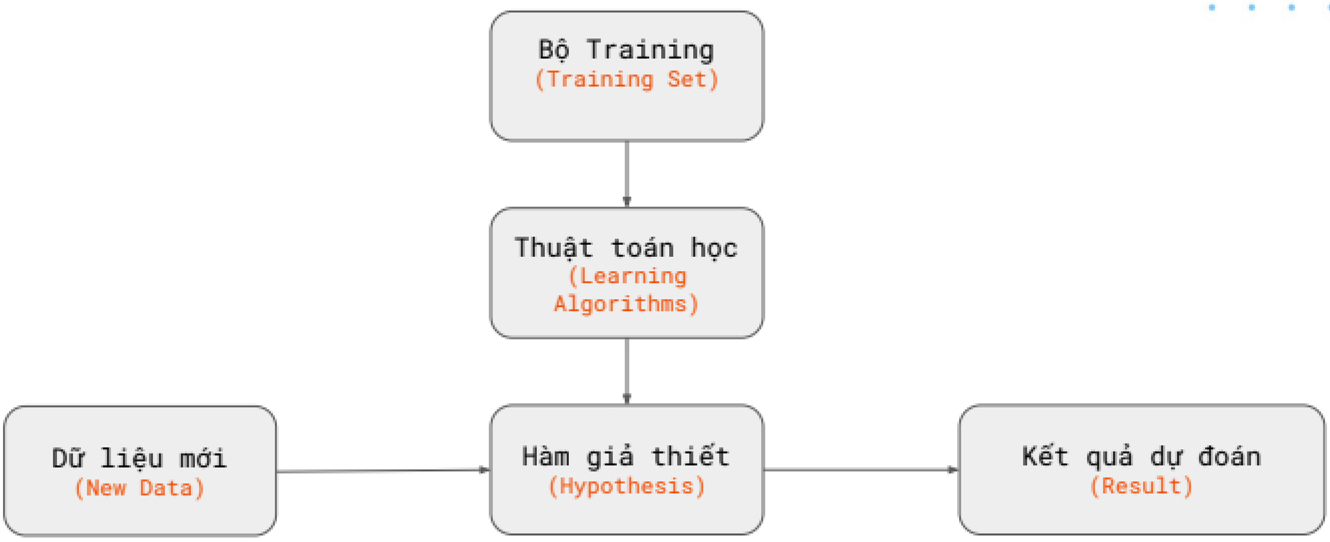
PROTONX

Nhắc lại Quá trình Training

Training Progress Recall



Nhắc lại Quá trình Training
Training Progress Recall



Vấn đề của Gradient Descent

Gradient Descent Problem

Hàm giả thiết

$$h_\theta(\mathbf{x}) = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n$$

Ở một vòng lặp (epoch)

Một phần tử của bộ tham số

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

Toàn bộ tham số

$$\theta := \theta - \alpha \nabla_\theta J(\theta)$$

$$\nabla_\theta J(\theta) = \frac{1}{m} X^T (X \cdot \theta - \mathbf{y})$$

Thuật toán Gradient Descent còn được gọi là **Batch Gradient Descent**

Vấn đề của Gradient Descent

Gradient Descent Problem

Ở một vòng lặp (epoch)

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

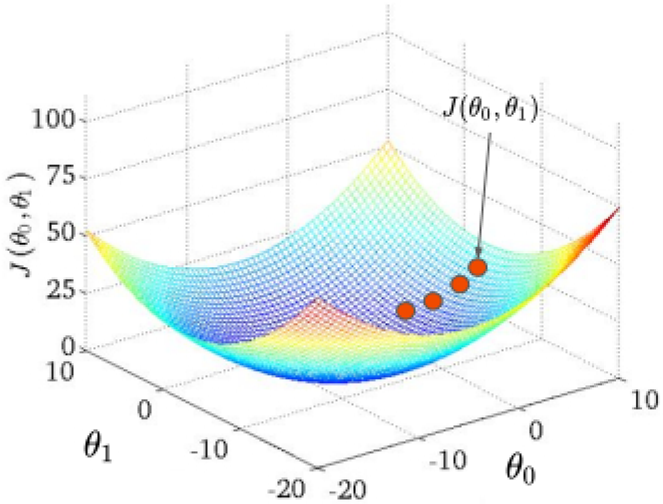
Tính toán Gradient trên toàn bộ tập dữ liệu sẽ **trở nên rất tốn kém** khi mà m lớn

