CS794/CO673: Optimization for Data Science

Lec 10: Accelerated Proximal Gradient

Yaoliang Yu

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS
DAVID R. CHERITON SCHOOL OF COMPUTER SCIENCE

October 21, 2022

Problem

Composite smooth minimization:

$$f_\star = \inf_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}), \quad \text{where} \quad f(\mathbf{w}) = \ell(\mathbf{w}) + r(\mathbf{w})$$

- ℓ: smooth and possibly nonconvex
- r: nonsmooth and possibly nonconvex
- The sum $f = \ell + r$ may not be smooth or convex
- Minimizer may or may not be attained
- Maximization is just negation

Problem

Composite smooth minimization:

$$f_\star = \inf_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}), \quad \text{where} \quad f(\mathbf{w}) = \ell(\mathbf{w}) + r(\mathbf{w})$$

- ℓ: smooth and possibly nonconvex
- r: nonsmooth and possibly nonconvex
- The sum $f = \ell + r$ may not be smooth or convex
- Minimizer may or may not be attained
- Maximization is just negation

Problem

Composite smooth minimization:

$$f_\star = \inf_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}), \quad \text{where} \quad f(\mathbf{w}) = \ell(\mathbf{w}) + r(\mathbf{w})$$

- ℓ: smooth and possibly nonconvex
- r: nonsmooth and possibly nonconvex
- The sum $f = \ell + r$ may not be smooth or convex
- Minimizer may or may not be attained
- Maximization is just negation

Problem

Composite smooth minimization:

$$f_\star = \inf_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}), \quad \text{where} \quad f(\mathbf{w}) = \ell(\mathbf{w}) + r(\mathbf{w})$$

- ℓ: smooth and possibly nonconvex
- r: nonsmooth and possibly nonconvex
- The sum $f = \ell + r$ may not be smooth or convex
- Minimizer may or may not be attained
- Maximization is just negation

Problem

Composite smooth minimization:

$$f_\star = \inf_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w}), \quad \text{where} \quad f(\mathbf{w}) = \ell(\mathbf{w}) + r(\mathbf{w})$$

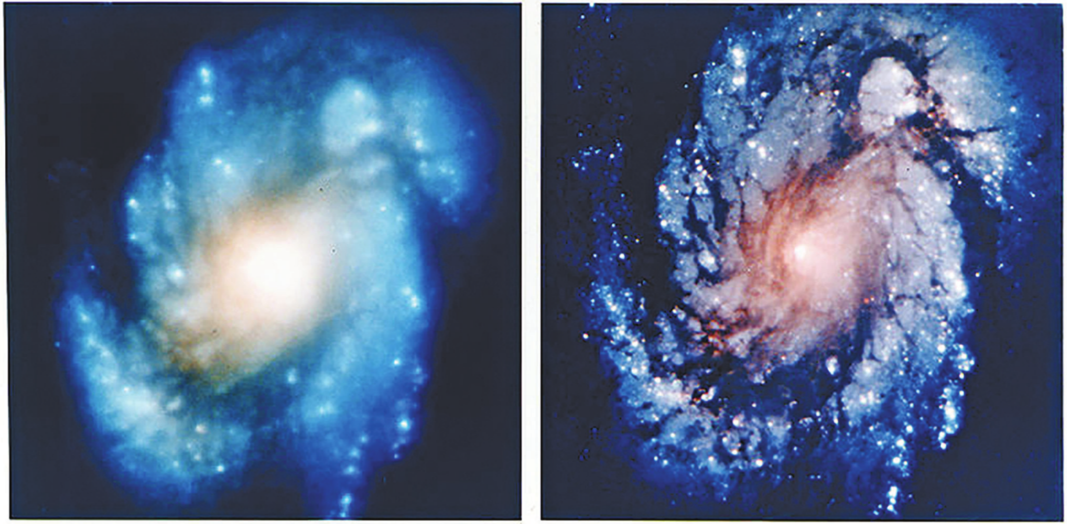
- ℓ: smooth and possibly nonconvex
- r: nonsmooth and possibly nonconvex
- The sum $f = \ell + r$ may not be smooth or convex
- Minimizer may or may not be attained
- Maximization is just negation

Problem

Composite smooth minimization:

$$f_\star = \inf_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}), \quad \text{where} \quad f(\mathbf{w}) = \ell(\mathbf{w}) + r(\mathbf{w})$$

- ℓ: smooth and possibly nonconvex
- r: nonsmooth and possibly nonconvex
- The sum $f = \ell + r$ may not be smooth or convex
- Minimizer may or may not be attained
- Maximization is just negation



