1 Outline

From Algorithm 1 to 2, and hopefully we can get Algorithm 3.

Algorithm 1 Standard proximal gradient descent (PG) algorithm

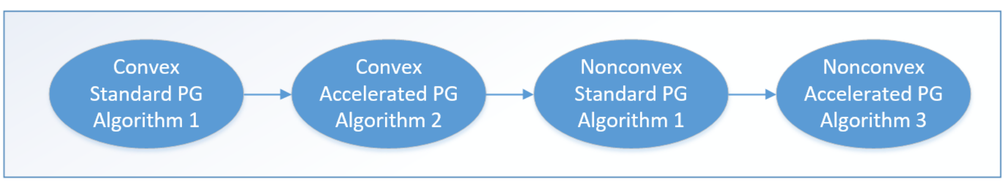
for $t = 1, \dots, T$ **do**
  $x_{t+1} = \text{Prox}_{\frac{1}{L}g}\left(x_t - \frac{1}{L}\nabla f(x_t)\right)$;
end for

Algorithm 2 Accelerated PG (convex)

for $t = 1, \dots, T$ **do**
  $y_t = x_t + \theta_t(x_t - x_{t-1})$ where $\theta_t = \frac{t-1}{t+2}$;
  $z_{t+1} = \text{Prox}_{\frac{1}{L}g}\left(y_t - \frac{1}{L}\nabla f(y_t)\right)$;
end for

Algorithm 3 Accelerated PG (nonconvex)

for $t = 1, \dots, T$ **do**
  $y_t = x_t + \theta_t(x_t - x_{t-1})$ where $\theta_t = \frac{t-1}{t+2}$;
  $z_{t+1} = \text{Prox}_{\frac{1}{L}g}\left(y_t - \frac{1}{L}\nabla f(y_t)\right)$;
  if $F(z_{t+1}) \leq F(x_t) - \delta \left\|y_t - z_{t+1}\right\|_2^2$ **then**
    $x_{t+1} = z_{t+1}$;
  else
    $x_{t+1} = \text{Prox}_{\frac{1}{L}g}\left(x_t - \frac{1}{L}\nabla f(x_t)\right)$;
  end if
end for



In each step ask yourself

Step 1

- What optimization PG algorithm can handle? Why it is popular in machine learning?
- What is the most important step for PG algorithm?

Step 2

- What is the acceleration?
- What are the convergence properties of PG algorithms under convex case?

Step 3

- What does nonconvexity mean? What kind of nonconvexity does PG algorithm allow?
- What is PG algorithm for nonconvex optimization?

Step 4

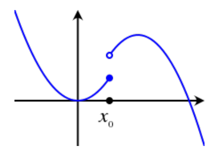
- What is acceleration PG algorithm for nonconvex optimization? Why there are such differences?

2 Assumptions

Optimization problem (composite optimization)

$$\min_x F(x) \equiv f(x) + g(x). \tag{1}$$

Most update to date assumptions



- f is Lipschitz smooth, i.e., $\|\nabla f(x) - \nabla f(y)\|_2 \leq L \|x - y\|_2$
- g is lower semi-continuous (see the right figure)
- F is bounded from below, i.e., $\inf F > -\infty$

3 Algorithms

In this section, we assume f and g are also both convex.

3.1 Standard PG algorithm

The next iterate x_{t+1} is generated as

$$\begin{aligned} x_{t+1} &= \arg\min_x f(x_t) + (x - x_t)^\top \nabla f(x_t) + \frac{L}{2} \|x - x_t\|_2^2 + g(x) \\ &= \arg\min_x \frac{1}{2} \left\| x - \left(x_t - \frac{1}{L} \nabla f(x_t) \right) \right\|_2^2 + \frac{1}{L} g(x) \\ &= \text{Prox}_{\frac{1}{L}g} \left(x_t - \frac{1}{L} \nabla f(x_t) \right) \end{aligned}$$

The most important step: **proximal step** (or proximal operator)

$$x^* = \text{prox}_{\lambda g}(z) \equiv \arg\min_x \frac{1}{2} \|x - z\|_2^2 + \lambda g(x).$$

It should have **cheap (better also closed-form)** solutions.