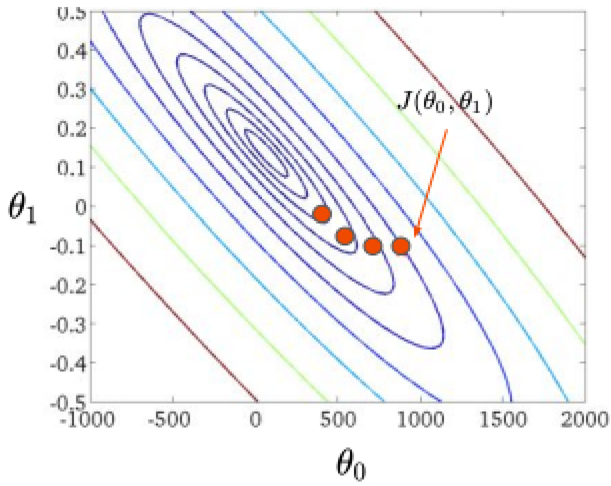
Ví dụ **$m = 10^9$** điểm dữ liệu

Quá trình training

Training Process

$J(\theta_0, \theta_1)$





Hàm giả thiết

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

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

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

Cách nào tốt hơn

What better strategy

Thi đại học

Lớp 12

.
.
.

Lớp 2

Lớp 1

Cậu bé lựa chọn **học hết 12 lớp** rồi mới quyết định thi đại học

Thi đại học

Lớp 12

.
.
.

Lớp 2

Lớp 1

Cậu bé lựa chọn **cứ kết thúc** một lớp lại đi thi đại học

Cách nào tốt hơn

What better strategy

Thi đại học

Lớp 12

.
.
.

Lớp 2

Lớp 1

Cậu bé lựa chọn **học hết 12 lớp** rồi mới quyết định thi đại học

Lợi	Bất lợi
Được đào tạo bài bản cho nên khả năng thi đỗ sẽ cao	Tốn kém thời gian và công sức. Kiến thức bị nhồi nhét, Dễ sinh tẩu hỏa.

Cách nào tốt hơn

What better strategy

Thi đại học

Lớp 12

.
.
.

Lớp 2

Lớp 1

Cậu bé lựa chọn **cứ kết thúc** một lớp lại đi thi đại học

Lợi	Bất lợi
Có thể thi đỗ ngay từ lớp 10 - rút ngắn thời gian phấn đấu. Nắm bắt được dạng đề sớm, chuẩn bị được tâm lý phòng thi tốt	Những năm đầu tiên có thể thất bại

Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD)

Batch Gradient Descent

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

Stochastic Gradient Descent

$$cost(\theta, (x^{(i)}, y^{(i)})) = \frac{1}{2}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Độ sai lệch ở một điểm dữ liệu

Chạy vòng lặp, ở mỗi vòng lặp (epoch)

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

Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD)

Batch Gradient Descent

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

$$\boxed{J(\theta) = \frac{1}{m} \sum_{i=1}^{m} cost(\theta, (x^{(i)}, y^{(i)}))}$$

Chạy vòng lặp, ở mỗi vòng lặp (epoch)

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

Stochastic Gradient Descent

$$cost(\theta, (x^{(i)}, y^{(i)})) = \frac{1}{2} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Độ sai lệch ở một điểm dữ liệu

Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD)

Batch Gradient Descent

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

$$\boxed{J(\theta) = \frac{1}{m}\sum_{i=1}^{m} cost(\theta, (x^{(i)}, y^{(i)}))}$$

Chạy vòng lặp, ở mỗi vòng lặp (epoch)