Wide Field Planetary Camera 1 — Wide Field Planetary Camera 2

https://www.ams.org/journals/notices/202208/noti2534/

Sparsity

$$\min_{\mathbf{w}} \underbrace{\frac{1}{n}\|\mathbf{w}\mathbf{X}-\mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_0}_{r}$$

- Balancing square error with sparsity
- ℓ is convex and L-smooth, r is nonsmooth and nonconvex

$$\min_{\mathbf{w}} \underbrace{\frac{1}{n}\|\mathbf{w}\mathbf{X}-\mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_1}_{r}$$

- Convex relaxation: r is now convex but remains nonsmooth (crucial)

R. Tibshirani. "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1 (1996), pp. 267–288.

Sparsity

$$\min_{\mathbf{w}} \underbrace{\frac{1}{n}\|\mathbf{w}\mathbf{X} - \mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_0}_{r}$$

- Balancing square error with sparsity
- ℓ is convex and L-smooth, r is nonsmooth and nonconvex

$$\min_{\mathbf{w}} \underbrace{\frac{1}{n}\|\mathbf{w}\mathbf{X} - \mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_1}_{r}$$

- Convex relaxation: r is now convex but remains nonsmooth (crucial)

R. Tibshirani. "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1 (1996), pp. 267–288.

Sparsity

$$\min_{\mathbf{w}} \; \underbrace{\frac{1}{n}\|\mathbf{w}\mathbf{X} - \mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_0}_{r}$$

- Balancing square error with sparsity
- ℓ is convex and L-smooth, r is nonsmooth and nonconvex

$$\min_{\mathbf{w}} \; \underbrace{\frac{1}{n}\|\mathbf{w}\mathbf{X} - \mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_1}_{r}$$

- Convex relaxation: r is now convex but remains nonsmooth (crucial)

R. Tibshirani. "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1 (1996), pp. 267–288.

Sparsity

$$\min_{\mathbf{w}} \underbrace{\frac{1}{n}\|\mathbf{wX} - \mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_0}_{r}$$

- Balancing square error with sparsity
- ℓ is convex and L-smooth, r is nonsmooth and nonconvex

$$\min_{\mathbf{w}} \underbrace{\frac{1}{n}\|\mathbf{wX} - \mathbf{y}\|_2^2}_{\ell} + \underbrace{\lambda \cdot \|\mathbf{w}\|_1}_{r}$$

- Convex relaxation: r is now convex but remains nonsmooth (crucial)

R. Tibshirani. "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1 (1996), pp. 267–288.

Input: $\mathbf{w}_0 \in \mathbb{R}^d$, smooth function $\ell : \mathbb{R}^d \to \mathbb{R}$, $r : \mathbb{R}^d \to \mathbb{R}$

for $t = 0, 1, \ldots$ **do**

  $\mathbf{z}_t \leftarrow \mathbf{w}_t - \eta_t \cdot \nabla \ell(\mathbf{w}_t)$ // gradient step w.r.t. ℓ

  $\mathbf{w}_{t+1} \leftarrow \mathrm{P}_r^{\eta_t}(\mathbf{z}_t)$ // proximal step w.r.t. r

---

Input: $\mathbf{w}_0, \mathbf{b} \in \mathbb{R}^d$, $A \in \mathbb{S}^d_{++} \in [\sigma, \mathsf{L}]$, $\gamma_0 = 2$, $\kappa = \frac{\mathsf{L}}{\sigma}$

$\mathbf{g}_0 \leftarrow A\mathbf{w}_0 - \mathbf{b}$

$\mathbf{w}_1 \leftarrow \mathbf{w}_0 - \eta_0 \mathbf{g}_0$ // $\eta_t \equiv \frac{2}{\mathsf{L}+\sigma}$

for $t = 1, 2, \ldots$ **do**

  $\mathbf{g}_t \leftarrow A\mathbf{w}_t - \mathbf{b}$ // gradient

  $\gamma_t \leftarrow \frac{4(\kappa+1)^2}{4(\kappa+1)^2 - (\kappa-1)^2 \gamma_{t-1}}$ // γ_t is the momentum size

  $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \gamma_t \cdot \eta_t \mathbf{g}_t + (\gamma_t - 1)(\mathbf{w}_t - \mathbf{w}_{t-1})$ // $\eta_t \equiv \frac{2}{\mathsf{L}+\sigma}$

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned}\mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1-\beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1-\beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t))\end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned} \mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1-\beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1-\beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t)) \end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned} \mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1-\beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1-\beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t)) \end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned} \mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1-\beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1-\beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t)) \end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned} \mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1-\beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1-\beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t)) \end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned}\mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1-\beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1-\beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t))\end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Heavy Ball

