Optimization With Theoretical Guarantees

2024年11月7日 星期四 13:01

Convexity

Definition: $f(\alpha x - (1-\alpha)y) \le \alpha f(x) + (1-\alpha) f(y)$ (No Assumption)

Alternative Notion:

1. $f(y) \ge f(x) + f^T(x)(y-x)$] All in Taylor theory (twice differentiable)
2. $\nabla^2 f(x) \succcurlyeq 0$, PSD (twice differentiable)
   or all eigenvalues are ≥ 0
3. $\nabla f(x)$ is monotone, $\langle \nabla f(x) - \nabla f(y), x-y \rangle \ge 0$ (once differentiable)

Convex set: $x \in C, y \in C \rightarrow \alpha x + (1-\alpha) y \in C$ (use in Constraint)

Convex function Convex set $\longrightarrow$ Local min = global min

Convex function $\nabla^2 f(x)$ always $\succcurlyeq 0$, as long as $\nabla f(x) = 0 \longrightarrow$ Local/global min

Gradient Descent Finds Optimality

Deriving GD Comes from ① Pick $\vec{v}$ Satisfy Taylor Theorem Such that $f(\vec{x} + \mu\vec{v}) - f(\vec{x}) < 0$
② Make Simplified assumption (Locally L-Smooth)

Gradient Descent

Taylor Theory: $f(\vec{y}) = f(\vec{x}) + \nabla f(\xi)^T (y-x)$

$f(\vec{x} + \mu\vec{v}) = f(\vec{x}) + \boxed{\nabla f(\xi)^T (\mu\vec{v})}$ Should be negative

equivalent $f(\vec{x} + \mu\vec{v}) - f(\vec{x})$ = negative

We want $f(\vec{x} + \mu\vec{v}) \le f(\vec{x})$, Let's make this happen Based on using Taylor theory

* Taylor theory Says the approximation of a different Point in function f

① $\vec{v} \in \mathbb{R}$ is descent direction at $\vec{x}$ when when $\vec{v} \cdot \nabla f|_x < 0$
This descent direction would hold true if $\vec{\mu}$ is small enough and f is Continuous
Then $\vec{v} \cdot \nabla f(\vec{x} + \vec{\mu}\vec{v})$ (new $\vec{x}$ point) follows Same direction
$\Rightarrow$ Always descent (Induction) $f(\vec{x} + \mu\vec{v})$ always $\le f(\vec{x})$

② Thus, let's ensure that $\vec{v} \cdot \nabla f(x)$ is < 0, this is when $\vec{v} = -\nabla f(x)$
and we move at $x^{(t+1)} = \vec{x}^{(t)} + \mu\vec{v}$

GD: $x^{(t+1)} = x^{(t)} - \mu^{(t)} \nabla f(x^{(t)})$

Educated Guess / Interpretation

We derived how we want to move ($\vec{v}$) Based on Wants to Satisfy Taylor Theorem such that $f(\vec{x} + \mu\vec{v}) - f(\vec{x}) < 0$

Deriving GD from Local Convexity + Traditional Calculus

We assume + Construct local Convexity by Armijo Property

$$f(z) = f(x^{(t)}) + \nabla f(x^{(t)})^T (z - x^{(t)}) + \frac{1}{2}(z - x^{(t)})^T \nabla^2 f(\xi)(z - x^{(t)})$$

For seen the future location by Taylor theory
We Construct our beliefs by Saying that Curvature $= 1/\mu$, Locally L-Smooth

Assume $\nabla^2 f(\xi) \longleftarrow \frac{1}{\mu} I$ and thus

$$g(z) = f(x^{(t)}) + \nabla f(x^{(t)})^T (z - x^{(t)}) + \frac{1}{2\mu} \| z - x^{(t)} \|^2$$

We Say $g(z)$ looks like $f(z)$, but importantly, it is Convex, let's work with this easier function

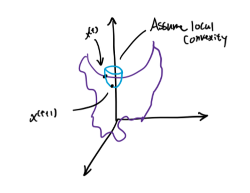
With this assumption, we can derive GD just like in Calculus $\longrightarrow \nabla_z g(z) = 0$

Global min Can be found

$\Rightarrow \nabla f(x^{(t)}) + \frac{1}{\mu}(z - x^{(t)}) = 0$

$\Rightarrow z^* = x^{(t)} - \mu \nabla f(x^{(t)})$

This is GD Algorithm, we make an educated guess Saying the function is Locally L-Smooth which hides away Complexity, then optimize on this Simple function
This feels much like EM