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

Với j từ 1,...,n

Stochastic Gradient Descent

$$cost(\theta, (x^{(i)}, y^{(i)})) = \frac{1}{2}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Độ sai lệch ở một điểm dữ liệu

1 → Chạy vòng lặp, ở mỗi vòng lặp **(epoch)**

1.1 → Xáo (Shuffle) ngẫu nhiên bộ dữ liệu

1.2 → Lặp i từ 1 đến m

1.2.1 Cập nhật theta

$$\theta_j := \theta_j - \alpha\frac{\partial cost(\theta, (x^{(i)}, y^{(i)}))}{\partial \theta_j}$$

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

Với j từ 1,...,n

Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent

$$cost(\theta, (x^{(i)}, y^{(i)})) = \frac{1}{2}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Độ sai lệch ở một điểm dữ liệu

1 → Chạy vòng lặp, ở mỗi vòng lặp **(epoch)**

1.1 → Xáo (Shuffle) ngẫu nhiên bộ dữ liệu

1.2 → Lặp i từ 1 đến m

1.2.1 Cập nhật theta

$$\theta_j := \theta_j - \alpha \frac{\partial cost(\theta, (x^{(i)}, y^{(i)}))}{\partial \theta_j}$$

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

Với j từ 1,...,n

Không giống như Gradient Descent tính Gradient trên toàn bộ tập dữ liệu rồi mới cập nhật theta,

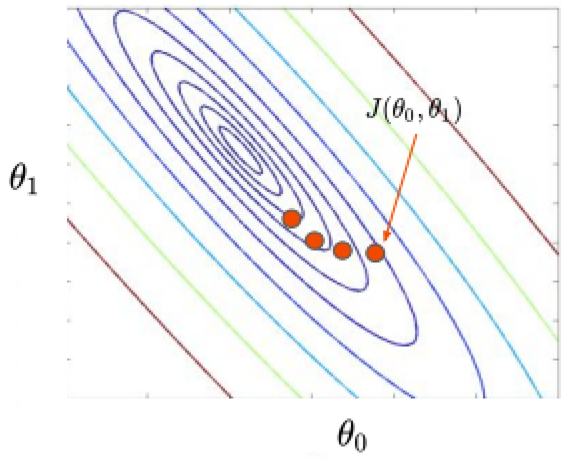
thì SGD tính Gradient luôn trên **một điểm dữ liệu** và cập nhật theta **ngay lập tức**.

Sau đó duyệt các điểm dữ liệu tiếp theo và làm tương tự.