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}} = \underbrace{(1+\beta_t)\mathbf{w}_t - \beta_t \mathbf{w}_{t-1}}_{\text{extrapolation}} - \eta_t \nabla f(\mathbf{w}_t)$$

- Typically $\mathbf{w}_1 = \mathbf{w}_0$ (so that at $t = 1$ we start with the usual gradient step)
- The underlying continuous analogue:

$$\begin{aligned} \mathbf{0} &= [(\mathbf{w}_{t+1} - \mathbf{w}_t) - (\mathbf{w}_t - \mathbf{w}_{t-1})] + (1 - \beta_t)(\mathbf{w}_t - \mathbf{w}_{t-1}) + \eta_t \nabla f(\mathbf{w}_t) \\ &\approx \ddot{\mathbf{w}}(t) + (1 - \beta_t)\dot{\mathbf{w}}(t) + \eta_t \nabla f(\mathbf{w}(t)) \end{aligned}$$

  - $\mathbf{w}(t)$ as the position of a heavy ball
  - $\dot{\mathbf{w}}(t)$ is the velocity; $\ddot{\mathbf{w}}(t)$ is the momentum
  - f acts as the potential energy
  - $\beta_t > 0$: extrapolation vs. $\beta_t < 0$:

B. T. Polyak. "Some methods of speeding up the convergence of iteration methods". *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5 (1964), pp. 791–803.

Nesterov's Momentum

- Simultaneous gradient update and extrapolation:

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}}$$

- Sequential gradient update and extrapolation:

$$\mathbf{z}_{t+1} = \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})$$
$$\mathbf{w}_{t+1} = \mathbf{z}_{t+1} - \eta_t \nabla f(\mathbf{z}_{t+1})$$

- Continuous analogue:

$$\ddot{\mathbf{w}}(t) + \frac{a}{t}\dot{\mathbf{w}}(t) + \nabla f\big(\mathbf{w}(t)\big) = \mathbf{0}$$

W. Su et al. "A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights". *Journal of Machine Learning Research*, vol. 17, no. 153 (2016), pp. 1–43.

Nesterov's Momentum

- Simultaneous gradient update and extrapolation:

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}}$$

- Sequential gradient update and extrapolation:

$$\mathbf{z}_{t+1} = \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})$$
$$\mathbf{w}_{t+1} = \mathbf{z}_{t+1} - \eta_t \nabla f(\mathbf{z}_{t+1})$$

- Continuous analogue:

$$\ddot{\mathbf{w}}(t) + \frac{a}{t}\dot{\mathbf{w}}(t) + \nabla f(\mathbf{w}(t)) = \mathbf{0}$$

W. Su et al. "A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights". *Journal of Machine Learning Research*, vol. 17, no. 153 (2016), pp. 1–43.

Nesterov's Momentum

- Simultaneous gradient update and extrapolation:

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}}$$

- Sequential gradient update and extrapolation:

$$\mathbf{z}_{t+1} = \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})$$
$$\mathbf{w}_{t+1} = \mathbf{z}_{t+1} - \eta_t \nabla f(\mathbf{z}_{t+1})$$

- Continuous analogue:

$$\ddot{\mathbf{w}}(t) + \frac{a}{t}\dot{\mathbf{w}}(t) + \nabla f(\mathbf{w}(t)) = \mathbf{0}$$

W. Su et al. "A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights". *Journal of Machine Learning Research*, vol. 17, no. 153 (2016), pp. 1–43.

Nesterov's Momentum

- Simultaneous gradient update and extrapolation:

$$\mathbf{w}_{t+1} = \underbrace{\mathbf{w}_t - \eta_t \nabla f(\mathbf{w}_t)}_{\text{gradient step}} + \underbrace{\beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})}_{\text{momentum}}$$

- Sequential gradient update and extrapolation:

$$\mathbf{z}_{t+1} = \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})$$
$$\mathbf{w}_{t+1} = \mathbf{z}_{t+1} - \eta_t \nabla f(\mathbf{z}_{t+1})$$

- Continuous analogue:

$$\ddot{\mathbf{w}}(t) + \frac{a}{t}\dot{\mathbf{w}}(t) + \nabla f\big(\mathbf{w}(t)\big) = \mathbf{0}$$

W. Su et al. "A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights". *Journal of Machine Learning Research*, vol. 17, no. 153 (2016), pp. 1–43.