Optimality Guaranteed

Convex + Taylor Theory + GD Proofed

$f(x)$ Can't go Crazy — $\nabla f(x)$ Can't go Crazy

① L-Lipschitz: $|f(x) - f(y)| \le L\|x - y\|$
L-Lip bound Grad: $\|\nabla f(x)\| \le L$ (if f stands)

L-Lip Convergence guarantee: 1. Convex
2. $\|x^{(0)} - x^*\| < R$
3. T iteration
4. $\mu = \frac{R}{L\sqrt{T}}$

$\Rightarrow f(\frac{1}{T}\sum_{s=0}^{T-1} x^{(s)}) - f(x^*) \le \frac{RL}{\sqrt{T}}$

Average iteration errors bounded

② L-Smooth: $\| \nabla f(x) - \nabla f(y) \| \le L \| x - y \|$
L-Smooth bound Grad: $0 \le V^T \nabla^2 f(x) V \le L \quad \forall x \in \Omega$ and $V \in \mathbb{R}^n, \|V\| = 1$
↑ norm matrix

L-Smooth Convergence Guarantee: $f(x^{(t+1)}) \le f(x^{(t)}) - \frac{\mu}{2}\|\nabla f(x^{(t)})\|^2$ At each step decrease

L-Smooth Grad steps: $\| \nabla f(x^{(t)}) \| \le \sqrt{\frac{2(f(x^{(0)}) - f(x^*))}{\mu T}}$ At least one $x^{(t)}$ must satisfy this within T iterations

* Strong, no Convexity assumption
only differentiable + L-Smooth (root makes a huge differences)

③ If we have $f(x)$ as a strongly Convex and smooth

$$f(x^{(t)}) - f(x^*) \le (1 - \frac{c}{L})^t (f(x^{(0)}) - f(x^*))$$

Twitch Some Components from theoritical Perspective

Every time we twitch Something, it is not just adding onto GD, but rather rederiving everything from a novel perspective, then finding Connection to GD (Solve the Problem You want to solve)

Maintaining Local Convexity

How to pick μ So we don't miss out the optimality point

If we can make a Projection line, We Can maintain such local Convexity

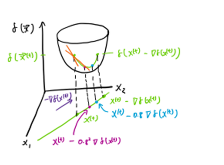
Armijo Condition: $f(x^{(t+1)}) = f(x^{(t)} - \mu^{(t)} \nabla f(x^{(t)})) \le f(x^{(t)}) - \mu^{(t)} \gamma \|\nabla f(x^{(t)})\|^2$

Convergence Guarantee: $f(x^{(t+1)}) - f(x^*) \le \frac{\|x^{(0)} - x^*\|^2}{2t \min(\mu^{(s)})}$

and $\min_{s=0\dots T}(\mu^{(s)}) = \min(1, \beta/L)$

Thus $f(x^{(t+1)}) - f(x^*) \le \frac{L}{2t\beta}\|x^{(0)} - x^*\|^2$

At each Step, error is bounded



Descent From Constraint's Perspective

We Can derive a different Version of GD With Norm in it

Taylor theorem first order expansion says that

$f(x^{(t+1)}) \approx f(x^{(t)}) + \nabla f(x)^T (x^{(t+1)} - x^{(t)})$

Since $x^{(t+1)} = x^{(t)} - \mu \nabla f(x)$

$\longrightarrow x^{(t+1)} - x^{(t)} = -\mu \nabla f(x)$

$\hookrightarrow f(x^{(t)}) + \nabla f(x)^T(-\mu \nabla f(x)) = f(x^{(t)}) - \mu \underbrace{\nabla f(x)^T \nabla f(x)}_{\text{This is dot product = norm}} \approx f(x^{(t)}) - \mu \|\nabla f(x)\|_2^2$

Thus, GD Can be seen like:

$$f(x^{(t+1)}) = f(x^{(t)}) - \mu \|\nabla f(x)\|_2^2$$

In addition, by adjusting the Projection of norm that GD uses, we have different Variants to GD

Norm forms a Constraint on the descent

$L_1 \rightarrow$ Sparse step direction (Coordinate descent) $\|\nabla f(x)\|_1^2$ $\quad \vec{P}$(Descent direction) $= -\operatorname{sign}(\frac{\partial f}{\partial x_j^*}) \begin{pmatrix} 0 \\ 0 \\ \vdots \\ f \\ \vdots \end{pmatrix}$ ← jth entry

