Natural Gradient Descent

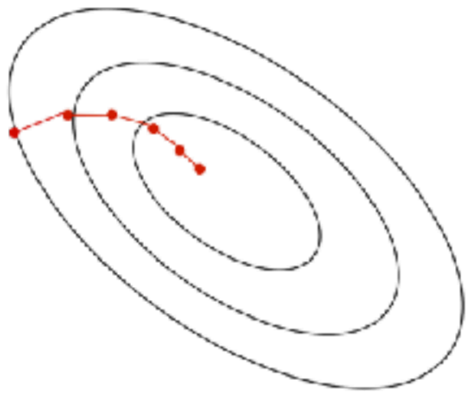
CMU 10-703 Recitation

Sep. 27, 2019

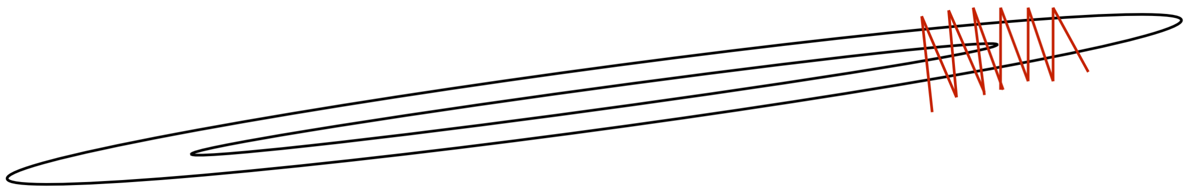
By Xingyu Lin

Issues with gradient descent

- When the curvature is ill conditioned, gradient descent will
  - bounce around in high curvature direction
  - make slow progress in low curvature direction



normal case



ill conditioned surface

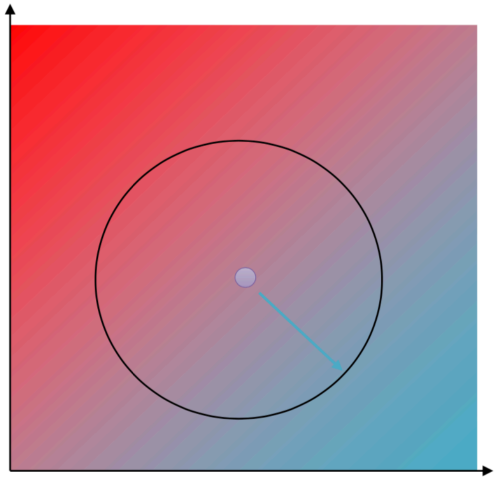
A different interpretation of gradient descent

- GD can be viewed as first linearizing the objective and then optimizing the objective under a constraint

$$\theta_{t+1} = \arg\min_{\theta} f(\theta_t) + \nabla f(\theta_t)^T(\theta - \theta_t)$$

$$\text{s.t.} \frac{1}{2}\|\theta - \theta_t\|_A^2 = \epsilon^2.$$

A-weighted norm $||x||_A = x^T A x$



- Solving the constraint optimization

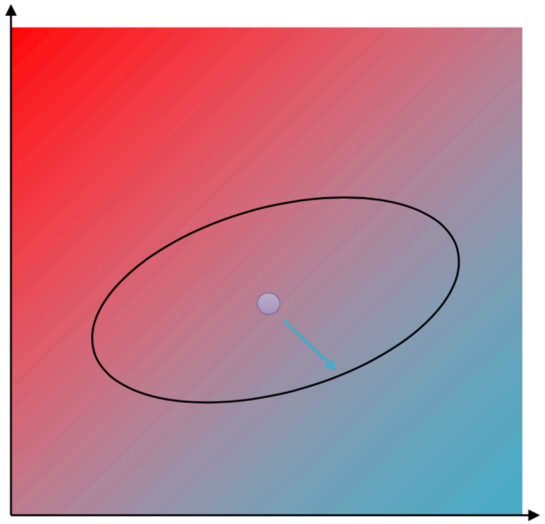
$$\theta_{t+1} = \theta_t - \frac{1}{\lambda} A^{-1} \nabla f(\theta_t).$$

Natural gradient

$$\theta_{t+1} = \arg\min_{\theta} f(\theta_t) + \nabla f(\theta_t)^T(\theta - \theta_t)$$
$$\text{s.t.} \frac{1}{2}\|\theta - \theta_t\|_A^2 = \epsilon^2.$$