Theorem: Optimal rate for Nesterov's momentum

Let $r = 0$ and ℓ be L-smooth convex. Then, with the momentum size choice

$$\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}, \quad \text{where} \quad \gamma_{t+1} = \frac{1 + \sqrt{1 + 4\gamma_t^2}}{2},$$

Nesterov algorithm satisfies:

$$f(\mathbf{w}_t) - f_\star \leq \frac{2L\|\mathbf{w}_0 - \mathbf{w}_\star\|_2^2}{\eta(t+2)^2},$$

where the constant step size $\eta \in (0, 1/L)$ and $\mathbf{w}_\star \in \operatorname{argmin} f$ with $f_\star = f(\mathbf{w}_\star)$.

Y. E. Nesterov. "A Method for Solving a Convex Programming Problem with Convergence Rate $O(1/k^2)$". *Soviet Mathematics Doklady*, vol. 27, no. 2 (1983), pp. 372–376.

Back to the Composite Problem

$$\min_{\mathbf{w}} \; \ell(\mathbf{w}) + r(\mathbf{w})$$

Algorithm 1: Accelerated Proximal Gradient, a.k.a. FISTA

Input: $\mathbf{w}_0 = \mathbf{z}_1, \gamma_1 = 1, \eta_0$

for $t = 1, 2, \ldots$ **do**

choose step size $\eta_t \leq \eta_{t-1}$ `// step size can only decrease`

$\mathbf{u}_t = \mathbf{z}_t - \eta_t \nabla \ell(\mathbf{z}_t)$ `// gradient step w.r.t.` ℓ

$\mathbf{w}_t = \mathrm{P}_r^{\eta_t}(\mathbf{u}_t) = \mathrm{argmin}_{\mathbf{u}} \frac{1}{2\eta_t} \|\mathbf{u}_t - \mathbf{u}\|_2^2 + r(\mathbf{u})$ `// proximal step w.r.t.` r

$\gamma_{t+1} = \frac{1+\sqrt{1+4\gamma_t^2}}{2}$

$\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ `// momentum size`

$\mathbf{z}_{t+1} = \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})$ `// extrapolation`

A. Beck and M. Teboulle. "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". *SIAM Journal on Imaging Sciences*, vol. 2, no. 1 (2009), pp. 183–202, Y. E. Nesterov. "Gradient Methods for Minimizing Composite Functions". *Mathematical Programming, Series B*, vol. 140 (2013), pp. 125–161.

Disccussions

- When $r \equiv 0$, FISTA reduces to the original algorithm of Nesterov.
- When $\gamma_1 = 1$, $\mathbf{w}_0$ does not really play any role: the first step is simply a proximal gradient step
- The smooth function ℓ needs to be defined over the entire space $\mathbb{R}^d$
- The proximal sequence $\mathbf{w}_t$ remains in $\operatorname{dom} r$ by construction
- The momentum choice $\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ is universal: given any sequence β_t,

$$\forall t \in [j, i] \text{ s.t. } \beta_t \neq 0, \quad \gamma_{t+1} = \frac{\gamma_t - 1}{\beta_t} = \frac{\gamma_j - 1 - \sum_{m=j}^{t-1} \prod_{k=j}^{m} \beta_k}{\prod_{k=j}^{t} \beta_k}.$$

In particular, the choice

$$\gamma_t = \frac{t + a - 2}{a - 1}, \quad \text{or equivalently} \quad \beta_t = \frac{t - 1}{t + a - 1}, \quad a \geq 3,$$

works equally well.

Disccussions

- When $r \equiv 0$, FISTA reduces to the original algorithm of Nesterov.
- When $\gamma_1 = 1$, $\mathbf{w}_0$ does not really play any role: the first step is simply a proximal gradient step
- The smooth function ℓ needs to be defined over the entire space $\mathbb{R}^d$
- The proximal sequence $\mathbf{w}_t$ remains in $\operatorname{dom} r$ by construction
- The momentum choice $\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ is universal: given any sequence β_t,

$$\forall t \in [j, i] \text{ s.t. } \beta_t \neq 0, \quad \gamma_{t+1} = \frac{\gamma_t - 1}{\beta_t} = \frac{\gamma_j - 1 - \sum_{m=j}^{t-1} \prod_{k=j}^{m} \beta_k}{\prod_{k=j}^{t} \beta_k}.$$

In particular, the choice

$$\gamma_t = \frac{t + a - 2}{a - 1}, \quad \text{or equivalently} \quad \beta_t = \frac{t - 1}{t + a - 1}, \quad a \geq 3,$$

works equally well.