$L_\infty \rightarrow$ Uniform step in each direction $\|\nabla f(x)\|_\infty^2$ $\quad \vec{P} = \|\nabla f(x^{(t)})\|_1 \begin{bmatrix} \operatorname{Sign}(\nabla f_1) \\ \vdots \\ \operatorname{Sign}(\nabla f_n) \end{bmatrix}$

Generally Projection: $x^{(t+1)} = x^{(t)} - \mu \nabla f(x^{(t)})$ S.t. $x \in \Omega$

$y^{(t+1)} = x^{(t)} - \mu \nabla f(x^{(t)})$

then $x^{(t+1)} = \pi_\Omega(y^{(t+1)}) \longleftarrow x^{(t+1)} = \operatorname{argmin}_{z \in \Omega} \| y^{(t+1)} - z \|$

Newton's Method

Using different way to derive GD

We used $f(x) \approx f(x^{(t)}) + \nabla f(x^{(t)})(x - x^{(t)})$ as deriving GD

now let's say $f(x) \approx f(x^{(t)}) + \nabla f(x^{(t)})(x - x^{(t)}) + \frac{1}{2}(x - x^{(t)}) \nabla^2 f(x^{(t)})(x - x^{(t)})$

Different from GD Derivation, let's set this to 0 directly with assumption that $\nabla f(x^*) = 0$

We will get a new method called Newton's method where

$$x^{(t+1)} = x^{(t)} - [\nabla^2 f(x^{(t)})]^{-1} \nabla f(x^{(t)})$$

Theoritical Suppose: ① $\|\nabla^2 f(x)^{-1}\| \le \frac{1}{h} \quad \forall h > 0 \longrightarrow |\lambda_{min}(A)| \ge h \longrightarrow$ ensure Inverse exist
② $\|\nabla^2 f(x) - \nabla^2 f(x^*)\| \le L\|x - x^*\| \longrightarrow$ Hessian bounded

Gaurantees: ① $\|x^{(t)} - x^*\| \le \frac{2h}{3L} \quad \forall t$
② $\|x^{(t+1)} - x^*\| \le \frac{3L}{2h}\|x^{(t)} - x^*\|^2$

① Converge exponentially fast if Converge
② Non-Contrastive mapping
$\|x^{(t)} - x^*\|$ not garanteed ≤ 1

Alternative Problem: $\underset{\substack{A \\ \text{matrix}}}{\nabla^2 f(x^{(t)})}\underset{\substack{\text{Unknown } x \\ \text{Vector}}}{(x - x^{(t)})} = \underset{\substack{b \\ \text{Vector}}}{-\nabla f(x^{(t)})} \Longrightarrow$ Solving $Ax + b = 0$

Inverting matrix A has a extremely high Computation Cost:

Quasi-Newton: Construct $\nabla^2 f(x^{(t)}) \approx B^{(t)}$ where $B^{(t)}$ remains Key insight (Dynamic learning rate per Variable)
but Computationally Cheap and Invertible

ADAM $\quad B^{(t)} = \operatorname{diag}|\nabla f(x^{(t)})_i|$

general $= \sqrt{\operatorname{diag}[J_{(x)} J_{(x)}^T]}$

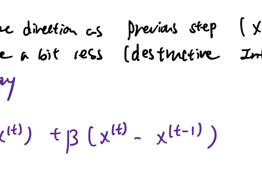
Momentum and Nestrov Acceleration

Momentum: If $-\nabla f(x^{(t)})$ is in the same direction as previous step $(x^{(t)} - x^{(t-1)})$, then move a bit further (Constructive Interference)
If opposite direction, move a bit less (destructive Interference)

Injecting Markovian memory

$$x^{(t+1)} = x^{(t)} - \mu \nabla f(x^{(t)}) + \beta(x^{(t)} - x^{(t-1)})$$

Smooth out noises in GD



Convergence property analyzed from Control's stability perspective

$$\begin{bmatrix} x^{(t+1)} \\ x^{(t)} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 - \mu\lambda + \beta & -\beta \\ 1 & 0 \end{bmatrix}}_{M} \begin{bmatrix} x^{(t)} \\ x^{(t-1)} \end{bmatrix}$$

← Use this to look at the eigenvalue thus deducing stability of GD w/m

For quadratic problems like $\frac{1}{2}x^T A x$, GD + M Converges to x^* at rate of $(\frac{k-1}{k+1})^t$ where $k = \frac{\lambda_{max}}{\lambda_{min}}$

