

SDS 385: Stat Models for Big Data

Lecture 4: GD with momentum.

Purnamrita Sarkar
Department of Statistics and Data Science
The University of Texas at Austin
<https://psarkar.github.io/teaching>

Polyak's heavy ball method

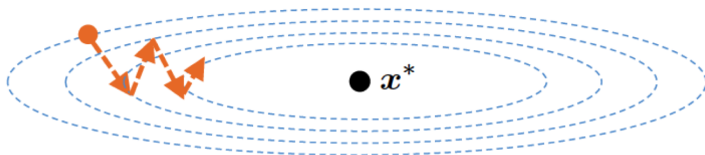
Figure 1: B. Polyak



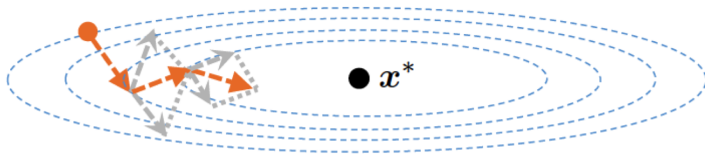
•

$$\beta_{t+1} = \beta_t - \alpha \nabla f(\beta_t) + \underbrace{\theta(\beta_t - \beta_{t-1})}_{\text{momentum term}}$$

Momentum



gradient descent



heavy-ball method

Recall GD?

- For a L smooth and μ convex optimization problem, i.e.
 $\mu I \preceq \|H\| \preceq LI$,

$$\|\beta_t - \beta^*\| \leq \left(\frac{\kappa - 1}{\kappa + 1} \right)^t \|\beta_0 - \beta^*\|$$

where $\kappa = L/\mu$ i.e. the condition number of the Hessian.

- For the same problem, using Polyak's method we can show that,

$$\left\| \begin{bmatrix} \beta_{t+1} - \beta^* \\ \beta_t - \beta^* \end{bmatrix} \right\| \leq \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^t \left\| \begin{bmatrix} \beta_1 - \beta^* \\ \beta_0 - \beta^* \end{bmatrix} \right\|$$

Momentum method

- Recall we have:

$$\begin{aligned}\beta_{t+1} - \beta^* &= (1 + \theta)(\beta_t - \beta^*) - \alpha \nabla f(\beta_t) - \theta(\beta_{t-1} - \beta^*) \\ &= ((1 + \theta)I - \alpha \nabla^2 f(z_t))(\beta_t - \beta^*) - \theta(\beta_{t-1} - \beta^*)\end{aligned}$$

- This gives the dynamic system:

$$\begin{bmatrix} \beta_{t+1} - \beta^* \\ \beta_t - \beta^* \end{bmatrix} \leq \begin{bmatrix} (1 + \theta)I - \alpha \nabla^2 f(z_t) & -\theta I \\ I & 0 \end{bmatrix} \begin{bmatrix} \beta_t - \beta^* \\ \beta_{t-1} - \beta^* \end{bmatrix}$$

Momentum method

- We need to upper bound the norm of

$$M := \begin{bmatrix} (1 + \theta)I - \alpha \nabla^2 f(z_t) & -\theta I \\ I & 0 \end{bmatrix}$$

- It can be shown that:

$$\begin{aligned} \|M\| &= \left\| \begin{bmatrix} (1 + \theta) - \alpha \Lambda & -\theta I \\ I & 0 \end{bmatrix} \right\| \\ &= \max_i \left\| \begin{bmatrix} (1 + \theta) - \alpha \lambda_i & -\theta \\ 1 & 0 \end{bmatrix} \right\| \end{aligned}$$

- Eigenvalues of the 2×2 matrix can be written as a solution of the following quadratic:

$$\sigma^2 - \sigma((1 + \theta) - \alpha \lambda_i) + \theta = 0$$

Momentum method

- If $((1 + \theta) - \alpha\lambda_i)^2 \leq 4\theta$, the roots are imaginary and the magnitude is $\sqrt{\theta}$
- This is satisfied if

$$\theta \in [(1 - \sqrt{\alpha\lambda_i})^2, (1 + \sqrt{\alpha\lambda_i})^2]$$

- But recall that $\lambda_i \in [\mu, L]$.
- If we set $1 - \sqrt{\alpha L} = -(1 - \sqrt{\alpha\mu})$, then we have

$$\alpha = \left(\frac{2}{\sqrt{L} + \sqrt{\mu}} \right)^2 \quad \theta = \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^2$$

- So the new contraction factor becomes $\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}$

Nesterov's Accelerated Gradient

- If we only assume that $\|\nabla^2 f(x)\| \leq L$ and not strong convexity, then in your homework you will prove that

$$f(\beta_t) - f(\beta^*) \leq c_L \frac{\|\beta_0 - \beta^*\|^2}{t}$$

- Note that this is much weaker than the linear convergence we saw before.
- Question is can we do better?