Disccussions

- When $r \equiv 0$, FISTA reduces to the original algorithm of Nesterov.
- When $\gamma_1 = 1$, $\mathbf{w}_0$ does not really play any role: the first step is simply a proximal gradient step
- The smooth function ℓ needs to be defined over the entire space $\mathbb{R}^d$
- The proximal sequence $\mathbf{w}_t$ remains in $\mathrm{dom}\, r$ by construction
- The momentum choice $\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ is universal: given any sequence β_t,

$$\forall t \in [j, i] \text{ s.t. } \beta_t \neq 0, \quad \gamma_{t+1} = \frac{\gamma_t - 1}{\beta_t} = \frac{\gamma_j - 1 - \sum_{m=j}^{t-1} \prod_{k=j}^{m} \beta_k}{\prod_{k=j}^{t} \beta_k}.$$

In particular, the choice

$$\gamma_t = \frac{t + a - 2}{a - 1}, \quad \text{or equivalently} \quad \beta_t = \frac{t - 1}{t + a - 1}, \quad a \geq 3,$$

works equally well.

Disccussions

- When $r \equiv 0$, FISTA reduces to the original algorithm of Nesterov.
- When $\gamma_1 = 1$, $\mathbf{w}_0$ does not really play any role: the first step is simply a proximal gradient step
- The smooth function ℓ needs to be defined over the entire space $\mathbb{R}^d$
- The proximal sequence $\mathbf{w}_t$ remains in $\operatorname{dom} r$ by construction
- The momentum choice $\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ is universal: given any sequence β_t,

$$\forall t \in [j, i] \text{ s.t. } \beta_t \neq 0, \quad \gamma_{t+1} = \frac{\gamma_t - 1}{\beta_t} = \frac{\gamma_j - 1 - \sum_{m=j}^{t-1} \prod_{k=j}^{m} \beta_k}{\prod_{k=j}^{t} \beta_k}.$$

In particular, the choice

$$\gamma_t = \frac{t + a - 2}{a - 1}, \quad \text{or equivalently} \quad \beta_t = \frac{t - 1}{t + a - 1}, \quad a \geq 3,$$

works equally well.

Disccussions

- When $r \equiv 0$, FISTA reduces to the original algorithm of Nesterov.
- When $\gamma_1 = 1$, $\mathbf{w}_0$ does not really play any role: the first step is simply a proximal gradient step
- The smooth function ℓ needs to be defined over the entire space $\mathbb{R}^d$
- The proximal sequence $\mathbf{w}_t$ remains in $\operatorname{dom} r$ by construction
- The momentum choice $\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ is universal: given any sequence β_t,

$$\forall t \in [j, i] \text{ s.t. } \beta_t \neq 0, \quad \gamma_{t+1} = \frac{\gamma_t - 1}{\beta_t} = \frac{\gamma_j - 1 - \sum_{m=j}^{t-1} \prod_{k=j}^{m} \beta_k}{\prod_{k=j}^{t} \beta_k}.$$

In particular, the choice

$$\gamma_t = \frac{t + a - 2}{a - 1}, \quad \text{or equivalently} \quad \beta_t = \frac{t - 1}{t + a - 1}, \quad a \geq 3,$$

works equally well.

Disccussions

- When $r \equiv 0$, FISTA reduces to the original algorithm of Nesterov.
- When $\gamma_1 = 1$, $\mathbf{w}_0$ does not really play any role: the first step is simply a proximal gradient step
- The smooth function ℓ needs to be defined over the entire space $\mathbb{R}^d$
- The proximal sequence $\mathbf{w}_t$ remains in $\operatorname{dom} r$ by construction
- The momentum choice $\beta_t = \frac{\gamma_t - 1}{\gamma_{t+1}}$ is universal: given any sequence β_t,

$$\forall t \in [j, i] \text{ s.t. } \beta_t \neq 0, \quad \gamma_{t+1} = \frac{\gamma_t - 1}{\beta_t} = \frac{\gamma_j - 1 - \sum_{m=j}^{t-1} \prod_{k=j}^{m} \beta_k}{\prod_{k=j}^{t} \beta_k}.$$

In particular, the choice

$$\gamma_t = \frac{t + a - 2}{a - 1}, \quad \text{or equivalently} \quad \beta_t = \frac{t - 1}{t + a - 1}, \quad a \geq 3,$$

works equally well.