- Assume that we are trying to optimize a probabilistic model $p(x;\theta_t)$
- We want to find a measure in the distribution space to constrain our optimization
- What measure to use?
  - KL divergence!

$$\theta_{t+1} = \arg\min_{\theta} f(\theta_t) + \nabla f(\theta_t)^T(\theta - \theta_t)$$
$$s.t. \frac{1}{2} KL(p(x;\theta_t)||p(x;\theta_t + \delta\theta)) \leq \epsilon^2$$



Fisher Information Matrix

$$\theta_{t+1} = \arg\min_{\theta} f(\theta_t) + \nabla f(\theta_t)^T (\theta - \theta_t)$$

$$s.t. \frac{1}{2} KL(p(x;\theta_t) || p(x;\theta_t + \delta\theta)) \leq \epsilon^2$$

- How to approximate a complex function with a quadratic function?
  - Taylor expansion!

$$\begin{aligned}
&\mathrm{KL}(\ p(x;\theta_t) \ \| \ p(x;\theta_t + \delta\theta) \ ) \\
&\quad \approx -\frac{1}{2}\delta\theta^T \left(\int p(x;\theta_t) \nabla^2 \log p(x;\theta_t) dx \right) \delta\theta \\
&\quad = -\frac{1}{2}\delta\theta^T \underbrace{\left(\int \nabla^2 p(x;\theta_t) dx \right)}_{=0} \delta\theta \\
&\qquad\quad + \frac{1}{2}\delta\theta^T \underbrace{\left(\int p(x;\theta_t) \left[\nabla \log p(x;\theta_t) \nabla \log p(x;\theta_t)^T \right] dx \right)}_{G(\theta_t)} \delta\theta.
\end{aligned}$$

Detailed derivation

$$
\begin{aligned}
&\frac{\partial^2}{\partial\theta_t^{(i)}\partial\theta_t^{(j)}}\left[\ \log p(x;\theta_t)\ \right] \\
&\qquad = \frac{\partial}{\partial\theta_t^{(i)}}\left(\frac{\frac{\partial}{\partial\theta_t^{(j)}}p(x;\theta_t)}{p(x;\theta_t)}\right) \\
&\qquad = \frac{p(x;\theta_t)\frac{\partial^2}{\partial\theta_t^{(i)}\partial\theta_t^{(j)}}p(x;\theta_t) - \frac{\partial}{\partial\theta_t^{(i)}}p(x;\theta_t)\frac{\partial}{\partial\theta_t^{(j)}}p(x;\theta_t)}{p(x;\theta_t)^2} \\
&\qquad = \frac{1}{p(x;\theta_t)}\frac{\partial^2}{\partial\theta_t^{(i)}\partial\theta_t^{(j)}}p(x;\theta_t) - \left(\frac{\frac{\partial}{\partial\theta_t^{(i)}}p(x;\theta_t)}{p(x;\theta_t)}\right)\left(\frac{\frac{\partial}{\partial\theta_t^{(j)}}p(x;\theta_t)}{p(x;\theta_t)}\right).
\end{aligned}
$$

$$
\nabla^2 \log p(x;\theta_t) = \frac{1}{p(x;\theta_t)}\nabla^2 p(x;\theta_t) - \nabla \log p(x;\theta_t)\nabla \log p(x;\theta_t)^T.
$$

Fisher Information Matrix

$$\theta_{t+1} = \arg\min_{\theta} f(\theta_t) + \nabla f(\theta_t)^T(\theta - \theta_t)$$

$$s.t. \frac{1}{2} KL(p(x;\theta_t)||p(x;\theta_t + \delta\theta)) \leq \epsilon^2$$

- How to approximate a complex function with a quadratic function?
  - Taylor expansion!

$$\begin{aligned}
&\mathrm{KL}(\ p(x;\theta_t)\ \|\ p(x;\theta_t+\delta\theta)\ ) \\
&\approx -\frac{1}{2}\delta\theta^T \left(\int p(x;\theta_t) \nabla^2 \log p(x;\theta_t) dx \right) \delta\theta \\
&= -\frac{1}{2}\delta\theta^T \underbrace{\left(\int \nabla^2 p(x;\theta_t) dx \right)}_{=0} \delta\theta \\
&\qquad + \frac{1}{2}\delta\theta^T \underbrace{\left(\int p(x;\theta_t) \left[\nabla \log p(x;\theta_t) \nabla \log p(x;\theta_t)^T \right] dx \right)}_{G(\theta_t)} \delta\theta.
\end{aligned}$$

$$\boxed{\theta_{t+1} = \theta_t - \eta_t G(\theta_t)^{-1} \nabla f(\theta_t).}$$