A convergence rate of $O(1/T)$ is guaranteed, i.e.,

$$F(x_t) - F(x_*) \leq O\left(\frac{1}{T} [F(x_t) - F(x_1)] \right)$$

where T is the number of iterations and x_* is an optimal solution

3.2 Accelerated PG algorithm

The next iterate x_{t+1} is generated as

$$y_t = x_t + \theta_t (x_t - x_{t-1}), \quad x_{t+1} = \text{Prox}_{\frac{1}{L}g} \left(y_t - \frac{1}{L} \nabla f(y_t) \right)$$

where θ_t is a coefficient and can be set as $\theta_t = (t-1)/(t+2)$.

A convergence rate of $O(1/T^2)$ is guaranteed.

- $O(1/T^2)$ is the best rate one can achieve for general convex problems with first order based optimization methods
- Acceleration has been extended to nonconvex problems and convergence can be guaranteed [9]. If standard PG algorithm convergence too slow, switch to accelerated one instead.

3.3 Important Tricks

- What if L is unknown - using line-search, e.g., [3, 9]
- Try larger stepsize - nonmonotonous updates, e.g., [6, 9]

4 Related Papers

Good monograph on proximal gradient descent algorithms.

- A survey paper by Boyd [11]
  - Sample codes and slides: `http://web.stanford.edu/~boyd/papers/prox_algs.html`
  - It mainly covers topics of PG algorithm for convex optimization
- Applications of PG algorithm on sparse learning problems - Section 3 [1]

Mile-stone papers

	convex	nonconvex
standard	[5, 13]	[6]
accelerated	[3]	[9]

Extensions of PG algorithms

- proximal gradient + Newton (second order method) [8]
- proximal average: $g(x) = \sum g_i(x)$, g does not have closed-form solution on proximal step, but each g_i does [16, 18]
- inexact PG algorithm [12, 15]

Some closed-form solutions on various proximal steps (convex g)

- group lasso [17]
- tree-structured lasso [10, 7]
- nuclear norm [4]

Some algorithms designed to solve proximal step (convex g), when no closed-form solutions

- overlapping group lasso [17]
- total variation [2]

For nonconvex regularizers (nonconvex g), we can directly handle proximal step with such g, or using transformation at [14] to convert them back to convex ones.

References

[1] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 4(1):1–106, 2012.

[2] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.

[3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[4] J.-F. Cai, E.J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[5] V.R. Combettes, P.L .and Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.

[6] P. Gong, C. Zhang, Z. Lu, J. Huang, and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *ICML*, pages 37–45, 2013.

[7] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for hierarchical sparse coding. *Journal of Machine Learning Research*, 12:2297–2334, 2011.

[8] J. D Lee, Y. Sun, and M.A. Saunders. Proximal newton-type methods for convex optimization. In *Advances in Neural Information Processing Systems*, volume 25, pages 827–835, 2012.

[9] H. Li and Z. Lin. Accelerated proximal gradient methods for nonconvex programming. In *NIPS*, pages 379–387, 2015.

[10] J. Liu and J. Ye. Moreau-Yosida regularization for grouped tree structure learning. In *Advances in Neural Information Processing Systems*, pages 1459–1467, 2010.

[11] N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):127–239, 2014.

[12] M. Schmidt, N.L. Roux, and F. Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. In *NIPS*, pages 1458–1466, 2011.

[13] S.J. Wright, R.D. Nowak, and M.A. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7):2479–2493, 2009.

[14] Q. Yao and J.T. Kwok. Efficient learning with a family of nonconvex regularizers by redistributing nonconvexity. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 2645–2654, 2016.

[15] Q. Yao and J.T. Kwok. More efficient accelerated proximal algorithm for nonconvex problems. *arXiv preprint arXiv:1612.09069*, 2016.

[16] Y.-L. Yu. Better approximation and faster algorithm using the proximal average. In *Advances in Neural Information Processing Systems*, pages 458–466, 2013.

[17] L. Yuan, J. Liu, and J. Ye. Efficient methods for overlapping group lasso. In *Advances in Neural Information Processing Systems*, pages 352–360, 2011.

[18] Wenliang Zhong and James T Kwok. Gradient descent with proximal average for nonconvex and composite regularization. In *The 31st AAAI Conference on Artificial Intelligence*, pages 2206–2212, 2014.