Theorem: Optimal rate for Nesterov's momentum

Suppose $\ell : \mathbb{R}^d \to \mathbb{R}$ is $\mathsf{L}^{[1]}$-smooth and convex, $r : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ is closed and convex, and $\eta_t \equiv \eta \le 1/\mathsf{L}^{[1]}$. Then, the proximal sequence $\{\mathbf{w}_t\}$ generated by FISTA satisfies: for all $\mathbf{w}$ and $t \ge 1$,

$$f(\mathbf{w}_t) \le f(\mathbf{w}) + \frac{\|\mathbf{w} - \mathbf{z}_1\|_2^2}{2\eta_t \gamma_t^2} \le f(\mathbf{w}) + \frac{2\,\|\mathbf{w} - \mathbf{z}_1\|_2^2}{\eta_t (t+1)^2}.$$

$$\frac{1}{2} + \gamma_t \le \frac{1 + 2\gamma_t}{2} \le \gamma_{t+1} \le \frac{1 + \sqrt{1 + 4\gamma_t^2}}{2} \le \frac{1 + 1 + 2\gamma_t}{2} = 1 + \gamma_t \implies$$

$$\tfrac{t-1}{2} + \gamma_1 \le \gamma_t \le t - 1 + \gamma_1$$

- FISTA is not monotonic: it could happen that $f(\mathbf{w}_{t+1}) > f(\mathbf{w}_t)$!

Theorem: Optimal rate for Nesterov's momentum

Suppose $\ell : \mathbb{R}^d \to \mathbb{R}$ is $\mathsf{L}^{[1]}$-smooth and convex, $r : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ is closed and convex, and $\eta_t \equiv \eta \leq 1/\mathsf{L}^{[1]}$. Then, the proximal sequence $\{\mathbf{w}_t\}$ generated by FISTA satisfies: for all $\mathbf{w}$ and $t \geq 1$,

$$f(\mathbf{w}_t) \leq f(\mathbf{w}) + \frac{\|\mathbf{w} - \mathbf{z}_1\|_2^2}{2\eta_t \gamma_t^2} \leq f(\mathbf{w}) + \frac{2\,\|\mathbf{w} - \mathbf{z}_1\|_2^2}{\eta_t (t+1)^2}.$$

$$\frac{1}{2} + \gamma_t \leq \frac{1 + 2\gamma_t}{2} \leq \gamma_{t+1} \leq \frac{1 + \sqrt{1 + 4\gamma_t^2}}{2} \leq \frac{1 + 1 + 2\gamma_t}{2} = 1 + \gamma_t \implies$$

$$\tfrac{t-1}{2} + \gamma_1 \leq \gamma_t \leq t - 1 + \gamma_1$$

- FISTA is not monotonic: it could happen that $f(\mathbf{w}_{t+1}) > f(\mathbf{w}_t)$!

Refinements

If we choose $\mathbf{w} \in \operatorname{argmin} f$, then we can make the following refinements:

- The extrapolation constants need only satisfy

$$\gamma_{t-1}^2 \geq \gamma_t^2 - \gamma_t.$$

  In particular, the choice for $\gamma_t = \frac{t+a-2}{a-1}, a \geq 3$ works and enjoys the same guarantee (with slightly worse constants).

- We can use Amijo's rule to adaptively choose η_t. However, the condition $\eta_t \leq \eta_{t-1}$ needs to be respected, meaning that each Amijo step should start with the step size from the previous iteration.

Refinements

If we choose $\mathbf{w} \in \operatorname{argmin} f$, then we can make the following refinements:

- The extrapolation constants need only satisfy

$$\gamma_{t-1}^2 \geq \gamma_t^2 - \gamma_t.$$

  In particular, the choice for $\gamma_t = \frac{t+a-2}{a-1}$, $a \geq 3$ works and enjoys the same guarantee (with slightly worse constants).

- We can use Amijo's rule to adaptively choose η_t. However, the condition $\eta_t \leq \eta_{t-1}$ needs to be respected, meaning that each Amijo step should start with the step size from the previous iteration.

Refinements

If we choose $\mathbf{w} \in \operatorname{argmin} f$, then we can make the following refinements:

- The extrapolation constants need only satisfy
$$\gamma_{t-1}^2 \geq \gamma_t^2 - \gamma_t.$$
In particular, the choice for $\gamma_t = \frac{t+a-2}{a-1}, a \geq 3$ works and enjoys the same guarantee (with slightly worse constants).

- We can use Amijo's rule to adaptively choose η_t. However, the condition $\eta_t \leq \eta_{t-1}$ needs to be respected, meaning that each Amijo step should start with the step size from the previous iteration.

