Machine Learning and Neural Networks (MATH3431) Epiphany term, 2023

Handout 3: Stochastic gradient descent

Lecturer & author: Georgios P. Karagiannis georgios.karagiannis@durham.ac.uk

Aim. To introduce the stochastic gradient descent (motivation, description, practical tricks, analysis in the convex scenario, and implementation).

Reading list & references:

- Shalev-Shwartz, S., & Ben-David, S. (2014). Understanding machine learning: From theory to algorithms. Cambridge university press.
- Bottou, L. (2012). Stochastic gradient descent tricks. In Neural networks: Tricks of the trade (pp. 421-436). Springer, Berlin, Heidelberg.

This is under development, it is subject to minor changes according to the Lecture, and it will be finalized around 1 day after the Lecture. It is given as guide before the lecture.

1. Motivations for stochastic gradient descent

Problem 1. Consider a learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$. Learning may involve the computation of the minimizer $w^* \in \mathcal{H}$, where $\mathcal{H}$ is a class of hypotheses, of the risk function (RF) $R(w) = \mathrm{E}_{z\sim g}(\ell(w, z))$ given an unknown data generating model $g(\cdot)$ and using a known tractable loss $\ell(\cdot, \cdot)$; that is

$$w^* = \arg\min_{\forall w \in \mathcal{H}} (R_g(w)) = \arg\min_{\forall w \in \mathcal{H}} (\mathrm{E}_{z\sim g}(\ell(w, z))) \tag{1.1}$$

Remark 2. Gradient descent (GD) cannot be directly utilized to address Problem 1 (i.e., minimize the Risk function) because g is unknown, and because (1.1) involves an integral which may be computationally intractable. Instead it aims to minimize the ERF $\hat{R}(w) = \frac{1}{n}\sum_{i=1}^{n} \ell(w, z_i)$ which ideally is used as a proxy when data size n is big (big-data).

Remark 3. The implementation of GD may be computationally impractical even in problems where we need to minimize an ERF $\hat{R}_n(w)$ if we have big data ($n \approx$big). This is because GD requires the recursive computation of the exact gradient $\nabla \hat{R}_n(w) = \frac{1}{n}\sum_{i=1}^{n} \nabla \ell(w, z_i)$ using all the data $\{z_i\}$ at each iteration. That may be too slow.

Remark 4. Stochastic gradient descent (SGD) aims at solving (1.1), and overcoming the issues in Remarks 2 & 3 by using an unbiased estimator of the actual gradient (or some sub-gradient) based on a sample properly drawn from g.

2. Stochastic gradient descent

2.1. Description.

Notation 5. For the sake of notation simplicity and generalization, we present Stochastic Gradient Descent (SGD) in the following minimization problem

$$w^{*}=\arg\min_{\forall w\in\mathcal{H}}\left(f\left(w\right)\right) \tag{2.1}$$

where here $f:\mathbb{R}^{d}\rightarrow\mathbb{R}$, and $w\in\mathcal{H}\subseteq\mathbb{R}^{d}$; $f\left(\cdot\right)$ is the unknown function to be minimized, e.g., $f\left(\cdot\right)$ can be the risk function $R_{g}\left(w\right)=\mathrm{E}_{z\sim g}\left(\ell\left(w,z\right)\right)$.

Algorithm 6. *Stochastic Gradient Descent (SGD) with learning rate $\eta_{t}>0$ for the solution of the minimization problem (2.1)*

For $t=1,2,3,...$ iterate:

(1) *compute*

$$w^{(t+1)}=w^{(t)}-\eta_{t}v_{t}, \tag{2.2}$$

where v_{t} is a random vector such that $E\left(v_{t}|w^{(t)}\right)\in\partial f\left(w^{(t)}\right)$

(2) *terminate if a termination criterion is satisfied, e.g.*

$$\textit{If }t\geq T\ \textit{ then }\ \textit{STOP}$$

Remark 7. If f is differentiable at $w^{(t)}$, it is $\partial f\left(w^{(t)}\right)=\left\{ \nabla f\left(w^{(t)}\right)\right\}$. Hence v_{t} is such as $\mathrm{E}\left(v_{t}|w^{(t)}\right)=\nabla f\left(w^{(t)}\right)$ in Algorithm 6 step 1.

Note 8. Assume f is differentiable (for simplicity). To compare SGD with GD, we can re-write (2.2) in the SGD Algorithm 6 as

$$w^{(t+1)}=w^{(t)}-\eta_{t}\left[\nabla f\left(w^{(t)}\right)+\xi_{t}\right], \tag{2.3}$$

where

$$\xi_{t}:=v_{t}-\nabla f\left(w^{(t)}\right)$$

represents the (observed) noise introduced in (2.2) by using a random realization of the exact gradient.

Remark 9. Given T SGD algorithm iterations, the output of SGD can be (but not a exclusively)

(1) the average (after discarding the first few iterations of $w^{(t)}$ for stability reasons)

$$w_{\text{SGD}}^{(T)}=\frac{1}{T}\sum_{t=1}^{T}w^{(t)} \tag{2.4}$$

(2) or the best value discovered

$$w_{\text{SGD}}^{(T)}=\arg\min_{\forall w_{t}}\left(f\left(w^{(t)}\right)\right)$$

(3) or the last value discovered

$$w_{\text{SGD}}^{(T)}=w^{(T)}$$

Note 10. SGD output converges to a local minimum, $w_{\text{SGD}}^{(T)}\rightarrow w_{*}$ (in some sense), under different sets of regularity conditions. Section **??** has a brief analysis. To achieve this, Conditions 11 on the learning rate are inevitable and should be satisfied.

Condition 11. Regarding the learning rate (or gain) $\{\eta_t\}$ should satisfy conditions

1. $\eta_t \geq 0$,
2. $\sum_{t=1}^{\infty} \eta_t = \infty$
3. $\sum_{t=1}^{\infty} \eta_t^2 < \infty$

Remark 12. The popular learning rates $\{\eta_t\}$ in Remark 9 in Handout 2 satisfy Condition 11 and hence can be used in SGD too.

Remark 13. Intuition on Condition 11. Assume that v_t is bounded. Condition 11((3)) aims at reducing the effect of the stochasticity in v_t (introduced noise ξ_t) because it implies $\eta_t \searrow 0$ as $t \to \infty$ and hence allows the chain to converge as

$$w^{(t+1)} - w^{(t)} = -\eta_t v_t \to 0.$$

Condition 11(2) prevents η_t from reducing too fast and allows the generated chain $\left\{w^{(t)}\right\}$ to be able to converge. E.g., after t iterations

$$\begin{aligned}\left\|w^{(t)} - w^*\right\| &= \left\|w^{(t)} \pm w^{(0)} - w^*\right\| \geq \left\|w^{(0)} - w^*\right\| - \left\|w^{(t)} - w^{(0)}\right\| \\ &\geq \left\|w^{(0)} - w^*\right\| - \sum_{t=0}^{\infty} \left\|w^{(t+1)} - w^{(t)}\right\| = \left\|w^{(0)} - w^*\right\| - \sum_{t=0}^{T-1} \|\eta_t v_t\|\end{aligned}$$

However if it was $\sum_{t=1}^{\infty} \eta_t < \infty$ it would be $\sum_{t=0}^{\infty} \|\eta_t v_t\| < \infty$ and hence $w^{(t)}$ will never converge to w^* if the seed $w^{(0)}$ is far enough from w^*.