Nesterov Acceleration: Going a little bit more first, then take the gradient

$y^{(t+1)} = x^{(t)} + \beta(x^{(t)} - x^{(t-1)})$

$x^{(t+1)} = y^{(t+1)} - \mu \nabla f(y^{(t+1)})$

N.A. is a less intuitive version of momentum, but it Can go into Continuous space and model as differential equation

$$\underset{\text{GD}}{\left(\frac{k+1}{k+1}\right)^t} \longrightarrow \underset{\text{GD+M}}{\left(\frac{\sqrt{k}-1}{\sqrt{k}+1}\right)^t} \longrightarrow \underset{\text{N.A.}}{\left(\sqrt{\frac{\sqrt{k}-1}{\sqrt{k}}}\right)^t}$$

For $\min_x \phi(x)$ where $\phi(x) = \frac{1}{2}x^T A x$

Conjugate Gradient Descent

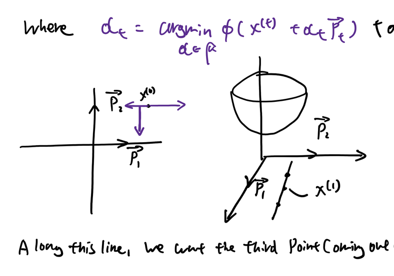
Reformulating Again, Solve $Ax^* - b = 0$, $\nabla \phi(x) = Ax - b$

Question: Can we ① Not invert A (expensive + unstable)
② Do it step wise

Conjugate (general notion of orthogonality): $P_i^T A P_j = 0 \quad \forall i \ne j$, $\{P_1 \dots P_n\}$ is the Conjugate of A (PD)
↑ $\langle P_i, P_j \rangle_A$

Maybe $x^{(t+1)} = x^{(t)} + \alpha_t \vec{P}_t$, we just step in 1 orthogonal direction at a time $\longrightarrow$ Once we update in one direction, it's done for operation, we don't need think about it

Where $\alpha_t = \operatorname{argmin}_{\alpha \in \mathbb{R}} \phi(x^{(t)} + \alpha_t \vec{P}_t)$ to ensure descent



Pick predefined direction,
Sort of like Coordinate descent, just Coordinate descent pick max direction

A long this line, we want the third Point Coming out of the Page to be minimized

Let's Solve this easier Problem $\alpha_t = a^* = \frac{(b - Ax^{(t)})^T P_t}{P_t^T A P_t}$

Think about what this is: $b - Ax^{(t)} = -\phi(x^{(t)})$, then $\alpha_t = \frac{-\nabla\phi(x^{(t)})^T P_t}{P_t^T A P_t}$

Almost like a Gradient descent Scheme
Step we ends up taking is just how P_t Project on our gradient. Take the Gradient In the Component direction of P_t

How to find P_t then? $P_i^T A P_j = 0$ is expensive to Calculate
Need to dynamically Choose Such P_t
Compute P_t Using only P_{t-1} (next direction from only Previous direction). Throw away all P_0 to P_{t-2}

Like Bellman Equation idea, if it Can be Conjugate to the Previous Vector and keep doing so, all are Conjugate

Let's Start with:

$$P_t = -\nabla\phi(x^{(t)}) + \beta_t P_{t-1}$$

Pick a new direction based on gradient and previous direction

$$P_{t-1}^T A P_t = -P_{t-1}^T A \nabla\phi(x^{(t)}) + \beta_t P_{t-1}^T A P_{t-1}$$

Make this zero, then P_{t-1}^T and P_t is Conjugate

This makes $\beta_t = \frac{P_{t-1}^T A \nabla\phi(x^{(t)})}{P_{t-1}^T A P_{t-1}}$ to Satisfy $P_{t-1}^T A P_t = 0$

Start with $\beta_0 = 0$ and we pick a P_0 that we don't Care what it is, then we Conjugate at the next iteration

CGD give strong Convergence

① For $\phi(x)$ as quadratic, for any $x^{(0)}$, the set of step $\{x^{(t)}\}$ from CGD Converge to x^* (Solution) in at most n steps where $A \in \mathbb{R}^{n \times n}$

② $\|x^{(t+1)} - x^*\|_A^2 \le \left(\frac{\lambda_{n-t} - \lambda_1}{\lambda_{n-t} + \lambda_1}\right)^2 \|x^{(0)} - x^*\|$

At every step pick up this factor that is related to the eigenvalue

$\|x^{(t)} - x^*\| \le 2\left(\frac{\sqrt{k}-1}{\sqrt{k}+1}\right)^t \|x^{(0)} - x^*\|_A$