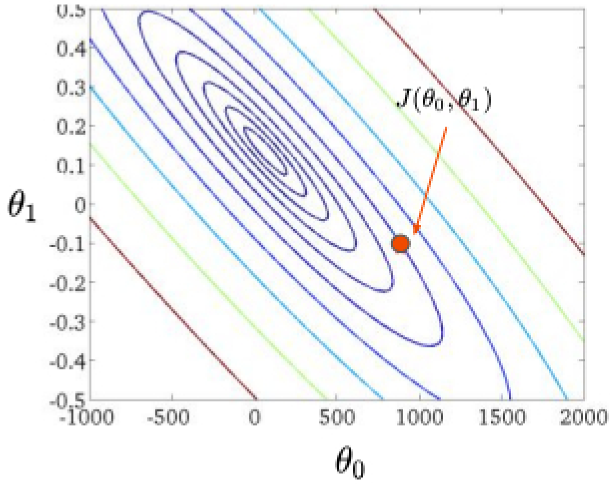
PROTON X

Quá trình training với SGD

Training Progress using SGD



Gradient Descent

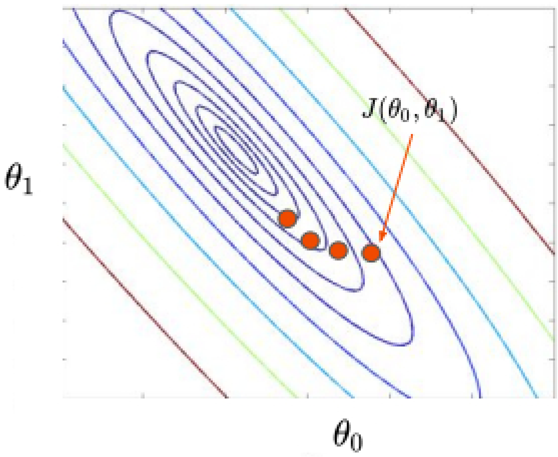


Stochastic Gradient Descent

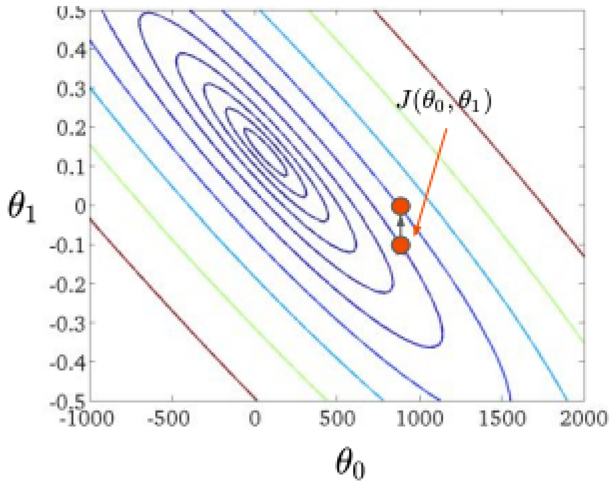
Ở mỗi một vòng lặp qua 1 điểm dữ liệu, thì theta được cập nhật chỉ với điểm dữ liệu đó. Nên đường đi của SGD sẽ bất định ở thời điểm đầu, theo nhiều hướng khác nhau.Nhưng nếu chúng ta duyệt đủ lâu, SGD có khả năng làm cho **mô hình hội tụ ở khu vực xung** quanh cực trị (global minimum).

Quá trình training với SGD

Training Progress using SGD



Gradient Descent

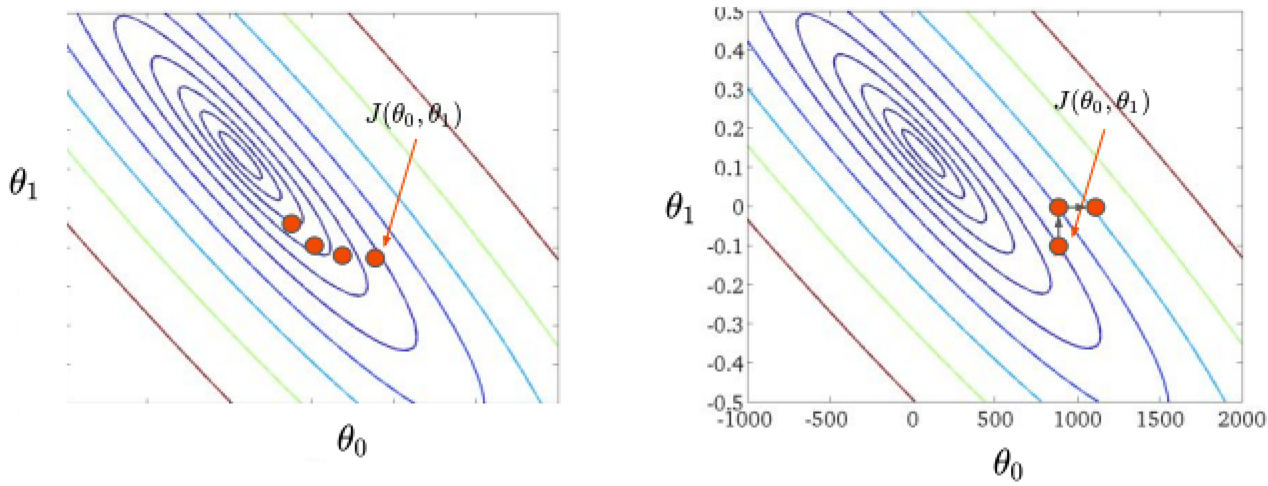


Stochastic Gradient Descent

Ở mỗi một vòng lặp qua 1 điểm dữ liệu, thì theta được cập nhật chỉ với điểm dữ liệu đó. Nên đường đi của SGD sẽ bất định ở thời điểm đầu, theo nhiều hướng khác nhau.Nhưng nếu chúng ta duyệt đủ lâu, SGD có khả năng làm cho **mô hình hội tụ ở khu vực xung** quanh cực trị (global minimum).

