Stochastic Gradient Descent

10701 Recitations 3

Mu Li

Computer Science Department
Cargenie Mellon University

February 5, 2013

The problem

- A typical machine learning problem has a penalty/regularizer + loss form

$$\min_w F(w) = g(w) + \frac{1}{n}\sum_{i=1}^{n} f(w; y_i, x_i),$$

  $x_i, w \in \mathbb{R}^p$, $y_i \in \mathbb{R}$, both g and f are convex
- Today we only consider differentiable f, and let $g = 0$ for simplicity
- For example, let $f(w; y_i, x_i) = -\log p(y_i|x_i, w)$, we are trying to maximize the log likelihood, which is

$$\max_w \frac{1}{n}\sum_{i=1}^{n} \log p(y_i|x_i, w)$$

Gradient Descent

- choose initial $w^{(0)}$, repeat

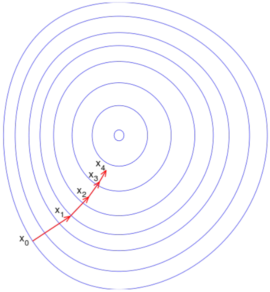
$$w^{(t+1)} = w^{(t)} - \eta_t \cdot \nabla F(w^{(t)})$$

  until stop
- η_t is the learning rate, and

$$\nabla F(w^{(t)}) = \frac{1}{n} \sum_i \nabla_w f(w^{(t)}; y_i, x_i)$$

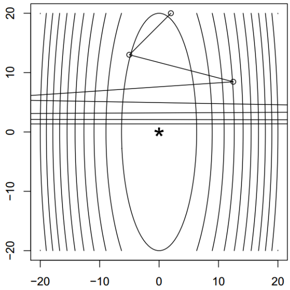
- How to stop? $\|w^{(t+1)} - w^{(t)}\| \leq \epsilon$ or $\|\nabla F(w^{(t)})\| \leq \epsilon$

Two dimensional example:

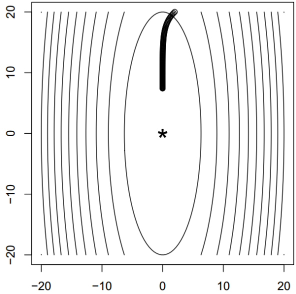


Learning rate matters

$\eta_t = t$, it is too big



too small η_t, after 100 iterations



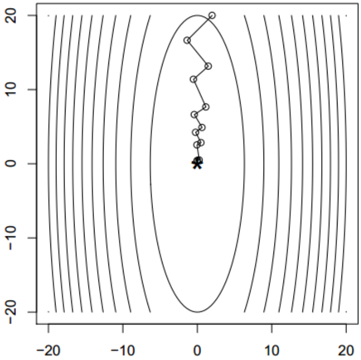
Backtracking line search

Adaptively choose the learning rate

- choose a parameter $0 < \beta < 1$
- start with $\eta = 1$, repeat $t = 0, 1, \ldots$
  - while
    $$L(w^{(t)} - \eta \nabla L(w^{(t)})) > L(w^{(t)}) - \frac{\eta}{2} \|\nabla L(w^{(t)})\|^2$$
    update $\eta = \beta \eta$
  - $w^{(t+1)} = w^{(t)} - \eta \nabla L(w^{(t)})$

Backtracking line search

A typical choice $\beta = 0.8$, converged after 13 iterations:



Stochastic Gradient Descent

- We name $\frac{1}{n}\sum_i f(w; y_i, x_i)$ the empirical loss, the thing we hope to minimize is the expected loss

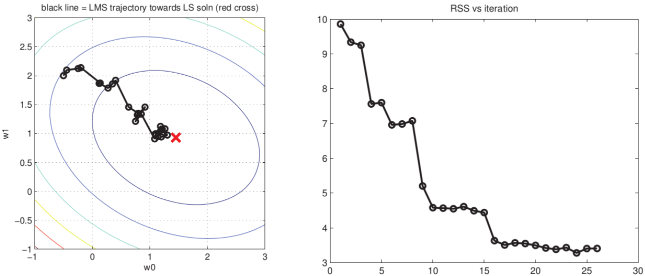
$$f(w) = \mathbb{E}_{y_i, x_i} f(w; y_i, x_i)$$

- Suppose we receive an infinite stream of samples (y_t, x_t) from the distribution, one way to optimize the objective is

$$w^{(t+1)} = w^{(t)} - \eta_t \nabla_w f(w^{(t)}; y_t, x_t)$$

- On practice, we simulate the stream by randomly pick up (y_t, x_t) from the samples we have
- Comparing the average gradient of GD $\frac{1}{n}\sum_i \nabla_w f(w^{(t)}; y_i, x_i)$

More about SGD



- the objective does not always decrease for each step
- comparing to GD, SGD needs more steps, but each step is cheaper
- mini-batch, say pick up 100 samples and do average, may accelerate the convergence

Relation to Perceptron

- Recall Perceptron: initialize w, repeat

$$w = w + \begin{cases} y_i x_i & \text{if } y_i \langle w, x_i \rangle < 0 \\ 0 & \text{otherwise} \end{cases}$$

- Fix learning rate $\eta = 1$, let $f(w; y, x) = \max(0, -y_i \langle w, x_i \rangle)$, then

$$\nabla_w f(w; y, x) = \begin{cases} -y_i x_i & \text{if } y_i \langle w, x_i \rangle < 0 \\ 0 & \text{otherwise} \end{cases}$$

  we derive Perceptron from SGD

Question?