Lecture 24: Deep neural networks

Professor Ilias Bilionis

The stochastic gradient decent and variants

The Robbins-Monro algorithm

Stochastic optimization algorithm

$$\min_{\vartheta} \mathbb{E}_z[\ell(\vartheta; z)] \quad (*)$$

1. Initialize ϑ_0.
2. Iterate:

$$\vartheta_{t+1} = \vartheta_t - \alpha_t \nabla_{\vartheta} \ell(\vartheta; z_t)$$

(α_t: learning rate (scalar))

where z_t is a sample of z. (i.i.d.)

RM theorem: the algorithm above will converge to a local minimum of the st. opt. $(*)$ if

$$\sum_{t=1}^{\infty} \alpha_t = +\infty \; ; \; \sum_{t=1}^{\infty} \alpha_t^2 < \infty$$

↳ (α_t goes to zero but not too fast)

$$\alpha_t = \frac{A}{(Bt+C)^{\rho}}, \quad 0.5 < \rho < 1$$

Application of Robbins-Monro to loss minimization

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - f(x_i;\theta)\right)^2$$

m (batch size), $I_1, \dots, I_m \sim \text{Categorical}(1/n, \dots, 1/n)$ ind.

$$\ell_m(\theta; I_{1:m}) = \frac{1}{m}\sum_{j=1}^{m}\left(y_{I_j} - f(x_{I_j};\theta)\right)^2$$

still have to define the right loss function

converges to local min of this

$$\min_\theta L(\theta) \iff \min_\theta \mathbb{E}_{I_{1:m}}\left[\ell_m(\theta; I_{1:m})\right]$$

R.M.

- Sample indices $i_1, \dots, i_m$ from $I_1, \dots, I_m$
- $\theta_{t+1} = \theta_t - \alpha_t \nabla_\theta \ell_m(\theta_t; i_{1:m})$

$$\rightarrow \theta_{t+1} = \theta_t - \alpha_t \nabla_\theta \sum_{j=1}^{m}\left\{y_{i_j} - f(x_{i_j};\theta_t)\right\}^2$$

$$= \theta_t - \alpha_t \sum_{j=1}^{m} 2\left(y_{i_j} - f(x_{i_j};\theta_t)\right)\cdot \nabla_\theta f(x_{i_j};\theta_t)$$

error of prediction

gradient of the model wrt to all parameters

same dimension as the parameters

Backpropagation

Tensorflow

Pytorch

PREDICTIVE SCIENCE LABORATORY

Advanced variants of stochastic gradient descent

↳ to avoid having to select the learning rate

- Stochastic gradient descent with momentum.
- AdaGrad.
- Adam (adaptive moment estimation).