Final Project for "Optimization Methods"

Zaiwen Wen
Beijing International Center for Mathematical Research
Peking University

December 13, 2022

1 Algorithms and analysis for shallow neural network

Let $f(\theta; x_i) \in \mathbb{R}^m$ be the output of a neural network and $\sigma(x)$ be one of the following element-wise activation function

$$\text{Relu}(x) = \max(0, x), \tag{1.1}$$

$$\text{Sigmoid}(x) = \frac{1}{1+\exp(-x)}, \tag{1.2}$$

$$\text{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{1.3}$$

For a given data set $\{\{x_i, y_i\}_{i=1}^N\}$, consider the supervised learning problem:

$$\min_{\theta} \quad h(\theta) = \sum_{i=1}^{N} L(y_i, f(\theta; x_i)), \tag{1.4}$$

where L is defined as one of the following cases:

- classification problem with m classes:

$$\text{Cross-entropy loss}: L(y_i, f(\theta; x_i)) = -\log\left(\frac{\exp(f(\theta; x_i)[y_i])}{\sum_{j=1}^m \exp(f(\theta; x_i)[j])}\right), \tag{1.5}$$

  where $y_i \in [1, 2, \ldots, m]$ is a class index and $f(\theta; x_i)[y_i]$ means the y_i-th element of the output $f(\theta; x_i)$.

- regression problem:

$$\ell_2 \text{ loss}: L(y_i, f(\theta; x_i)) = \frac{1}{2}\|y_i - f(\theta; x_i)\|_2^2, \tag{1.6}$$

  where y_i and $f(\theta; x_i)$ are vectors with the same sizes.

Examples of the neural network are listed as follows.

1. The two-layer feed-forward neural network is

$$f(\theta; x) := f_{W,v}(x) = \frac{1}{\sqrt{p}} v^\top \sigma(Wx).$$

where $x \in \mathbb{R}^d$ is the input data, $W \in \mathbb{R}^{p\times d}$, $v \in \mathbb{R}^p$ are weight matrices ($m = 1$ in this case). The parameters are concatenated into one column vector $\theta = [\mathrm{vec}(W)^\top, v^\top]^\top$, where $\mathrm{vec}(W)$ transforms the matrix W into a vector by stacking all the columns of W one underneath the other.

2. Adding one more activation layer gives

$$f(\theta; x) := f_{W,V}(x) = \sigma(V\sigma(Wx)),$$

where $x \in \mathbb{R}^d$ is the input, $W \in \mathbb{R}^{p\times d}$, $V \in \mathbb{R}^{m\times p}$ are weight matrice, and σ is an element-wise nonlinear function. All parameters are concatenated into one vector $\theta = [\mathrm{vec}(W)^\top, \mathrm{vec}(V)^\top]^\top$.

In the following questions, the neural network, the activation and loss function can be any of their combinations. Of course, you can consider multiple choices of them or even cover all of them.

1. Compute the gradient (or subgradient) of $h(\theta)$ with respect to θ and estimate its Lipschitz constant. Check whether the gradient can be written as the form of $\mathrm{vec}(uv^\top)$ for some vectors u and v.

   Reference on matrix calculus and backpropagation:

   - Stanford: CS224n, Christopher Manning, Gradients by hand (matrix calculus) and algorithmically (the backpropagation algorithm), `https://web.stanford.edu/class/cs224n/slides/cs224n-2020-lecture04-neuralnets.pdf`
   - Stanford: CS224n: Natural Language Processing with DeepLearning, `https://web.stanford.edu/class/cs224n/readings/cs224n-2019-notes03-neuralnets.pdf`

2. Compute the Hessian of $h(\theta)$ with respect to θ if it is available and estimate the upper and lower bound of its eigenvalues. Check whether each block of the Hessian can be written as the form $A \otimes B$ for some matrices A and B.

3. Compute the empirical Fisher matrix and compare the differences to the Hessian matrix. See the following paper on the the empirical Fisher matrix:

   - New Insights and Perspectives on the Natural Gradient Method, `https://jmlr.org/papers/v21/17-678.html`

4. Will the function $h(\theta)$ be strongly convex in a small neighbourhood of the global optimal solution? Either try numerical experiments on a few small examples or try to establish certain theoretical results.

5. Check if $h(\theta)$ satisfies the so-called Polyak-Łojasiewicz condition, i.e., there exists a constant c such that

$$h(\theta) - h_{inf} \le c\|\nabla h(\theta)\|,$$

where h_{inf} is the optimal value of (1.4).

6. Suppose that **the gradient descent method** with certain line search schemes is applied to solve (1.4). Write down the method and the corresponding convergence results from the classic textbooks on nonlinear programming such as

   - "Numerical Optimization", Jorge Nocedal and Stephen Wright, Springer

- "Optimization Theory and Methods", Wenyu Sun, Ya-Xiang Yuan

Is it possible that this method converges to a global optimal solution of (1.4)? Either try numerical experiments on a few examples or try to establish certain theoretical results.

7. Suppose that **the stochastic gradient method** is applied to solve (1.4). Write down the method and the corresponding convergence results from a few literatures. Is it possible that this method converges to a global optimal solution of (1.4)? Either try numerical experiments on a few examples or try to establish certain theoretical results, for example, see section 6 in the following paper:

   - Conservative set valued fields, automatic differentiation, stochastic gradient methods and deep learning, `https://link.springer.com/article/10.1007/s10107-020-01501-5`

8. Read sec. 5.2.1 and 5.2.2. in the following paper. Write a summary of the main results applied to (1.4).

   - On the Finite-Time Complexity and Practical Computation of Approximate Stationarity Concepts of Lipschitz Functions, `https://proceedings.mlr.press/v162/tian22a.html`

Requirement:

1. Answer at least five questions at your own preferences.
2. Test problems (If you do numerical experiments):
   - random examples created by yourself
   - MNIST or Cifar-10.
3. Prepare a report including
   - detailed answers to each question
   - numerical results and their interpretation if there are numerical experiments
4. Pack all of your codes in one file named as "proj-neu-name-ID.zip" and send it to TA:

   pkuopt@163.com
5. If you get significant help from others on one routine, write down the source of references at the beginning of this routine.