Nesterov's Accelerated Gradient

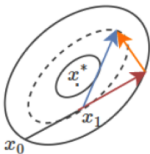
Figure 2: Y. Nesterov



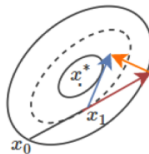
- Keep track of two vectors x_t and y_t
- $x_{t+1} = y_t - \alpha_t \nabla f(y_t)$
- $y_{t+1} = x_{t+1} + \underbrace{\frac{t}{t+3}}_{\mu_{t+1}} (x_{t+1} - x_t)$

Nesterov's Accelerated Gradient

Polyak's Momentum



Nesterov Momentum



- Can be re-written as:

$$x_{t+1} = x_t + \mu(x_t - x_{t-1}) - \alpha_t \nabla f(x_t + \mu_t(x_t - x_{t-1}))$$

- Very much like the momentum method, but computes the derivative at a future step.

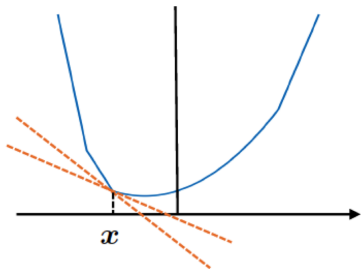
Nesterov's Accelerated Gradient

- Not a descent method.
- If f is convex and L smooth and the learning rate is $1/L$, this obtains the optimal $O(1/t^2)$ error after t steps.
- Proof is complicated, but can be simplified using intuitions from differential equations.

Subgradient methods

- So far we have assumed differentiable f .
- What if f is not differentiable?
- Instead of a gradient we will define a subgradient.

Subgradient methods



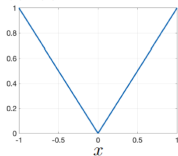
- We will say that g is a subgradient of f at point x if

$$f(z) \geq f(x) + g^T(z - x), \quad \forall z$$

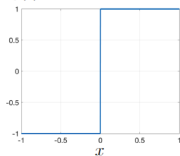
- Set of all gradients is called the sub-differential of f at point x and is denoted by $\partial f(x)$

Example

$$f(x) = |x|$$

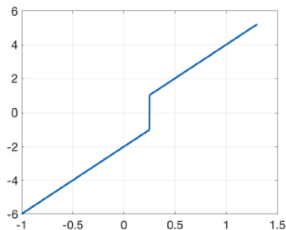
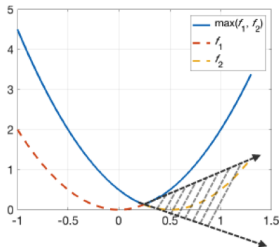


$$\partial f(x)$$



$$f(x) = |x| \quad \partial f(x) = \begin{cases} \{-1\} & \text{if } x < 0 \\ [-1, 1] & \text{if } x = 0 \\ \{1\} & \text{if } x > 0 \end{cases}$$

Example



$f(x) = \max\{f_1(x), f_2(x)\}$ where f_1 and f_2 are differentiable

$$\partial f(x) = \begin{cases} \{f'_1(x)\}, & \text{if } f_1(x) > f_2(x) \\ [f'_1(x), f'_2(x)], & \text{if } f_1(x) = f_2(x) \end{cases}$$

Behaves very much like a gradient;

- $\partial(\alpha f) = \alpha \partial f$ for $\alpha \geq 0$
- $\partial(f + g) = \partial f + \partial g$
- For convex f , if $g(x) = f(Ax + b)$, $\partial g(x) = A^T \partial f(Ax + b)$

$$f(\mathbf{x}) = \|\mathbf{x}\|_1 = \sum_{i=1}^n \underbrace{|x_i|}_{:=f_i(\mathbf{x})}$$

since

$$\partial f_i(\mathbf{x}) = \begin{cases} \operatorname{sgn}(x_i) \mathbf{e}_i, & \text{if } x_i \neq 0 \\ [-1, 1] \cdot \mathbf{e}_i, & \text{if } x_i = 0 \end{cases}$$

we have

$$\sum_{i: x_i \neq 0} \operatorname{sgn}(x_i) \mathbf{e}_i \in \partial f(\mathbf{x})$$

Lets talk about Lasso

$$\hat{\beta}_{LASSO} = \min_{\beta} (\mathbf{y} - \mathbf{x}\beta)^{\top} (\mathbf{y} - \mathbf{x}\beta) + \lambda \sum_{j=1}^p |\beta_j| \quad (1)$$

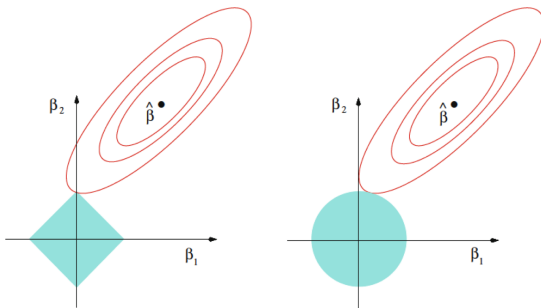


FIGURE 3.11. Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions $|\beta_1| + |\beta_2| \leq t$ and $\beta_1^2 + \beta_2^2 \leq t^2$, respectively, while the red ellipses are the contours of the least squares error function.

