ECE 901: Large-scale Machine Learning and Optimization **Fall 2016**

Lecture 12 — 10/13

Lecturer: Dimitris Papailiopoulos *Scribe: Yunyang Xiong*

12.1 SGD for Nonconvex Problems and Overparameterized NNs

12.1.1 Motivation

Deep neural network is a very powerful tool in many areas, like computer vision, artifial intelligence and so on so forth. SGD is widely used in deep neural network. Obviously, We want to know if SGD is convergent for solving nonconvex optimization problems and if overparameterization will help neural network avoid bad local minima?

12.1.2 Convergence of SGD for β-smooth nonconvex function

We will investigate the convergence result of SGD for β-smooth nonconvex function.

Theorem 12.1. *If function is β-smooth nonconvex and the gradient of $f(x)$ is bounded, $E||\nabla f_{sk}(x_k)|| = ||\nabla f_k(x_k)|| \leq \mu$, then*

$$E||\nabla f(x_k)|| \to 0, k \to \infty \tag{12.1}$$

[Ghadimi and Lan(2013)]

Proof: In SGD setting, we have

$$x_{k+1} = x_k - \lambda \nabla f_{sk}(x_k)$$

With respect to β-smooth nonconvex function, we know

$$f(x) \leq f(x0) + < \nabla f(x), x - x0 > + \frac{\beta}{2}||x - x_0||^2$$

Thus, we get

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + < \nabla f(x_k), x_{k+1} - x_k > + \beta/2||x_{k+1} - x_k||^2 \\ &= f_k - \lambda < \nabla f(x_k), \nabla f_{sk}(x_k) > + \beta/2||x_{k+1} - x_k||^2 \end{aligned}$$

The gradient of $f(x)$ is bounded. Then we have

$$\begin{aligned} Ef_{k+1} &\leq Ef_k - \lambda E||\nabla f_s k(x_k)||^2 + \frac{\beta}{2}\lambda^2\mu^2 \\ E||\nabla f(x_k)||^2 &\leq \frac{E(f_k - f_{k+1}}{\lambda} + \frac{\beta}{2}\lambda\mu^2 \end{aligned}$$

Summing T iterations, it leads to,

$$\begin{aligned}
&\sum_{i=1}^{T} E||\nabla f(x_i)||^2 \leq \sum_{i=1}^{T} \frac{1}{\lambda} E(f_i - f_{i+1}) + \frac{T\beta}{2}\lambda\mu^2 \\
&= \frac{f_1 - f_{T+1}}{\lambda} + \frac{T\lambda\beta\mu^2}{2} \\
&\leq \frac{f_1 - f^\star}{\lambda} + \frac{T\lambda\beta\mu^2}{2}
\end{aligned}$$

Then we have,

$$\begin{aligned}
T \min_{1\leq i\leq T} E||\nabla f(x_i)||^2 &\leq \frac{f_1 - f^\star}{\lambda} + \frac{T\lambda\beta\mu^2}{2} \\
\min_{1\leq i\leq T} E||\nabla f(x_i)||^2 &\leq \frac{f_1 - f^\star}{\lambda T} + \frac{\lambda\beta\mu^2}{2} \\
\min_{1\leq i\leq T} E||\nabla f(x_i)||^2 &\leq \sqrt{\frac{(f_1 - f^*)\beta\mu^2}{2T}}
\end{aligned}$$

Therefore, when $T \to \infty$, it leads to

$$E||\nabla f(x_k)|| \to 0, k \to \infty$$

□

12.1.3 Does overparameterizaiton help avoid bad local minima?

We will investigate overparameterization for neural networks.

Theorem 12.2. *If the last layer of neural networks has more activation nodes than samples, then training error* $= 0$.

[Soudry and Carmon(2016)]

Proof: Define dataset $S = (x_1, y_1), \cdots, (x_n, y_n)$, $X = [x_1, \cdots, x_n] \in R^{d\times n}$, Leaky ReLu activation function is used,

$$\sigma(a) = \begin{cases} a, a \geq 0 \\ s * a, a < 0 \end{cases}$$

where s is a small positive number. $u_l^i =$ input of $l-th$ layer, $v_l^i =$ output of $l-th$ layer. Thus, we have

$$\begin{aligned}
u_l^i &= w_l \times v_{l-1} \\
v_l^i &= \sigma(u_l^i)
\end{aligned} \tag{12.2}$$

The loss function of this neural network is,

$$\min_{\omega} \frac{1}{n} \sum_{i=1}^{n} (h_{l-1}^{i} \cdot \omega - y_i)^2 \tag{12.3}$$

Take derivative of loss function with respect to w,

$$\begin{aligned} &\nabla_w \frac{1}{n} \sum_{i=1}^{n} (y_i - h(\omega_i x_i))^2 \\ &\iff h(wx_i) = y_i \end{aligned} \tag{12.4}$$

We only consider a NN with one single hidden layer,

$$\begin{aligned} e_i &= y_i - w_2^T \sigma(w_1 x) = y_i - w_2^T diag(a_i) w_1 x \\ e_i &= y_i - a_i^T diag(w_2) w_1 x \end{aligned} \tag{12.5}$$

Take derivative of $\sum_{i=1}^{n} e_i^2$ with respect to w,

$$\begin{aligned} \nabla_w \sum_{i=1}^{n} e_i^2 &= \sum_i 2 e_i \frac{de_i}{dw} \\ &= \sum_i 2 e_i [a_i x_i^T]_{d_1 \times d} = 0_{d_i \times d} \end{aligned} \tag{12.6}$$

Therefore, $\sum_i e_i(a_i) \otimes x_i = 0_{d_1 \times d}$, it means that $[a_1 \otimes x_1, \cdots a_n \otimes x_n] e_i = 0$. Define

$$G = \begin{bmatrix} a_1^1 x_1, \ldots, a_n^1 x_n \\ \ldots, \ldots, \ldots \\ a_1^d x_1, \ldots, a_n^d x_n \end{bmatrix}$$

Then $Ge = 0, e = [e_1, \cdots, e_n]$. If $d \times d_1 \geq n$, then matrix G is full rank, the null space of G is empty. Therefore the training error will be 0. □

Bibliography

[Ghadimi and Lan(2013)] Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[Soudry and Carmon(2016)] Daniel Soudry and Yair Carmon. No bad local minima: Data independent training error guarantees for multilayer neural networks. *arXiv preprint arXiv:1605.08361*, 2016.