Quá trình training với SGD

Training Progress using SGD

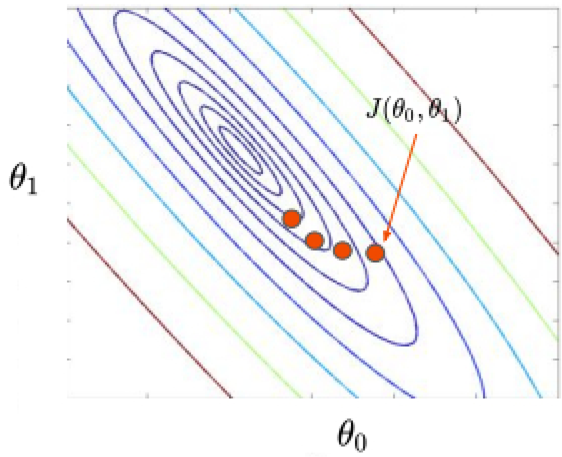


Gradient Descent Stochastic Gradient Descent

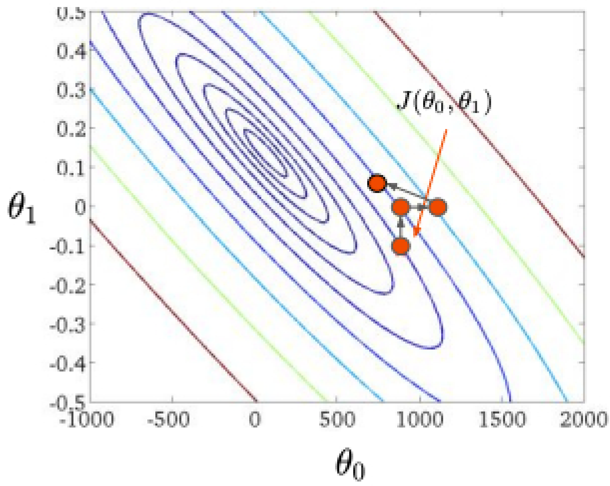
Ở mỗi một vòng lặp qua 1 điểm dữ liệu, thì theta được cập nhật chỉ với điểm dữ liệu đó. Nên đường đi của SGD sẽ bất định ở thời điểm đầu, theo nhiều hướng khác nhau.Nhưng nếu chúng ta duyệt đủ lâu, SGD có khả năng làm cho **mô hình hội tụ ở khu vực xung** quanh cực trị (global minimum).