Alternative formulation

$$\hat{\beta}_{ridge} = \min_{\beta} (\mathbf{y} - \mathbf{x}\beta)^{\top} (\mathbf{y} - \mathbf{x}\beta) \quad \text{Subject to } \beta^{\top} \beta \leq \tau^2 \quad (2)$$

$$\hat{\beta}_{lasso} = \min_{\beta} (\mathbf{y} - \mathbf{x}\beta)^{\top} (\mathbf{y} - \mathbf{x}\beta) \quad \text{Subject to } \|\beta\|_1 \leq \tau \quad (3)$$

Optimality condition

- For differentiable f

$$f(x^*) = \min_x f(x) \leftrightarrow \nabla f(x^*) = 0$$

- For convex f that may not be differentiable,

$$f(x^*) = \min_x f(x) \leftrightarrow 0 \in \delta f'(x^*)$$

- Just plug into the definition of a subgradient!

$$f(y) \geq f(x^*) + 0^T (y - x^*) = f(x^*)$$

- Consider the easier problem

$$x = \arg \min \frac{1}{2} \|y - x\|^2 + \lambda \|x\|_1$$

- Show that the soft thresholding operator $x^* = S_\lambda(y)$ is the solution to this.

$$S_\lambda(y) = \begin{cases} y_i - \lambda & \text{if } y_i > \lambda \\ 0 & \text{if } y_i \in [-\lambda, \lambda] \\ y_i + \lambda & \text{if } y_i < -\lambda \end{cases}$$

Sub-gradient method

- $\beta_{k+1} = \beta_k - \alpha_k g_{k-1}$
- Here g_{k-1} is any subgradient at the β_{k-1}
- Note that subgradient direction is not always a direction of descent
- So we do

$$f(\beta_k^{best}) = \min_{i=1,\dots,k} f(\beta_i)$$

- We did not cover backtracking line search, but since this is not a descent method we cannot choose the step size adaptively.
- We can choose it as
 - Fixed, i.e. $t_k = \alpha$
 - Or diminishing such that $\sum_k t_k^2 < \infty, \sum_k t_k = \infty$

Convergence

Assume that f convex, $\text{dom}(f) = \mathbb{R}^n$, and also that f is Lipschitz continuous with constant $G > 0$, i.e.,

$$|f(x) - f(y)| \leq G\|x - y\|_2 \quad \text{for all } x, y$$

Theorem: For a fixed step size t , subgradient method satisfies

$$\lim_{k \rightarrow \infty} f(x_{\text{best}}^{(k)}) \leq f^* + G^2 t / 2$$

Theorem: For diminishing step sizes, subgradient method satisfies

$$\lim_{k \rightarrow \infty} f(x_{\text{best}}^{(k)}) = f^*$$

Regularized logistic regression

- Let $\{x_i, y_i\}_{i=1}^n$ with $x_i \in \mathbb{R}^p$ and $y_i \in \{-1, 1\}$
- The logistic regression loss is:

$$f(\beta) = \sum_i \left(-y_i x_i^T \beta + \log(1 + \exp(x_i^T \beta)) \right)$$

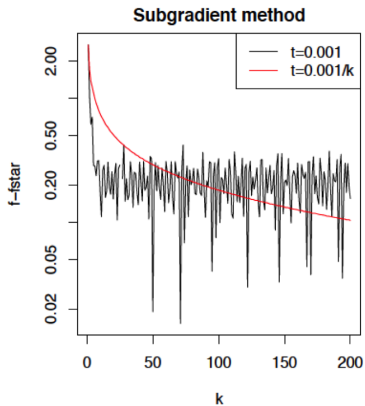
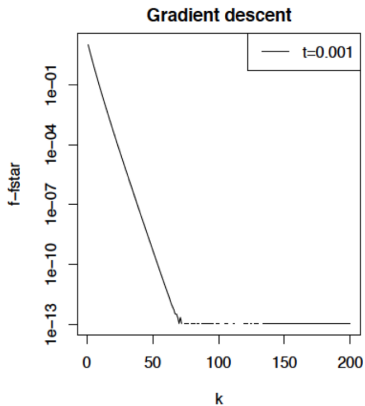
- With lasso regularization we have:

$$\hat{\beta} = \arg \min f(\beta) + \lambda \|\beta\|_1$$

- So, use

$$\Delta_{\beta} = \underbrace{\sum_i (y_i - p_i(\beta)) x_i}_{\text{gradient}} + \underbrace{\partial \|\beta\|_1}_{\text{subgradient}}$$

Convergence



Convergence

- Gradient descent takes $1/\epsilon$ time to converge, whereas subgradient descent with variable step-size takes $1/\epsilon^2$ time to converge.

Theorem

For any $k \leq n - 1$ and starting point $\beta^{(0)}$, there is a function such that any non-smooth first order method satisfies:

$$f(\beta^{(k)}) - f^* \geq \frac{G \|\beta^{(0)} - \beta^*\|}{2(1 + \sqrt{k+1})}$$

-
- So it seems like we cant really improve on sub-gradient methods.

Acknowledgment

Y. Chen's large scale Optimization class at Princeton and Hastie and Tibshirani's book, Ryan Tibshirani's class