Algorithm 2: Monotonic FISTA

Input: $\mathbf{w}_0 = \mathbf{z}_1, \gamma_1 = 1, \eta_0$

for $t = 1, 2, \ldots$ **do**

choose step size $\eta_t \le \eta_{t-1}$ `// step size can only decrease`

$\mathbf{u}_t = \mathbf{z}_t - \eta_t \nabla \ell(\mathbf{z}_t)$ `// gradient step w.r.t.` ℓ

$\tilde{\mathbf{w}}_t = \mathrm{P}_r^{\eta_t}(\mathbf{u}_t) = \operatorname{argmin}_{\mathbf{u}} \frac{1}{2\eta_t} \|\mathbf{u}_t - \mathbf{u}\|_2^2 + r(\mathbf{u})$ `// proximal step w.r.t.` r

choose $\mathbf{w}_t$ such that $f(\mathbf{w}_t) \le f(\tilde{\mathbf{w}}_t)$ `// local improvment`

$\gamma_{t+1} = \frac{1+\sqrt{1+4\gamma_t^2}}{2}$

$\mathbf{z}_{t+1} = \mathbf{w}_t + \frac{\gamma_t - 1}{\gamma_{t+1}}(\mathbf{w}_t - \mathbf{w}_{t-1}) + \frac{\gamma_t}{\gamma_{t+1}}(\tilde{\mathbf{w}}_t - \mathbf{w}_t)$ `// extrapolation`

- Can also restart the algorithm: roll back to the previous $\mathbf{w}_{t-1}$
  - does the algorithm simply repeat and get stuck?
  - what to do with γ_t?

A. Beck and M. Teboulle. "Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems". *IEEE Transactions on Image Processing*, vol. 18, no. 11 (2009), pp. 2419–2434.

Algorithm 3: Monotonic FISTA

Input: $\mathbf{w}_0 = \mathbf{z}_1, \gamma_1 = 1, \eta_0$

for $t = 1, 2, \ldots$ **do**

  choose step size $\eta_t \leq \eta_{t-1}$ // step size can only decrease

  $\mathbf{u}_t = \mathbf{z}_t - \eta_t \nabla \ell(\mathbf{z}_t)$ // gradient step w.r.t. ℓ

  $\tilde{\mathbf{w}}_t = \mathrm{P}_r^{\eta_t}(\mathbf{u}_t) = \mathrm{argmin}_{\mathbf{u}} \frac{1}{2\eta_t} \|\mathbf{u}_t - \mathbf{u}\|_2^2 + r(\mathbf{u})$ // proximal step w.r.t. r

  choose $\mathbf{w}_t$ such that $f(\mathbf{w}_t) \leq f(\tilde{\mathbf{w}}_t)$ // local improvment

  $\gamma_{t+1} = \frac{1+\sqrt{1+4\gamma_t^2}}{2}$

  $\mathbf{z}_{t+1} = \mathbf{w}_t + \frac{\gamma_t - 1}{\gamma_{t+1}}(\mathbf{w}_t - \mathbf{w}_{t-1}) + \frac{\gamma_t}{\gamma_{t+1}}(\tilde{\mathbf{w}}_t - \mathbf{w}_t)$ // extrapolation

- Can also restart the algorithm: roll back to the previous $\mathbf{w}_{t-1}$
  - does the algorithm simply repeat and get stuck?
  - what to do with γ_t?

A. Beck and M. Teboulle. "Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems". *IEEE Transactions on Image Processing*, vol. 18, no. 11 (2009), pp. 2419–2434.

Algorithm 4: Monotonic FISTA

Input: $\mathbf{w}_0 = \mathbf{z}_1, \gamma_1 = 1, \eta_0$

for $t = 1, 2, \ldots$ **do**

  choose step size $\eta_t \leq \eta_{t-1}$ // step size can only decrease

  $\mathbf{u}_t = \mathbf{z}_t - \eta_t \nabla \ell(\mathbf{z}_t)$ // gradient step w.r.t. ℓ

  $\tilde{\mathbf{w}}_t = \mathrm{P}_r^{\eta_t}(\mathbf{u}_t) = \mathrm{argmin}_{\mathbf{u}} \frac{1}{2\eta_t}\|\mathbf{u}_t - \mathbf{u}\|_2^2 + r(\mathbf{u})$ // proximal step w.r.t. r

  choose $\mathbf{w}_t$ such that $f(\mathbf{w}_t) \leq f(\tilde{\mathbf{w}}_t)$ // local improvment

  $\gamma_{t+1} = \frac{1+\sqrt{1+4\gamma_t^2}}{2}$

  $\mathbf{z}_{t+1} = \mathbf{w}_t + \frac{\gamma_t - 1}{\gamma_{t+1}}(\mathbf{w}_t - \mathbf{w}_{t-1}) + \frac{\gamma_t}{\gamma_{t+1}}(\tilde{\mathbf{w}}_t - \mathbf{w}_t)$ // extrapolation

- Can also restart the algorithm: roll back to the previous $\mathbf{w}_{t-1}$
  - does the algorithm simply repeat and get stuck?
  - what to do with γ_t?