Quá trình training với SGD

Training Progress using SGD



Gradient Descent

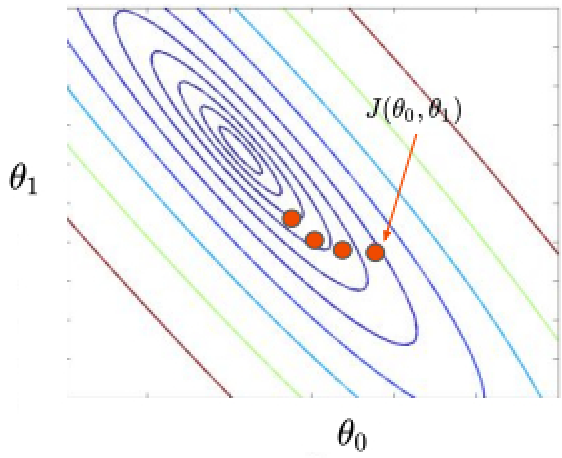


Stochastic Gradient Descent

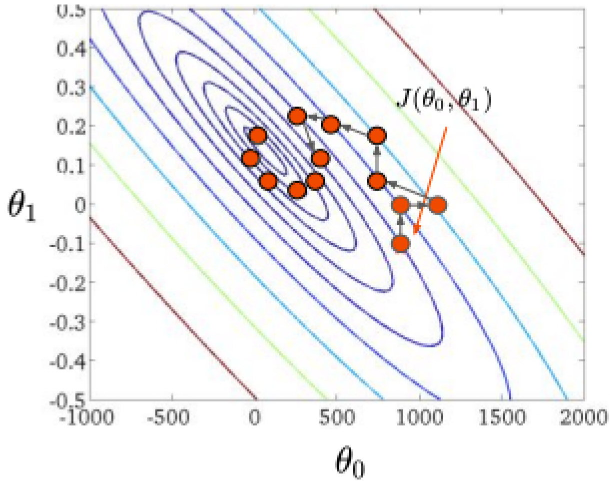
Ở mỗi một vòng lặp qua 1 điểm dữ liệu, thì theta được cập nhật chỉ với điểm dữ liệu đó. Nên đường đi của SGD sẽ bất định ở thời điểm đầu, theo nhiều hướng khác nhau.Nhưng nếu chúng ta duyệt đủ lâu, SGD có khả năng làm cho **mô hình hội tụ ở khu vực xung** quanh cực trị (global minimum).

Quá trình training với SGD

Training Progress using SGD



Gradient Descent



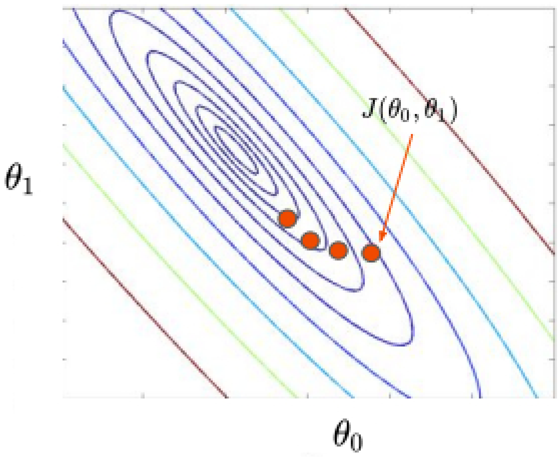
Stochastic Gradient Descent

Ở mỗi một vòng lặp qua 1 điểm dữ liệu, thì theta được cập nhật chỉ với điểm dữ liệu đó. Nên đường đi của SGD sẽ bất định ở thời điểm đầu, theo nhiều hướng khác nhau.Nhưng nếu chúng ta duyệt đủ lâu, SGD có khả năng làm cho **mô hình hội tụ ở khu vực xung** quanh cực trị (global minimum).

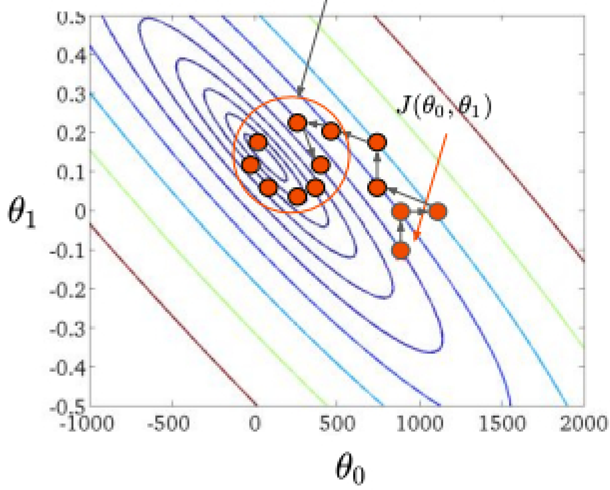
Quá trình training với SGD

Training Progress using SGD

khu vực xung quanh cực trị (global minimum)



Gradient Descent



Stochastic Gradient Descent

Ở mỗi một vòng lặp qua 1 điểm dữ liệu, thì theta được cập nhật chỉ với điểm dữ liệu đó. Nên đường đi của SGD sẽ bất định ở thời điểm đầu, theo nhiều hướng khác nhau.Nhưng nếu chúng ta duyệt đủ lâu, SGD có khả năng làm cho **mô hình hội tụ ở khu vực xung** quanh cực trị (global minimum).