Similar Convergence Property to GD + M

Theoretical Boosting: Strongly Convex

Properties of function that Can make many things easier and generalizable

Strongly Convex

Swap out $\nabla^2 f(x)$ in Taylor theorem with Smallest Eigenvalue $\nabla^2 f(\xi) \ge CI$

$f(y) \ge f(x) + \nabla f(x)^T (y - x) + \frac{c}{2}\|x - y\|^2$ (Strongly Convex) $\quad 0 < c \le \lambda_{min}(\nabla^2 f(\xi))$

Instead of tangent line, we actually have a tangent Curve
Size of the gradient tells you how close you are to the x^* when your Condition is under Strongly Convex, and the relation is Squared

$$\Rightarrow f(x) - f(x^*) \le \frac{\|\nabla f(x)\|^2}{2c}$$

Strongly Convex Yields very strong Convergence rate

$$f(x^{(t+1)}) - f(x^*) \le (1 - \frac{c}{L})(f(x^{(t)}) - f(x^*)) = \underset{\text{Exponential!}}{(1 - \frac{c}{L})^t} (f(x^{(0)}) - f(x^*))$$

When GD + ① Strong Convexity
② L-Smooth $\|\nabla^2 f(x)\| \le L$
③ $\mu_t = 1/L$

Strongest Power of Strongly Convex is that with this Condition Satisfied, all Previous method that we need quadratic format Can generalize to any equation satisfying strongly Convex

If $f(w)$ is Convex and $R(w)$ is C-strongly Convex, then $f(w) + R(w)$ is C-strongly Convex
This give room for regularizors to show its power (i.e. Ride regression)

PL Condition

$f: \mathbb{R}^n \to \mathbb{R}$ Satisfies μ-PL-Condition if $\forall x \in \mathbb{R}^n$

$\frac{1}{2}\|\nabla f(x)\|_2^2 \ge \mu(f(x) - f(x^*))$ Looks like Strong Convexity!
When far from x^*, the gradient is big

SC $\rightarrow$ PL, PL $\not\rightarrow$ SC

① PL-Condition Can hold for non-Convex functions
Acts as a Stand in for non-Strongly-Convex functions

② Importantly, if $f(x)$ is L-Smooth and is μ-PL-Condition, then gradient descent with a step-size of $\eta = 1/L$ Converges at a rate of

$$f(x^{(t)}) - f(x^*) \le (1 - \frac{\mu}{L})^t (f(x^{(0)}) - f(x^*))$$

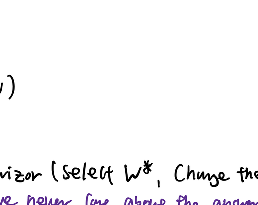
exponentially fast, just like Strong Convex

Notice that this does not need Convex assumption, just L-Smooth + μ-PL

With over Parametrized Neural Network, we Can write the Neural Tangent Kernel as

$\|\nabla L(w)\|^2 \ge 2\mu L(w)$ NN is PL-condition

And Neural network Converges at a rate of $(1 - \frac{\mu}{\beta})^t$



OverParametrized at start to Converge is exponentially fast.
With PL-Condition, even for non-Convex function, all previous analysis Can be done

Data Science is about Constraint

$$\min_{w \in \mathbb{R}^p} \frac{1}{N}\sum_{i=1}^{N} f((x_i, y_i); w) + \lambda R(w)$$

Model Parameter

Loss Function

Training Data

Regularizor (Select W^*, Change the answer)
- We never Care about the answer, how about Changing it to Something more reasonable, maybe with many 0, maybe W^* is Small

For non-Convex problem, there are many W^*, regularizor put a weight on the W^* with some requirement

$\rightarrow$ Small $\|.\|_2$
$\rightarrow$ Small $\|.\|_1$
$\rightarrow$ Small Entropy

More importantly, we Can Reframe Constraint optimization into regularized unconstraint Problem

Original Constrain (Correctness) $\longrightarrow$ Optimization Target f (Hinge Loss)
Origin Optimization Target (margin) $\longrightarrow$ Constrain

$\min_{w,b} \frac{1}{2}\|w\|_2^2$ Subject to Constraint of Correctness of labels

Now the Constraint is a function, the whole Soft margin SVM Can be framed as:

$$\min_{w,b} \frac{1}{N}\sum_{i=1}^{N} \underbrace{(1 - y_i(\langle w, x_i \rangle + b))_+}_{f(w;x)} + \frac{\lambda}{2}\|w\|_2^2$$

Convex Function $\longrightarrow$ every where you go $\nabla_w^2 f(w;x) = 0$ PSD, no more W term at 2nd derivative