Natural Gradient Descent Algorithm

Algorithm: Natural Gradient Descent

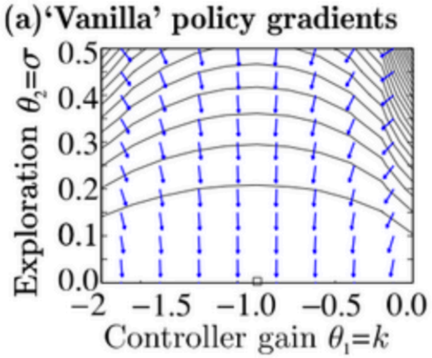
1. Repeat:
    1. Do forward pass on our model and compute loss $\mathcal{L}(\theta)$.
    2. Compute the gradient $\nabla_\theta \mathcal{L}(\theta)$.
    3. Compute the Fisher Information Matrix $\mathbf{F}$, or its empirical version (wrt. our training data).
    4. Compute the natural gradient $\tilde{\nabla}_\theta \mathcal{L}(\theta) = \mathbf{F}^{-1} \nabla_\theta \mathcal{L}(\theta)$.
    5. Update the parameter: $\theta = \theta - \alpha \tilde{\nabla}_\theta \mathcal{L}(\theta)$, where α is the learning rate.
2. Until convergence.

In practice, inverse of F is usually approximated

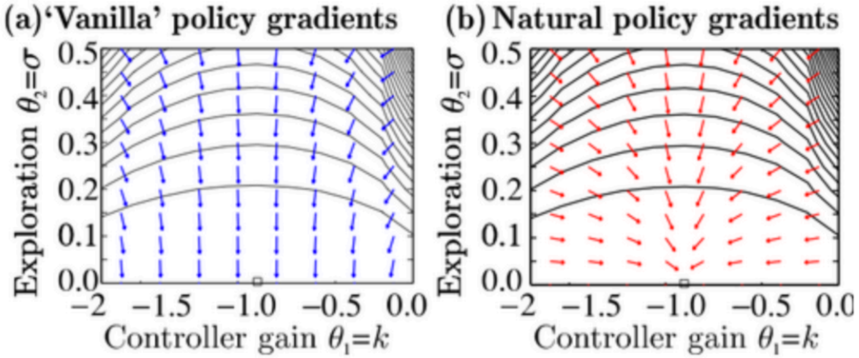
Is this a problem for RL?

$$r(\mathbf{s}_t, \mathbf{a}_t) = -\mathbf{s}_t^2 - \mathbf{a}_t^2$$

$$\log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t) = -\frac{1}{2\sigma^2}(k\mathbf{s}_t - \mathbf{a}_t)^2 + \text{const} \qquad \theta = (k, \sigma)$$

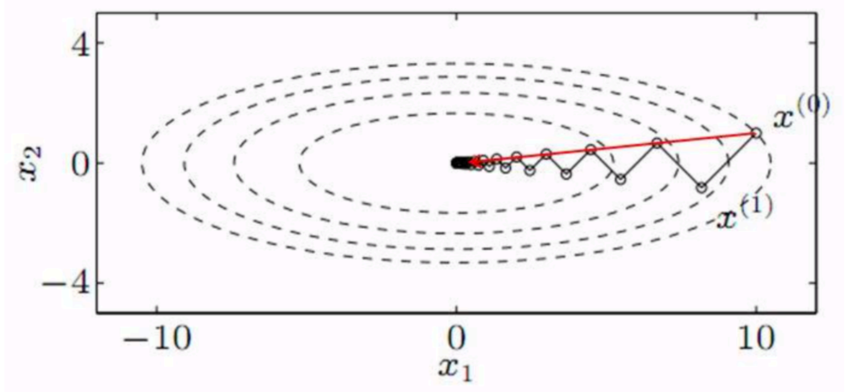


(image from Peters & Schaal 2008)



(figure from Peters & Schaal 2008)

Essentially the same problem as this:



From Sergey Levine's slide

Reference

- Information Geometry and Natural Gradients, Nathan Ratliff, 2013
- Natural Gradient Descent, Agustinus Kristiadi's Blog
- Berkeley CS285, Sergey Levine, Lecture 5

Questions?