Mini Batch Gradient Descent

Mini Batch Gradient Descent

Batch Gradient Descent

Ở một epoch, cập nhật theta thông qua Gradient độ sai lệch **trên toàn tập dữ liệu**

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

Stochastic Gradient Descent

Ở một epoch,lặp qua các điểm dữ liệu và cập thật theta thông qua Gradient độ sai lệch **trên từng điểm dữ liệu**

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

Mini Batch Gradient Descent

Ở một epoch,lặp qua các điểm dữ liệu và cập thật theta thông qua Gradient độ sai lệch **trên c điểm dữ liệu**

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

Mini Batch Gradient Descent

Mini Batch Gradient Descent

Ở một epoch,lặp qua các bước (step)
Trong một bước cập thật theta thông qua Gradient độ sai lệch **trên c điểm dữ liệu**

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

$c = 5$ **Độ sai lệch của c điểm dữ liệu**

1 → Chạy vòng lặp, ở mỗi vòng lặp **(epoch)**

1.1 → Lặp i trong khoảng 1, 6, 11, …,m

1.1.1 Cập nhật theta

$$\theta_j := \theta_j - \alpha \frac{1}{5} \sum_{k=i}^{i+4} (h_\theta(x^{(k)}) - y^{(k)}) x_j^{(k)}$$

Với j từ 1,...,n

[Nâng cao] Tại sao SGD hoạt động

[Advanced] Why does SGD works?

Kỳ vọng có điều kiện

$$\mathbf{E}[\nabla cost(\theta, (x^{(\tilde{i})}, y^{(\tilde{i})})|\theta] = \sum_{i=1}^{m} \nabla cost(\theta, (x^{(i)}, y^{(i)})\mathbf{P}(\tilde{i} = i|\theta)$$
$$= \sum_{i=1}^{m} \nabla cost(\theta, (x^{(i)}, y^{(i)}))\frac{1}{m} = \nabla_\theta J(\theta)$$

Kỳ vọng hàm mất mát trên từng điểm dữ liệu

Phân phối đều (Uniform)

Đạo hàm trên toàn bộ tập dữ liệu

The insight of SGD is that the $\boxed{\text{gradient is an expectation}}$. The expectation may be approximately estimated using a small set of samples. Specifically, on each step of the algorithm, we can sample a **minibatch** of examples $\mathbb{B} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m')}\}$ drawn uniformly from the training set. The minibatch size m' is typically chosen to be a relatively small number of examples, ranging from one to a few hundred. Crucially, m' is usually held fixed as the training set size m grows. We may fit a training set with billions of examples using updates computed on only a hundred examples.

Khi ta train SGD đủ lâu, kỳ vọng các đạo hàm trên từng điểm dữ liệu sẽ xấp xỉ đạo hàm trên toàn bộ tập dữ liệu

Lúc này, kết quả của SGD sẽ tiệm cận kết quả của GD

https://www.deeplearningbook.org/contents/regularization.html

[Nâng cao] Tại sao SGD hội tụ xung quanh cực trị

[Advanced] Why does SGD coverge aroud the optimal point?

Giá trị hàm mất mát tại điểm xuất phát

Kỳ vọng cường độ đạo hàm

Giá trị cực tiểu

Giá trị dương này làm cho đạo hàm của SGD rất khó về 0

$$\mathbf{E}[||\nabla J(\theta_t)||^2] = \frac{J(\theta_0) - J^*}{\alpha T} + \boxed{\alpha G}$$

Số dương G > 0

Số Epochs

Thời điểm t

Tốc độ học

Chứng minh:

- Sử dụng định lý Taylor