A. Beck and M. Teboulle. "Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems". *IEEE Transactions on Image Processing*, vol. 18, no. 11 (2009), pp. 2419–2434.

Algorithm 5: Monotonic FISTA

Input: $\mathbf{w}_0 = \mathbf{z}_1, \gamma_1 = 1, \eta_0$

for $t = 1, 2, \ldots$ **do**

  choose step size $\eta_t \leq \eta_{t-1}$ `// step size can only decrease`

  $\mathbf{u}_t = \mathbf{z}_t - \eta_t \nabla \ell(\mathbf{z}_t)$ `// gradient step w.r.t.` ℓ

  $\tilde{\mathbf{w}}_t = \mathrm{P}_r^{\eta_t}(\mathbf{u}_t) = \mathrm{argmin}_{\mathbf{u}} \frac{1}{2\eta_t}\|\mathbf{u}_t - \mathbf{u}\|_2^2 + r(\mathbf{u})$ `// proximal step w.r.t.` r

  choose $\mathbf{w}_t$ such that $f(\mathbf{w}_t) \leq f(\tilde{\mathbf{w}}_t)$ `// local improvment`

  $\gamma_{t+1} = \frac{1+\sqrt{1+4\gamma_t^2}}{2}$

  $\mathbf{z}_{t+1} = \mathbf{w}_t + \frac{\gamma_t - 1}{\gamma_{t+1}}(\mathbf{w}_t - \mathbf{w}_{t-1}) + \frac{\gamma_t}{\gamma_{t+1}}(\tilde{\mathbf{w}}_t - \mathbf{w}_t)$ `// extrapolation`

- Can also restart the algorithm: roll back to the previous $\mathbf{w}_{t-1}$
  - does the algorithm simply repeat and get stuck?
  - what to do with γ_t?

A. Beck and M. Teboulle. "Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems". *IEEE Transactions on Image Processing*, vol. 18, no. 11 (2009), pp. 2419–2434.

Algorithm 6: Optimized gradient descent

Input: $\mathbf{w}_0 = \mathbf{z}_1, \gamma_1 = 1, \eta_0$

for $t = 1, 2, \ldots, T$ **do**

  choose step size $\eta_t \leq \eta_{t-1}$ `// step size can only decrease`

  $\mathbf{w}_t = \mathbf{z}_t - \eta_t \nabla \ell(\mathbf{z}_t)$ `// gradient step w.r.t.` ℓ

  $\gamma_{t+1} = \frac{1+\sqrt{1+4(1+[\![t=T]\!])\gamma_t^2}}{2}$

  $\mathbf{z}_{t+1} = \mathbf{w}_t + \frac{\gamma_t - 1}{\gamma_{t+1}}(\mathbf{w}_t - \mathbf{w}_{t-1}) + \frac{\gamma_t}{\gamma_{t+1}}(\mathbf{w}_t - \mathbf{z}_t)$ `// extrapolation`

$$f(\mathbf{z}_{T+1}) - f_\star \leq \frac{\|\mathbf{z}_1 - \mathbf{w}_\star\|_2^2}{2\eta\gamma_{T+1}^2} \leq \frac{\|\mathbf{z}_1 - \mathbf{w}_\star\|_2^2}{\eta(T+1)(T+1+\sqrt{2})}, \quad \eta_t \equiv \eta \leq 1/\mathsf{L}^{[1]}$$

$$f(\mathbf{w}_t) - f_\star \leq \frac{\|\mathbf{z}_1 - \mathbf{w}_\star\|_2^2}{4\eta\gamma_t^2} \leq \frac{\|\mathbf{z}_1 - \mathbf{w}_\star\|_2^2}{\eta(t+1)^2}$$

D. Kim and J. A. Fessler. "Optimized first-order methods for smooth convex minimization". *Mathematical Programming*, vol. 159 (2016), pp. 81–107, D. Kim and J. A. Fessler. "On the Convergence Analysis of the Optimized Gradient Method". *Journal of Optimization Theory and Applications*, vol. 172 (2017), pp. 187–205.

Algorithm 7: Proximal point algorithm for minimization

Input: $\mathbf{w}_0 \in \mathbb{R}^d$, function $f : \mathbb{R}^d \to \mathbb{R}$

for $t = 0, 1, \ldots$ **do**

  $\mathbf{w}_{t+1} \leftarrow \mathrm{P}_f^{\eta_t}(\mathbf{w}_t)$ `// ` η_t `is the step size`